Volume 46 Number 1 March 2022

ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

Editorial Boards

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor from the Editorial Board can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the list of referees. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board are confirmed by the Executive Editors.

The coordination necessary is made through the Executive Editors who examine the reviews, sort the accepted articles and maintain appropriate international distribution. The Executive Board is appointed by the Society Informatika. Informatica is partially supported by the Slovenian Ministry of Higher Education, Science and Technology.

Each author is guaranteed to receive the reviews of his article. When accepted, publication in Informatica is guaranteed in less than one year after the Executive Editors receive the corrected version of the article.

Towards a Feasible Hand Gesture Recognition System as Sterile Non-contact Interface in the Operating Room with 3D Convolutional Neural Network

Roy Amante A. Salvador and Prospero C. Naval, Jr.
E-mail: rasalvador1@up.edu.ph, pcnaval@up.edu.ph
Computer Vision and Machine Intelligence Group, Department of Computer Science
College of Engineering, University of the Philippines, Diliman, Quezon City, Philippines

Keywords: hand gesture recognition system, computer vision, 3d convolutional neural network, operating room

Received: February 16, 2021

Operating surgeons are constrained when interacting with computer systems as they traditionally utilize hand-held devices such as keyboard and mouse. Studies have previously proposed and shown the use of hand gestures is an efficient, touchless way of interfacing with such systems to maintain a sterile field. In this paper, we propose a Deep Computer Vision-based Hand Gesture Recognition framework to facilitate the interaction. We trained a 3D Convolutional Neural Network with a very large scale dataset to classify hand gestures robustly. This network became the core component of a prototype application requiring intraoperative navigation of medical images of a patient. Usability evaluation with surgeons demonstrates the application would work and a hand gesture lexicon that is germane to Medical Image Navigation was defined. By completing one cycle of usability engineering, we prove the feasibility of using the proposed framework inside the Operating Room.

Povzetek: Prispevek skuša dokazati izvedljivost uporabe globokega računalniškega sistema za prepoznavanje kretenj z roko v operacijski sobi.

1 Introduction

Hand Gesture Recognition (HGR) Systems for interfacing is now more interesting and relevant than ever. The development and deployment of contactless technology could be part of community preparedness and response during disease outbreaks. With the ongoing Coronavirus Disease pandemic (COVID-19), it behooves us to observe guidelines such as frequent hand sanitation, social distancing, and the wearing of personal protective attire to reduce the spreading of pathogens and contamination. This aseptic setting is more strictly enforced in the Operating Room (OR) domain.

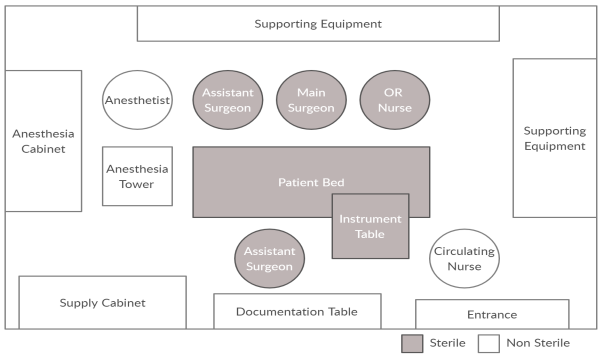


Figure 1: A typical operating room layout [1] depicting personnel and objects which should remain sterile during operations.

During operations, surgeons are not able to physically touch unsterile objects due to safety and health regulations but may need to control a medical system such as navigating to the patient's medical images, viewing for reference, and manipulating them on the screen. A simple hand gesture like swiping in the air would be more convenient for the surgeon and still a compliant way of interfacing with the system. Such a touchless system driven by modalities like hand gestures helps maintain sterility. Figure 1 shows a visualization of the Operating Room setting in terms of sterility. Wipfli et al. [2] compared and evaluated the gesture-controlled approach versus assistant-controlled with guided instructions from the surgeon when it comes to manipulating images in a surgery setting. The former received significantly higher ratings on efficiency and surgeon satisfaction.

Building a Hand Gesture Recognition System is non-trivial. It may be challenging to develop a quick and reliable method of detecting and recognizing the human hand in dynamic environments such as the operating room. For example, the hand may vary in size, color, illumination, position, and orientation to the camera. Many local and global invariant features have been manually designed and engineered to cope with these variabilities such as Histogram of Gradients (HOG) [3], Countour Description [4], Hu Invariant Moments [5], Fourier Descriptors [6], and Karhunen-Loeve Transform [7]. As with the no free lunch theorem, none of the features are better all the time from the others but there are factors and situations where they perform

relatively better.

Deep Learning is a branch of machine learning based on a set of techniques that attempt to model high-level data abstractions using multiple processing layers composed of complex structures and transformations. It has been applied and proven useful in fields including but not limited to computer vision, natural language processing, speech recognition, and knowledge representation. As a subset of Representation learning [8], Deep Learning has the advantage of learning features automatically from data whereas rule-based and classical machine learning approaches need manual engineering and extracting of hand-designed features. It can also eliminate the need for pre-processing steps such as Hand Tracking and Region Segmentation making the processing pipeline simpler and straight-forward (raw data in; prediction out).

We approach the issue of maintaining sterility in the Operating Room by proposing the usage of a Real-time Computer Vision and Deep Learning-based Hand Gesture Recognition framework. The main contributions of our work are the following:

1. Demonstration of the feasibility of the proposed framework inside the Operating Room by designing and building a basic Medical Image Navigation prototype that is positively evaluated by surgeons.

2. Definition of a Hand Gesture Lexicon that is suitable for Medical Image Navigation application and consequently also appropriate to use inside the operating room.

3. A new Dynamic Hand Gesture dataset we've collected during the development of the prototype application. It is medium-sized containing more hand gesture samples than the Sheffield Kinect Gesture (SKIG) [9] dataset.

2 Hand gesture recognition systems in the operating room

Pioneering work in this arena heavily applied traditional computer vision techniques for performing image preprocessing, hand detection, and hand tracking and used finite-state machine for gesture classification [10, 11]. Some of them had poor usability and caused fatigue for the users [12]. A classical machine learning approach was taken by Achacon et al. [13]. Their system called *REALISM* included only a few gesture classes. They first performed hand detection with Haar-like features and cascade classifier then employed Principal Component Analysis and Euclidean Distance matching from the samples of the classes to perform classification.

Jacob et al. [14] defined a set of gestures for navigating medical images in consultation with veterinary surgeons. They used the 3D trajectory (3Dt) of the hand as the feature and Hidden Markov Model (HMM) as the classifier. The computation of the hand trajectory relied however on the Skeletal Tracking feature of Microsoft Kinect. They also used the skeletal information to compute head and torso orientation for determining intentional gestures. Several other HGR systems in the operating room [1, 15, 16, 17] have used and depended on the Microsoft Kinect device. Another popular device is the Leap Motion Controller (LMC) which uses proprietary drivers to process and format data into frames of objects like hands and fingers [18]. Park et al. [19] developed a message hooking program called *GestureHook* to convert gestures into mouse and keyboard functions to their medical system. [20] applied a rule-based classification based on the 3D hand movement (trajectory) using the points returned by the LMC device.

With their ability to project holograms that can be accessed interactively with hand gestures, mixed reality headsets are now making their way inside the operating room to support surgical procedures. A study by Galati et al. [21] found that they can increase the surgeon's productivity but highlighted that the battery autonomy and the weight of the device which can cause physical stress and discomfort are points for improvement. Furthermore, these devices cost significantly higher than the other capture devices.

3 Proposed framework

We propose a Hand Gesture Recognition System using Deep Computer Vision to act as an interface of the surgeon to a medical image navigation application. To do this successfully, we aimed at training a deep network and developing a framework which classifies static and dynamic gestures:

1. *With High Accuracy* - Classification performance must be invariant to the user and background. It must be at least comparable in performance with baseline systems.

2. *In Real-time* - The system must be able to complete the processing pipeline at most 250 milliseconds. The average reaction time for visual stimuli ranges from 250 to 350 milliseconds [22]. Taking longer than this range would introduce a noticeable lag visually.

3.1 Data capture

Microsoft Kinect was first explored for capturing the hand gestures as it provides depth modality. With depth information, one can easily determine which objects or pixels are in the foreground and the background. Moreover, the depth sensor of Kinect is an infrared camera so the effect of lighting conditions, user's skin color and clothing, and the background were assumed to have small to no impact on system performance. After the introduction of the first-generation Microsoft Kinect in 2010, there have been recognition systems developed and research using the device [23, 24, 25]. The simpler, cheaper and more available way is to use a

single regular camera to capture the user's gesture. Many laptops have webcams and they are much widely available especially in developing countries. Usage of a single webcam also works in the proposed framework.

3.2 Network architecture

We determined the separation of networks for static hand postures and dynamic gestures is unnecessary as static hand gestures can also be seen as dynamic. Even though a static hand posture may not move in space, it always moves forward in time. With this, we chose a 3D Convolutional Neural Network as they are well-suited for systems involving spatiotemporal feature learning [26, 27, 28]. Specifically, it is a slightly modified version of the C3D model which was used for action recognition, action similarity labeling, and scene classification [29]. It resembles a VGG-11 [30] architecture replacing all 2D convolutional layers with 3D convolutional layers followed by Batch Normalization. The input to the network is the 16 RGB and depth fused frames, sized 96 x 96. The sizing of the input and network depended on the capacity of the GPU (NVIDIA GTX 970M graphics card) of the demo laptop machine used. Prediction of one sample took around 60 ms on battery and 40 ms when plugged in on the mentioned machine. We also verified the network with the SKIG [9] dataset using RGB only, and both RGB and depth (RGBD) on three-fold cross-validation. Results can be seen in Table 1. There might be other network architectures that are arguably better but our purpose is to only be able to predict robustly and quickly based on our defined guidelines for accuracy and real-time processing speed.

3.3 System actions / functionalities

The system actions include the ten functionalities listed by Jacob et al. [14]. The set was suggested by surgeons who were asked to give the most common functions they perform with medical images during surgeries. These are actions for navigation, brightness, orientation, and zoom level manipulation. Furthermore, we've added auxiliary functionalities for usability guided by consultation with a surgeon. These are locking/unlocking the system, panning, and animation of the images in the medical series. The Lock and Unlock actions signal to the system the user's intent to use the system. Locking the system reduces the possibility of recognizing unintended hand gestures performed by the user. The Panning action is seen as a supporting functionality when zoning into regions of interest. While the series animation functionalities not only enable viewing of the medical series but also help with navigating to a specified image as a series can contain hundreds or thousands of images/slices.

3.4 Medical image navigation interface

Developed in OpenCV [31], the interface is designed as a Medical Image/DICOM Viewer-like application. Digital Imaging and Communications in Medicine (DICOM) is the international standard format for storing medical imaging and information. In consultation with radiologists and surgeons, we were given and chose anonymized medical images taken from a real case with Appendicitis. We extracted and organized its images for the application to display as sample patient data.

3.5 Processing pipeline

As a real-time system, the network continuously gives its prediction for every new frame it receives from the camera but we need to treat the gestures of the user as discrete actions. We expect the user to perform each intentional gesture for about 1 second. For System Actions to be triggered, the prediction for the particular gesture class must be sustained by a set duration - gestures must continuously be predicted by the network in τ timesteps. If this condition is not met, we treat them as unintended gestures hence won't trigger any action. The workflow for executing System Actions can be summarized by the diagram in Figure 4. On our demo laptop with 16 GB memory, Intel Core i-7 processor, and NVIDIA GTX 970M graphics card, the application logic and rendering took around 50 ms running on battery and around 25 ms when running on power. Combining this with the network's prediction time, one cycle takes around 110 ms on battery, and 65 ms when running on power. This means the system's performance is in the range of about 8 to 15 fps and the suitable duration threshold τ is within this range.

4 Training our hand gesture recognition network

In this section, we discuss the efforts carried out in training the network used in our prototype application. Due to the size of the datasets, all network training efforts were performed in a Google Cloud Instance with 24 GB of memory and P100 GPU. We used Lasagne [32] deep learning library to train the networks using Stochastic Gradient Descent (SGD) with Nesterov Momentum with learning rate from $1e^{-2}$ to $1e^{-6}$ annealed by a factor of 0.1 whenever performance on the validation set did not improve. Moreover, we employed heavy Data Augmentation by manipulating the frames of the video samples - scaling inward and outward, random brightness and contrast, applying Gaussian noise, and random and multiple frame sampling of the training video clips.

4.1 Collecting and using our dataset

We first naively collected our dataset with the Microsoft Kinect camera. Some example RGB frames with their cor-

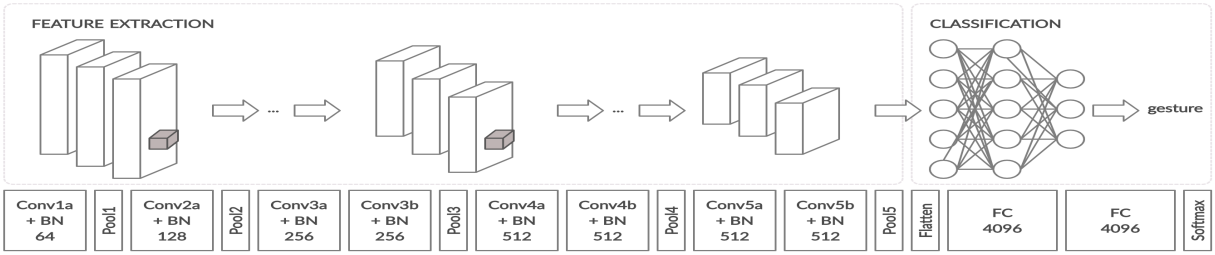


Figure 2: Modified C3D [29]. The network contains eight 3D convolutional layers (Conv*xx*) having a kernel size of $3 \times 3 \times 3$ with a stride of 1 in all dimensions. The number of filters is indicated in their corresponding boxes. Each convolutional layer is followed by Batch Normalization (BN). There are five 3D max-pooling layers (Pool*x*) each with pooling kernel size and stride of $2 \times 2 \times 2$, except for *Pool1* which is $1 \times 2 \times 2$. The series of convolutional, batch normalization and pooling layers act as the feature extraction phase producing 3D feature maps. The network is closed off with the classification phase by two fully connected layers (FC) with 4096 neurons each and the output softmax layer whose size is the number of gesture classes.

Table 1: Accuracy of the Modified C3D Model on the SKIG Dataset. We first validated our chosen network architecture (Modified C3D Model) with the Sheffield Kinect Gesture (SKIG) [9] dataset. Data is divided into three folds each fold contains all samples from two subjects - Fold A (subjects 1 and 2), Fold B (subjects 3 and 4), and Fold C (subjects 5 and 6). For RGB only, and RGB with depth (RGBD) modality, the accuracy is more than our benchmark accuracy of around 93%.

	Fold A	Fold B	Fold C	Avg±Stdev
RGB	92.22%	94.17%	95.00%	93.8%±1.43%
RGBD	93.06%	93.61%	98.33%	95.0%±2.90%

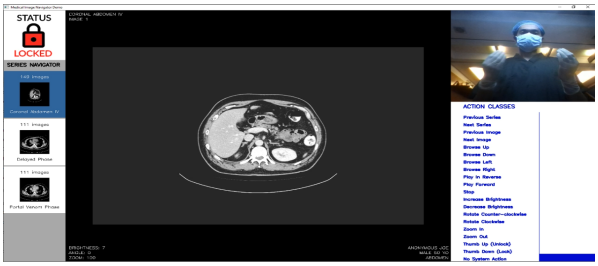


Figure 3: The Medical Image Navigation Application Interface. It is designed as a DICOM viewer-like application. The status (whether Locked or Unlocked) is displayed at the top-left corner. On the left-hand side is the list of available DICOM series for viewing. The center panel displays the current image/slice of interest of the current series. We can see the video stream and real-time visualization of the prediction of our system on the right-hand side of the application.

responding depth frames can be seen in Figure 5. Samples from four users were incrementally used for training and samples from the remaining user were used for validation. Adding more users to the training set was seen to increase the classification performance, however, at four users, the quality of the network is still very poor. The collection of new data and annotation is a long and tedious process so it was decided to look for a public dataset that can help boost performance via Transfer Learning.

4.2 Leveraging very large scale hand gesture dataset

The 20BN-JESTER [33] dataset is a very large scale hand gesture recognition dataset containing more than a hundred thousand densely-labeled clips performed by a huge number of crowd workers in front of a webcam or laptop camera. With transfer learning in mind, the selected network architecture is trained with the Jester dataset applying all previously mentioned regularization techniques. Since we have 4 channels as input to our chosen network and the Jester dataset only has RGB, a blank white image is fused in place of the depth frame. This is because, in Microsoft Kinect's depth images, white represents background. Samples in the dataset have varying lengths so, during prediction time, 16 frames sampled at equal intervals are extracted to represent the entire clip. Our trained network scored an overall accuracy of 94.52% on the official Jester validation set and around this value for the test set.

4.3 Fine-tuning

We initialized the network with the weights of our network trained with Jester Dataset and performed finetuning with our dataset. A sharp increase in performance as opposed to training from scratch is seen; however, the resulting System Accuracy is still quite poor at 67.58%. This might be because in the Jester dataset the gestures are performed directly in front of the camera so the hand is a dominant part

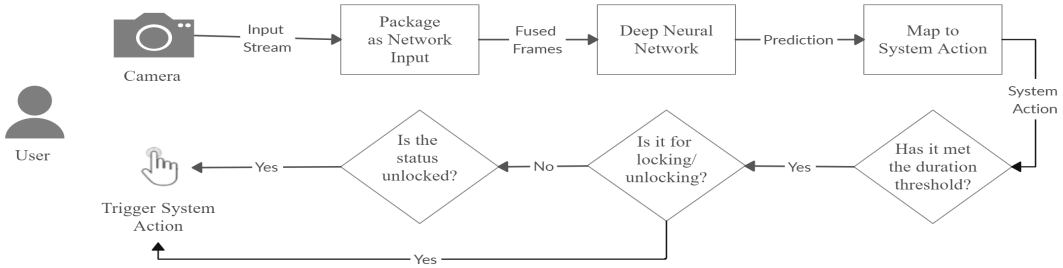


Figure 4: Overview of the Processing Pipeline. The camera continuously sends the video frames and the system collects the last 16 frames, resizes, and organizes them as input to our deep neural network. Every time the network performs an inference, the predicted gesture class is mapped to a system action. We check if the current system action has been sustained long enough for an actual gesture to take place. We further reduce the occurrence of triggering system actions for unintended gestures from the user by checking the application status (whether Locked or Unlocked).

Figure 5: Our Dynamic Hand Gesture Dataset. Microsoft Kinect is used to capture the gestures of 5 users in RGB and depth modality. There are 16 gestures classes, 10 of which were taken from Jacob et al. [14], and a no system action class. Sequences are recorded with 3 varying backgrounds (blue, green and pink) and 3 different scales (user distance from the camera - 1, 2, 3 steps from the camera) performed each on their left and right hand unless the gesture needs two hands to perform. For each take, the users were asked to dress differently and apply different hairdos to increase variability. The dataset includes 1400+ gesture samples.

of the frame. In our dataset, gestures are performed standing with some distance from the Microsoft Kinect camera, making the hand relatively small compared to everything else in the frame. Another factor could be the Jester dataset only has the RGB modality. If it also has depth information, that might have further helped with boosting performance.

4.4 Utilizing Jester-trained network

At this point, we have trained a robust network capable of predicting for different users on different types of complex backgrounds. We assumed it would also work for a surgeon inside the Operating Room. Instead of forcing the use of our gesture lexicon, we used the ones in the Jester dataset and mapped the most intuitively matching gestures into our System Actions. Table 2 and Figure 6 details the mapping and its performance respectively. Table 3 summarizes the performance of training with the SKIG dataset, our dataset, and the Jester dataset. Compared to working with our dataset, utilizing the Jester-trained network meets the criterion for robust gesture classification.

5 Evaluation

5.1 Baseline systems

We evaluated against prior work on HGR systems for OR usage. The baseline systems include: *REALISM* [13], 3Dt+HMM without and with contextual cues (WC) [14], *GestureHook* [19], binarized (b-) and raw depth+2D-CNN [34], 3Dt+rule-based (RB) classification [20], and IR+ CapsNet [35]. They are listed in Table 4. We note that we compare the systems as a whole in which the number of action classes, types of gesture, and capture device are packaged and designed as a system. We addressed the collective weaknesses of these systems which include the need for manual engineering of features or pre-processing procedures to achieve a feasible recognition performance [13, 14, 34, 20], constraints on the hand gesture lexicon containing only static or movement hand gestures only but unable to process both due to methodology [13, 14, 34, 35], and reliance on the capabilities of the data capture device which are more expensive and may not be readily available in developing countries as an ordinary webcam would [14, 19, 34, 20, 35].

5.2 Test setup in the operating room

We were given a brief timeframe and have attempted to validate our system by testing it in an actual operating room. We recorded and annotated a sequence of gestures to see how it performs in real-time. Figure 7 shows the confidence over time of our trained network which is the output of the final softmax layer. Visually, the parts where there is sustained prediction confidence at 1.0 coincide with the ground truth. Quantitatively, the continuous recognition results in System Action Accuracy (SAA) of 94.95%, No Action Precision (NAP) of 96.54%, and No Action Recall (NAR) of 63.90% for the test sequence. The low NAR is attributed to confidence spikes for other gestures that corre-

Table 2: Jester Mapped Actions. Each gesture in the Jester dataset is intuitively mapped with our system actions.

Our System Action	**Jester Gesture**
Previous Series	Swiping Up
Next Series	Swiping Down
Previous Image	Swiping Left
Next Image	Swiping Right
Browse Up	Sliding Two Fingers Up
Browse Down	Sliding Two Fingers Down
Browse Left	Sliding Two Fingers Left
Browse Right	Sliding Two Fingers Right
Play Series In Reverse	Rolling Hand Backward
Play Series	Rolling Hand Forward
Stop Playing Series	Stop Sign Pushing Hand Away
Increase Brightness	Pushing Two Fingers Away
Decrease Brightness	Pulling Two Fingers In
Rotate Counter-clockwise	Turning Hand Counterclockwise
Rotate Clockwise	Turning Hand Clockwise
Zoom In	Zooming In With Full Hand Zooming In With Two Fingers
Zoom Out	Zooming Out With Full Hand Zooming Out With Two Fingers
Thumbs Up (Unlock)	Thumb Up
Thumbs Down (Lock)	Thumb Down
No System Action	No gesture Doing other things Pulling Hand In Drumming Fingers Shaking Hand

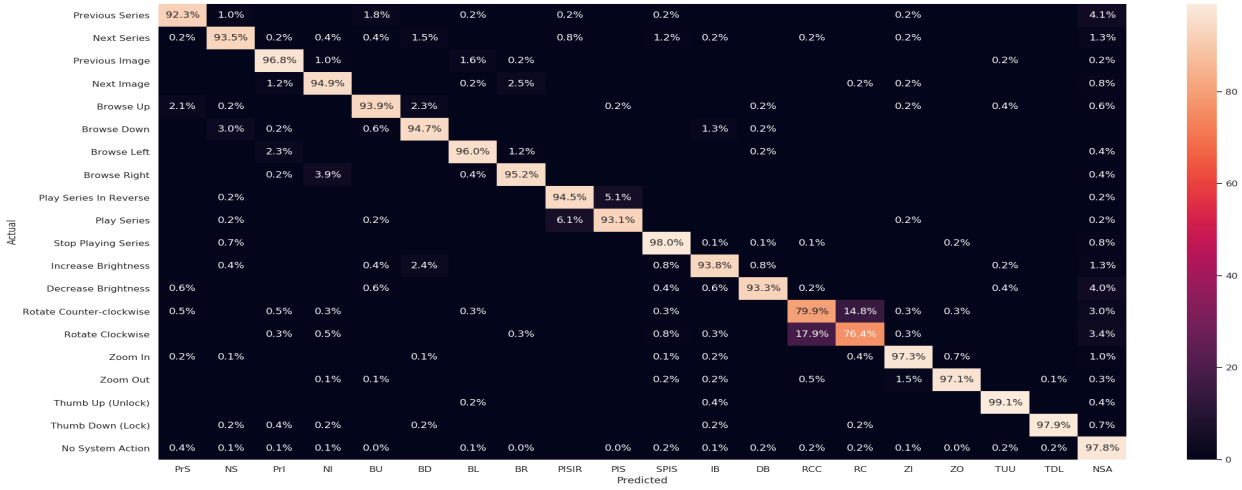


Figure 6: Jester Mapped Actions Performance. The confusion matrix of the proposed mapping.

spond to unintended gestures. They generally do not trigger system actions since we employed a set duration threshold to mitigate.

5.3 Usability evaluation with surgeons

We created an online survey form for the evaluation of the hand gestures as well as the overall usability of the application. There were 11 survey respondents from 7 in-

Table 3: Hand Gesture Recognition Results. We quantified the robustness of the trained network by the following: System Action Accuracy (SAA) tells us how good the network is in identifying system action classes, No Action Precision (NAP) shows how resilient the network is against classifying intended gestures as unintended (classified as no system action), and No Action Recall (NAR) denotes how resilient the network is against classifying unintended gestures as intended.

	SAA	**NAP**	**NAR**
SKIG Dataset (RGB)	93.80%	-	-
SKIG Dataset (RGBD)	95.00%	-	-
Own Dataset (1 user)	0.96%	79.18%	97.37%
Own Dataset (2 users)	23.58%	83.59%	75.38%
Own Dataset (3 users)	29.87%	86.15%	82.05%
Own Dataset (4 users)	43.57%	90.80%	88.93%
Jester Validation Set	94.52%	-	-
Jester Test Set	94.26%	-	-
Jester pre-training + Own Dataset (4 users)	67.58%	94.65%	91.44%
Jester Mapped Actions	94.48%	96.62%	97.80%

Table 4: Comparison with Baseline Systems. We relate the performance of our work with their reported System Action Accuracy (SAA) along with the type of gestures used (whether static, dynamic, or both), and the number of System Actions/commands.

	Capture Device	**Gesture Type**	**Number of System Actions**	**System Action Accuracy (SAA)**
REALISM [13]	webcam	static	5	$< 80\%$
3Dt+HMM [14]	Kinect	dynamic	10	93.60%
3Dt+HMM WC [14]	Kinect	dynamic	10	92.58%
GestureHook [19]	LMC	dynamic	8	$\approx 92\%$
depth+2D-CNN [34]	ToF	static	10	94.86%
b-depth+2D-CNN [34]	ToF	static	10	92.07%
3Dt+RB [20]	LMC	dynamic	10	95.83%
IR+CapsNet [35]	LMC	static	5	86.46%
Ours	webcam	both	19	94.48%

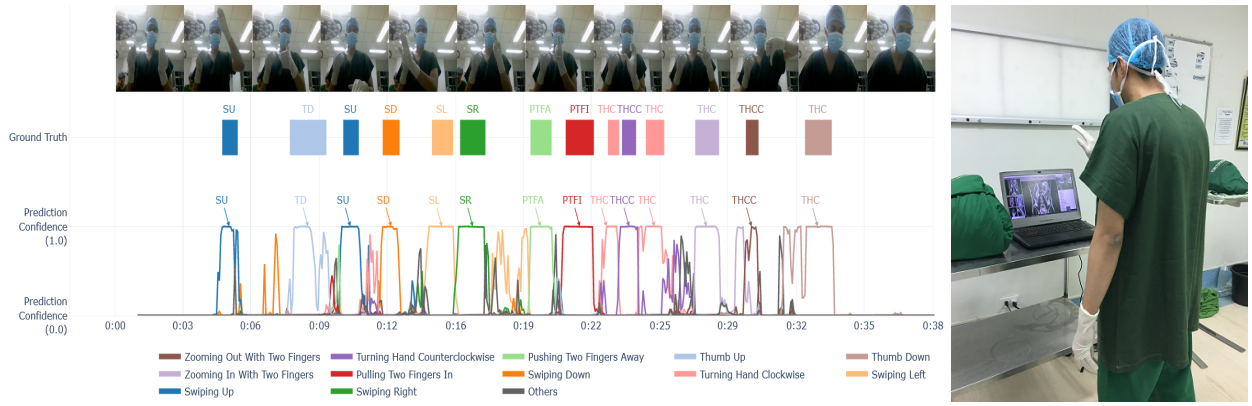
stitutions with varying experience ranging from fellows in post-residency training to attending physicians from different specialties such as General Surgery, Pediatric Surgery, Surgical Oncology, and Surgical Endoscopy. To avoid biased feedback, our main consulting surgeon did not participate. Participants were first introduced with the purpose of the research and then walked through the application by showing them images of the setup and video clip recordings of each system action being triggered in the system by their corresponding hand gesture. We asked them to choose from Strongly Disagree, Disagree, Neutral, Agree, and Strongly Agree for each criterion and we converted their answers numerically from 1 to 5 respectively for analysis. At the end of the form, they were also asked to evaluate and provide their overall thoughts on the usability of the system.

5.4 Hand gesture lexicon

For the hand gestures to be feasible, they should be positively rated on the following criteria [12]: Intuitiveness (Gesture Intuitively Matches the System Action Performed), Ease of Use (Gesture Is Easy and Comfortable to Perform), and Memorability/Ease of Remembrance. Additionally, we also included Appropriateness to use inside the OR. Table 5 shows the detailed ratings of the surgeons in our evaluation survey. All of the hand gestures received a mean rating greater than 3 (Neutral), and the majority of them greater than 4 (Agree). Most surgeons did not provide any suggestions to improve implying they were content with the gestures. With this, we can say that we have a workable initial set of hand gestures for the application just by using the ones in the Jester dataset.

A few of the hand gestures particularly those for brightness and orientation manipulation and playing the series received a relatively lower rating across all criteria. Some of the comments mention a preference for gestures involving smaller movements. Some suggested highly subjective gestures and additional functionalities with no agreement with other respondents. To determine the final set of proposed gestures for a Medical Image Navigation application in the operating room, we perform the following:

1. If the gesture received consistently lower scores across all criteria, we replace it with:
   (a) The suggested improved gesture provided by at



(a) (b)

Figure 7: Test inside the operating room. (a) Visualization of the continuous hand gesture recognition during the test. (b) The system hardware (a laptop with a webcam in this case) is placed in a convenient location adhering to sterility rules. To consult with the patient's medical images, the surgeon moves in front of the camera and gestures to the system.

least two surgeons in agreement in the evaluation; else with

(b) The suggested improved gesture provided by at least one surgeon in agreement with Jacob et al [14].

2. If there are no suggestions or if the gesture received satisfactory ratings, we keep the gesture.

The resulting improved hand gesture lexicon can be found in Table 6. We note that gestures can have different meanings across cultures hence some of them might only be suitable in our local setting.

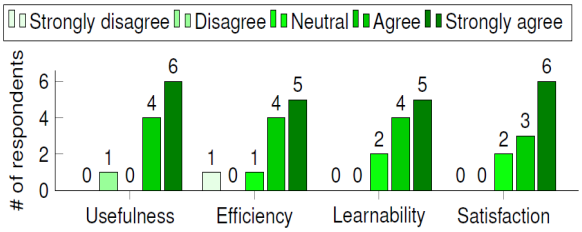


Figure 8: Evaluation survey overall usability rating distribution. Most of the respondents were generally satisfied, agreeing with the system's usefulness, efficiency, and learnability, suggesting high viability of the application.

5.5 Overall usability

For the overall usability of the application, we measure with the following criteria: Usefulness, Efficiency, Learnability, and Satisfaction. Figure 8 depicts the ratings given by the survey respondents. Feedback from surgeons is strongly positive with a mean score between 4 and 5 (from Agree to Strongly Agree) for all of our criteria. At 95% probability, the confidence intervals of the mean rating using t-distribution are 4.36 $\pm$ 0.62 for Usefulness, 4.09 $\pm$ 0.82 for Efficiency, 4.27 $\pm$ 0.56 for Learnability, and 4.36 $\pm$ 0.54 for Satisfaction. Only one surgeon disagreed with the usefulness of the application and the efficiency it would bring to productivity. The consensus is that the system is easily learnable and quite useful. The majority did not leave any further comments but some left suggestions on additional functionalities such as contrast management, measuring, incorporating voice commands, and using additional monitor screens to display the interface and video feed.

5.6 Limitations

Due to restrictions on physical meetings brought about by ongoing local policies regarding community quarantines, evaluating the system with multiple surgeons in an OR session has been a roadblock. However, we believe the results on the Jester evaluation set (a highly variable dataset which contains 14,000+ samples) coupled with our brief test in an operating room translates to a generally feasible hand gesture recognition performance.

For the framework to be seamlessly applied to any application in the operating room or any domain, there should be a capability for defining custom gestures to use [1]. With the current approach, it is difficult to achieve satisfactory performance for new gestures without acquiring a large enough number of samples as shown in Table 3. An extension of this work that could mitigate this issue is to integrate a One or Few-Shot Learning mechanism of hand gestures. [36] had executed this by using a pre-trained network for feature extraction then employing some distance measurement.

6 Conclusion

In this paper, we were able to demonstrate that a straightforward Deep Learning and Computer Vision-based framework is a viable solution in maintaining sterility in the op-

Table 5: Mapped Jester gestures evaluation results. Participants were asked to choose among strongly disagree (1), disagree (2), neutral (3), agree (4), and strongly agree (5) for each criterion for each gesture. The values displayed are the confidence intervals of the mean hand gesture ratings using t-distribution at 95% probability.

System Action	**Hand Gesture**	**Appropriate-ness**	**Intuitive-ness**	**Ease of Use**	**Memorability**
Unlock	Thumb Up	4.00±0.74	3.54±0.70	4.09±0.36	4.27±0.31
Lock	Thumb Down	3.91±0.76	3.91±0.76	4.18±0.27	4.27±0.31
Previous Series	Swipe Up	4.09±0.36	4.09±0.36	3.91±0.56	4.09±0.36
Next Series	Swipe Down	4.00±0.42	4.00±0.42	3.91±0.47	4.00±0.42
Previous Image	Swipe Left	4.36±0.34	4.27±0.43	4.27±0.43	4.36±0.34
Next Image	Swipe Right	4.36±0.34	4.18±0.50	4.09±0.56	4.36±0.34
Play Series	Roll Hand Forward	3.73±0.68	3.73±0.61	3.45±0.35	3.64±0.69
Play Series In Reverse	Roll Hand Backward	3.91±0.56	3.64±0.62	3.45±0.76	3.64±0.69
Stop Series	Palm Facing Screen	4.45±0.35	4.18±0.72	4.36±0.45	4.27±0.53
Pan Up	Slide Two Fingers Up	4.18±0.27	3.91±0.47	4.18±0.27	4.09±0.36
Pan Down	Slide Two Fingers Down	4.18±0.27	4.00±0.42	4.09±0.36	4.09±0.36
Pan Left	Slide Two Fingers Left	4.27±0.31	4.18±0.41	4.27±0.31	4.27±0.85
Pan Right	Slide Two Fingers Right	4.27±0.31	4.18±0.41	4.27±0.31	4.27±0.31
Increase Brightness	Push Two Fingers Away	3.82±0.72	3.27±0.74	3.73±0.75	3.55±0.87
Decrease Brightness	Pull Two Fingers In	3.82±0.72	3.45±0.72	3.73±0.82	3.45±0.92
Rotate Counter-clockwise	Turn Hand Counter-clockwise	4.18±0.27	3.73±0.61	3.91±0.63	3.64±0.75
Rotate Clockwise	Turn Hand Clockwise	4.00±0.52	3.55±0.82	3.73±0.61	3.36±0.69
Zoom In	Open Two Fingers / Hand	4.45±0.35	4.45±0.35	4.36±0.45	4.45±0.35
Zoom Out	Close Two Fingers / Hand	4.45±0.35	4.45±0.35	4.36±0.45	4.45±0.35

erating room. We implemented an end-to-end, real-time, robust Hand Gesture Recognition System applied for usage inside the operating room in the form of a Medical Image Navigation application that is not dependent on the capture device and is positively evaluated by surgeons. General feedback from our local surgeons shows receptiveness and willingness to apply this technology. Furthermore, we defined a set of suitable hand gestures for the application. This set coupled with the framework can serve as a foundation for building and deploying Hand Gesture-controlled applications in our operating rooms as well as in other more lenient settings requiring sterility maintenance.

Acknowledgement

We would like to thank the following medical professionals who helped us with this work: Dr. Alvin Caballes from the University of the Philippines Philippine General Hospital (UP PGH), Department of Surgery, our primary surgeon in consultation with during the creation and evaluation of the prototype application; Dr. Cecilia Amicis and the Radiology Department of Veterans Memorial Medical Center (VMMC) who provided anonymized medical images from areal case study; and Dr. Jose Abellera and the Surgery Department at National Children's Hospital (NCH) who helped and allowed us to perform a brief test setup in one of their operating rooms.

Table 6: Suggested hand gesture lexicon for a medical image navigation application. An initial set of gestures was defined by mapping the System Actions with the Jester dataset [33] gestures. This was refined based on the results of our evaluation survey coupled with the ethnographic study conducted by Jacob et al [14]. It was raised that if there are added functionalities for confirmation, the thumb up and down gestures would be more appropriate for answering yes and no respectively.

System Action	**Hand Gesture**	**Details**
Unlock	Thumb Up	Thumb Up, other fingers tucked in.
Lock	Thumb Down	Thumb Down, other fingers tucked in.
Go To Previous Series	Swipe Up	Palm facing upward. Smaller movement of four fingers pivoting upward.
Go To Next Series	Swipe Down	Palm facing downward. Smaller movement of four fingers pivoting downward.
Go To Previous Image	Swipe Left	Palm facing camera or side. Move hand to the left.
Go To Next Image	Swipe Right	Palm facing camera or side. Move hand to the right.
Play Series	Swipe Down (Side View)	Side view of the hand facing screen. Palm facing downward. Smaller movement of four fingers pivoting downward.
Play Series In Reverse	Swipe Up (Side View)	Side view of the hand facing screen. Palm facing upward. Smaller movement of four fingers pivoting upward.
Stop Playing Series	Stop Sign	Open hand palm facing Screen.
Pan Up	Slide Two Fingers Up	Point index and middle finger. Move hand or two fingers upward.
Pan Down	Slide Two Fingers Down	Point index and middle finger. Move hand or two fingers downward.
Pan Left	Slide Two Fingers Left	Point index and middle finger. Move hand or two fingers towards the left.
Pan Right	Slide Two Fingers Right	Point index and middle finger. Move hand or two fingers towards the right.
Increase Brightness	Push Hand Away	Palm facing camera, move open hand towards camera. Taken from [14].
Decrease Brightness	Pull Hand In	Back of hand facing camera, move open hand away from camera. Taken from [14].
Rotate Counter-clockwise	Swipe Counter-clockwise	Palm facing camera, wave hand to the left (counter-clockwise). Taken from [14].
Rotate Clockwise	Swipe Clockwise	Palm facing camera, wave hand to the right (clockwise). Taken from [14].
Zoom In	Open Two Fingers	Index and thumb touching initially, other fingers tucked in. Move index and thumb away from each other.
Zoom Out	Close Two Fingers	Index and thumb away from each other initially, other fingers tucked in. Move index and thumb towards each other until they touch.

References

[1] A. Bigdelou, *Operating Room Specific Domain Model for Usability Evaluations and HCI Design*. PhD thesis, Technical University Munich, 2012.

[2] R. Wipfli, V. Dubois-Ferrière, S. Budry, P. Hoffmeyer, and C. Lovis, "Gesture-Controlled Image Management for Operating Room: A Randomized Crossover Study to Compare Interaction Using Gestures, Mouse, and Third Person Relaying," *PLoS ONE*, vol. 11, no. 4, p. e0153596, 2016.

https://doi.org/10.1371/journal.pone.0153596.

[3] W. T. Freeman and M. Roth, "Orientation histograms for hand gesture recognition," 1995.

[4] C.-C. Chang, I.-Y. Chen, and Y.-S. Huang, "Hand pose recognition using curvature scale space," vol. 2, pp. 386 – 389 vol.2, 02 2002. https://doi.org/10.1109/ICPR.2002.1048320.

[5] P. Premaratne, *Human Computer Interaction Using Hand Gestures*. Springer Science+Business Media Singapore, 2014. https://doi.org/10.1007/978-3-642-14831-6_51.

[6] S. Conseil, S. Bourennane, and L. Martin, "Comparison of fourier descriptors and hu moments for hand posture recognition," in *2007 15th European Signal Processing Conference*, pp. 1960–1964, 2007. https://doi.org/10.5281/zenodo.40606.

[7] J. Singha and K. Das, "Hand gesture recognition based on karhunen-loeve transform," *Mobile and Embedded Technology International Conference (MECON)*, 06 2013.

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. https://www.deeplearningbook.org.

[9] L. Liu and L. Shao, "Learning discriminative representations from rgb-d video data," in *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pp. 1493–1500, AAAI Press, 2013.

[10] C. Grätzel, T. Fong, S. Grange, and C. Baur, "A non-contact mouse for surgeon-computer interaction," *Technology and health care : official journal of the European Society for Engineering and Medicine*, vol. 12, pp. 245–57, 02 2004. https://doi.org/10.3233/THC-2004-12304.

[11] J. Wachs, H. Stern, Y. Edan, M. Gillam, C. Feied, M. Smith, and J. Handler, "Gestix: A doctor-computer sterile gesture interface for dynamic environments," in *Soft Computing in Industrial Applications* (A. Saad, K. Dahal, M. Sarfraz, and R. Roy, eds.), (Berlin, Heidelberg), pp. 30–39, Springer Berlin Heidelberg, 2007. https://doi.org/10.1007/978-3-540-70706-6_3.

[12] A. Hurstel and D. Bechmann, "Approach for intuitive and touchless interaction in the operating room," *J*, vol. 2, pp. 50–64, 01 2019. https://doi.org/10.3390/j2010005.

[13] M. Achacon, David Louis Jr, D. M Carlos, M. Kaye Puyaoan, C. T Clarin, and P. Naval, "Realism: Real-time hand gesture interface for surgeons and medical experts," 09 2010.

[14] M. G. Jacob, J. P. Wachs, and R. A. Packer, "Hand-gesture-based sterile interface for the operating room using contextual cues for the navigation of radiological images," *J Am Med Inform Assoc*, vol. 20, pp. e183–186, Jun 2013. https://doi.org/10.1136/amiajnl-2012-001212.

[15] G. C. Ruppert, L. O. Reis, P. H. Amorim, T. F. de Moraes, and J. V. da Silva, "Touchless gesture user interface for interactive image visualization in urological surgery," *World J Urol*, vol. 30, pp. 687–691, Oct 2012. https://doi.org/10.1007/s00345-012-0879-0.

[16] M. Strickland, J. Tremaine, G. Brigley, and C. Law, "Using a depth-sensing infrared camera system to access and manipulate medical imaging from within the sterile operating field," *Can J Surg*, vol. 56, pp. 1–6, Jun 2013. https://doi.org/10.1503/cjs.035311.

[17] J. H. Tan, C. Chao, M. Zawaideh, A. C. Roberts, and T. B. Kinney, "Informatics in Radiology: developing a touchless user interface for intraoperative image control during interventional radiology procedures," *Radiographics*, vol. 33, no. 2, pp. 61–70, 2013. https://doi.org/10.1148/rg.332125101.

[18] B. Pavaloiu, "Leap motion technology in learning," pp. 1025–1031, 05 2017. https://doi.org/10.15405/epsbs.2017.05.02.126.

[19] B. J. Park, T. Jang, J. W. Choi, and N. Kim, "Gesture-Controlled Interface for Contactless Control of Various Computer Programs with a Hooking-Based Keyboard and Mouse-Mapping Technique in the Operating Room," *Computational and Mathematical Methods in Medicine*, vol. 2016, p. 5170379, 2016. https://doi.org/10.1155/2016/5170379.

[20] P. Sa-nguannarm, T. Charoenpong, C. Chianrabutra, and K. Kiatsoontorn, "A method of 3d hand movement recognition by a leap motion sensor for controlling medical image in an operating room," in *2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)*, pp. 17–20, 2019. https://doi.org/10.1109/ICA-SYMP.2019.8645985.

[21] R. Galati, M. Simone, G. Barile, R. De Luca, C. Cartanese, and G. Grassi, "Experimental Setup Employed in the Operating Room Based on Virtual and Mixed Reality: Analysis of Pros and Cons in Open Abdomen Surgery," *J Healthc Eng*, vol. 2020, p. 8851964, 2020. https://doi.org/10.1155/2020/8851964.

[22] J. Shelton and G. Kumar, "Comparison between auditory and visual simple reaction times," *Neuroscience & Medicine*, vol. 1, pp. 30–32, 01 2010. https://doi.org/10.4236/nm.2010.11004.

[23] Y. Li, "Multi-scenario gesture recognition using kinect," in *Proceedings of the 2012 17th International Conference on Computer Games: AI, Animation, Mobile, Interactive Multimedia, Educational & Serious Games (CGAMES)*, CGAMES '12, (Washington, DC, USA), pp. 126–130, IEEE Computer Society, 2012. https://doi.org/10.1109/CGames.2012.6314563.

[24] A. Tang, K. Lu, Y. Wang, J. Huang, and H. Li, "A real-time hand posture recognition system using deep neural networks," *ACM Trans. Intell. Syst. Technol.*, vol. 6, pp. 21:1–21:23, Mar. 2015. https://doi.org/10.1145/2735952.

[25] C. Yang, Y. Jang, J. Beh, D. Han, and H. Ko, "Gesture recognition using depth-based hand tracking for contactless controller application," in *2012 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 297–298, Jan 2012. https://doi.org/10.1109/ICCE.2012.6161876.

[26] J. Li, S. Zhang, and T. Huang, "Multi-scale 3d convolution network for video based person re-identification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 8618–8625, 07 2019. https://doi.org/10.48550/arXiv.1811.07468.

[27] D. Cheng, S. Xiang, C. Shang, Y. Zhang, F. Yang, and L. Zhang, "Spatio-temporal attention-based neural network for credit card fraud detection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 362–369, 04 2020. https://doi.org/10.1109/ICIP.2019.8803152.

[28] P. Pandey, A. P. Prathosh, M. Kohli, and J. Pritchard, "Guided weak supervision for action recognition with scarce data to assess skills of children with autism," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 463–470, 04 2020. https://doi.org/10.48550/arXiv.1911.04140.

[29] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, (Washington, DC, USA), pp. 4489–4497, IEEE Computer Society, 2015. https://doi.org/10.1109/ICCV.2015.510.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[31] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[32] S. Dieleman, J. Schlüter, C. Raffel, E. Olson, S. K. Sønderby, D. Nouri, *et al.*, "Lasagne: First release.," Aug. 2015. https://doi.org/10.5281/zenodo.27878.

[33] J. Materzynska, G. Berger, I. Bax, and R. Memisevic, "The jester dataset: A large-scale video dataset of human gestures," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 2874–2882, 2019. https://doi.org/10.1109/ICCVW.2019.00349.

[34] E. Nasr-Esfahani, N. Karimi, S. M. R. Soroushmehr, M. H. Jafari, M. A. Khorsandi, S. Samavi, and K. Najarian, "Hand gesture recognition for contactless device control in operating rooms," *CoRR*, vol. abs/1611.04138, 2016. https://doi.org/10.48550/arXiv.1611.04138.

[35] A.-r. Lee, Y. Cho, S. Jin, and N. Kim, "Enhancement of surgical hand gesture recognition using a capsule network for a contactless interface in the operating room," *Computer Methods and Programs in Biomedicine*, vol. 190, p. 105385, 2020. https://doi.org/10.1016/j.cmpb.2020.105385.

[36] Z. Lu, S. Qin, X. Li, L. Li, and D. Zhang, "One-shot learning hand gesture recognition based on modified 3d convolutional neural networks," *Machine Vision and Applications*, vol. 30, 08 2019. https://doi.org/10.1007/s00138-019-01043-7.

Improving Modeling of Stochastic Processes by Smart Denoising

Jakob Jelenčič and Dunja Mladenić
E-mail: jakob.jelencic@ijs.si, dunja.mladenic@ijs.si
Jozef Stefan Institute, Ljubljana, Slovenia
Jozef Stefan International Postgraduate School, Ljubljana, Slovenia

Keywords: de-noising, deep learning, stochastic process

Received: December 16, 2021

This paper proposes a novel method for modeling stochastic processes, which are known to be notoriously hard to predict accurately. State of the art methods quickly overfit and create big differences between train and test datasets. We present a method based on smart noise addition to the data obtained from unknown stochastic process, which is capable of reducing data overfitting. The proposed method works as an addition to the current state of the art methods in both supervised and unsupervised setting. We evaluate the method on equities and cryptocurrency datasets, specifically chosen for their chaotic and unpredictable nature. We show that with our method we significantly reduce overfitting and increase performance, compared to several commonly used machine learning algorithms: Random forest, General linear model and LSTM deep learning model.

Povzetek: V tem članku je predstavljena metoda, s katero se lahko omeji preprileganje nevronskih mrež na časovnih vrstah.

1 Introduction

Time series prediction has always been an interesting challenge. Deep learning structures that are designed for time series are prone to overfitting. Especially if the underlying time series is stochastic by nature. Every young researcher's first attempt when dealing with time series, was trying to learn a time series model that will predict future prices; whether in equities, commodities, forex or cryptocurrencies. Unfortunately it is not that simple. One can easily build a near perfect model on the train dataset just to find it is completely useless on the test dataset.

Overfitting is a difficult problem, especially in deep learning models with many variables, where optimizers based on gradient descent are prone to it. There are many ways to limit it, like regularizes in loss functions that punish high values in variables, or random masking of training data and adding noise it. However most efficient way to prevent overfit is to obtain more data. Unfortunately sometimes this is not easily done or even possible.

We have proposed a novel method that is capable of effectively combatting the overfitting [5], especially this proves to be a difficult task when one is dealing with a problem directly applicable in practical situations. In this work we expand on the denoising part of our method and provide additional parameter analysis. The main idea is to add noise from the same distribution as the training data, which is applicable for both supervised and unsupervised problems. The longer the training goes, the lower is the amplitude of noise and the less focus is on the denoising process.

We have evaluated the proposed method on an equities dataset and a cryptocurrency dataset, in both cases achieving extraordinary results on the test dataset. We have also shown the importance of noise distribution and how the denoising fails if the distributions of the data and noise do not align.

The rest of the paper is organised as follows. Section 2 describes the data we were using. In section 3 we introduce the proposed method. In section 4 we present empirical results. In section 5 we conclude by pointing out the main results and defining guidance for the future work.

2 Data

The proposed method works well for stochastic processes. Equities are supposed to follow some form of stochastic process [10], either the Black-Scholes one or some more complex process with unknown formulation. In order to evaluate our method, we have collected daily data of more than 5000 equities listed on NASDAQ from 2007 on. The data is freely available on the Yahoo Finance website [2]. We transformed the data using technical analysis [11] and for test set took every instance that happened after 2019. We calculated moving average using 10 days closing price then tried to predict the direction of the change of this trendline. As a second experiment we tried to predict change of the equity in the following day.

The smoothed equity data turned out to be a little bit timid, not chaotic enough to demonstrate the full ability of the proposed method in the unsupervised part of the experiment. This is why we also collected minute data of cryptocurrencies Ethereum and Bitcoin and used the method

on them as well. Data is available on the crypto exchange Kraken [1]. We used the same transformation as for the equities, but with a bit quicker trend. For the test set we took every instance that has time stamp after December 2020.

The reader should note that the end goal is not to accurately predict future equity price, since that is next to impossible. As soon there is a pattern, someone will profit from it and then the pattern will change. By predicting the future trend line, one can obtain a significant confidence interval and estimates of where the price could be, and then design for example a derivative strategy that searches for favourable risk versus rewards trades.

3 Proposed method

We propose the method designed for prediction of stochastic processes. The method achieves significant results improving the metrics and loss functions on unseen data, where standard deep learning is prone to over-fit. The main advantage is reducing the gap between training data and testing data, sometimes to a degree where one sacrifices a little bit on the train side to actually have the model outperforming it on test data. This is very important in time series, where a prediction model is usually just one part of a bigger strategy and where the train over-fit is the biggest issue. For example, designing a trading strategy on over-fitted predictions, that kind of mistake can lead to huge capital losses.

The proposed method can be broken down into 2 important parts: normalization and noise addition. Each part can be easily integrated into an already existing pipeline of both supervised or unsupervised settings.

3.1 Empirical normalization

Normalization plays an important role in deep learning models. It was shown that normalization significantly speeds up the gradient descent, almost independently of where normalization takes place. It can be weight normalization [12] during the actual optimization, or it can be the batch normalization [9], or just normalization of the whole input data [8].

In the proposed method it is important that the 3 dimensional input data comes from the same distribution as the generated noise. Since it is fairly straightforward to sample data from a 3 dimensional normal distribution, we normalize input data using an empirical cumulative distribution function [13] and empirical copula [4] [6]. We align all central moments of the unknown distribution to the ones from centered and standardised normal distribution. The normalization takes place before the data is reshaped to 3 dimensional tensor.

3.2 Noise addition

Introduction of the noise is not new in unsupervised learning and it was shown that it has a positive effect [15]. Adding noise to input data and then forcing the model to learn how to ignore it has a lot of success in generative adversarial networks [3], where convergence can be very tricky to achieve. We transformed that idea and embedded it into supervised learning procedure. The noise addition is described in Algorithm 1.

In Algorithm 1 we will use the following abbreviations.

- $X = [bs, ts, np]$ stands for the input tensor with 3 dimensions; batch size, time steps and number of features used for predictions.
- α, β are parameters that control how fast noise will decrease during the training procedure. They should be between 0 and 1, where lower value correspond to a faster decrease in the amplitude of the added noise.
- mvn stands for function sampling from a two dimensional correlated Gaussian distribution, where Σ is the covariance. $matmul$ stands for matrix multiplication.

Algorithm 1 Noise definition

Inputs: X, α, β, $epoch$
$Y = [ts, ts, np]$ ▷ Array for holding Cholesky decompositions of time correlation matrices.
for $t \in \{1, \ldots, np\}$ **do**
    $\Sigma_t = cov(X[, , t])$
    $Y[, , t] = chol(\Sigma_t)$ ▷ In practice the closest positive definite matrix of Σ_t is computed before the Cholesky decomposition.
end for
$Z = [bs, ts, np]$ ▷ Array for holding noise samples.
for $i \in \{1, \ldots, ts\}$ **do**
    $\Sigma_i = cov(X[, i,])$
    $Z[, i,] = mvn(bs, \Sigma_i)$
end for
for $j \in \{1, \ldots, np\}$ **do**
    $Z[, , j] = matmul(Z[, , j], Y[, , j])$ ▷ Correcting initially independent noise samples with respect to time.
end for
for $w \in \{1, \ldots, ts\}$ **do**
    $Z[, w,] = Z[, w,] * ((\beta^{ts-w} \cdot \alpha^{epoch}) \cdot sd)$ ▷ Decrease the noise during the training procedure.
end for
$R = X + Z$
Return R.

The most common issue with deep learning optimization is falling into a local optimum and being unable to move past it [14]. We expect that the addition of noise will force the model to learn how to ignore the noise that we added and the noise that is already in the data by nature of the stochastic process [16] and with that escaped some of the local optimum and converged deeper. It is important to tune the noise amplitude by fine tuning the α parameter. We optimized the model using the Adam optimizer [7].

4 Results

We have divided the results section into 3 parts: unsupervised, supervised and parameter analysis. In the first we demonstrate why the noise distribution is important. For the unsupervised part, due to hardware constraints, we have only used the cryptocurrency dataset since we deemed it more demanding than the equity one. In the second, we demonstrate how the our method increases test metric on both datasets and provide some additional parameter analysis. In the third we focus on how the method parameters affect method's performance.

4.1 Unsupervised learning results

In order to test the efficiency of distributed noise versus just random noise, we created 3 models. The baseline model was a deep learning model with 3 stacked LSTM layers, encoded layer, then again 3 stacked LSTM for decoded output. We have used Adam as optimizer. As loss function we used mean-squared error. We have stopped the learning after there was no improvement for 25 epochs on the validation set. The validation set was randomly taken out of the train set. Parameters α and β were both set to 0.99 and sd was initially set to 1.25. The noise decreases with learning procedure. Interestingly keeping noise constant did not achieve any results.

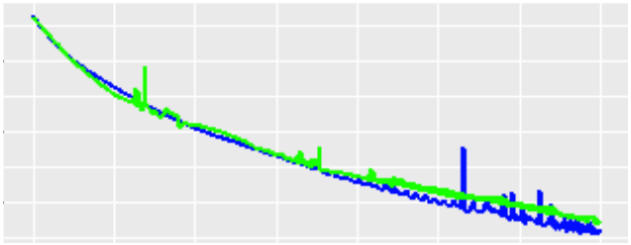


Figure 1: Test loss of autoencoder model with random noise (green) versus no noise (blue).

Initially we have tested baseline model versus de-noising model but with uncorrelated noise. In the Figure 1 is plotted the de-noising test loss function in green colour and the baseline test loss function in blue. Training was stopped relatively early compared to Figure 2 and it is also obvious that de-noising test loss is even worse than that of the classic autoencoder.

In the second example we switched from uncorrelated noise to the noise with same distribution as input data. As is apparent on Figure 2, where again we have de-noising test loss plotted with green and classic test loss with blue, the de-noising autoencoder achieved lower test loss than the classic one.

What we expected is that then the train and validation losses will be worse than with the classic autoencoder. Surprisingly, that was not the case. With the de-noising autoencoder using noise with the same distribution as the input data, both train and validation losses were better than

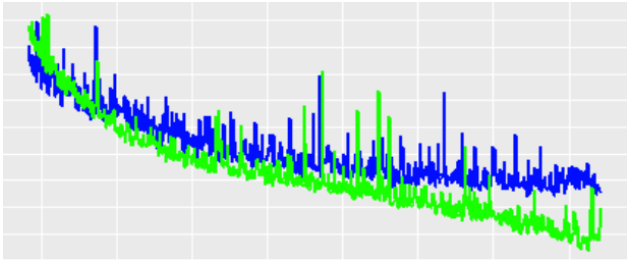


Figure 2: Test loss of autoencoder model with correlated noise (green) versus no noise (blue).

with classic one. This result is definitely worth further investigation and experimentation.

4.2 Supervised learning results

In the previous section we have shown that the distribution of the noise matters in the unsupervised setting. In this section we will show that the addition of the noise significantly improves metrics on unseen data. Each model has been ran 10 times, and in the tables below we present the average results. Since we operate under the stochastic assumption, this was done to eliminate any doubt that the presented results are good due to luck. We believe 10 runs is enough and more was simply not possibly due to hardware and time constraints. Similarly as before, α and β were both set to 0.99 and sd was initially set to 1.25 when comparing proposed method to existing state of the art structures.

Since we now operate in a supervised environment, we can compare our models to the majority class. But to really demonstrate the effectiveness of the method, we chose to compare the following models:

- Majority class, which serves as a sanity check.
- Random Forest with 500 trees.
- Generalized linear model.
- Deep learning model with 3 stacked LSTM layers.
- Deep learning model with 3 stacked LSTM layers and correlated noise addition.

The two deep learning models are identical, both are optimized with Adam and categorical cross entropy was used as a loss function. Initially we have only tested the models on equities data, but it turned out that the equities were not chaotic enough. By that we mean that especially with deep learning models the difference between train and test loss was not so big that it would be problematic. From previous work experience we know that overfit is a big issue in cryptocurrency dataset, so then we decided to test that dataset in a supervised setting as well.

In Table 1 we show the results from the equity dataset. Our method managed to improve test accuracy (from 0.672 to 0.677) without decreasing train accuracy (0.679). Maintaining test accuracy and keeping it comparable to test one

is important if one needs to build additional strategy upon predictions.

Table 1: Supervised results on equity dataset.

Method	Train Acc.	Test Acc.
Majority	0.513	0.537
Random Forest	0.651	0.653
GLM	0.664	0.655
LSTM	0.679	0.672
noise LSTM	0.679	**0.677**

In Table 2 we show results from the cryptocurrency dataset. Similar as on the equity dataset, our method behaves as intended on the cryptocurrency dataset as well. We can see reduced overfitting that is apparent in the normal LSTM model. With those results we can conclude that the proof of concept works, but for additional claims we will need more testing and deeper parameter analysis.

Table 2: Supervised results on cryptocurrency dataset.

Method	Train Acc.	Test Acc.
Majority	0.512	0.556
Random Forest	0.692	0.690
GLM	0.682	0.695
LSTM	0.749	0.693
noise LSTM	0.702	**0.706**

4.3 Parameter analysis

So far we have demonstrated that the noise addition increases test metric and reduces overfitting. But we have demonstrated that with fixed $\alpha = 0.99$ and with fixed $sd = 1.25$. For now we will leave β fixed and focus on analysis of noise amplitude and its decrease over the learning process. We have chosen a combination of α, sd and noise correlation to show how the method behaves with respect to those parameters. For this experiment we have used equity dataset, where the target variable was change of the corresponding equity in the next day. So we have created extremely difficult problems, very likely to overfit in order to really test the method.

We did this only on one deep learning structure, where we evaluate each parameter setting 25 times, and display the average result in the table 3. To further demonstrate the necessity of the noise structure, we have evaluated example where the added noise were random and constant trough out the learning process.

The results in Table 3 confirm the assumptions we draw earlier. It is clear that some combination of noise and its decrease reduces overfitting compared to model without noise or the model without the amplitude decrease.

Table 3: Parameter analysis on raw equity dataset.

SD	Alpha	Train Acc.	Test Acc.
0.5	0.9	0.576	0.527
0.75	0.9	0.577	0.526
1.25	0.9	0.578	0.525
1	0.99	0.577	0.524
1	0	0.577	0.524
1.25	0	0.578	0.524
1.25	0.99	0.578	0.523
0	0	0.579	0.521

5 Conclusions and future work

In this work we have introduced and demonstrated how the addition of noise reduces overfitting on time series data. In the unsupervised case we have shown that the distribution of the noise matters and the input data must align to achieve maximum effect from the noise addition.

In the future work we have to estimate the effect of the newly introduced parameters on method's convergence. At the same time we need to explore how the method behaves when embedded into larger models, transformers for example. We also need to evaluate the method in datasets that are by nature stochastic but do not come from the financial domain. Finally, we need to evaluate our method on a dataset that is not stochastic.

6 Acknowledgments

This work was supported by the Slovenian Research Agency. We also wish to thank prof. dr. Ljupčo Todorovski for his help, especially with unsupervised results.

References

[1] *Kraken exchange*. `https://www.kraken.com/`.

[2] *Yahoo Finance*. `https://finance.yahoo.com/`.

[3] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.

[4] P. Jaworski, F. Durante, W. K. Hardle, and T. Rychlik. *Copula theory and its applications*, volume 198. Springer, 2010.

[5] J. Jelencic and D. Mladenic. Modeling stochastic processes by simultaneous optimization of latent representation and target variable. 2020.

[6] H. Joe. *Dependence Modeling with Copulas*. CRC Press, 2014.

[7] D. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2014. https://arxiv.org/abs/1412.6980.

[8] K. Y. Levy. The power of normalization: Faster evasion of saddle points. *arXiv preprint arXiv:1611.04831*, 2016.

[9] M. Liu, W. Wu, Z. Gu, Z. Yu, F. Qi, and Y. Li. Deep learning based on batch normalization for p300 signal detection. *Neurocomputing*, 275:288–297, 2018.

[10] R. C. Merton. Option pricing when underlying stock returns are discontinuous. *Journal of financial economics*, 3(1-2):125–144, 1976.

[11] J. J. Murphy. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance Series. New York Institute of Finance, 1999.

[12] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *Advances in neural information processing systems*, 29:901–909, 2016.

[13] B. W. Turnbull. The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 38(3):290–295, 1976.

[14] R. Vidal, J. Bruna, R. Giryes, and S. Soatto. Mathematics of deep learning. *arXiv preprint arXiv:1712.04741*, 2017.

[15] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.

[16] N. Wax. *Selected papers on noise and stochastic processes*. Courier Dover Publications, 1954.

Automatic Fabric Inspection using GLCM-based Jensen-Shannon Divergence

V. Asha
E-mail: v_asha@live.com; asha.gurudath@gmail.com
Department of Master of Computer Applications, New Horizon College of Engineering, Bangalore, India
ORCID: 0000-0003-4803-099X

Keywords: cluster, co-occurrence matrix, defect, periodicity, Jensen-Shannon, divergence

Received: November 25, 2019

Jensen-Shannon divergence is one of the powerful information-theoretic measures that can capture mutual information between two probability distributions. In this paper, a machine vision algorithm is proposed for automatic inspection on dot patterned fabric using Jensen-Shannon divergence based on gray level co-occurrence matrix (GLCM). Input defective images are split into several periodic blocks based on their periodicities extracted using superposition of distance matching functions and the gray levels are quantized from 0-255 to 0-63 to keep the GLCM compact and to reduce the computation time. Symmetric Jensen-Shannon divergence metrics are calculated from the GLCMs of each periodic block with respect to itself and all other periodic blocks to get a dissimilarity matrix that satisfies true metrics conditions. This dissimilarity matrix is subjected to hierarchical clustering to automatically identify defective and defect-free blocks. Results from experiments on real fabric images with defects such as broken end, hole, thin bar, thick bar, netting multiple and knot show the effectiveness of the proposed method for fabric inspection.

Povzetek: Razvita je metoda, ki na osnovi Jensen-Shannovove divergence omogoča iskanje pokvarjenih vzorcev na ponavljajoči se shemi.

1 Introduction

Periodically patterned texture images are often found in day-to-day life in various applications such as ceramic tiles, wallpapers, and textile fabrics. Inspection of these products plays a major role with regard to quality control of the products in industries. Conventional human-vision based inspection are being followed in many of the industries. Lack of repeatability and reproducibility of inspection results due to fatigue and subjective nature of humans and imperfect inspection due to complicated design in the texture patterns are common in the conventional human-vision based inspection systems. Automated machine-vision based inspection can promote the production rate of the products by decreasing the inspection time compared to the traditional human-vision based inspection especially in industries such as textile industries. Periodic textures which we come across in our life include several natural patterns and made-made / artificial patterns. Plenty of patterns are used in textile fabric design. These patterns, in turn, include either random structures or periodic structures, making the fabric appealing to our eyes. Inspection on fabric images having periodic texture patterns is more complicated than that on plain and twill fabric images due to complexity in the design, existence of numerous categories of patterns, and similarity between the defect and background [1]. For the inspection on periodically patterned fabrics, there are methods in literature that depend on training stage with numerous defect-free samples for obtaining decision-boundaries or thresholds prior to detection of defects [1]-[11]. However, unsupervised method of identifying defects in a patterned texture image is quite challenging [12]-[17]. In this paper, a method of inspection on periodically patterned fabrics is proposed without any training stage with the help of texture-periodicity and Jensen-Shannon divergence metric based on gray level co-occurrence matrix. The main contributions of this research can be summarized as follows:

- There is no training stage with defect-free samples for decision boundaries or thresholds unlike other methods.
- Due to the absence of training stage with defect-free samples, the proposed method does not need huge memory space for storage of defect-free samples.
- Identification of defective and defect-free periodic units of the fabrics is automatically carried out based on cluster analysis without human intervention.

The program for the proposed algorithm is written in Matlab-7.0. The organization of this paper is as follows: Section-2 presents a brief review on Jensen-Shannon divergence, gray level co-occurrence matrix and the proposed algorithm for fabric inspection along with illustration and results from experiments on various real fabric images with defects. Section-3 has the conclusions.

2 Proposed algorithm for fabric inspection

Jensen-Shannon divergence is one of the effective measures for capturing mutual information between two probability distributions [12]. In this paper, it is intended to employ Jensen-Shannon divergence metric to discriminate between defective and defect-free periodic blocks as feature extraction (e.g. [18],[19]). Following the fact that human vision perception makes use of second-order statistics for texture discrimination [20], histogram constructed from second-order statistics is used to discern between defective and defect-free periodic blocks. Since the time of proposal of 14 features derived from histogram of second order statistics or gray-level co-occurrence matrices (GLCM) by Haralick *et al*. **[21]**, several authors have employed GLCM features for various texture analysis applications (e.g.[22]-[26]). Instead of extracting GLCM features, information from every pixel-pair of a periodic block with that of other periodic block is used to compute a distance metric using Jensen-Shannon divergence. The uniqueness of this Jensen-Shannon divergence metric based on GLCM is that the distance measure is considered for every pair of gray levels between two periodic blocks unlike the conventional GLCM features that are computed for the entire GLCM.

2.1 Brief review on Jensen-Shannon divergence

Jensen-Shannon divergence is a symmetrized, smoothed version of Kullback-Leibler divergence which is the most important divergence measure of information theory [18]. Given two classes $p(\omega_1)$ and $p(\omega_2)$ with a common feature vector *x*, the Kullback-Leibler divergence or the relative entropy in terms of ratio of the probability distributions of these classes conveys useful information concerning the capability of discriminating the two classes. This class-separability measure over class ω_1 is given as [27]

$$\lambda(\omega_1, \omega_2) = \int_{-\infty}^{\infty} p(\omega_1) log\left(\frac{p(\omega_1)}{p(\omega_2)}\right) dx \quad (1)$$

Similarly, the class-separability measure over class ω_2 is given as

$$\lambda(\omega_2, \omega_1) = \int_{-\infty}^{\infty} p(\omega_2) log\left(\frac{p(\omega_2)}{p(\omega_1)}\right) dx \quad (2)$$

Based on the probability distributions $p(\omega 1)$ and $p(\omega_2)$ and their average, a symmetric divergence called Jensen-Shannon divergence Λ is given as

$$\Lambda = \int_{-\infty}^{\infty} \left\{ p(\omega_1) \log\left(\frac{0.5 \times p(\omega_1)}{p(\omega_1)+p(\omega_2)}\right) + p(\omega_2) \log\left(\frac{0.5 \times p(\omega_2)}{p(\omega_1)+p(\omega_2)}\right) dx \right\} \quad (3)$$

Since Kullback-Leibler divergence λ (ω_1,ω_2) or λ (ω_1,ω_2) can be interpreted as the inefficiency of assuming that the true distribution is $p(\omega_2)$ or $p(\omega_1)$ when it is really $p(\omega_1)$ or $p(\omega_2)$, Jensen-Shannon divergence can be considered as a minimum inefficiency distance.

2.2 Brief review on Gray Level Co-occurrence Matrix

Gray level co-occurrence matrices (GLCM) have been employed in extracting texture features for various applications for a long time. It is a measure of gray-level dependencies in a local neighborhood for a given pixel displacement and orientation. In other words, a GLCM is a matrix showing the number of times a pixel with gray level *I* occurs at position vector from a pixel with gray level *j*. Mathematically, the GLCM for an image$I(x, y)$ of size (M, N) at a given offset $(\Delta x, \Delta y)$ is given as

$$C_{\Delta x,\Delta y} = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \begin{cases} 1, if\ I(x,y) = i\ and \\ I(x+\Delta x, y+\Delta y) = j; \\ 0, otherwise. \end{cases} \quad (4)$$

where, L is the total number of gray values in the image. For an-bit image, the value of L is 2n with dynamic range of gray values being [0, L-1]. The offset (Δx, Δy) characterizes the pixel displacement and the orientation for the co-occurrence matrix.

2.3 Description of the algorithm

There are three major assumptions in the proposed algorithm as follows:

- Test image contains at least two periodic units in horizontal direction and two in vertical direction whose row and column dimensions are known a priori.
- Number of defective periodic units is always less than the number of defect-free periodic units.
- Test images are from imaging system oriented perpendicular to the surface of the product to be inspected. This assumption is due to the fact that in an inspection system in industries, the imaging system is always oriented perpendicular to the plane of the surface of the products such as tiles, wallpapers, and textile fabrics.

An image under inspection may have fractional periods also. Hence, the concept of analyzing the periodic blocks extracted from all four corners of the test image is used for identifying the defects followed by the concept of defect fusion proposed in [28]. Four cropped images are obtained from the defective test image by cropping the input image from all four corners (top-left, bottom-left, top-right and bottom-right) to get complete number of periodic blocks. If g is an image of size M × N with row periodicity P_r (i.e., number of columns in a periodic unit) and column periodicity P_c (i.e., number of rows in a periodic unit), size of the cropped images is M_{crop} × N_{crop}, where M_{crop} and N_{crop} are measured from top-left, bottom-left, top-right and bottom-right corners and are given by the following equations:

$$M_{crop} = floor\left(\frac{M}{P_c}\right) \times P_c \quad (5)$$

$$N_{crop} = floor\left(\frac{N}{P_r}\right) \times P_r \quad (6)$$

Each cropped image is split into several periodic blocks of size Pc x Pr. The periodicities are estimated using Superposition of distance matching function proposed in [29]. The decision that one has to make now

is how many levels are needed to construct a GLCM for effective texture analysis. The number of gray levels is an important factor in the computation of GLCM. The more levels included in the computation, the more accurate the extracted textural information, with, of course, a subsequent increase in computation costs and too less gray levels will result in loss of information due to quantization. In general, the effect of false-contouring starts predominating in an image if the gray levels are quantized below the dynamic range of gray values 0-63 [30]. In the proposed method, each 8-bit test image having a dynamic range of gray values 0-255 is linearly quantized to 6-bit image having a dynamic range 0-63 and rotation-invariant GLCMs (sum of GLCMs over 8 directions $\theta \in \{0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/8\}$) calculated for a unit pixel displacement is utilized. If p(i, j) and q(i, j) represent the summed up GLCMs of two periodic blocks A and B respectively whose dynamic range of gray values is between 0 and L-1 for a unit pixel displacement, the Jensen-Shannon divergence can be computed as

$$\Lambda_{A,B} = \Lambda_{B,A} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\left\{p(i,j)\log\left(\frac{0.5\times p(i,j)}{p(i,j)+q(i,j)}\right)+q(i,j)\log\left(\frac{0.5\times q(i,j)}{p(i,j)+q(i,j)}\right)\right\} \quad (7)$$

The square root of this divergence is a true metric obeying the conditions of a true metric in 2D Euclidean space [18], viz.,$\sqrt{\Lambda_{i,j}} \geq 0$ for all i and j (non-negativity), $\sqrt{\Lambda_{i,j}} = 0$ for all $i = j$ (self-distance) and $\sqrt{\Lambda_{i,j}} = \sqrt{\Lambda_{j,i}}$ for all i and j (symmetry), and $\sqrt{\Lambda_{i,j}} \leq \sqrt{\Lambda_{i,k}} + \sqrt{\Lambda_{k,j}}$ for $i \neq j \neq k$ (triangular inequality). Following Eq. (7) based on Jensen-Shannon divergence for each periodic block with respect to itself and all other periodic blocks, a dissimilarity matrix D containing true distance metrics can be obtained as below:

$$D = \begin{pmatrix} \sqrt{\Lambda_{1,1}} & \sqrt{\Lambda_{1,2}} & \cdots & \sqrt{\Lambda_{1,n-1}} & \sqrt{\Lambda_{1,n}} \\ \sqrt{\Lambda_{2,1}} & \sqrt{\Lambda_{2,2}} & \cdots & \sqrt{\Lambda_{2,n-1}} & \sqrt{\Lambda_{2,n}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \sqrt{\Lambda_{n-1,1}} & \sqrt{\Lambda_{n-1,2}} & \cdots & \sqrt{\Lambda_{n-1,n-1}} & \sqrt{\Lambda_{n-1,n}} \\ \sqrt{\Lambda_{n,1}} & \sqrt{\Lambda_{n,2}} & \cdots & \sqrt{\Lambda_{n,n-1}} & \sqrt{\Lambda_{n,n}} \end{pmatrix} \quad (8)$$

It should be noted that if a cropped image has n number of periodic blocks, then the size of the dissimilarity matrix is n x n. Because the divergence measure of a periodic block with itself is zero and the divergence measure between ith periodic block and jth periodic block is same as the divergence measure between jth periodic block and ith periodic block, the dissimilarity matrix becomes a diagonally symmetric matrix with diagonal elements being zero as below:

$$D = \begin{pmatrix} 0 & & & & \\ \sqrt{\Lambda_{2,1}} & 0 & & & \\ \vdots & \vdots & \vdots & & \\ \sqrt{\Lambda_{n-1,1}} & \sqrt{\Lambda_{n-1,2}} & \cdots & 0 & \\ \sqrt{\Lambda_{n,1}} & \sqrt{\Lambda_{n,2}} & \cdots & \sqrt{\Lambda_{n,n-1}} & 0 \end{pmatrix} \quad (9)$$

It may be noted that because the matrix is similar about the diagonal, the upper diagonal elements are not filled for the sake of simplicity. This dissimilarity matrix is directly given as input to the Ward's cluster algorithm [28],[31] to automatically get defective and defect-free periodic blocks from each cropped image. Detection of defective periodic blocks from each cropped image does not give an overview of the total defects in the input defective image. Hence, in order to get the overview of the total defects in the input image, defect-fusion proposed in [28] is used which involves fusion of the boundaries of the defective periodic blocks identified from each cropped image, morphological filling and Canny edge extraction.

2.4 Illustration of the algorithm

In order to illustrate the proposed algorithm for fabric inspection, let us a defective box-patterned fabric image as shown in Fig. 1 (a). Following Eqs. (5) and (6), four cropped images containing complete number of periodic blocks are obtained as shown in Fig. 1 (b) - (e), from the test image with the help of periodicities known a priori.

Each cropped image is split into several blocks of size same as the size of the periodic unit and the gray values are quantized from 0-255 to 0-63. Jensen-Shannon divergence metrics are calculated for each periodic block with respect to itself and all other periodic blocks and a dissimilarity matrix is obtained from the GLCMs calculated over 8 directions for unit pixel displacement. The dissimilarity matrices thus obtained are shown in Fig. 2 in gray-scale form by scaling the matrix elements linearly in the range 0-255. Dark pixels indicate closeness among the periodic blocks and bright pixels indicate high dissimilarity. It may be noted from Fig. 2 that the diagonal elements in the dissimilarity matrix clearly indicate that the periodic blocks are of zero dissimilarity with themselves and that the dissimilarity matrix is symmetric.

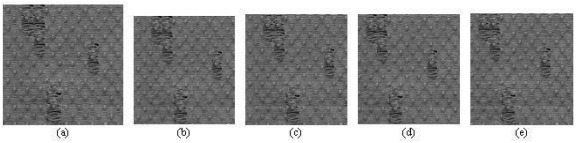


Figure 1: (a) Input defective image; (b) Cropped image obtained from top-left corner of the input image; (c) Cropped image obtained from bottom-left corner of the input image; (d) Cropped image obtained from top-right corner of the input image; (e) Cropped image obtained from bottom-right corner of the input image.

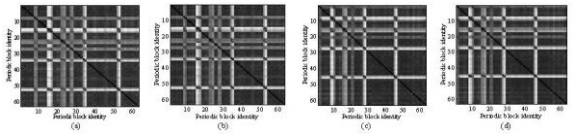


Figure 2: Dissimilarity matrix derived from the Jensen-Shannon divergence metrics for the cropped image obtained from (a) top-left (b) bottom-left (c) top-right and (d) bottom-right corners of the test image shown in gray-scale.

The dendrograms obtained from the dissimilarity matrices through Ward's hierarchical clustering are shown in Fig. 3 along with the defective periodic blocks identified by the clustering for all cropped images. Boundaries of the defective periodic blocks thus identified from each cropped image are highlighted using white pixels and shown in Fig. 4. Boundaries of the defective periodic blocks identified from each cropped image are shown superimposed on the original image in Fig. 5 (a) and separately on a plain background in Fig. 5 (b). Fig. 5 (c) shows the result of morphological filling and Fig. 5 (d) shows the edges of the total defects superimposed on the original defective image after Canny edge detection. Thus, an overview of the total defects on the original image itself can be obtained following the concept of fusion, morphological filling and edge detection. It may be noted that though the number of periodic blocks taken from a defective input image is same for all of its cropped images, the number of defective periodic blocks identified does not need to be same for all cropped images. This is because the contribution of defect in each periodic block may differ for different cropped images. Nevertheless, fusion of defects from all 4 cropped images helps in getting an overview of total defects in the input image. This is clearly evident from the example image used for illustration of the proposed algorithm.

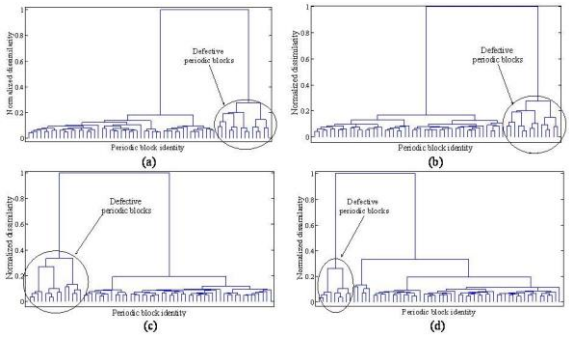


Figure 3: Dendrogram obtained from cluster analysis of the matrix containing Jensen-Shannon divergence metrics obtained from the test image by cropping it from (a) top-left (b) bottom-left (c) top-right and (d) bottom-right corners. Defective blocks identified from these cropped images are (8, 16, 27, 9, 17, 28, 24, 35, 53, 15, 34, 52, 22, and 23), (8, 16, 27, 24, 35, 53, 9, 17, 28, 15, 34, 52, 22, and 23), (28, 46, 17, 9, 8, 27, 45, 15, 16, 1, 20, 2, 10, and 21) and (28, 46, 17, 9, 8, 27, 45, 15, and 16).

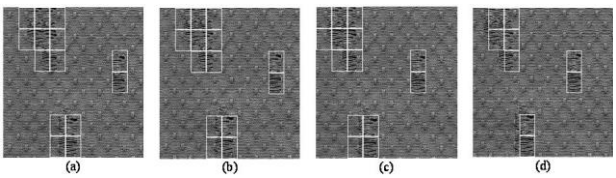


Figure 4: Defective periodic blocks identified from the cluster analysis of dissimilarity matrix derived from the Jensen-Shannon divergence metrics of the cropped images obtained from (a) top-left (b) bottom-left (c) top-right and (d) bottom-right corners of the test image with their boundaries highlighted using white pixels.

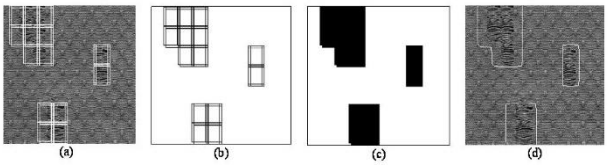


Figure 5: Defective periodic blocks identified from the cluster analysis of dissimilarity matrix derived from the Jensen-Shannon divergence metrics of the cropped images obtained from (a) top-left (b) bottom-left (c) top-right and (d) bottom-right corners of the test image with their boundaries highlighted using white pixels. Illustration of defect fusion: (a) Boundaries of the defective blocks identified from each cropped image shown super-imposed on the original image; (b) Boundaries of the defective blocks shown separately on plain background; (c) Result of morphological filling; (d) Edges identified through Canny edge operator shown superimposed on original defective image using white pixels.

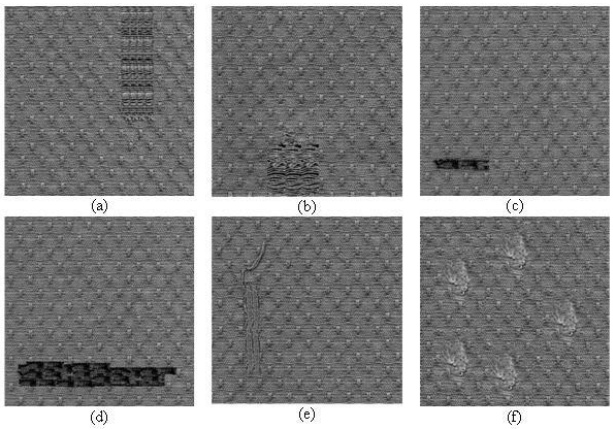


Figure 6: Sample fabric images with defect - (a) broken end, (b) hole, (c) thin bar, (d) thick bar, (e) netting multiple, and (f) knot.

2.5 Experiments on real fabric images with different types of defects

Periodically patterned fabric images with defects such as broken end, thin bar, thick bar, netting multiple and knot are tested to study the performance of the proposed algorithm of fabric inspection. Fig. 6 shows the defective fabric images and Fig. 7 shows the final result of defect identification after fusion, morphological filling and edge detection.

2.6 Performance evaluation and comparison with other methods

Precision, recall and accuracy are the widely used performance parameters in several retrieval applications. In order to access the performance of the proposed method, these performance parameters are evaluated in terms of true positive (TP), true negative (TN), false positive (FP), and false negative (FN), [31], [33], [34], [35]. True positive is defined as the number of defective periodic blocks identified as defective. True negative is defined as the number of defect-free periodic blocks

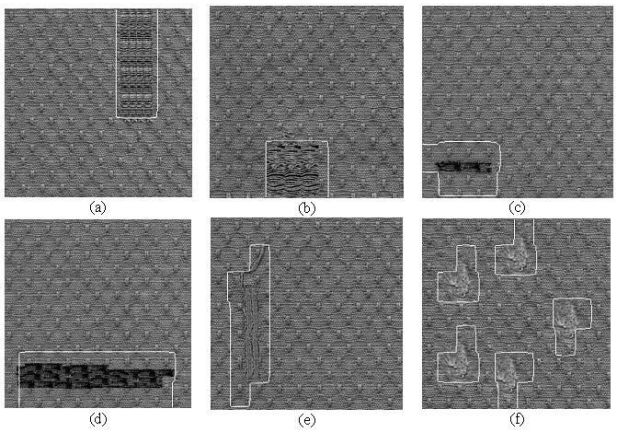


Figure 7: (a) Boundaries of the defective blocks identified from each cropped image shown superimposed on the original image; (b) Boundaries of the defective blocks shown separately on plain background; (c) Result of morphological filling; (d) Edges identified through Canny edge operator shown superimposed on original defective image using white pixels. Final result of defect identification after defect fusion, morphological filling and edge detection for (a) defective fabric image with broken end; (b) defective fabric image with hole; (c) defective fabric image with thin bar; (d) defective fabric image with thick bar; (e) defective fabric image with netting multiple; (f) defective fabric image with knot.

Table 1: Summary of performance parameters of the proposed algorithm.

Image defect	**Precision (%)**	**Recall (%)**	**Accuracy (%)**
Hole (in illustration)	100	82.1	95.6
Broken end	100	80.0	96.8
Hole	100	85.7	98.4
Thin bar	100	87.5	99.2
Thick bar	100	91.7	97.6
Netting multiple	100	77.5	97.2
Knot	100	80.4	95.6
Average	**100**	**83.6**	**97.2**

identified as defect-free. False positive is defined as the number of defect-free periodic blocks identified as defective. False negative is defined as the number of defective periodic blocks identified as defect-free. Precision is defined as the number of periodic blocks correctly labelled as belonging to the positive class divided by the total number of periodic blocks labelled as belonging to the positive class and is calculated as TP/(TP+FP). Recall is defined as the number of true positives divided by the sum of true positives and false negatives that are periodic blocks not labelled as belonging to the positive class but should have been and is calculated as TP/(TP+FN). Accuracy is the measure of success rate that takes into account the detection rates of defective and defect-free periodic blocks and is calculated as (TP+TN)/(TP+TN+FP+FN). These performance parameters are averaged for all cropped images and given in Table 1. The average precision, recall and accuracy for all the defective images (based on a total of 1764 number of periodic blocks) are 100%, 83.6%, and 97.2% respectively. The accuracy of a few other methods available in literature for defect detection on patterned textures varies from 88% to 99% [36]. Relatively less recall rates in the proposed method indicate that there are few false negatives identified by the proposed method. However, as the proposed method yields high precision and accuracy, the proposed method can contribute to automatic inspection in industries such as fabric industries. In general, the computational time complexity of a full GLCM is $O(M^2N^2)$. As the proposed method is based on reduced GLCM from 0-255 to 0-63, the time complexity is reduced by one-third.

3 Conclusion

In this paper, texture-periodicity and Jensen-Shannon divergence metrics have effectively been used for the development of the automated inspection on periodically patterned fabrics. Absence of training stage with defect-free test samples for obtaining decision-boundaries or thresholds, unsupervised method of identifying defects using cluster analysis, and high success rates are the novelties of the proposed method. Thus, the proposed method can contribute to the development of computerized inspection and quality control in industries such as fabric industries.

Acknowledgement

The author would like to thank Dr. Henry Y. T. Ngan, Research Assistant Professor, Department of Mathematics, Hong Kong Baptist University, Kowloon, for providing the database of patterned fabrics.

References

[1] H.Y.T Ngan, Pang and G.H.K. Regularity Analysis for Patterned Texture Inspection. IEEE Trans. Autom. Sci. Eng., Vol. 6 (1) (2009), pp. 131-144. (DOI: 10.1109/TASE.2008.917140)

[2] H.Y.T Ngan, Pang, G.H.K. and N.H.C Yung. Performance Evaluation for Motif-Based Patterned Texture Defect Detection. IEEE Trans. Autom. Sci. Eng. Vol. 7 (1) (2010) pp. 58-72. (DOI:10.1109/TASE.2008.2005418)

[3] H.Y.T Ngan, Pang and G.H.K. Novel method for patterned fabric inspection using Bollinger bands. Opt. Eng. Vol. 45 (8) (2006), pp. 087202-1-15. (DOI:10.1117/1.2345189)

[4] F Tajeripour, E Kabir and A Sheikhi. Fabric Defect Detection Using Modified Local Binary Patterns. In: Proc. of the Int. Conf. on Comput. Intel. And Multimed. Appl., (2007) pp. 261-267. (https://doi.org/10.1155/2008/783898)

[5] H.Y.T Ngan, Pang, G.H.K and N.H.C Yung. Motif-based defect detection for patterned fabric. Pattern

Recognit. Vol. 41, (2008), pp.1878-1894. (https://doi.org/10.1016/j.patcog.2007.11.014)

[6] H.Y.T Ngan, Pang, G.H.K. and N.H.C Yung. Ellipsoidal decision regions for motif-based patterned fabric defect detection. Pattern Recognit. Vol. 43, (2010) pp.2132-2144. (DOI:10.1016/j.patcog.2009.12.001)

[7] M.K Ng, H.Y.T Ngan, X.Yuan and W. Zhang. Patterned Fabric Inspection and Visualization by the Method of Image Decomposition. IEEE Trans. Autom. Sci. Eng. Vol. 11 (3) (2014), pp. 943-947. (DOI: 10.1109/TASE.2014.2314240)

[8] J Liang, C Chen, L. Jiuzhen, L and H Zhenjie. Fabric defect inspection based on lattice segmentation and Gabor filtering. Neurocomput. Vol. 238 (5), (2017), pp. 84-102. (https://doi.org/10.1016/j.neucom.2017.01.039)

[9] S Mei, Y. Wang, and G. Wen., Automatic Fabric Defect Detection with a Multi-Scale Convolutional Denoising Auto-encoder Network Model. Sensors. Vol. 18 (4), (2018), 1064-1074. (DOI:10.3390/s18041064)

[10] C. Li, G. Gao, Z. Liu, D. Huang and J Xi. Defect Detection for Patterned Fabric Images Based on GHOG and Low-Rank Decomposition. IEEE Access, Vol 7, (2019), pp. 83962-83973. DOI: 10.1109/ACCESS.2019.2925196

[11] R.A.L Morales, R.E.S Yanez and R.B. Serrato. Defect detection on patterned fabrics using texture periodicity and the coordinated clusters representation, Textile Research J., Vol 87 (15), (2016), pp. 1869-1882. https://doi.org/10.1177/0040517516660885

[12] V. Asha, N.U. Bhajantri and P. Nagabhushan. Automatic Detection of Defects on Periodically Patterned Textures, Journal of Intelligent Systems, vol. 20 (3), (2011), pp. 279-303. (https://www.degruyter.com/document/doi/10.1515/jisys.2011.015/html)

[13] V. Asha, N.U. Bhajantri and P. Nagabhushan. GLCM-based chi-square histogram distance for automatic detection of defects on patterned textures. International Journal of Computational Vision and Robotics (IJCVR), vol. 2 (4), (2011), pp. 302-313. (DOI: 10.1504/IJCVR.2011.045267)

[14] V. Asha, N.U. Bhajantri and P. Nagabhushan. Automatic Detection of Texture-defects using Texture-periodicity and Jensen-Shannon Divergence. Journal of Information Processing Systems, vol. 8 (2), (2012) pp. 359-374, 2012. (https://doi.org/10.3745/JIPS.2012.8.2.359)

[15] V. Asha, N.U. Bhajantri and P. Nagabhushan. Similarity measures for automatic defect detection on patterned textures. International Journal of Information and Communication Technology, vol. 4 (2/3/4), (2012), pp. 118-131. (https://www.inderscienceonline.com/doi/abs/10.1504/IJICT.2012.048758)

[16] V. Asha. Singular Value Decomposition of Images of Structural Textures and their Reconstruction using Periodicity. International Journal of Tomography and Simulation (IJTS), vol 31, Issue 4, (2018), pp 86-97. (http://www.ceser.in/ceserp/index.php/ijts/article/view/5696)

[17] V. Asha. Texture Defect Detection using Human Vision Perception based Contrast. International Journal of Tomography and Simulation (IJTS), vol 32, Issue 3, (2019), pp 86-97. (http://www.ceser.in/ceserp/index.php/ijts/article/view/6017)

[18] H. Jiang, Defect Features Recognition in 3D Industrial CT Images, Informatica – An Int. J. of Comput. and Inform., Vo. 42, No. 3 (2018). (https://doi.org/10.31449/inf.v42i3.2454)

[19] D.M Endres and J.E Schindelin. A New Metric for Probability Distributions. IEEE Trans. on Info. Theory, Vol 49 (7), (2003), pp.1858-1860. (DOI: 10.1109/TIT.2003.813506)

[20] T Caelli, B. Julesz and E. Gilbert. On Perceptual Analyzers Underlying Visual Texture Discrimination, Part II. Biol. Cybern. Vol 29(4), (1978), pp. 201-214. (DOI: 10.1007/BF00337138)

[21] R.M. Haralick, K. Shanmugam and I. Dinstein. Textural features for image classification. IEEE Trans. On Syst., Man and Cybern. Vol 3(6), (1973), pp. 610-621. (DOI:10.1109/TSMC.1973.4309314)

[22] R.M Haralick. Statistical and structural approaches to texture. In: Proc. of the IEEE, Vol 67(5), (1979) pp. 786 – 804. (DOI: 10.1109/PROC.1979.11328)

[23] A.L Amet, A. Ertuzun and A. Ercil. Texture defect detection using subband domain co-occurrence matrices. Image and Vis. Comp. Vol 18, (2000), pp. 543-553. (DOI: 10.1109/IAI.1998.666886)

[24] Fernandos, J.C.A., Neves, J.A.B.C, Couto and C.A.C. Defect Detection and Localization in Textiles using Co-occurrence Matrices and Morphological Operators. In:Proc. of the 6th Int.

Conf. onMechatron. Mach. Vis. In Pract. (M2VIP'99), Ankara, Turkey. (1999)

[25] C.F.J Kuo and T.L Su. Gray Relational Analysis for Recognizing Fabric Defects. Textile Res. J. Vol 73(5), (2003), pp. 461-465 (https://doi.org/10.1177/004051750307300515)

[26] L.H Soh and C Tsatsoulis. Texture Analysis of SAR SeaIce Imagery Using Gray Level Co-occurrence Matrices. IEEE Trans. On Geosci. Remote Sens. Vol 37(2), (1999), pp. 780-795. (DOI: 10.1109/36.752194)

[27] S. Theodoridis and K. Koutroumbas, Pattern Recognition, Fourth Edition, Academic Press, CA (2009)

[28] V. Asha, N.U. Bhajantri and P. Nagabhushan. Automatic Detection of Texture Defectsusing Texture-Periodicity and Gabor Wavelets. In: Venugopal, K.R., and Patnaik, L.M. (Eds.): ICIP 2011, Communication and Computer Information Series (CCIS) 157, Springer-Verlag, Berlin Heidelberg, (2011), pp. 548-553.

[29] V. Asha, P. Nagabhushan and N.U. Bhajantri. Automatic extraction of texture-periodicity using superposition of distance matching functions and their forward differences. Pattern Recog. Lett. 33 (5), (2012), pp. 629-640. (https://doi.org/10.1016/j.patrec.2011.11.027)

[30] R.C. Gonzalez, R.E. Woods. Digital Image Processing. Pearson Prentice Hall, New Delhi. (2008)

[31] J. Pisanki and T. Pisanki, The use of collaboration distance in scheduling conference talks, Informatica – An Int. J. of Comput. and Inform., Vo. 43, No. 4 (2019). (https://doi.org/10.31449/inf.v43i4.2832)

[32] T. Fawcett. An introduction to ROC analysis, Pattern Recognit. Lett. 27, (2006), pp. 861-874. (https://doi.org/10.1016/j.patrec.2005.10.010)

[33] C.D Brown and H.T. Davis. Receiver operating characteristics curves and related decision measures: A tutorial. Chemom. And Intell. Lab. Syst. 80, (2006), pp. 24-38. (https://doi.org/10.1016/j.chemolab.2005.05.004)

[34] E. K. Ampomah, Z. Qin, G. Nyame, F. E. Botchey, Stock market decision support modeling with tree-based AdaBoost ensemble machine learning models, Informatica – An Int. J. of Comput. and Inform., Vol. 44, No. 4 (2020). (https://doi.org/10.31449/inf.v44i4.3159)

[35] R. Patel, S. Tanwani, C. Patida, Relation Extraction between Medical Entities using Deep Learning Approach, Informatica – An Int. J. of Comput. and Inform., Vol. 45 No. 3 (2021). (https://doi.org/10.31449/inf.v45i3.3056)

[36] T. Czimmermann, G. Ciuti, M. Milazzo, M. Chiurazzi, S. Roccella, C.M. Oddo and P. Dario., Visual-Based Defect Detection and Classification Approaches for Industrial Applications—A SURVEY, Sensors 20, Article No. 1459. (DOI:10.3390/s20051459)

A Complete Traceability Methodology Between UML Diagrams and Source Code Based on Enriched Use Case Textual Description

Wiem Khlif, Dhikra Kchaou and Nadia Bouassida
E-mail: Wiem.khlif@gmail.com, Dhikra.Kchaou@fsegs.rnu.tn, nadia.bouassida@isimsf.rnu.tn
Sfax University, Mir@cl Laboratory, Tunisia

Keywords: traceability, UML diagrams, use case, enriched textual description, control structure.

Received: September 9, 2020

Abstract: Traceability in software development proves its importance in many domains like change management, customer's requirements satisfaction, model slicing, etc. Existing traceability techniques trace either between requirement and design or between requirement and code. However, none of the existing approaches achieved reliable results when dealing with traceability between requirements, design models and source code. In this paper, we propose an improvement and an extension of our design traceability approach in order to tackle the traceability between design, requirement and code. The fine-tuning of our methodology stems from considering an expanded textual description. A pre-treatment step is added in order to divide the textual description of system functionalities into different parts, each of which represents a specific goal. In fact, the extension consists in extracting an expanded textual description from a natural language text in order to trace between related elements belonging to requirement, design and code while using an information retrieval technique. The proposed method is based on different scenarios (nominal, alternatives and errors), particularly on concepts related to control structures to establish the traceability between artefacts. Furthermore, we implemented our method in a tool allowing the evaluation of its performance. The evaluation is performed on real existing applications that consist in comparing results found by our approach with results found by experts. Our method achieves an average precision of 0.84 and a recall of 0.91 in traceability between requirement, design and code. Besides its promising performance outcomes, our automated method has the merit of generating a traceability report describing the correspondence between different artefacts.

Povzetek: Prispevek opisuje novo metodo za sledenje povezavam med UML diagrami in izvirno kodo.

1 Introduction

Traceability quality is defined as the degree to which existing artefacts of a software development project are traceable as mandated by the project's traceability stakeholders. The Unified Modelling Language (UML) is used for specifying, constructing, and documenting these artefacts. It is composed of a set of diagrams grouping structural and semantic dependencies between UML elements [1]. Based on the unified process, UML diagrams are produced iteratively and incrementally from use case diagram (UCD) to code. An iteration generates a baseline that comprises a partially complete version of the final system. Each one results in an increment, which is a release of the system that contains added or improved functionality compared with the previous release. Each iteration goes through five activities that specify what needs to be done: requirements, analysis, design, implementation and test. Requirements are modelled by (UCD) and their textual descriptions while the design is modelled through UML diagrams (class, sequence, etc.). These diagrams are strongly related either within one iteration or between iterations and consequently the lack of traceability between them makes any change difficult and expensive. Determining and keeping traceability between UML models is important for many reasons. For instance, in the context of change impact analysis, a change in one iteration often leads to changes in the following iterations. Certainly, the major challenge when developing a requirement change consists in creating traceability links between heterogeneous artefacts produced at different abstraction levels [2]. For example, adding data and actions in a use case (UC) description leads to add the corresponding methods and attributes in the class diagram and in the code. In fact, tracing change inter-UML diagrams into the source code is crucial to maintain the consistency and coherence. However, creating accurate and complete traceability is costly and remains a practical challenge [2]. In fact, we focus in this paper on determining traceability by considering structural and behavioural aspects. Furthermore, it is crucial to keep traceability between UML models since it allows checking the conformance between safety requirements and design decisions through model slicing. Thus, traceability definition is used to extract design slices that filter out irrelevant design details and keep information to inspect compliance between requirements and design [3, 4]. The recent literature on traceability shows two trends of approaches: those centred on traceability inter-UML models [4, 5, 6, 7, 8, 9], and those based on traceability from requirement and design to code [10, 11, 12, 13, 14, 15]. The first type of approaches

tackles the traceability within a set of models elements, particularly from requirements modelled by a UCD to design diagrams [6]. For instance, [16] deals with traceability between software architectural models and extra-functional results such as performance and security. Kchaou et al., [6] present traces between requirements modeled by a UCD and UML design diagrams. On the other hand, [4] illustrates the traceability between Use Case Maps and UML diagrams and [8] identifies the traceability between requirement and design models modeled with SysML. The second type of traceability approaches defines links between different models (requirements, design, test cases, etc.) and source code. These works differ in terms of the used techniques. These traceability approaches use exclusively either information retrieval techniques [17, 18, 19], a meta-model [20], Natural Language Processing (NLP) techniques [21] or machine learning techniques [22]. However, none of the existing approaches deals with traceability between requirements, design models and source code by covering all the concepts that can be determined in all levels (control structures, how activities are carried out, etc). That is, the so-far proposed approaches neglect additional semantic and/or structural information that can be extracted. The lack of this information may reduce the scope of possible analyses that can be made and possible traceability links that may be found. In addition, in the literature, traceability from requirement and design to the source code is based generally on the class diagram, which does not produce all necessary information such as control structure. Consequently, class diagrams allow engineers to understand its structure but it does not show the behavior of the software [5]. To understand its behaviour, dynamic models are needed, such as sequence, activity or state transitions diagrams [1]. Moreover, while the existing approaches use a semantic technique to compute similarities between different artifacts based on specific and common terms (e.g., actors, actions, etc), they do not cover all kinds of terms like behavioral elements (Parallel, alternative, loop, etc.), type of result, functional call, etc.

In this paper, we first show how the approach, initially presented in [6], that traces the elements of design diagrams, can be improved, fine-tuned and automated in order to discover correlated structural and semantic information and to trace between different UML diagrams, and between these diagrams and the source code. So, we have improved our previous work by defining an Enriched Textual Description (ETD) of a UC. The latter is extracted from a text written in a natural language and describing a software. In addition, the defined ETD allows tracing between the design and code. Unlike existing works (e.g. [4, 8, 21], we propose a method called TRADIAC Quality (TRAceability for UML DIAgrams and Code) that proceeds in three phases: "Pre-processing Natural language", "Traceability Inter-UML diagrams" and "Traceability from requirement and design to code". The "Pre-processing" phase receives as input the whole textual description of a software written in natural language. Then, the textual description is split into parts that achieve a specific goal expressing each one a functionality (use case). After that, each part is specified by using an enriched template that encapsulates the semantic information pertinent to the functional and behavioural aspects. In this work, we enrich the used textual description template [6] by basic control structures (BCS) (loop, if, switch, etc.) and a set of key words (e.g. PARALLEL expressing how activities are carried out) which take into account many important concepts in the design and code. This template is used for the requirements specification as a mean to document a UC. Compared to the presented template in [6], the enriched one provides more comprehensive traceability. For instance, in [6], the proposed approach does not determine which UC corresponds to which function in the code. In addition, it does not focus on details in alternative behavioural elements such as control structures. The second phase of our method "Traceability process inter-UML diagrams" is composed of traceability rules identification and similarity calculation. Traceability rules detect the relationships between requirements and design models. They distinguish between two traceability levels: structural and semantic. Structural traceability determines structural relationships between UML diagrams. Semantic traceability, which discriminates our method, is useful by considering that use case diagrams and their textual descriptions are based on a well-structured text. It searches the meaning of words contained in these descriptions and their synonyms to find similarities with terms used in the rest of UML diagrams. We note that the semantic traceability between the enriched textual description associated to a UC and other diagrams is based on an information retrieval technique. More specifically, it uses the Latent Semantic Indexing (LSI) similarity measure to estimate the similarity between corresponding elements. The choice of this measure is based on evaluations presented in [6] which showed that LSI is better suited to measure the semantic similarity. In its third phase, our method determines the traceability from requirement and design to code. It allows keeping traceability links from requirements into design and code by adding implementation details. To do so, it uses the traceability process from requirement to code which applies the defined traceability rules specific to details in the source code and calculates the similarity between the selected fragment in the textual description of a UC and code.

To show the advantages and limits of our method, we conduct an experimental evaluation thanks to TRADIAC (TRAceability for UML DIAgrams and Code) tool, which implements all of the method phases. For the herein presented evaluation, we applied a set of measurements (precision, recall, F-measure) to examine the conformity degree between corresponded elements generated by our method with the corresponded elements where traceability is evaluated by experts. This experimentation aims at proving that these models have similar quality values. For these quantitative evaluations, we used two case studies related to different domains. Our method shows an average precision of 84,1%, and an average recall of 91%. The results showed the efficiency of our method in terms of finding correct traceability reports.

The remainder of this paper is organized as follows: Section 2 overviews existing works that define traceability relationships between requirement and other diagrams, and from requirement and design to code. Section 3 presents our method in two subsections: the first subsection presents the pre-processing phase and the enriched textual description to document use cases based on basic control structures. The second subsection is composed of two parts: the first one identifies the traceability rules to facilitate first the transition from the requirement to design level by deriving other diagrams, particularly dynamic diagrams, and then derive code. The second part illustrates the LSI similarity which determines traceability between UML diagrams. To show the improvements gained by applying the traceability rules, we evaluate in section 4 our method and we consider threats to validity of the study and the results. Section 5 presents the tool support and illustrates the method through an example. Finally, Section 6 summarizes the presented work and outlines its extensions.

2 Related work

Several works cope with traceability based on different axes: covered artefacts (e.g. Horizontal vs. vertical) [17, 24, 25] representing the purpose of the traceability (e.g. finding inconsistency among artefacts, impact analysis, knowing the dependencies among artefacts, reuse) [25, 26, 27, 28], challenges and solutions [14, 29, 30, 31], etc.

As highlighted in the introduction, existing traceability approaches adopt either horizontal or vertical approaches. Horizontal traceability determines artifact dependencies at the same abstraction level (requirement, or design or code), while vertical traceability traces artifacts between different models at different abstraction levels. In this paper, we focus on vertical traceability which is classified into two categories: The first one focuses on traceability inter-UML models (requirement and design) and the second one determines traceability between requirements, UML models and code.

2.1 Traceability inter-UML models

Traceability inter-UML models approaches tackles the traceability within UML diagrams elements, particularly from requirements to design diagrams.

Adopting this type of approach, [4] considers the traceability relationships between Use Case Maps (UCMs) and UML diagrams. The proposed approach generates UML diagrams from UCMs notation to describe the system at high abstraction level. This work neglects several concepts that relate UML diagrams such as repetitive and conditional treatment.

In [14], the authors present an approach that supports the automatic maintenance of traceability relations between requirements, analysis and design models of a software systems expressed in UML. It followed two major phases: Recognition phase and maintenance. The first phase consists in capturing elementary changes to model elements and recognizing the compound development activity applied to the model element. The second phase, “Maintenance” consists on updating the traceability relations associated with the changed model element. A prototype called Trace Maintainer has been implemented to evaluate the approach.

In [32], an approach is presented to specify semantic relationships between system-level requirements, functional specifications, and architectures in terms of their subsystem specifications. This approach is based on logic predicate to present artifacts and their relations at different abstraction levels (Requirements, specification and architecture). The logical representation of each artifact is used by the authors to formalize relationships between these artifacts.

Adopting an abstract approach in defining traceability between software requirements and UML design, [20] proposes FUTOR (From Uml TO Requirement) guideline, which includes meta-model and process step. The meta-model expresses relationships between requirements and the UML model at the meta-level. For each meta-requirement, the author adds a “REQTYPE” attribute to decide which UML diagram shall be used for the traceability. Steps of the FUTOR guideline include: (1) writing requirements (2) annotate the requirement (3) start software design based on requirements, (4) check the traceability between requirements and UML models. This approach neglects information existing between requirements presented as textual documents and UML diagrams at the instance level.

In addition, [8] proposes a hybrid approach that combines graphs and information retrieval techniques to identify the requirement change impact on design models modeled with Systems Modeling Language (SysML). This approach is limited to traceability between requirements modeled with SysML and behavioral diagram modeled with the activity diagram. In addition, many behavioral aspects in the activity diagrams are not assigned like Join node, Fork node, etc.

For the purpose of reuse, [33] depicts an approach that derives systematically a standard functional model from a use case diagram, a structure diagram and a transition diagram. By decomposing the existing functional model into model components, traceability links are recovered based on guidelines that allow a mapping of model components to non-functional requirements. This approach is limited to use cases names without referring to use case descriptions.

Adopting an Information Retrieval (IR) technique to identify traceability between requirement and design, [34] proposes a method that uses graphs to model the structural dependencies. The Information Retrieval technique is used to handle the semantic traceability between the use case documentation and the sequence diagram. This approach is based on a structural textual description of a use case to express requirements. However, this description lacks structural controls which are used in UML behavioural diagrams.

On the other hand, [6] proposes a method that uses graphs to model the structural dependencies and an information retrieval technique to handle the semantic traceability between the use case documentation and the sequence diagrams. This approach is based on a structural

textual description of a use case diagram to express requirements; however, this description lacks structural controls which are used frequently in the UML behavioural diagrams. In fact, it is not possible to trace between behavioural elements in design and control structures in code functions such as loop, switch, etc. Additionally, the limitation of this approach lies in its incapability to determine the nature of functions/ methods that corresponds to a use a case textual description.

Furthermore, several approaches adopt a Natural language processing approach (NLP). For instance, [35] determines basic elements of a class diagram from natural language requirements. Requirements are presented in English and the designed tool (Natural language Processing for Class NLPC) applies NLP methods to analyze the given input. Natural language text is semantically analyzed to obtain classes, data members and member functions. NLPC uses pre-processing, Part of Speech (POS) Tagging, Class Identification, Attribute and Function identification to plot the classes.

[5] extracts class diagrams from natural language requirements using NLP techniques such as WordNet, OpenNLP parser, class extraction engine, etc. Moreover, the authors proposed a system based on rules to extract details related to the object oriented concepts like generalization, association and dependency from natural language requirements specification.

Furthermore, [35] adopts a NLP approach to show that natural language requirements are semantically analysed to obtain classes, data members and member functions.

Based on a combination between NLP and artificial neural networks, [36] proposes a new approach to automatically identify actors and actions in a natural language based requirements description of a system. They used an NLP parser with a general architecture for text engineering, producing lexicons, syntaxes, and semantic analyses. An artificial neural networks (ANN) was developed using five different use cases, producing different results due to their complexity and linguistic formation.

2.2 Traceability from requirement and design to code

Besides traceability inter UML models, the vertical traceability approaches tackle also the relationships between requirement, design and code [20, 37, 38, 39].
In this context, to support traceability between requirement and source code, [20] proposes a meta-model based approach that defines traceability links between different artifacts (requirements, test cases, etc.) and source code. The authors propose an editor to visualize traceability between the source code stored as an Abstract Syntax Tree (AST) and other possible artifacts. However, the use of an AST causes foreign problems like the existence of syntax errors and comments in the source code which loses traceability links.

In [37], the focus is on the traceability between requirement and source code in the context of version control system. Specifically, the authors study the link between issues (i.e. new requests), commits (change set), and source code files. They train a classifier to identify missing issue tags in commit messages to generate missing links.

Besides, in the purpose of supporting traceability between requirement and source code, [40] introduces a solution for automating the evolution of bidirectional trace links between source code classes or methods and requirements. The solution depends on a set of heuristics coupled with refactoring detection tools and informational retrieval algorithms to detect predefined change scenarios that occur across contiguous versions of a software system.

To trace between requirements documents, UML class diagrams, and source code, [41] [42] use graph and XML format to capture links between artifact elements.

Based on a set of policies, [38] [39] describe an approach which allows maintaining traceability of evolving architecture to implementation links. They develop a tool “ArchTrace” which maintain existing traceability link. These links have to be created manually by the developers or by a traceability recovery method. In addition, the authors distinguish between four classes of rules depending on the level where the change occurs. For instance, architectural element evolution policies trigger when an architect makes modifications to an architecture. An example of an architectural policy is illustrated in the case of creating a new version of an architectural element [39]. This new version of this element should inherit all traceability links from its ancestor based on a copy of all traceability links from its previous version.

By referring to machine learning techniques, [22] presents a process to recover traceability links between Java programs entities and elements in a use case diagram. This solution, which is called LEarning and ANAlyzing Requirements Traceability (LeanArt), combines program analysis, run-time monitoring, and machine learning to search similarities between the names and values of program entities, and the elements names of use case diagrams. This work is only based on traceability between use case name and source code. Nonetheless, it does not take into account the different scenarios that can be found in a use case textual description.

Likewise, [14] proposes an approach called TRAIL (TRAceability lInk cLassifier) that applies Traceability Link Recovery (TLR) as a binary classification problem for automating traceability maintenance. It uses historically collected traceability information (i.e., existing traceability links between pairs of artifacts) to train a machine learning classifier which is then able to classify the link between any new or existing pair of artifacts as valid (i.e., the two artifacts are related) or invalid (i.e., the two artifacts are unrelated) [29]. To determine the validity of the link between two artifacts, TRAIL introduces three types of features: IR Ranking, Query Quality, and Document Statistics.

[43] proposes a neural network architecture that utilizes word embedding and Recurrent Neural Network (RNN) technique to automatically generate trace links. Word embedding learns word vectors that represent knowledge of the domain corpus and RNN uses these

word vectors to learn the sentence semantics of requirements artifacts. The authors use an existing training set of validated trace links from the domain to train the RNN to predict the likelihood of a trace link existing between two software artifacts. For each artifact (i.e. requirement, source code file, etc.), each word is replaced by its associated vector representation learned in the word embedding training phase and then sequentially fed into the RNN. The final output of RNN is a vector that represents the semantic information of the artifact. The tracing network then compares the semantic vectors of two artifacts and outputs the probability that they are linked.

IR techniques are used also to define traceability between models and the source code. [17, 18, 19] use the Latent semantic indexing (LSI) to recover traceability between different artifacts. For instance, [17] uses LSI to recover traceability links between software artefacts produced during the different phases of a development project (use case diagrams, interaction diagrams, test cases and code). [7] utilizes comments and identifier names within the source code to match them with sections of corresponding documents. [13] establishes traceability between requirement and other software elements (code elements, API documentation, and comments) by taking into account the change frequency, and the semantic similarity (TF-IDF) between the requirement description and the software element.

In order to improve IR-based traceability recovery, [44] combines IR techniques with closeness analysis. Specifically, the work quantifies and utilizes the "closeness" for each call and data dependency between two classes to improve rankings of traceability candidate lists. In [45], the authors propose an improvement of the previous approach by introducing user feedback into the closeness analysis on call and data dependencies in code. Specifically, the approach iteratively asks users to verify a chosen candidate link based on the quantified functional similarity for each code dependency (which they called closeness) and the generated Information Retrieval values. The verified link is then used as the input to re-rank the unverified candidate links.

Based on NLP techniques, [25] defines an enhanced framework of software artefact traceability management which is implemented in the "SATAnalyzer" tool. NLP techniques are used to extract information from artefacts produced during software development process. The tool supports the traceability between requirements, UML class diagrams, and corresponding Java code. [15] extends the SAT-Analyzer tool to consider traceability among other stages of development life cycle such as testing and deployment with enhanced visualization suitable for DevOps practices and continuous integration.

In order to evaluate their graph-based traceability approach, [46, 47] use also the SAT-Analyser tool with a "Sale system Point" case. They present phases such as software artefact identification, data preprocessing, data extraction and traceability establishment methodologies presented with a graph. The tool traces software requirement artifact in natural language, only UML class diagram as design artefact and the Java source code artifact. The traceability graph construction is based on similarity algorithms (Jaro Winkler Distance and Levenshtein Distance) between requirements, classes, methods, attributes and the relationships inheritance, association and generalization.

Using a model-based approach, [25, 48] derive a quality model to present traceability (Traceability Assessment Model (TAM)) that specifies per element (class, link, path) the acceptable state (Traceability Gate) and unacceptable deviations (Traceability Problem) from this state. The authors describe how both, the acceptable states and the unacceptable deviations can be detected to systematically assess their project's traceability. In order to improve the previous works, [2] defines a system allowing to ensure that the software delivered meets all requirements and thus avoids failures by using data traceability management.

In summary, existing works tackled the traceability either between UML diagrams at the same abstraction level (or similar notations) or between UML models (requirements, design, etc.) and the source code, at different abstraction levels. However, none of the existing approaches deal with traceability between requirements presented with an enriched template that covers the whole

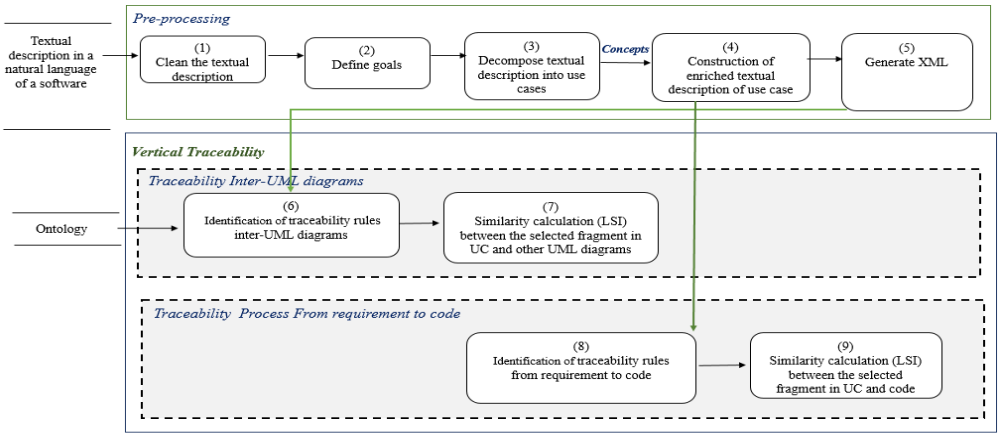


Figure 1: The proposed method for tracing UML code based on textual description of use cases.

concepts, design models and source code. In addition, all traceability techniques [49, 50] rely on either the structural and/or semantic information. For example, [20, 21, 41] determine traceability between heterogeneous terms existing in models (text in requirements, classes name, methods name, etc.). These works are purely structure-based; they ignore the remaining aspects of UML diagrams elements, which do affect the traceability between them.

The purpose of the proposed method focus on enriching the requirement template presented in [6] to cope with the control structures and orient our traceability. Furthermore, it combines both structural and semantic aspects in order to determine the traceability between all elements at different abstraction levels and detects the relationships between the requirements, design (modelled with sequence (SD), class (CD), activity (AD) and state transition (STD) diagrams (first phase), and the source code (second phase).

3 A new traceability method

Figure 1 depicts our method for determining vertical traceability. It followed three major phases: "Pre-processing Natural language", "traceability inter-UML diagrams" phase and "traceability from requirement and design to code" phase.

The "Pre-processing Natural language" phase during which the software analyst receives a textual description of a software written in a natural language. The description is cleaned based on simple NLP technique (i.e. Stanford CoreNLP tool) [51]. Then, the software analyst uses the output to identify the goals that are used to divide the textual description into different parts. The proposed decomposition guides and improves the generation of description parts and the corresponding fragments related to design diagrams in a more systematic, rigorous, and consistent way. For each description part, the software analyst prepares its textual description according to a specific template. To handle this requirement, we define an enriched template that can be written in a specific format. The template is used to generate its corresponding XML file. The second phase, "Traceability inter-UML diagrams" receives the produced file which will be considered as the input to the traceability process.

The latter is composed of traceability rules identification and similarity calculation between the selected fragment in the use case and its corresponding in UML design diagrams (class, sequence, activity and state transition). This process uses the identification of traceability rules and semantic traceability results. The identification of traceability rules explicitly represents the relationships (structural aspect) among the diagrams' elements. It is based on an ontology for the semantic analysis of the textual description template. To identify the semantic traceability between the structured textual documentation and UML design diagrams, traceability process inter-UML diagrams apply the LSI technique.

The third phase is based on the traceability process from requirement to code which apply the traceability rules defined in the first phase on the code and calculate the similarity between the selected fragment in UC and the code.

3.1 Natural language pre-processing

The most important challenge we are facing when trying to generate the enriched format from the textual description is the complexity of natural language. Consequently, we used natural language processing concepts that are syntax parsing.

The syntax parsing consists in obtaining a structured representation of the software knowledge. Therefore, the software analyst has first to clean the textual description by using the Stanford CoreNLP tool [51] and second to organize it according to a specific template's structure. Stanford CoreNLP tool is used to obtain a more manageable and readable text. The tool relies on the following methods:

- Tokenization is the task of breaking a character sequence up into pieces (words/phrases) called tokens, and perhaps at the same time throw away certain characters such as punctuation marks [52].
- Filtering aims to remove some stop words from the text. Words, which have no significant relevance and can be removed from the documents [53].
- Lemmatization considers the morphological analysis of the words, i.e. grouping together the various inflected forms of a word so they can be analysed as a single item.
- Stemming aims at obtaining stem (root) of derived words. Stemming algorithms are indeed language dependent [54].
- Part of Speech Tagging tags for each word (whether the word is a noun, verb, adjective, etc.), then finds the most likely parse tree for a piece of text.

The cleaned file is then used to identify the goals. By goal, we mean a collection of functionalities that are related to describe a functional process of the software. Each goal will correspond to a textual description of a use case.

To guide and improve the generation of a software in a more systematic way, the software analyst associates to each textual description of a part, a template that is described by a set of linguistic patterns. The template is easy to understand and validated by stakeholders. It covers the semantic, behavioural, functional and organizational information. It is composed of three blocks (See Table 1).

The first block gives an executive summary of the textual description block in terms of the name of the UC, purpose of the use case and actors. The second block describes the main, alternative, and error scenarios. The use case description contains also pre-condition for execution, post-condition (success/failure), and relationships with parts successors. These scenarios respect a linguistic syntax pattern:

<NumAction><From Actor><To Actor> <Type of Result> <Action Description> <In-Parameter><Out-Parameter> <IsConsidered><IsIgnored> <IsNegative>

Table 1 depicts the expanded description template with alternative behavioral elements based on control structures such as IF-THEN statement and iterative

elements, e.g. <for><number of iterations>). In addition, the extended template expresses how the actions are executed: in a parallel <Parallel>, or sequence way <Sequential>, etc. Besides the common elements, we proposed an extension of the UC textual description with behavioral elements and keywords, such as:

- <In-Parameter> and <Out-Parameter> expressing the input and output of the action.
- <Type of Result > which determines if the result has a simple value or it represents an entity. In the case of a simple value, it can be represented as an attribute. However, in the case of an entity, it can be transformed to a class in the class diagram.
- <From Actor><To Actor> which represents the sender and receiver of the action.
- <If><Else if><Else>represents a choice or behaviour alternatives.
- <Parallel> expresses parallel execution of the actions.
- <For> <number of iterations> represents the loop which is repeated a number of times.
- <Loop>: an iterative behaviour that englobes one or several actions.
- <Break> represents an exceptional situation corresponding to a scenario of rupture.
- <Functional Call> is an action that calls another action or use case.
- <IsIgnored> reflecting that the actions types can be considered insignificant and are implicitly ignored.
- <IsConsidered> determines which actions should be considered within this textual description, meaning that any other action will be ignored.
- <IsNegative> describes actions of traces that are defined to be negative (invalid). Negative traces occur when the system has failed. It can represent an exception.

The added behavioral elements and keywords are organized according to the use case scenario. The main scenario contains sequential or parallel actions. It can also contain a functional call; while the alternative and error scenario are based on conditional (opt, If Else, etc.) or iterative (Loop) control structures that can be expressed in one or more levels (nested levels). For instance, it is possible to determine an iterative block nested in a conditional block and vice versa. These control structure types can be followed by parallel or sequence blocs.

Name of the Use Case (UC):<unique name assigned to a use case>
Purpose of the use case:< a summary of a UC purpose>
Actors:<Primary actor>: actor that initiates the use case>
<Secondary actor>: actor that participate within the use case>

Pre-condition for execution:<A list of conditions that must be true to initialize the UC>
Post-condition (success/failure):<state of the system if the goal is achieved/abandoned>
Relationships: <include>: <UC in relation with this UC by include>
<Extend>: < use cases in relation with this use case by "extend">
<Super use case>: <list of subordinate uses cases of this use case>
<Sub use case>: <list of all uses cases that specialize this use case>
Begin
Main scenario <steps of the scenario to goal>
Begin
//sequential actions
<NumAction><From Actor><To Actor> <Type of Result> <Action Description> <In-Parameter><Out-Parameter> <IsConsidered><IsIgnored>
<IsNegative>
//parallel actions
Parallel
<NumAction><From Actor><To Actor><Type of Result ><Action Description><In-Parameter><Out-Parameter><IsConsidered> <IsIgnored> <IsNegative>
<NumAction><From Actor><To Actor><Type of Result ><Action Description><In-Parameter><Out Parameter> <IsConsidered> <IsIgnored> <IsNegative>
// Functional call
Functional Call
<NumAction><From Actor><To Actor><Type of Result> <Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored> <IsNegative>
End
Alternative scenario
SA1
Begin<Event, condition>
begin at <Num "action number"> <Return "action number">
List of actions
//sequential actions
<NumAction><From Actor><To Actor><Type of Result> <Action Description> <In-Parameter><Out-Parameter> <IsConsidered> <IsIgnored><IsNegative>
//parallel actions
<NumAction><From Actor><To Actor><Type of Result> <Action Description> <In-Parameter><Out-Parameter> <IsConsidered> <IsIgnored><IsNegative>
<NumAction><From Actor><To Actor><Type of Result> <Action Description> <In-Parameter><Out-Parameter> <IsConsidered> <IsIgnored><IsNegative>
//alternative control structure in the first level
<IF><condition>
<NumAction><From Actor><To Actor><Type of Result ><Action Description><In-Parameter><Out Parameter> <IsConsidered> <IsIgnored> <IsNegative>
End IF
//iterative control structure in the first level
<Loop><Min Number of Iterationxfcyws, Max Number of Iterations >
<NumAction><From Actor><To Actor><Type of result ><Action Description><In-Parameter><Out-Parameter><IsConsidered> <IsIgnored> <IsNegative>
End Loop
End SA1
SA2
Begin<Event, condition>
begin at <Num "action number"> <Return "action number">
List of actions
// Loop nested in an alternative control structures
< IF><condition>
<NumAction> <From Actor><To Actor><Type of result ><Action Description><In-Parameter><OutParameter><IsConsidered> <IsIgnored> <IsNegative>
<Else>
<Loop><Min Number of Iterations, Max Number of Iterations >
<NumAction> <From Actor><To Actor><Type of result ><Action Description><In- Parameter> <Out Parameter><IsConsidered> <IsIgnored> <IsNegative>
End Loop
End IF
<Functional Call>
<NumAction> <From Actor><To Actor><Type of Result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored> <IsNegative>
End
End SA2
Error scenario
SE1// Treat the error and return to the action
Begin<Event, condition>
begin at <Num "action number"> <Return "action number">
List of actions
<NumAction> <From Actor><To Actor><Type of Result ><Action Description><In-

Parameter><OutParameter><IsConsidered> <IsIgnored> <IsNegative> End SE1 End Use case
Critical situations of execution of the activity Special requirement: <non Functional requirement><Project requirement and constraints>

Table 1: Enriched textual description of a use case.

3.2 Traceability process

In this subsection, we define traceability rules which are applicable to the first and second phase of our method. They are used to determine correspondences between the requirement modeled with the use case diagram based on the enriched textual description and design diagrams modeled with SD, CD, AD and STD.

3.2.1 Traceability rules

R1: For each <In-Parameter> and <Out-Parameter> expressing the input and output of the action, there is:
- **SD:** an object in a sequence diagram.
- **CD:** a class corresponding to each parameter, and an attribute corresponding to an argument.
- **AD:** an object node that corresponds to InputPin and OutputPin. We note that InputPin and OutptPin can be related to the same or more than one objectNode.
- **Code:** a class corresponding to <In parameter> and an attribute corresponding to an argument.

R2: For each action's sequence in a use case, there is:
- **SD:** a sequence of sent or received message which preserves the action order in the scenario.
- **AD:** a sequence of ordered activities.
- **STD:** a sequence of ordered states in the state diagram. If the action in a STD respects the renaming pattern: « Action verb + DataObject| NominalGroup », then the state of the action will be: Data object + past participle.
- **Code:** a sequence of lines of code that respect the ordered actions.

We note that this rule cannot be expressed in the CD.

R3: For each actor expressing the sender and the receiver of the action in the use case scenario, there is:
- **SD:** an object corresponding to each participant (actor) in the SD.
- **CD:** a class corresponding to each participant in the CD.
- **AD:** a swimlane having the actor name which performs a group of activities.
- **STD:** the actor has no corresponding in the STD.
- **Code:** a class in the code.

R4: For each action in the use case scenario, there is:
- **SD:** a message in a SD having a synonym name.
- **CD:** a method in a class corresponding to the action.
- **AD:** an executable node represented by 'Action' having the same name and the same parameters.
- **STD:** If the action in a textual description respects the renaming pattern: « Action verb + Object | Nominal Group », then the state will be : object + past participle.
- **Code:** a method in the code having the synonym name, the same parameters.

R5: For each pre-condition/post-condition of the use case scenario, there is:
- **SD:** a precondition/post-condition of the first message sent by an object in the sequence diagram.
- **AD:** a guard of the corresponding action [55]
- **STD:** a pre-condition associated to a transition which is necessary to define a state.
- **Code:** a precondition under which a method may be called and expected to produce correct results [56].

We note that the precondition and the post-condition have no corresponding in the class diagram.

R6: For each parallel scenario (PARALLEL), there is:
- **SD:** a parallel combined fragment in a sequence diagram.
- **AD:** a set of parallel actions between a fork node and a join node.
- **STD:** a fork pseudo state vertices and a join state.
- **Code:** a multi-threaded program in java.

We note that the parallelism is not expressed in the CD. R6 is illustrated in Table 2.

R7: For each alternative scenario in a use case where instructions begin with alternative behavioural elements (IF-THEN Statement ELSE Statement), there is:

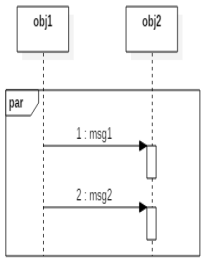
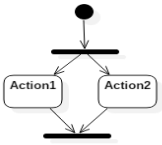
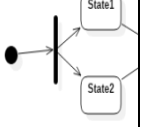
Use case	Sequence Diagram	Activity Diagram	State transition diagram	Code
PARALLEL < <NumAction><Pre-condition> <From Actor><To Actor><Action Type><Type of Result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored><IsNegative> <NumAction><Pre-condition> <From Actor><To Actor><Action Type><Type of Result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored><IsNegative>>				**public class myClassimplements** Runnable{ Thread UnThread ; MyClass ( ) {*//..initialisation of myClass constructor* UnThread = **new** Thread (**this** , "thread secondaire"); UnThread.start(); **}** **public void** run ( ) { *//....second thread actions here* **}}**

Table 2: R6 illustration.

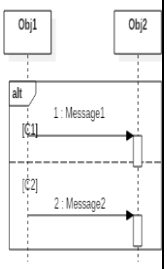
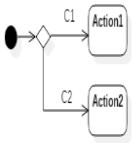
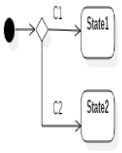
Use case	Sequence Diagram	Activity Diagram	State transition diagram	Code
<IF><condition> <NumAction><Pre-condition> <From actor><To actor><Action Type><Type of result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored><IsNegative> **<Else >** <NumAction><Pre-condition> <From actor><To actor><Action Type><Type of result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered><IsIgnored> <IsNegative> **END**				If (condition) { operation 1; else operation 2; }

Table 3: R7 illustration.

- **SD:** an ALT combined fragment with the interaction operator "ALT" and two alternative interactions in a SD.
- **AD:** a decision node with two outgoing edges with guards in the activity diagram or a conditional node is a structured activity that represents an exclusive choice between two alternatives.
- **STD:** a decision point leading to two different states in the state transition diagram.
- **Code:** a basic control structure corresponding to "IF condition THEN treatment 1 ELSE treatment2".

R7 is illustrated in Table 3.

R7.1: For each alternative Scenario where instructions begin with the alternative behavioral elements (<if > condition <else if >…..<else if>…<else>…), (SWITCH), there is:

- **SD:** an Alt Combined Fragments: Interaction operator "alt" with more than two alternatives in a SD.
- **AD:** a decision node with more than two outgoing edges in an activity diagram.
- **STD:** a decision point leading to n different states in a STD or a conditional node is a structured activity that represents an exclusive choice among some number of alternatives.
- **Code:** a basic control structure corresponding to switch.

R7.2: For each alternative scenario in a use case where instructions begin with the alternative behavioral elements (<if > condition <then> treatment.), there is:

- **SD:** an opt combined fragment in a sequence diagram. We recall that the opt (optional) operator is a non-alternative (otherwise) test statement.
- **AD:** a decision node with two outgoing edges: one to execute an action and the second is related to the final activity in the activity diagram.
- **STD:** a decision point leading to one state and one final state.
- **Code:** a basic control structure corresponding to "IF condition THEN treatment".

R7.3: For each alternative scenario in a use case where instructions contain the alternative behavioral elements (<if > condition <break>) in an iterative bloc, there is:

- **SD:** a break combined fragment in a loop fragment that belongs to a sequence diagram
- **AD:** a decision node which one is related to a final activity by an outgoing edge and another outgoing edge which is related to a final node in the alternative scenario
- **STD:** a decision point leading to 1 state and one final state (Transition to terminate pseudostate).

R7.4: For each error scenario in a use case where instructions begin with the alternative behavioral elements <IF><condition> Return, there is:

- **SD:** a break combined fragment in a sequence diagram which can be used to express an error scenario.
- **AD:** an interruptible region which contains activity nodes in the error scenario
- **STD:** a decision point leading to 1 state and one final state (Transition to terminate pseudostate). A break can be also expressed by an Exit point pseudostate which is an exit point of a state machine or composite state.

The exit point is typically used if the process is not completed but has to be escaped for some error or other issue.

R7.4 is illustrated in Table 4.

R8: For each alternative/error Scenario in a use case where instructions begin with the iterative behavioural elements (<For><[num of iterations]>…), there is:

- **SD:** a loop combined fragment in a sequence diagram.
- **AD:** a decision node with one of the outgoing edges is a precedent activity in an activity diagram.
- **STD:** a reflective transition or transition path.
- **Code:** a basic control structure corresponding to For-do, DO while (post-test), While do (pre-test).

R8 is illustrated in Table 5.

R9: For each functional call (an action that calls another action or use case), there is:

- **SD:** a ref fragment expressing the reference to an interaction in another sequence diagram.
- **AD:** a call Behaviour: An activity is invoked by using the 'Call Behavior Action' node, which means that the invoked activity is defined in more details in another AD.
- **STD:** A Composite state which encloses refinements of the given state. We note that the composite state corresponds to the object that can realize the functional call or an entry point of a state machine or composite state which allows you to specify an activity that occurs when you enter the state.

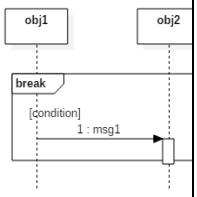
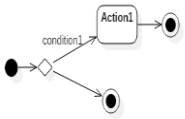
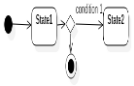
Use case	Sequence Diagram	Activity Diagram	State transition diagram	Code
<IF><condition> <NumAction><Pre-condition> <From actor><To actor><Action Type><Type of result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored><IsNegative> <Else> return				If (condition1) operation 1; Else return;

Table 4: R7.4 illustration.

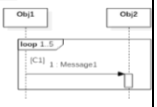
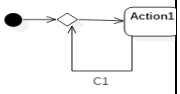
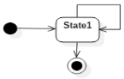
Use case	Sequence Diagram	Activity Diagram	State transition diagram	Code
<For><Min Number of Iterations, Max Number of Iterations > <NumAction><Pre-condition> <From actor><To actor><Action Type><Type of result ><Action Description> <In-Parameter> <Out-Parameter> <IsConsidered> <IsIgnored> <IsNegative> **End For**				For(i=1, i<=5,i++) operation 1; }

Table 5: R8 illustration.

- **Code:** a call of a class or a method.

R10: For each action that represents an invalid interaction/exception <IsNegative>, there is:
- **SD:** A Negative combined fragment in the sequence diagram which defines invalid traces.
- **AD:** An event representing an error (exception) that interrupts the flow or a break which are most commonly used to model exception handling.
- **STD:** a transition to an error state. This error state may be terminal, i.e. aborts further event handling.
- **Code:** a basic control structure corresponding to "Exception". This corresponds to a try-catch.

R11: For each action that should be considered (respectively ignored) within the scenario, there is:
- **SD:** messages that are considered as significant (respectively insignificant) within the "consider" (respectively ignore) combined fragment.
- **AD:** considered (respectively ignored) messages are shown in the activity diagram.
- **STD:** The states corresponding to the considered (respectively ignored) actions.
- **Code:** The method should be considered (ignored) as significant in the code.

3.2.2 Similarity calculation

Based on the proposed rules, we apply the similarity measure "Latent Semantic Indexing" (LSI) which is defined to the traceability process inter-UML diagrams and to the traceability process from requirement to code.

The first step in calculating the LSI is to assign term weights and construct the term-document matrix A and query matrix. The m by n document-matrix A is presented as follows where:

$$a_{ij} = w_{ij} = \text{term weights} \quad (1)$$

In the second step, LSI applies singular value decomposition (SVD) to the A matrix which consists in decomposing the A matrix into three matrices: the U, S and V. One component matrix describes the original row entities as vectors of derived orthogonal factor values, another describes the original column entities in the same way, and the third is a diagonal matrix containing scaling values such that when the three components are matrix-multiplied, the original matrix is re-constructed. The third step represents the dimensionality reduction, which consists in computing U_k, S_k, V_k and V_k^T. For instance, implementing a rank 2 Approximation (K=2) by keeping the first two columns of U and V and the first two columns and rows of S. The fourth step consists in finding the new document vector coordinates in this reduced 2-dimensional space. Rows of V hold eigenvector values. These are the coordinates of individual document vectors. The fifth step finds the new query vector coordinates in the reduced 2-dimensional space as follows:

$$q = q_T U_k S_k^{-1} \quad (2)$$

Finally, the last step ranks documents in order to decrease the order of query-document cosine similarities using the following equation:

$$\text{sim}(q,d) = \frac{q.d}{|q|\,|d|} \quad (3)$$

The document which has a higher score is closer to the query vector than the other vectors.

We note that, in this paper, LSI is used to compute similarities between the selected fragment in a use case and the corresponding ones in other UML diagrams (SD, CD, AD and STD), and then the corresponding fragment in the code while in [6] the LSI is used only to compute similarities between actions in UC and messages in sequence diagrams. The choice of LSI amongst other similarity measures is justified by its capacity in retrieving hidden, semantic relations between terms when searching

for similar terms between queries extracted from a fragment in the UC and the documents containing the information in other UML diagrams. In fact, LSI does not rely on words but rather on concepts; that is, words having same contexts can be revealed similar. This propriety expresses the difference between LSI and other IR techniques. Henceforth, the similarity measure can be properly calculated between queries and documents even when they do not share enough words.

4 Traceability evaluation

The evaluation phase expresses the performance of the proposed method revealed by two steps: experimental evaluation and result interpretation. The first step in the evaluation phase compares corresponded elements generated by our method with the corresponded elements where traceability is evaluated by experts. Particularly, we present two UML projects containing a set of UML diagrams (including use cases and their textual descriptions) and the source code (projects are implemented using JAVA language) to five experts having years of experience studying and developing UML projects. The expert should determine traceability by detecting the corresponding elements. The solution presented by these experts was compared with our solution (constructed by our tool). The projects source codes are available as well as their design (i.e. UML diagrams). Table 6 provides some information about these projects. Besides, for experimental evaluation purposes, we refer to the recall and precision measures:

$$\text{Precision} = TP/(TP+FP) \quad (4)$$

$$\text{Recall} = TP/(TP+FN) \quad (5)$$

where:

- True positive (TP) is the number of existing real corresponded elements generated by our tool;
- False Positive (FP) is the number of non existing real corresponded elements generated by our tool;
- False Negative (FN) is the number of existing real corresponded elements not generated by our tool.

4.1 Evaluation results and interpretation

High scores for both ratios show that our traceability approach returns both accurate corresponding elements of UML diagram (high precision) and the majority of all relevant corresponding elements (high recall). It means that the generated traceability links cover the whole domain precisely in accordance to the experts' perspective.

As illustrated in Table 7, precision, whose average is 0.84, indicates that we found some false positive corresponding elements (i.e. incorrect detected corresponding elements). The false positives corresponding elements are not significant value when we compare them to the true positives found by our method. The recall, whose average value is 0.91, expresses that there are also some false negatives corresponding elements (i.e. true corresponding elements are not detected). These false negatives can be explained by the fact that our method uses "threads" to detect parallelism in the source code however parallelism in JAVA can be implemented using different ways (fork/join framework, threads, Agregate operations, etc.). Source code in the used projects uses the aggregate operations and parallel streams to express parallelism and our method uses threads to detect parallelism. This is why parallel fragments are not traced and we found some false negatives.

The true positives and the false negatives are equal to the total number of actual corresponding elements. All the false negatives are corresponding elements associated to elements in UC textual description diagram that have corresponding impacts on other UML diagrams which are not detected.

4.2 Threats to validity

This section discusses the potential issues that may threaten the validity of our study, including the internal and external validity [57].

The internal validity threats in the case of traceability identification are related to user requirements [58]. They are related to three issues: The first issue is due to the use of the enriched textual description of a use case which may not always be available. The second problem is addressed when there is a diversity of requirements description. In this case, which one can be used to describe the functional requirements?

Furthermore, if the functional requirements are clearly stated, then our method generates well matched elements; otherwise, the quality of the derived traceability elements is not guaranteed in terms of dependencies between elements. The third issue is related to the impact of an error-prone generation of UML diagrams and code. This case may lead to inconsistency between the requirement, design models and source code.

The external validity threats deal with the possibility to generalize this study results to other case studies. The limited number of case studies used to illustrate the proposed approach could not generalize the results. In addition, the traceability between all levels increases the detection and localization of consistency errors.

5 TRADIAC tool

To facilitate the application of our method, we have developed a tool for determining the traceability at different abstraction levels, named TRADIAC Quality (TRAceability for UML DIAgrams and Code). Our tool is implemented as an EclipseTM plug-in [59]. It is composed of four main modules (see Figure 2): Pre-processing Natural language, Traceability inter-UML diagrams, Traceability from requirement to code, and traceability evaluator.

5.1 Pre-processing natural language module

The pre-processing engine is composed of the cleaner and the XML generator.

PROJECT NAME	#Use cases	# CLASSES	# METHODS	KLOC
Car rental system	9	98	252	108
Customer Relationships system	7	65	124	96

Table 6: Characteristics of the studied projects.

Evaluation Measures	TP	FP	FN	Precision= TP/(TP+FP)	Recall=TP/ (TP+FN)
Results	62	9	5	0.84	0.91

Table 7: Evaluation results.

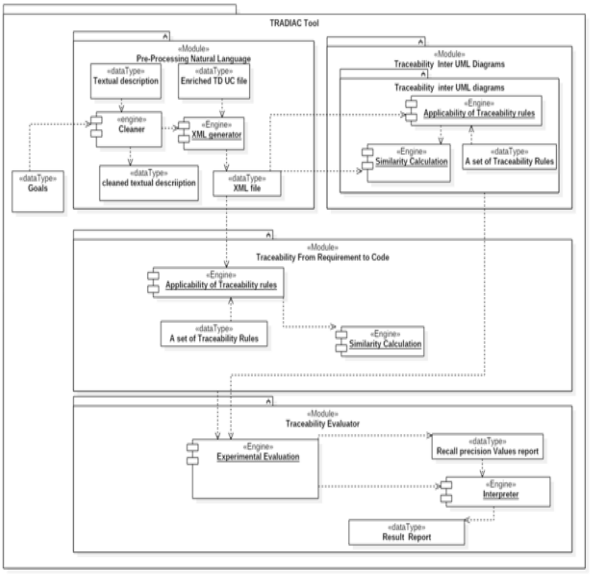


Figure 2: Software architecture of TRADICAC Quality Tool.

Goals: The purpose is to make a reservation by a customer from a car rental branch.

Textual description: The use case begins when a customer decides to make a reservation and introduce himself in the car rental branch to an available Clerk. The clerk asks the customer for his/her ID and introduces it.

The system checks if the customer is a person who has had contact with EU-Rent. If he/she exists, the system verifies that the customer is not in the black list otherwise it introduces a new EU-rent costumer/driver. The clerk introduces the reservation ID, the period desired and countries planned to visit. He specifies and verifies the period validity and that there is no overlap with other customer reservations and 3) the availability of the specified car model for the period indicated.

If there are no cars to rent corresponding to the desired model in the selected period, the system displays an error message to the user and suggests if it is possible to change the reservation period or the car type. The clerk asks the customer to validate the reservation.

If the customer validates the reservation, the clerk creates the reservation agreement and offers a discount to the customer. The rental is confirmed and a new rental agreement is created with the indicated parameters.

Figure 3: Goal definition.

5.1.1 Cleaner

The cleaner uses as input the textual description of the software written in a natural language. It cleaned the file using the Stanford CoreNLP tool. The cleaned file is used by the software analyst to define manually goals. Then, the latter associates each goal to its corresponding textual description part.

In order to illustrate the functioning of this module, we apply it to the “make a reservation” textual description. For instance, Figure 3 illustrates the goal definition and its description. The software analyst creates the enriched template corresponding to each textual description part. Table 8 illustrates enriched textual description for the use case "UC-ETD" “make a reservation” from a car rental system [60].

5.1.2 XML generator

XML generator takes as input the enriched textual description of UC introduced by the user. The purpose interface of "UC-ETD" is presented in Fig. 4. It is composed respectively of five tabs illustrating the identification purposes "identification purpose", the nominal scenario "Main Scenario", the alternative scenario (s) "Alternative Scenario", the error scenario (s) "Error scenario (s)" and the generator of the XML file corresponding to the textual description. The "identification purpose" tab contains the name of the UC, its purpose, the primary and secondary list of the actors, the pre-condition and the post-condition of the UC in the textual description and the use case's relationships: include, extend and generalize. The list expresses use cases in relation with the corresponding one by “include”, use cases in relation with the corresponding use case by “extend”, subordinate uses cases of the super UC and the list of all uses cases that specialize the sub use case. The three other tabs express the details of the different UC scenarios being documented. The last tab expresses the XML file corresponding to the textual description of the whole UC. In the rest of the section, we detail these tabs through the use case “make a reservation” from the case study “Car Rental” [60]. The enriched textual description for the use case “make a reservation” is presented in Table 8 describing the purpose (See Figure 4) of the UC, Figure 5, Figure 6 and Figure 7 presenting respectively the main, alternative and error scenarios, and Figure 8 illustrates the corresponding XML file based on the enriched template of the “make a reservation” UC.

- Addition a nominal scenario NS: The “Main Scenario” (see Figure 5) shows the list of actions in the main scenario which can be classified on two blocs: sequential or parallel actions. Each bloc indicates how these actions are executed. It is composed of seven columns representing respectively: a) NumAction that indicates an automatic number identifying an action, b) Fom actor and c)To actor which allows to specify who is responsible for the action, d)Type of result which determines if the result is a simple value or it represents an entity, and e)Action description representing a field specifying the action text, f) In-

Name of the UC : <Make a reservation >
Purpose: <A customer makes a reservation from an EU-Rent branch >
Principal Actors : < Clerk > , <**Secondary actor**> : < Customer>

Pre-condition for execution:< when a customer decides to make a reservation and inform the clerk>
Post-condition (success): < The rental is confirmed and a new rental agreement is created with the indicated characteristics>
Post-condition (failure): <The indicated characteristics are not satisfied and a rental is canceled>
Relationships:
<include>: < Offer discount>< Offer special advantages>
<Extend>: < -- >;
<Super use case>: < -->;
<Sub use case>: <-->
Begin

******Normal scenario******

<steps of the scenario of the trigger to goal>
Begin NS
<NumAction 1> < From Actor Clerk >< To Actor Customer> < Type of Result Simple: Integer> < Action Description Asks the custumer for hisID > <In-Parameter: IDCustomer > <Out-Parameter IDCustomer >< IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<NumAction 2> < From Actor The Clerk >< To Actor The Customer > < Type of Result Simple: Boolean>< Action Description Checks if the customer is a person who had contact with EU-Rent> <In-Parameter IDcustomer > <Out-Parameter Exists: Yes> < IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<NumAction 3>< From Actor Customer> < To Actor clerk>< Type of Result Entity>< Action Description Tells information about the reservation to the clerk><In-Parameter Reservation (IDRes,StartResDat, End ResDat, DepartureCity, Arrival city)> <Out-Parameter Reservation (IDRes , StartResDat, End ResDat, Departure/Arrival City, registration number car) >< IsConsidered 1>< IsIgnored 0><IsNegative 0>
<NumAction 4> < From Actor Clerk >< To Actor The reservation> < Type of Result Entity>< Action Description Introduces the reservation ID, the period desired and countries planned to visit ><In-Parameter IDRes, StartResDat, End ResDat, DepartureCity, Arrival city, registration number car ><Out-Parameter Reservation (IDRes, StartResDat, End ResDat, Departure/ArrivalCity, registration number car)> < IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<NumAction 5> < From Actor The Clerk >< To Actor The reservation><Type of Result Simple: boolean><Verify that the period is correct, that there is no overlap with other reservations of the customer and the availability of the specified car model for the period indicated>
<In-Parameter: Reservation> <Out-Parameter availability car > < IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<Parallel>
<NumAction 6>< From Actor The Clerk >< To Actor The agreement> < Type of Result Entity>< Action Description Create the reservation agreement ><In-Parameter rental agreement: ID customer, price, ID reservation> <Out-Parameter Rental agreement >< IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<NumAction 7> < From Actor The Clerk >< To Actor The agreement>< Type of Result Entity>< Action Description Offer a discount to the customer ><In-Parameter Discount rental agreement : Discount,ID agreement, ID customer> <Out-Parameter Discount rental agreement >< IsConsidered 1>< IsIgnored 0>< IsNegative 0>
End

******Alternative scenario******

AS1
Begin <Event, begin at Num 2 in SN>
<IF>< the customer does not exists >
<NumAction 1> < From Actor Clerk >< To Actor customer> < Type of Result Customer (name, ID, birthdate, address, phone)>< Action Description Introduce a new customer> <In-Parameter Customer (name, ID, birthdate, address, telephone)> <Out-Parameter Customer > < IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<Else > <restart at num 3 in SN>
End AS1
AS2
<begin at Num 3 in SN>
<Do>
<NumAction 1>< From Actor clerk ><To Actor reservation><Type of Result Simple>< Action Description Specifies the period ><In Parameter period > <Out-Parameter period >< IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<NumAction 2> < From Actor clerk><To Actor reservation> <Type of Result correct yes/no>< Action Description Verify the period>] [<In-Parameter period>] [<Out-Parameter correct yes/no >]< IsConsidered 1>< IsIgnored 0> < IsNegative 0>
<While><period is correct &does not overlaps with other reservations >
<restart at Num "6" in SN>
End AS2
AS3
<begin at Num 5 in SN>
<Opt><needs confirmation>
<NumAction 6> < From Actor Clerk ><To Actor Customer><Type of Result Simple: Boolean > <Action Description Asks the customer if he validates the reservation> <In-Parameter Reservation><Out-Parameter IsValidated >< IsConsidered 1>< IsIgnored 0>< IsNegative 0>
<restart at Num "6" in SN>
End AS3
****** Error scenario ******
ES1
<begin at Num "5">
<IF><There is no cars to rent having the desired model in the selected period>
<NumAction 6> < From Actor The system >< To Actor The Clerk >< Type of Result Entity >< Action Description displays an error message to the user and suggests if it is possible to change the reservation period or the car type ><In-Parameter: Car model, period > <Out-Parameter Error message> < IsConsidered 0> < IsIgnored 0> < IsNegative 1>
Return
End

Table 8: Enriched textual description for the use case "make a reservation".

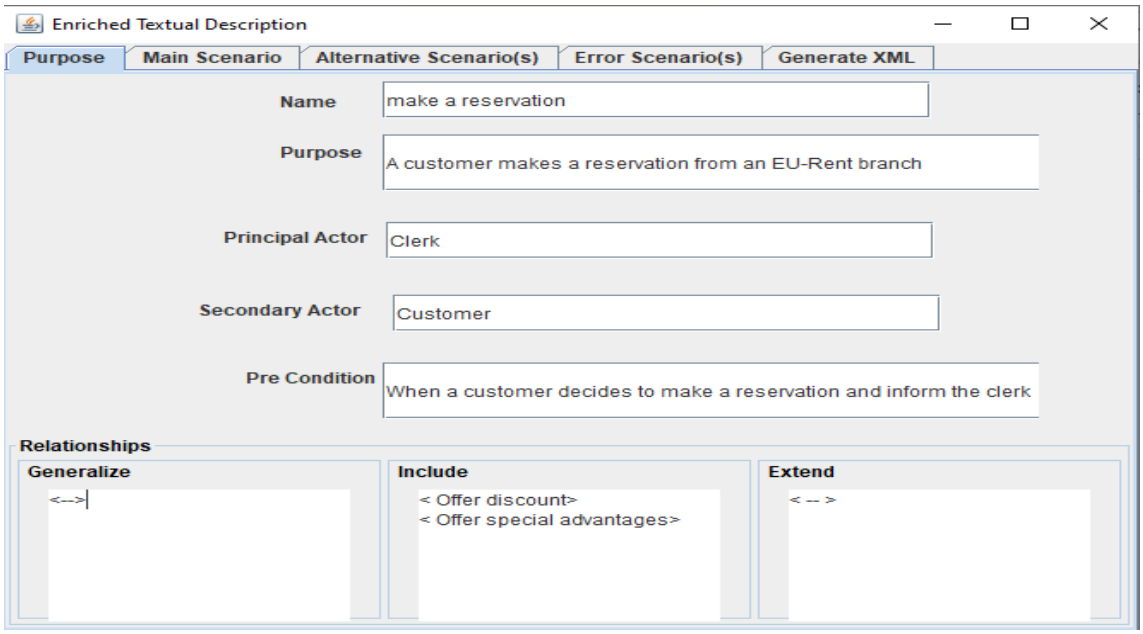


Figure 4: UC-ETD “make a reservation” purpose interface.

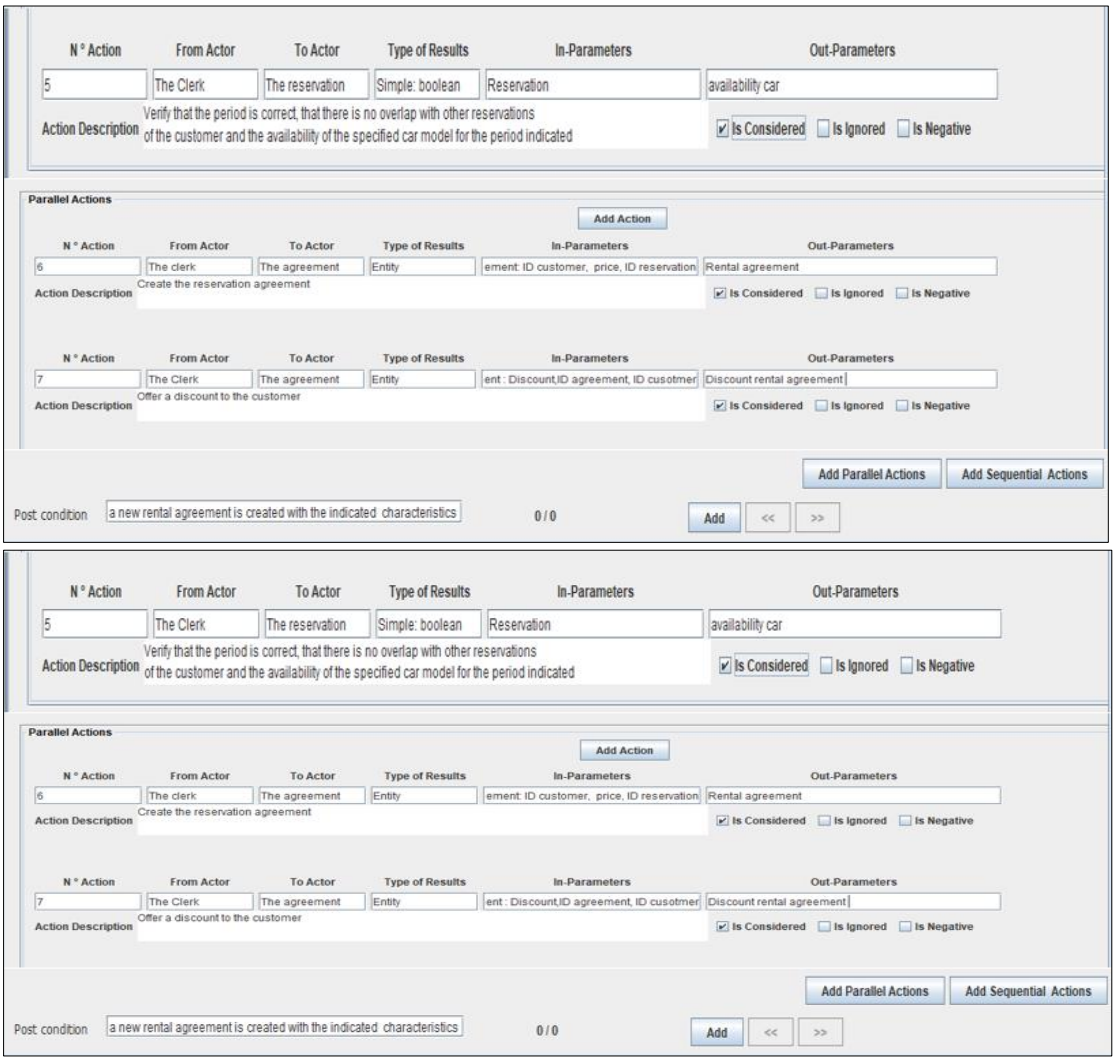


Figure 5: Main scenario of the "make a reservation" use case.

Parameter expressing the input of the action g)Out Parameter expressing the output of the action, and h) boolean value corresponding to each state of the action that can be Considered, IsIgnored, IsNegative. To add a nominal scenario in a specified bloc, click on the "Add Parallel Actions" button or "Add Sequential Actions" in the corresponding bloc.

– Addition of an alternative scenario and /or errors: Figure 6 and Figure 7 illustrate, respectively, the alternative and error scenarios. Each alternative or error scenario is composed of two blocks where the

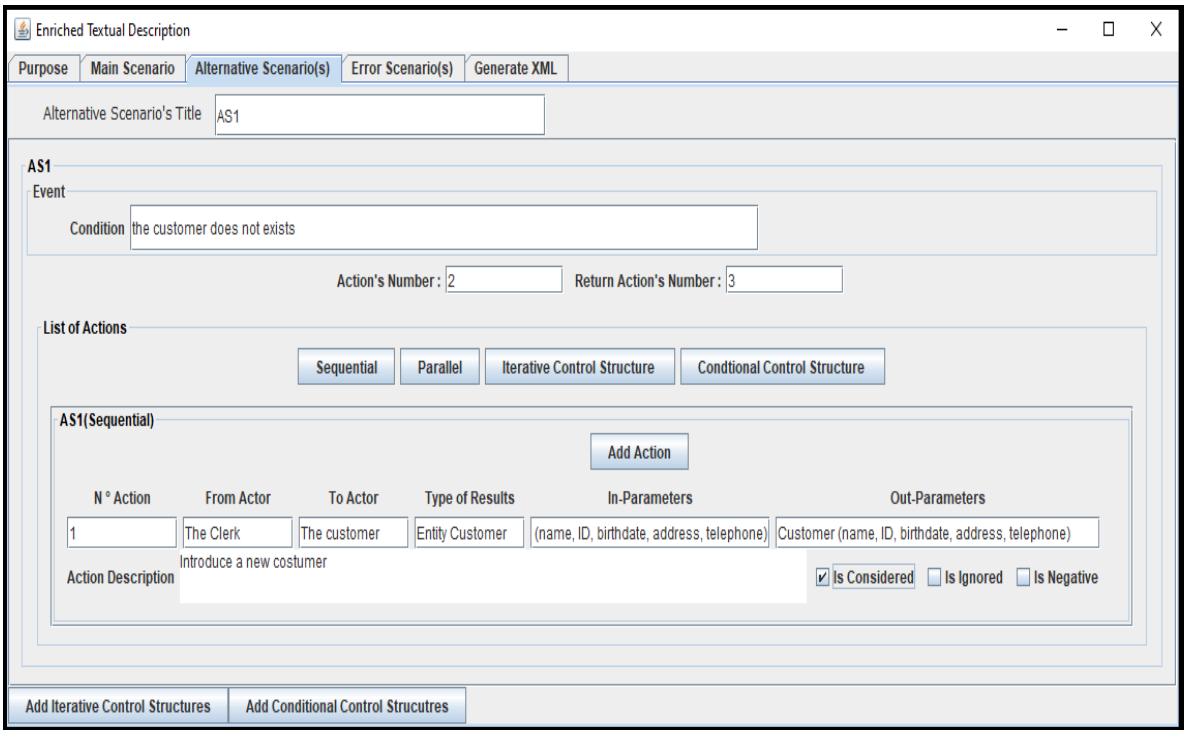


Figure 6: Alternative scenario of the "make a reservation" use case.

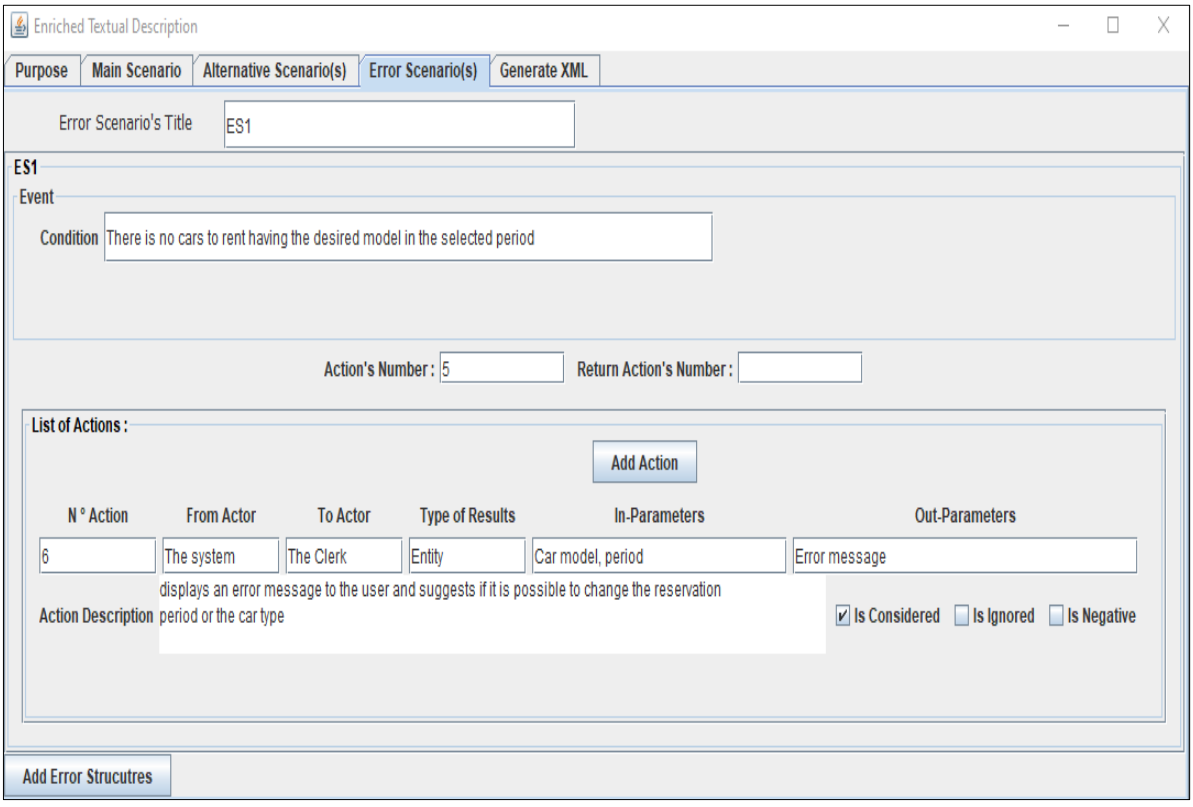


Figure 7: Error scenario of the "make a reservation" use case.

user enters the following information: The first bloc contains the scenario title, guard condition of the event triggering the scenario, the start action at the alternative scenario level and the return action number if it exists. We note that the alternative scenario may contain conditional and/or iterative control structures. In addition, it is possible to depict nested blocs. For instance, an iterative control structures can be nested in an alternative control structures and vice versa. Besides, each bloc includes the list of actions executed in a parallel or sequential way. For each action, the user enters the corresponding information as presented in the nominal scenario (from a to h). To add a new alternative scenario, click on the "Add Conditional Control Structures" button or "add Iterative Control Structures". Similarly, to add an error scenario, click on the "Add Error Structures" button.

After entering the enriched textual description of the make a reservation use case, the XML generator module produces the XMI document as illustrated in Figure 8. To generate the XML file corresponding to the obtained XMI document, this module uses the standard template of Star UML definition. For example, we present as follows the generated XML file corresponding to the documentation of the "Make a reservation" use case of our “Car rental” case study.

5.2 Traceability process module

The traceability module is composed of two engines: applicability of traceability rules and calculation similarity.

5.2.1 Applicability of Traceability rules

In the traceability detection module, the user firstly imports the UML project. In this step, the designer chooses a UC from a list of use cases. Then, the designer can choose a specific fragment to be traced from the selected UC. The presented fragments represent specific concepts which we added in the UC textual description (e.g., parallel, sequence, loop, conditional, break, etc.). For instance, the designer needs to trace the ‘parallel’ fragment in the enriched textual description by checking the list of parallel fragments which are available in a list box as shown in Figure 9. Next, we apply the similarity measure LSI between the XML of the selected parallel fragment and the related UML diagrams; and the source code based on the defined traceability rules.

5.2.2 Similarity calculator

The similarity calculator uses the XML files to determine the traceability inter-UML diagrams where the module computes the similarity between the selected fragment and other UML diagrams (CD, AD, STD and SD), and the traceability between UML diagrams and code where the module detects the corresponding elements between the UML diagrams and the code.

We offer to the designer a pairwise traceability (two by two) from the use case diagram to the other diagrams. For instance, when the designer chooses Use case-sequence, the system calculates the similarity between the parallel fragment in the use case diagram and each parallel fragment in the sequence diagrams. To end this purpose, the similarity calculator determines the score of resemblances between the fragment elements in the enriched description and all the corresponding parallel fragments in the sequence diagram (i.e., actor/action in a use case diagram and object/message in the sequence diagrams).

The fragment having a higher score is considered as the most similar one. To decide upon the obtained score value, the constant threshold of 0.70 is widely used in the literature [17]. Consequently, we assume that a similarity value greater than or equal to 0.7 indicates a high similarity between fragments. Otherwise, the designer should verify the quality of the corresponding UML-diagrams. Besides, we calculate in the same manner the similarity between the selected fragment in the use case and the source code.

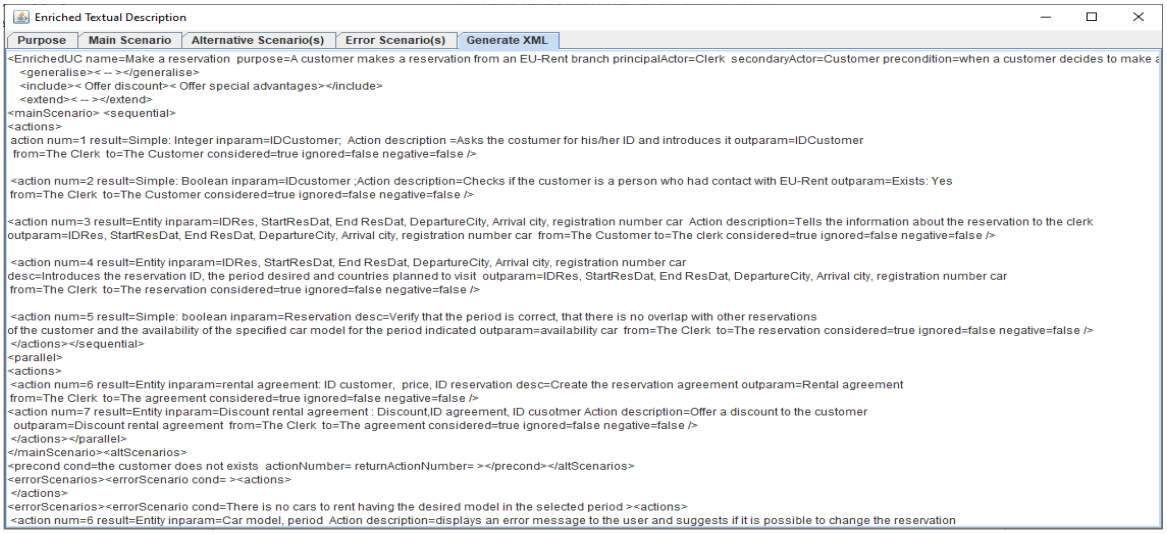


Figure 8: The generated XML file corresponding to the documentation of the "Make a reservation" use case.

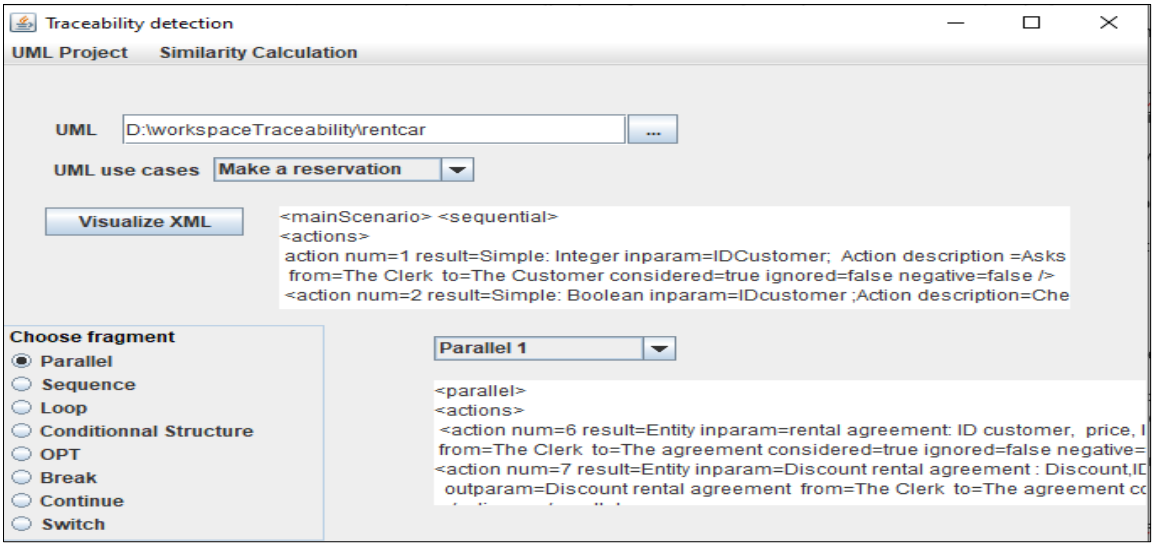


Figure 9: Traceability detection interface.

Figure 10: Traceability inter-UML Diagram.

Table 9 presents the correspondence between control structure fragments (OPT, ATL, WHILE, BREAK, ...) in the use case, activity, sequence, state transition diagram and code. Figure 10 shows traceability between the selected parallel fragment in the main scenario which includes the 6th and the 7th actions in the use case "make a reservation" and its corresponding one in the sequence, activity, class and state transition diagrams as well as the source code.

6 Conclusion

In this paper, we proposed a new method that determines the traceability at different abstraction levels. The traceability is based on the mapping between an enriched textual description of a use case and UML diagrams (class, sequence, activity and state transition diagrams) and between UML diagrams and the code. This correspondence is focused on the control structures defined in the use case textual description and the combined fragment used in the sequence diagrams.

In our future works, the following points will be taken into consideration:

- Representing data in textual description to derive directly the object and class diagram.
- Studying the possibility to derive the implementation diagrams from textual description.
- Determine the traceability from code to functional requirements based on the code-Requirement Traceability Matrix (CRTM) information.

References

[1] Y.Wang, Formal description of the UML architecture and extendibility, in: journal L'object: Software, Databases, Networks, 2000, Vol.6, No.3.

[2] P. Rempel, P. Mader, Continuous Assessment of Software Traceability, in: 38th IEEE/ACM Conference on Software Engineering Companion, May, Austin, TX, USA, 2016, pp. *747-748*, DOI: http://dx.doi.org/10.1145/2889160.2892657

[3] L. Briand, D. Falessi, S. Nejati, M. Sabetzadeh, T. Yue, Traceability and SysML Design Slices to Support Safety Inspections: A Controlled Experiment, Simula Research Laboratory, in: journal of ACM Transactions on Software Engineering and Methodolog, February, 2014, No.9. https://doi.org/10.1145/2559978

[4] A. Lawgali, Traceability of unified modeling language diagrams from use case maps, in: international Journal of Software Engineering & Applications (IJSEA), Vol.7, No.6, November, 2016, pp.89-100. doi:10.5121/ijsea.2016.7607

[5] V. Adhav, D. Ahire, A. Jadhav, D. Lokhande, Class Diagram Extraction from Textual Requirements Using NLP, in: Second International Conference on Computer Research and Development, (2015), vol.17, No 2, pp. 27-29. DOI**:** 10.1109/ICCRD.2010.71

[6] D. Kchaou, N. Bouassida, H.Ben-Abdallah, Uml models change impact analysis using a text similarity technique. In journal of IET Software, Vol 11, Issue 1, No 2, February, 2017, pp. 27-37. DOI: 10.1049/iet-sen.2015.0113

[7] P. Mader, O. Gotel, Towards automated traceability maintenance, in: Journal of Systems and Software, vol. 85, no. 10, 2012, pp. 2205–2227. https://doi.org/10.1016/j.jss.2011.10.023

[8] S. Nejati, M. Sabetzadeh, C. Arora, L.C.Briand, F.Mandoux, Automated change impact analysis between sysml models of requirements and design, in: Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, ACM, New York, USA, 2016, pp 242-253. https://doi.org/10.1145/2950290.2950293

[9] Min, H.S.: 'Traceability Guideline for Software Requirements and UML Design'. in: Journal of Software Engineering and Knowledge Engineering, 26, (01), 2016, pp. 87-113.

[10] M. Rahimi, J. Cleland-Huang, Evolving software trace links between requirements and source code, in: international journal of Empirical Software Engineering, Vol. 23, 2018, pp.2198–2231. DOI: https://doi.org/10.1007/s10664-017-9561-x

[11] G. Antoniol, G. Canfora, G. Casazza, A. De Lucia and E. Merlo, "Recovering traceability links between code and documentation," in IEEE Transactions on Software Engineering, vol. 28, no. 10, pp. 970-983, Oct. 2002, doi: 10.1109/TSE.2002.1041053.

[12] A. Ghabi, A. Egyed, Exploiting traceability uncertainty among artifacts and code, The Journal of Systems and Software, Vol. 108, October 2015, pp. 178–192. http://dx.doi.org/10.1016/j.jss.2015.06.037

[13] A. Ghannem, H. Mohamed Salah, M. Kessentini, H.A. Hany, Search-Based Requirements Traceability Recovery: A Multi-Objective Approach, in: IEEE Congress on Evolutionary Computation (CEC), San Sebastian, Spain, 5-8 June, 2017, **DOI:** 10.1109/CEC.2017.7969440

[14] C. Mills, C., J. Javier Escobar-Avila, S. Haiduc, Automatic Traceability Maintenance via Machine Learning Classification, in: IEEE International Conference on Software Maintenance and Evolution, Madrid, Spain, 2018, pp. 369-380. DOI: 10.1109/ICSME.2018.00045

[15] S. Palihawadana, C. H. Wijeweera, M. G. T. N. Sanjitha, V. Liyanage, I. Perera, D. Meedeniya, Tool support for traceability management of software artefacts with DevOps practices, in: Proceedings of the Moratuwa Engineering Research Conference, IEEE, 2017, pp. 129-134. DOI: 10.1109/MERCon.2017.7980469

[16] C. Trubiani, A. Ghabi, A. Egyed, Exploiting traceability uncertainty between software architectural models and extra-functional results, in: Journal of Systems and Software, Vol 125, March 2017, , 2017, pp.15-34. https://doi.org/10.1016/j.jss.2016.11.032

[17] A. D. Lucia, , F.Fasano, , R.Oliveto, , G.Tortora, Recovering traceability links in software artifact management systems using information retrieval methods, in: ACM Transactions on Software Engineering and Methodology, Vol 16, No4, 2007, pp.13-63. https://doi.org/10.1145/1276933.1276934

[18] M. Lormans, A. van Deursen, Can LSI help Reconstructing Requirements Traceability in Design and Test? In: Proceedings of the 10th European Conference on Software Maintenance and Reengineering, IEEE Computer Society, 2006, pp. 47-56. DOI**:** 10.1109/CSMR.2006.13

[19] A. Marcus, and J. I. Maletic, Recovering documentation-to-source-code traceability links using latent semantic indexing, in: Proceedings of the 25th International Conference on Software Engineering, IEEE Computer Society, Washington, USA, May, 2003, pp.125–135.

[20] M. Eyl, C. Reichmann, and K. Müller-Glaser, Traceability in a Fine Grained Software Configuration Management System, in: international conference on software quality, LNBIP 269, 2017, pp. 15–29. DOI: 10.1007/978-3-319-49421-0_2

[21] A. Shanmugathasan, S., Ratnavel, S., Thiyagarajah, V., Perera, I., Meedeniya, D., Balasubramaniam, D.: 'Support for traceability management of software artefacts using Natural Language Processing'. Moratuwa Engineering Research Conf., 2016. pp. 18-23.

[22] M. Grechanik, KS. McKinley, DE. Perry, Recovering and using use-case-diagram-to-source-code traceability links, in: Proceedings of the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering, September, 2007, pp.95–104, https://doi.org/10.1145/1287624.1287640

[23] A. Cockburn, *j. Highsmith*, Agile Software Development: The People Factor. EEE Computer, Volume 34, 2001, pp. 131-133.

[24] O. S. Dawood, A. E. K. Sahraoui, From Requirements Engineering to UML using Natural Language Processing – Survey Study. European Journal of Engineering Research and Science, 2, (1), January, 2017, pp. 44-50.

[25] K. Swathine, N. Sumathi, Study on Requirement Engineering and Traceability Techniques in Software Artefacts, in: international Journal of Innovative Research in Computer and Communication Engineering, Vol. 5, Issue 1, January 2017. DOI: 10.15680/IJIRCCE.2017. 0501016

[26] H. Kaiya, A. Hazeyama, S. Ogata, T. Okubo, N. Yoshioka, H. Washizaki,Towards A Knowledge Base for Software Developers to Choose Suitable Traceability Techniques, *in:* Proceedings of the 23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, *2019, pp. 1075-1084,* https://doi.org/10.1016/j.procs.2019.09.276

[27] H. Kaiya, R.Satoa, A.Hazeyamab, S.Ogatac, T.Okubod, T.Tanakae, N.Yoshiokaf, H. Washizakig, Preliminary Systematic Literature Review of Software and Systems Traceability, in: 2th International Conference on Knowledge Based and Intelligent Information and Engineering of the 10th European Conference on Software Maintenance and Reengineering, IEEE Computer Society, 2006, pp. 47-56. DOI**:** 10.1109/CSMR.2006.13

[28] C. Trubiani, A. Ghabi, A. Egyed, Exploiting traceability uncertainty between software architectural models and extra-functional results, in: Journal of Systems and Software, Vol 125, March 2017, , 2017, pp.15-34. https://doi.org/10.1016/j.jss.2016.11.032

[29] S. Maro, A. Anjorin, R. Wohlrab, J.P. Steghöfer*:* Traceability maintenance: factors and guidelines, *in:* Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering, Singapore, September 3-7, 2016, *pp. 414-425.*

[30] S. Maro, J.P. Steghöfer, M. Staron, Software traceability in the automotive domain: Challenges and solutions, i*n: Journal of Systems and Software, Vol 141, 2018, pp.85-110.* https://doi.org/10.1016/j.jss.2018.03.060

[31] R.Wohlrab, J.-P. Steghöfer, E. Knauss, S. Maro, A. Anjorin: Collaborative Traceability Management: Challenges and Opportunities, in: 24th IEEE International Requirements Engineering Conference, Beijing, China, September 12-16, 2016, pp. 216-225. DOI: 10.1109/RE.2016.1

[32] M., Broy, A logical approach to systems engineering artifacts: semantic relationships and dependencies beyond traceability - from requirements to functional and architectural views, in: journal of Software and Systems Modeling, pp.365-393, Vol.17, Issue 2, 2018, pp. 365-393. https://doi.org/10.1007/s10270-017-0619-4

[33] Yazawa, Y. , Ogata, S., Okano, K., Kaiya, H., Washizaki, H.: 'Traceability Link Mining - Focusing on Usability'.41st IEEE Annual Computer Software and Applications Conference, Italy, 2, 2017, pp 286-287**.**

[34] K.S. Divya, R. Subha, , S. Palaniswami, Similar words identification using naive and tf-idf method'. Information Technology and Computer Science Journal, pp. 42-47, 2014.

[35] Kothari, P.R.: 'Processing Natural Language Requirement to Extract Basic Elements of a Class'. Journal of Applied Information Systems, USA 3, (7), 2012, pp. 39-42.

[36] A. T. Imam, A. A. Hroob , R. A. Heisa, The use of artificial neural networks for extracting actions and actors from requirements document'. journal of Information and Software Technology, 2018, pp.1-15.

[37] Rath, M., Rendall, J., Guo, J. L.C., Cleland-Huang, J., Mader, P., 'Traceability in the Wild: Automatically Augmenting Incomplete Trace Links'. IConf on Software Engineering, May 27-June 3, Sweden, 2018, pp. 834–845.

[38] L. G. P. Murta, A. van der Hoek, C. M. L.Werner, Archtrace: policy-based support for managing evolving architecture-to implementation traceability links, in: 21st IEEE/ACM International Conference on Automated Software Engineering, Tokyo, Japan, 2006, pp. 135–144. DOI: 10.1109/ASE.2006.16

[39] L. G. P. Murta, A. van der Hoek, C. M. L.Werner, Continuous and automated evolution of architecture-to-implementation traceability links, in: Automated Software Engineering Journal, vol. 15, no. 1, 2008, pp. 75–107. https://doi.org/10.1007/s10515-007-0020-6

[40] I. D. D. Rubasinghe, A. Meedeniya, I. Perera, Towards TraceabilityManagement in Continuous Integration with SAT Analyser, in: Proceedings of the 3rd International Conference on Communication, and Information Processing, 2017, ACM, Tokyo. DOI: 10.1145/3162957.3162985

[41] I, Pete, D., Balasubramaniam, Handling the Differential Evolution of Software Artefacts A Framework for Consistency Management, in: 22nd IEEE International Conference on Software Analysis Evolution and Reengineering, 2015, pp.599-600, doi: 10.1109/SANER.2015.7081889

[42] M. Grechanik, KS. McKinley, DE. Perry, Recovering and using use-case-diagram-to-source-code traceability links, in: Proceedings of the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering, September,

2007, pp.95–104, https://doi.org/10.1145/1287624.1287640

[43] Guo, J., Cheng, J. Cleland-Huang, J.: 'Semantically Enhanced Software Traceability Using Deep Learning Techniques'. Conf on Software Engineering, May 2017, pp. 3-14.

[44] Kuang, H., Nie, J., Hu, H., Rempel, P., Lü, J., Egyed, A., Mäder, P. : 'Analyzing Closeness of Code Dependencies for Improving IR-Based Traceability Recovery'. Software Analysis Evolution & Reengineering, 2017, pp. 68-78.

[45] Kuang, H., Gao, H., Hu, H., Ma, X. , Lü, J., Mäder, P. , Egyed, A.: 'Using Frugal User Feedback with Closeness Analysis on Code to Improve IR-Based Traceability Recovery'. IEEE/ACM 27th Inter. Conf. on Program Comprehension, pp. 369-379, 2019.

[46] I. D. D. Rubasinghe, A. Meedeniya, I. Perera, Towards TraceabilityManagement in Continuous Integration with SAT Analyser, in: Proceedings of the 3rd International Conference on Communication, and Information Processing, 2017, ACM, Tokyo. DOI: 10.1145/3162957.3162985

[47] I. D. D. Rubasinghe, A. Meedeniya, I. Perera, Software Artefact Traceability Analyser: A Case-Study on POS System, in: Proceedings of the 6th International Conference on Communications and Broadband Networking, February 24 - 26, 2018, pp.1-5, DOI:10.1145/3193092.3193094

[48] H. Tufail, M. F. Masood, B. Zeb, F. Azam, A Systematic Review of Requirement Traceability Techniques and Tools, in: 2nd International Conference on System Reliability and Safety (ICSRS), 20-22 December, Milan, Italy, 2017, DOI: 10.1109/ICSRS.2017.8272863.

[49] O. Rahmaoui, K.Souali, M. Ouzzif, Improving Software Development Process using Data Traceability Management, *in:* international Journal of Recent Contributions from Engineering, Science & IT, 2019, pp.52-58. https://doi.org/10.3991/ijes.v7i1.10113.

[50] K. Souali, *O.* Rahmaoui, *M.* Ouzzif, An overview of traceability: Definitions and techniques. 4th IEEE Colloquium on Information Science and Technology, Morocco, October, 2016, *pp.789-793.*

[51] *C.D* Manning, *M.* Surdeanu, *J.* Bauer, *J.R* Jenny Rose Finkel, *S.* Bethard, *D.* McClosk, The Stanford CoreNLP Natural Language Processing Toolkit. *In* Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, June 22-27, *2014*, pp.*55-60.*

[52] J.J. Webster, C. Kit, Tokenization as the initial phase in nlp. In *Proceedings of the 14th conference on Computational linguistics*, Association for Computational Linguistics, Volume 4, 1992, pp. 1106-1110.

[53] H. Saif, M. Fernandez, Y. He, H. Alani, On stopwords, filtering and data sparsity for sentiment analysis of twitter'. In LREC'14 Proceedings of the Ninth International Conference on Language Resources and Evaluation, European Language Resources Association, Reykjavik, Iceland, May 26-31, 2014, pp. 810-817.

[54] J.B. Lovins, Development of a stemming algorithm. In Mechanical Translation and Computational Linguistics, Vol 11, No.1-2, March, June, 1968, pp. 22-31.

[55] OMG-UML :OMG-UML, 2015. *OMG Unified Modeling Language (OMG UML*). formal/2015-03-01. [Online].

[56] D. Bailey, Java Structures: Data Structures in Java for the Principled Programmer, 2end edition, (2007) pp. 528, McGraw-Hill Science/Engineering/Math.

[57] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, A. Wesslén 'Experimentation in Software Engineering: An Introduction, 2000.

[58] N. Mustafa, Y. Labiche, D. Towey, Mitigating Threats to Validity in Empirical Software Engineering: A Traceability Case Study. 43rd Annual Computer Software and Applications Conference, USA, July, 2019, pp. 324-329.

[59] Eclipse Specification. 2011, Available from: http://www.eclipse.org/

[60] L. Frias, A. Queralt, A. Oliv, EU-Rent car rentals specification. Technical report, 2003.

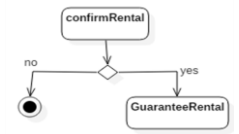
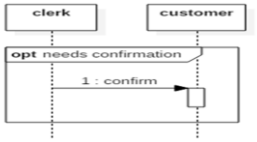
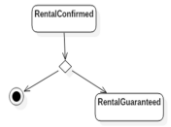
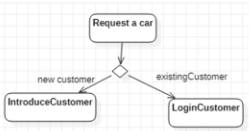
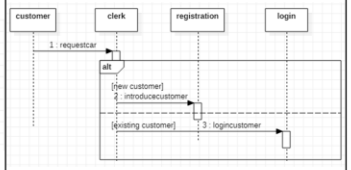
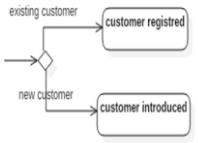
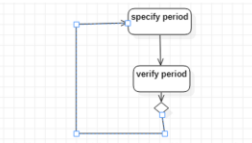
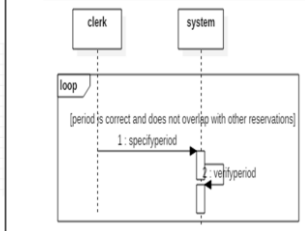
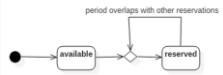
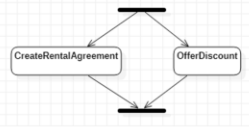
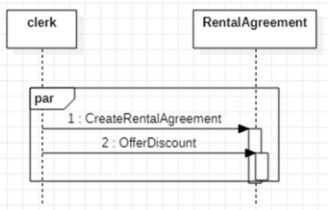
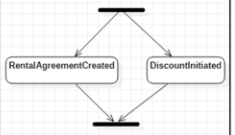
	USE CASE DIAGRAM	ACTIVITY DIAGRAM	SEQUENCE DIAGRAM	STATE TRANSITION DIAGRAM	Code
OPT	<Opt><needs confirmation> <NumAction 6> < From Actor The Clerk ><To Actor Customer><Type of Result Simple: Boolean > <Action Description Asks the customer if he/she wants to guarantee the reservation> <In-Parameter Reservation> <Out-Parameter IsValidated >< IsConsidered 1>< IsIgnored 0>< IsNegative 0> <restart at Num "6" in SN>				`If (rentalconfirmed)` `{ guaranteerental();`
ALT	Begin <Event, begin at Num 2 in SN> <IF>< the customer does not exists> <NumAction 1> < From Actor The Clerk >< To Actor The customer>< Type of Result Customer (name, ID, birthdate, address, telephone)>< Action Description Introduce a new costumer> <In-Parameter Customer (name, ID, birthdate, address, telephone)> <Out-Parameter Customer >< IsConsidered 1>< IsIgnored 0>< IsNegative 0> <Else > <restart at num 3 in SN>				`String username, pword;` `if (userObj.getType()` `!=UserType.MEMBER)` `{ introducecustumer();` `else` `Confirmlogin();}`
Do while	<begin at Num 3 in SN> <Do> <NumAction 1>< From Actor the clerk ><To Actor the reservation><Type of Result Simple>< Action Description Specifies the period ><In-Parameter period > <Out-Parameter period >< IsConsidered 1>< IsIgnored 0>< IsNegative 0> <NumAction 2> < From Actor The clerk><To Actor The reservation><Type of Result correct yes/no>< Action Description Verify the period>] [<In-Parameter period>] [<Out-Parameter correct yes/no >]< IsConsidered **1**>< IsIgnored **0**>< IsNegative **0**> <While><period is correct and does not overlaps with other reservations > <restart at Num "6" in SN>				`Do` `Specifyperiod();` `Verifyperiod();` `While (period is correct and does not overlaps with other reservations)`
PARALLEL	<Parallel> <NumAction 6>< From Actor The Clerk >< To Actor The agreement> < Type of Result Entity>< Action Description Create the reservation agreement ><In-Parameter rental agreement: ID customer, price, ID reservation> <Out-Parameter Rental agreement >< IsConsidered 1>< IsIgnored 0>< IsNegative 0> <NumAction 7> < From Actor The Clerk >< To Actor The agreement>< Type of Result Entity>< Action Description Offer a discount to the customer ><In-Parameter Discount rental agreement : Discount,ID agreement, ID cusotmer> <Out-Parameter Discount rental agreement >< IsConsidered 1>< IsIgnored 0>< IsNegative 0>				`public class rentalagreementThread` `implements Runnable {` `Thread rentalagreementThread;` `Public rentalagreementThread (int agreement, int IDloc, int IDMAT, date dat)` `{this.agreement= agreement;` `This.idloc=IDloc;` `This.IDMAT;` `This.dat=dat;}` `rentalagreementThread = new Thread ( this , " OfferDiscountthread");` `UnThread.start(); }` `public void run ( ) {` `//....second thread actions here` `public OfferPointsPayment(); }}`

Table 9: Traceability between control structure fragments in the use case, Activity, sequence state transition diagrams and code

A Global COVID-19 Observatory, Monitoring the Pandemics Through Text Mining and Visualization

M. Besher Massri[1,2], Joao Pita Costa, Marko Grobelnik, Janez Brank, Luka Stopar and Andrej Bauer[3]
E-mail: besher.massri@ijs.si, andrej.bauer@andrej.com
[1]Jožef Stefan Institute, Slovenia
[2]Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[3]University of Ljubljana, Slovenia

Keywords: text mining, data analytics, data visualisation, public health, Coronavirus, COVID-19, epidemic intelligence

Received: November 23, 2020

The global health situation due to the SARS-COV-2 pandemic motivated an unprecedented contribution of science and technology from companies and communities all over the world to fight COVID-19. In this paper, we present the impactful role of text mining and data analytics, exposed publicly through IRCAI's Coronavirus Watch portal. We will discuss the available technology and methodology, as well as the ongoing research based on the collected data.

Povzetek: Opisana je vloga rudarjenja besedil in podatkovne analitike na primeru portala IRCAI's Coronavirus Watch.

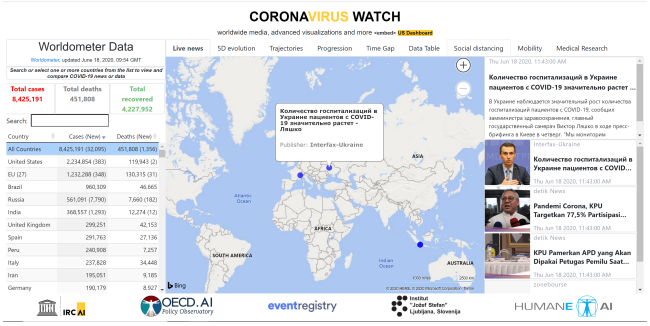


Figure 1: Coronavirus Watch portal

1 Introduction

When the World Health Organization (WHO) announced the global COVID-19 pandemic on March 11th 2020 [36], following the rising incidence of the SARS-COV-2 in Europe, the world started reading and talking about the new Coronavirus. The arrival of the epidemic to Europe scaled out the news published about the topic, while public health institutions and governmental agencies had to look for existing reliable solutions that could help them plan their actions and the consequences of these. Technological companies and scientific communities invested efforts in making available tools (e.g. the GIS [2] later adopted by the World Health Organisation (WHO)), challenges (e.g. the Kaggle COVID-19 competition [22]), and scientific reports and data (e.g. the repositories medRxiv [24] and Zenodo [38]). In this paper we discuss the Coronavirus Watch portal [21], made available by the UNESCO AI Research Institute (IRCAI), comprehending several data exploration dashboards related to the SARS-COV-2 worldwide pandemic (see the main portal in Figure 1). This platform aims to expose the different perspectives on the data generated and trigger actions that can contribute to a better understanding of the behavior of the disease.

2 Related work

With with growing dimension of the public health problem caused by the SARS-COV-2, the global effort multiplied worldwide: from the publicly shared research in, e.g., Zenodo [38] medRxiv [24] and IEEE Xplore [20]; to a diversity of datasets made available by the global agencies (as e.g., the European Commission [3] and the ECDC [6]), the tech giants (as, e.g., Google [7] and Facebook [15]), institutional initiatives (as, e.g., the OxCOVID19 project [34]), and private initiatives acting globally (e.g. [37]) and locally (e.g., the slovenian initiative [32]). With similar aim to contribute for the global effort, many platforms have been made publicly available over the internet to monitor aspects of the COVID-19 pandemics are mostly focusing on data visualization based on the incidence of the disease and the death rate worldwide (e.g., the CoronaTracker [4]). The limitations of the available tools are potentially due to the lack of resolution of the data in aspects like the geographic location of reported cases, the commodities (i.e., other diseases that also influence the death of the patient), the frequency of the data, etc. On the other hand, it was not common to monitor the epidemic through the worldwide news (with some exceptions as, e.g., the Ravenpack Coronavirus News Monitor [31]). This is a rather important perspective mostly due to the great importance that the related misinformation - known as *infodemia*[1] - has been playing a role in the context of this pandemics.

The Coronavirus Watch portal suggests the association of reported incidence with worldwide published news per country, which allows for real-time analysis of the epidemic situation and its impact on public health (in which specific topics like mental health and diabetes are important related matters) but also in other domains (such as economy, social inequalities, etc.). This news monitoring is based on state-of-the-art text mining technology aligned with the validation of domain experts that ensures the relevance of the customized stream of collected news.

Moreover, the Coronavirus Watch portal offers the user other perspectives of the epidemic monitoring, such as the insights from the published biomedical research that will help the user to better understand the disease and its impact on other health conditions. While related work was promoted in [22] in relation with the COVID-19, and is offered in general by MEDLINE mining tools (e.g., MeSH Now [25]), there seems to be no dedicated tool to the monitoring and mining of COVID-19 - related research as that presented here.

3 Description of data

The nature of the proposed global observatory system entails a range of heterogeneous data sources that entangle to provide a perspective as complete as possible on the matters of the COVID-19 pandemics. In the following section we discuss these data sources and their potential contributions.

3.1 Historical COVID-19 data

To perform an analysis of the growth of the coronavirus, we use the historical data on the daily number of cases of infections and deaths. This data is retrieved from a GitHub repository made available by the John Hopkins University [5]. The data source is based mainly on the official data from the World Health Organization (WHO)[28] along with some other sources provided by, e.g., the Center for Disease and Control[17], and Worldometer[37], among others. This data provides the basis for all the system's functionality that depended on the statistical information about SARS-COV-2 incidence.

3.2 Live data from worldometer

Apart from historical data, live data about the COVID-19 number of infection cases, deaths, recovered individuals, and tests made is retrieved from the Worldometer web portal [37]. Although the information might not be taken as official as the one provided by John Hopkins University (which is based on WHO data), this source is updated many times per day providing the latest up-to-date data about COVID-19 statistics at all times. Thus its usefulness in providing the system with an almost real-time perspective on the pandemics worldwide.

3.3 Live news about coronavirus

The live news is retrieved from the global news engine Event Registry [16], which is a media-intelligence platform that collects news media from around the world in many languages. The service analyzes news from more than 30 thousand news, blog posts, and press releases daily, over around 75 thousand news sources in more than 60 languages. The multilingual capabilities of the system allow for the identification of topics across languages, ensuring a global coverage with local granularity [18]. With this data we are able to access what is the awareness of the media on aspects and key figures in the pandemics.

3.4 Google COVID-19 community mobility data

Google's Community Mobility [19] data compares mobility patterns that date from before the COVID-19 crisis, describing the situation on a weekly basis. Mobility patterns are measured as changes in the frequency of visits to six location types: retail and recreation; grocery and pharmacy; parks; transit stations; workplaces; and residential areas. The data is provided on a country level as well as on a province level. These reports are available for a limited amount of time, limited to their usefulness in supporting the work of public health officials. This data provides us with a wide perspective on the exchanges with potential of contamination based on mobility.

3.5 MEDLINE: medical research open dataset

In 2020 the MEDLINE dataset [23] contains more than 30 million citations and abstracts of the biomedical literature dating back to 1966. Over the past ten years, an average of a million articles were added each year. Around 5% of MEDLINE is on published research on infections, with cancer research being the most prevalent occuping 12% of this body of knowledge. Most scientific articles in this dataset are hand-annotated by health experts using 16 major categories and a maximum of 13 levels of deepness. The labeled articles are hand-annotated by humans based on their main and complementary topics, and on the chemical substances that they relate to. It is widely used by the biomedical research community through the well-accepted search engine PubMed [29]. The richness of this data allows us for insight on aspects of the disease as well as approaches and best practices from the published research.

4 Coronavirus watch dashboard

The main layout of the dashboard displayed in figure 1 consists of two sides. On the left side, the dashboard is split into the summarized information on the incidence across countries, where a simple table of statistics is provided about countries along with the total numbers of cases, death

cases, and recovered cases. On the right side, there is a navigation panel with tabs, each representing a functionality. Each of these tabs is a view on the pandemics, based on its own specific data sources, and answers some questions and provides insights about a certain type of data focusing a specific aspect.

4.1 Coronavirus data table

The data table functionality is a straightforward data visualisation that shows the basic statistics about the new coronavirus. It's sourced from Worldometer as it's the most frequently updated source for coronavirus. This data table extends the one that is consistently shown in most views, a summarized version, to the table on the left containing the full information details. These include the new infection cases and the new death cases, in the day of the visit to the portal, as well as the total number of recovered cases, the active cases and the critical cases. The cumulative counts include the total number of infected, dead and tests. The latter are provided in absolute and per one million capita, taking into consideration the different population per country.

4.2 Coronavirus live news

The news monitoring functionality offers a live news feed about coronavirus-related topics from sources around the world. The feed is provided by the multilingual news engine Event Registry over an API. The ingested news article sample is generated by querying the system for articles that are annotated with concepts and keywords related to the coronavirus. The user can check for a country's specific news (based on the news source in that country) by clicking on the country name on the left table, as seen in figure 1.

This feature is a differentiator to most COVID-19 observatories, allowing the visitor to access real-time information about the pandemics, either at a global scale, in the European context or at country level. It also allows for a perspective on the public awareness on aspects of the pandemics.

4.3 Exploratory analytics

The following set of data visualization modules aim at displaying the statistical data about COVID-19 cases and deaths. While they all provide countries comparison, each one focus on a different perspective. Some are more complex and focused on the big picture (5D evolution), and some others are simple and focus on a unique aspect (Progression and Trajectory). Moreover, all of them have configuration options to tweak the visualization, like the ability to change the scale of the axes to focus on the top countries, or to focus on the long tale. Some offer a slider to manually move through the days for further inspection. Furthermore, the default view compares all the countries or the

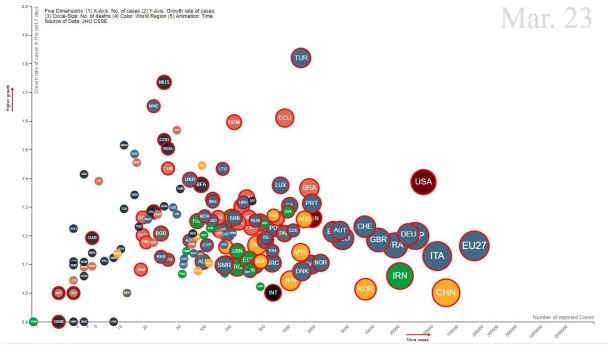


Figure 2: A snapshot of the 5D Visualization on March 23rd. Countries that were at the peak in terms of growth are shown high up.

top N countries, depending on the visualization selected. However, it is possible to track a single country or a set of countries and compare them together for a more specific view. This is done when selecting the main country by clicking on it on the left table and proceeding to select more countries by pressing the ctrl key while clicking on the additional country.

4.3.1 5D evolution

5D Evolution is a visualization that displays the incidence of the virus in the population through time. It is named 5D since it encompasses five dimensions: x-axis, y-axis, bubble size, bubble color, and time, as seen in figure 2. By default, it illustrates the evolution of the pandemic in countries based on N cases (x-axis). The growth factor of N Cases (y-axis), N Deaths (bubble size), and country region (bubble color) through time. In addition, a red ring around the country bubble is drawn whenever the first death appears. The growth rate represents how likely the numbers are increasing with respect to the day before. A growth rate of 2 means that the numbers are likely to double in the next day. The growth rate is calculated using the exponential regression model. At each day the growth rate is based on the N cases from the previous seven days. The goal of this visualization is to show how countries relate to each other, which are "exploding in numbers" and which ones managed to "flatten the curve" (since flattening the curve means less growth rate). It is intended to be one visualization that gives the user a big picture of the situation.

4.3.2 Progression

The progression visualization displays the line graph comparing Date vs N cases/deaths. It helps to provide a simplistic view of the situation and compare countries based on the raw numbers only. The user can display the cumulative numbers where each day represents the current counts, or daily where at each date we count the cases/deaths on that day only.

4.3.3 Trajectory

While the progress visualization displays the normal date vs N cases/deaths, this visualization seeks to compare how the trajectory of the countries differ starting from the point where they detect cases. This visualization helps to compare countries' situations if they all start having cases on the same date. The starting point has been set to the day the country reaches 100 cases, so we would compare countries when they started gaining momentum.

4.4 Time gap

The time gap functionality tries to estimate how the countries are aligned and how many days each country is behind the other, whether that is in the number of cases or deaths. This assumes that the trajectory of the country will continue as is without taking much more strict/loose measurements, which is a rough assumption. It helps to estimate how bad or good is the situation in terms of the number of days. To see the comparison, a country has to be selected from the table on the left. However, not all countries are comparable, having very different trajectories or growth rates.

The growth of each country is represented as an exponential function, the base is calculated using linear regression on the log of the historical values (that is, exponential regression). Based on that, the duplication N days, or the N days representing the number of cases/deaths will double is determined. Two countries are comparable if they have a reasonable difference in the base or doubling factor. If they are comparable, we see where the country with the smaller value fits in the historical values of the country with the larger numbers. We use linear interpolation if the number is not exact, hence the decimal values.

4.5 Mobility

The mobility visualization is based on google community mobility data that describe how communities in each country are moving based on 6 parameters: retail and recreation, grocery and pharmacy, parks, transit stations, workplaces, and residential areas. The data is then reduced to 2-dimensional data while keeping the Euclidean proximity nearly the same. The visualization can indicate that the closer the countries are on the visualization, the similar the mobility patterns they have. The visualization uses the T-SNE algorithm for dimensionality reduction [35], which reduces high dimensional data to low dimensional one while keeping the distance proximity between them proportionally the same as possible. The algorithm works in the form of iterations, at each iteration, the bubbles representing the country are drawn. We used those iterations to provide animation to the visualization.

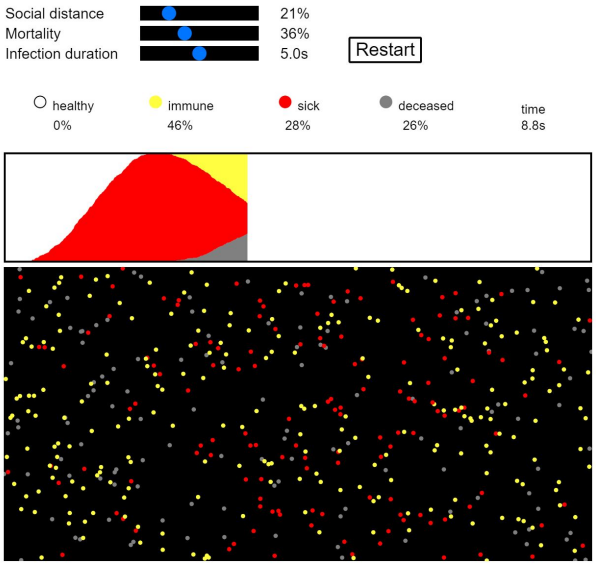


Figure 3: A snapshot of the Social Distancing Simulator. The canvas show a representation of the population. with red dots representing sick people, yellow dots representing immunized people, and grey dots represent deceased people.

4.6 Social distancing simulator

The Social Distancing simulator is displayed in figure 3. Each circle represents a person who can be either healthy (white), immune (yellow), infected (red), or deceased (gray). A healthy person is infected when they collide with an infected person. After a period of infection, a person either dies or becomes permanently immune. Thus the simulation follows the Susceptible-Infectious-Recovered-Deceased (SIRD) compartmental epidemiological model.

The simulator is controlled by three parameters. Firstly, the Social distancing that controls to what extent the population enforces social distancing. At 0% there is no social distancing and persons move with maximum speed so that there is a great deal of contact between them. At 100% everyone remains still and there is no contact at all. Secondly, mortality is the probability that a sick person dies. If you set mortality to 0% nobody dies, while the mortality of 100% means that anybody who catches the infection will die. Finally, the infection duration determines how long a person is infected. A longer time gives an infected person more opportunities to spread the infection. Since the simulation runs at high speed, time is measured in seconds.

4.7 Biomedical research explorer

To better understand the disease, the published biomedical science is the source that provides accurate and validated information. Taking into consideration a large amount of published science and the obstacles to access scientific information, we made available a MEDLINE explorer where the user can query the system and interact with a

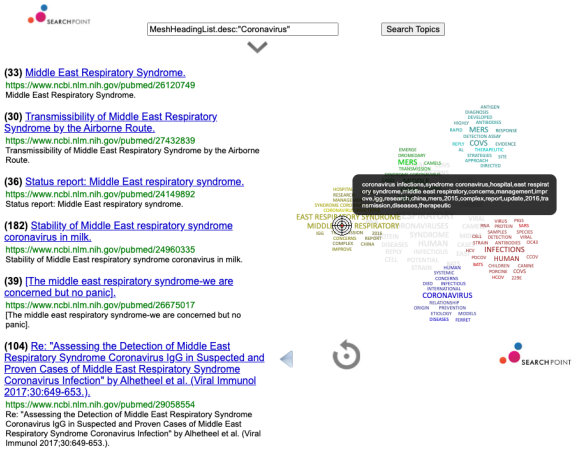


Figure 4: The interactive MEDLINE data exploration showing a scientific article originally positioned as 182nd, now in the 4th place.

pointer to specify the search results (e.g., obtaining results on biomarkers when searching for articles hand-annotated with the MeSH class "Coronavirus").

To allow for the exploration of any health-related texts (such as scientific reports or news) we developed an automated classifier [8] that assigns to the input text the MeSH classes it relates to. The annotated text is then stored in Elasticsearch [27], from where it can be accessed through Lucene language queries, visualized over easy-to-build dashboards, and connected through an API to the earlier described explorer (see figure 4 and read [12], [30] and [26] for more detail).

The integration of the MeSH classifier with the worldwide news explorer Event Registry allows us to use MeSH classes in the queries over worldwide news promoting an integrated health news monitoring [13] and trying to avoid bias in this context [11]. An obvious limitation is a fact that the annotation is only available for news written in the English language, being the unique language in MEDLINE.

5 Conclusion and future work

In this paper, we presented the coronavirus watch dashboard as a use-case of observing pandemic. However, this methodology can be applied to other kinds of diseases given the availability of similar data. Having this global system in place we are collecting data on the usual numbers accounted on the pandemics by other platforms (as in e.g., Worldometer [37] or ECDC [6]) but within an integrated approach (as e.g. in the OxCOVID19 project [34]) aiming for data interoperability enhancing the business intelligence generated by the opservatory For further development, we plan to implement a local dashboard for other countries as well, which would provide local data in the local language. This would provide the local coverage (as in, e.g., Slednik [32]) that can bring the global problems addressed to a specific and more usefu context (as was initiated in the context of the MIDAS project [12]). In addition, given the existence of more than seven months of historical data, we would like to build some predictive models to predict the number of cases/deaths in the next days.

Moreover, we are using the StreamStory technology [33] [14] in order to: (i) compare the evolution of the disease between countries by comparing their time-series of incidence; (ii) investigate the correlation between the incidence of the disease with weather conditions and other impact factors; and (iii) analyze the dynamics of the evolution of the disease based on incidence, morbidity, and recovery. This technology allows for the analysis of dynamical Markov processes, analyzing simultaneous time-series through transitions between states, offering several customization options and data visualization modules.

Furthermore, following the work done in the context of the Influenza epidemic in [9], we are using Topological Data Analysis methods in [10] to understand the behavior of COVID-19 in the European context [10] through the behaviour of the data it is reflected in. This is done analysing simultaneously a variety of time-series representing the several aspects of the pandemics, in a high dimension space. In it, we examine the structure of data through its topological structure, which allows for comparison of the evolution of the epidemics within countries through the encoded topology of their incidence time series.

Acknowledgement

The first author has been supported by the Knowledge 4 All foundation and the H2020 Humane AI project under the European research and innovation programme under GA No. 761758), while the second author was funded by the European Union research fund 'Big Data Supporting Public Health Policies', under GA No. 727721. The third author acknowledges that this material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-17-1-0326.

References

[1] H. Allahverdipour. 2020. Global challenge of health communication: infodemia in the coronavirus disease (covid-19) pandemic. *J Educ Community Health*, 7, 2, 65.–67.

[2] ArcGIS. 2020. ArcGIS who covid-19 dashboard. `https://covid19.who.int/`. (2020).

[3] European Commission. [n. d.] The european covid-19 data portal. `https://www.covid19dataportal.org/`. Accessed in: April 2021. ().

[4] CoronaTracker. 2020. CoronaTracker. `https://www.coronatracker.com/analytics/`. (2020).

[5] CSSE. 2020. Covid-19 data repository by the center for systems science and engineering (csse) at johns hopkins university. https://github.com/CSSEGISandData/COVID-19. (2020).

[6] ECDC. [n. d.] Ecdc covid-19 datasets. https://www.ecdc.europa.eu/en/covid-19/data. Accessed in: April 2021. ().

[7] B. Xu et al. 2020. Epidemiological data from the covid-19 outbreak, real-time case information. *Scientific data*, 7, 1, 1–6. DOI: 10.1038/s41597-020-0448-0.

[8] J. Pita Costa et al. 2021. A new classifier designed to annotate health-related news with mesh headings. *Artificial Intelligence in Medicine*, 114, 102053. DOI: 10.1016/j.artmed.2021.102053.

[9] J. Pita Costa et al. 2019. A topological data analysis approach to the epidemiology of influenza. In *Proceedings of the Slovenian KDD conference*.

[10] J. Pita Costa et al. 2020. A topological data analysis perspective on the covid-19 pandemics. In *In preparation*.

[11] J. Pita Costa et al. 2019. Health news bias and its impact in public health. In *Proceedings of the Slovenian KDD conference*.

[12] J. Pita Costa et al. 2020. Meaningful big data integration for a global covid-19 strategy. *Computer Intelligence Magazine*, 15, 4, 51–61. DOI: 10.1109/MCI.2020.3019898.

[13] J. Pita Costa et al. 2017. Text mining open datasets to support public health. In *WITS 2017 Conference Proceedings*.

[14] L Stopar et al. 2018. Streamstory: exploring multivariate time series on multiple scales. *IEEE transactions on visualization and computer graphics*, 25, 4, 1788–1802. DOI: 10.1109/TVCG.2018.2825424.

[15] T. Kuchler et al. 2020. The geographic spread of covid-19 correlates with the structure of social networks as measured by facebook. *National Bureau of Economic Research*, w26990. DOI: 10.1016/j.jue.2020.103314.

[16] EventRegistry. 2020. Event Registry. https://eventregistry.org. (2020).

[17] Center for Disease Control. 2020. Center for Disease Control and Prevention. https://www.cdc.gov/coronavirus/2019-ncov/index.html. (2020).

[18] J. Brank G. Leban B. Fortuna and M. Grobelnik. 2014. Event registry: learning about world events from news. In *In Proceedings of the 23rd International Conference on World Wide Web*, 107–110. DOI: 10.1145/2567948.2577024.

[19] Google. 2020. Google COVID-19 Community Mobility Report. https://www.google.com/covid19/mobility/. (2020).

[20] IEEE. [n. d.] Ieee xplore covid-19 resources. https://ieeexplore.ieee.org/Xplore/home.jsp. Accessed in: April 2021. ().

[21] IRCAI. 2020. IRCAI coronavirus watch portal. http://coronaviruswatch.ircai.org/. (2020).

[22] Kaggle. 2020. Kaggle covid-19 open research dataset challenge. https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge. (2020).

[23] MEDLINE. 2020. MEDLINE description of the database. https://www.nlm.nih.gov/bsd/medline.html. (2020).

[24] medRxiv. [n. d.] Covid-19 sars-cov-2 preprints from medrxiv and biorxiv. https://connect.medrxiv.org/relate/content/181. Accessed in: April 2021. ().

[25] MeSHNow. 2020. MeSHNow. https://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/MeSHNow/. (2020).

[26] MIDAS. 2020. MIDAS COVID-19 portal. http://www.midasproject.eu/covid-19/. (2020).

[27] Elastic NV. 2020. Elasticsearch portal. https://www.elastic.co/. (2020).

[28] World Health Organisation. 2020. WHO Coronavirus portal. https://www.who.int/emergencies/diseases/novel-coronavirus-2019. (2020).

[29] PubMed. 2020. PubMed biomedical search engine. https://pubmed.ncbi.nlm.nih.gov/. (2020).

[30] Quintelligence. 2020. Quintelligence COVID-19 portal. http://midas.quintelligence.com/. (2020).

[31] Ravenpack. 2020. Ravenpack coronavirus news monitor. https://coronavirus.ravenpack.com/. (2020).

[32] Slednik. [n. d.] Covid-19 sledilnik slovenija. https://www.sledilnik.org. Accessed in: April 2021. ().

[33] Luka Stopar. 2020. StreamStory. http://streamstory.ijs.si/. (2020).

[34] Oxford University. [n. d.] Oxford covid-19 (oxcovid19) project. https://covid19.eng.ox.ac.uk/. Accessed in: April 2021. ().

[35] Laurens van der Maaten and Geoffrey Hinton. 2008. Viualizing data using t-sne. *Journal of Machine Learning Research*, 9, (November 2008), 2579–2605.

[36] WHO. 2020. World Health Organization who director-general's opening remarks at the media briefing on covid-19 - 11 march 2020. https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020. (2020).

[37] WorldoMeters. 2020. WorldoMeters. https://www.worldometers.info/coronavirus/. (2020).

[38] Zenodo. [n. d.] Zenodo coronavirus disease research community - covid-19. https://zenodo.org/communities/covid-19/. Accessed in: April 2021. ().

A Novel Fuzzy Modifier Interpolation Rule for Computing with Words

Tanvi Dadu, Swati Aggarwal* and Nisha Aggarwal
E-mail: tanvid.co.16@nsit.net.in, swati1178@gmail.com, nishaaggarwal2810@gmail.com
Netaji Subhas University of Technology, Dwarka, New Delhi 110078, India

Keywords: type-1 fuzzy sets, interval type 2 fuzzy sets, fuzzy modifiers, interpolation rule, computing with words

Received: June 15, 2021

Computing with words is a concept that is used to solve problems with input in natural language. Modifiers are transformation functions with predefined labels used extensively in decision-making to specify the desired value of a linguistic variable defined by fuzzy sets.
In past years, few efforts have been made to study the application of Computing with Words (CW) in many domains ranging from fraud detection systems to diagnosis systems in medicine. However, the application of CW in these fields with modified Fuzzy sets did not give satisfactory results. When applied to modified Fuzzy sets, the existing interpolation rule does not cover the extreme left and extreme right-shifted fuzzy sets. Hence, there is a need to introduce a new interpolation rule when working with modifiers. This paper introduces a new Fuzzy Modifier Interpolation Rule to Type-1 Fuzzy sets and Interval Type-2 (IT-2) Fuzzy Sets to enhance the quality of results obtained when modifiers are applied.

Povzetek: V prispevku je predstavljeno novo interpolacijsko pravilo za mehke modifikatorje v mehkih nizih tipa 1 in intervalnega tipa 2.

1 Introduction

Computing with Words (CW) is a methodology that allows using words in place of numbers for computing and reasoning. It was introduced as an extension to Fuzzy Logic Systems by Zadeh [1][2][3]. It is a system of computation that offers the capability to compute with information present in a natural language. The advancement in CW has allowed a certain degree of fuzziness to the input and the propositions present in the database of the CW inference engine. The propositions in CW are specified using linguistic variables [4][5][6], whose values are words in natural language. CW engine consists of Rule Base, Fuzzy Inference Engine, and Output Generator. The IF-THEN rules in the CW engine are specified using natural language, which is modeled using either Type - 1 or Type - 2 Fuzzy sets (Interval Type-2 and General Type-2) [7].

Computing with Words (CW) has a wide variety of applications ranging from household appliances like fraud detection systems to biomedical instrumentation [8]. Medicine is one of the domains where the applications of Computing with Words has been recognized [8]. The uncertainty found in these applications is appropriately captured by fuzzy set theory. Therefore, in the past few years, new techniques in CW have been extensively applied in decision-making systems.

Modifiers are transformation functions with predefined labels applied to a fuzzy set defined on a linguistic variable to specify the desired value of a variable. Modifiers help us define input when it is not present in the given fuzzy sets. Modifiers like VERY, EXTREMELY, and connective like NOT, OR can be used to enhance the capability of the meaning specified by Fuzzy sets [9][10][11]. However, the direct application of the Fuzzy Interpolation rule on the modified fuzzy set gives poor results. While computing results, it does not incorporate the contribution from fuzzy sets shifted to the extreme left or extreme right on the application of modifiers as detailed in Section 4.

This paper introduces a novel Fuzzy Modifier Interpolation rule for Type-1 Fuzzy sets and Interval Type-2 Fuzzy sets to get better results and reduce errors when modifiers are applied to the fuzzy sets. In Section 2, we present recent relevant works followed by the proposed Fuzzy Modifier Interpolation rule for Type-1 and Interval Type-2 Fuzzy sets in Section 3. In Section 4, we present results of experiments performed on the UCLA dataset for heart disease [27] to show improvement in graphs or inference obtained upon application of our new Fuzzy Modifier Interpolation Rule. In Section 5, we discuss the results obtained for our proposed Fuzzy Modifier Interpolation rule with Fuzzy Interpolation rule. Finally, we discuss the advantages and limitations of our proposed work in Section 6 and Section 7, respectively.

2 Related works

The concept of modifier to represent linguistic hedges employing a mathematical transformation of membership functions was first introduced by Zadeh [12]. The author proposed using linguistic hedges as a power functions based operator, which applied to fuzzy sets, represents

*Corresponding author

the meaning of its operands. While applying transformations, the author used pure post modification of membership functions to represent linguistic hedges. The modifiers proposed by the author did not cover all forms of linguistic hedges. To mitigate this, another type of modifier called shifting modifiers was put forward in [13]. These modifiers are translatory modifiers that involve pure pre-modification of the membership functions.

However, traditional modifiers like powering and shifting modifiers could not handle similar categories distinguished by subtle differences. To mitigate this, De Cock and Kerre [14] introduced a new form of fuzzy modifiers, where weakening adverbs (*more or less*, *roughly*) and intensifying adverbs (*very*, *extremely*) are modeled in the inclusive and the non-inclusive interpretation. Another enhancement on powering and shifting modifiers was L-fuzzy modifiers introduced in [15] to model linguistic hedges in the L-fuzzy sets. They proposed using context through L-fuzzy relations to ensure L-fuzzy modifiers are endowed with clear inherent semantic. The proposed modifiers outperformed the traditional ones from the semantic point of view.

Since their advent, fuzzy modifiers have been used extensively in fuzzy rule-based and control systems [18, 19], analogy-Based Reasoning systems [16] and image processing or image-based retrieval systems [21]. They are used to obtain more interpretable results, limit the number of premises, dynamically modify the shape of membership functions and reduce the rule base [17, 20]. They have been used in designing databases as well, like SQLf - a well-known query language database [23, 22]. Fuzzy Modifiers have a wide range of practical applications, and therefore, it is imperative to account for the edge cases when modifiers are applied in real-world applications.

3 Approach

3.1 Preliminaries

Modifiers are a necessary part of Computing with Words. They allow us to define values in between given N Fuzzy sets. *Very*, *extremely*, *slightly*, *relatively*, *somewhat*, *quiet*, and *rather* are some of the frequently used modifiers. They are generally denoted by m.

$$X\,is\,mA \rightarrow X\,is\,f(A) \tag{1}$$

Modifier Type	Function $f(x)$
Extremely/Highly	x^3
Very/Rather	x^2
Relatively/Quite	$x^{3/2}$
Slightly	$x^{1/2}$
Somewhat	$x^{1/3}$

Table 1: Functions of The Modifiers

where f(A) is used to modify the behaviour of Fuzzy sets. The f(x) for frequently used modifiers is given in Table 1.

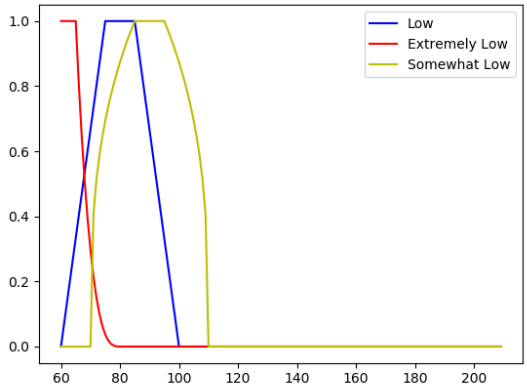


Figure 1: Modifier Applied to Type-1 Fuzzy Set

Figure 1 shows the Modifiers mentioned in Table 1 i.e. $Extremely$ and $Somewhat$ applied to Fuzzy set Low. The $f(x)$ is x^3 in case of Extremely and $x^{1/3}$ in case of Somewhat. The membership curve has shifted towards left for $extremely$ modifier, and it has shifted towards the right for $somewhat$ modifier. This is in accordance with the context of the application of modifiers.

Modifiers are extensively used in decision systems, especially in the domain of medicine. The interpolation rule for the Fuzzy set is unable to effectively deal when modifiers are applied. It cannot take contributions from the extreme left and extreme right-shifted fuzzy sets on the application of modifiers. Therefore a new Fuzzy Modifier Interpolation Rule was introduced for Type-1 Fuzzy sets.

3.2 Proposed fuzzy modifier interpolation rule on type-1 fuzzy set

In this subsection, we introduce our proposed Fuzzy Interpolation rule for Type-1 Fuzzy sets. The structure of the proposed rule consists of:

1. A set of IF-THEN rules, which is the database for inference mechanism.

2. An Antecedent with fuzzy modifier applied on it.

3. The proposed Fuzzy Modifier Interpolation rule, which consists of two cases.

   (a) **Edge Case**: It refers to the fuzzy sets having values near the boundaries of the domain.

   (b) **Non-Edge Case**: It refers to fuzzy set not having values near the boundaries of the domain.

Our proposed Fuzzy Modifier Interpolation technique predominantly deals with edge cases-Left edge case and the right edge case. Block diagram for proposed Fuzzy Modifier Interpolation rule is given in Fig 2 where m in mA_i antecedent is a modifier applied to the Fuzzy set input A_i. There are two separate cases - edge case and non-edge case, which gives Y is B as consequent.

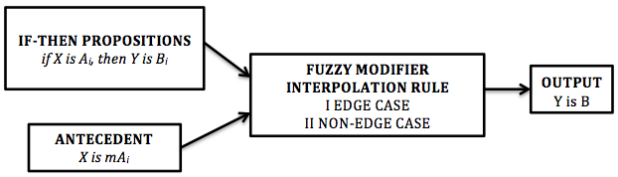


Figure 2: Block diagram for the application of Fuzzy Modifier Interpolation rule

3.2.1 Edge case

The edge case is defined as the case where the fuzzy sets have values near the boundaries of the domain. Formally, it is defined to be present when the following condition is satisfied:

$$If\, mA_i > A_i\, when\, i = n\, or\, mA_i < A_i\, when\, i = 1 \tag{2}$$

The condition (2) ensures that after application of modifiers on Type-1 fuzzy sets, the modified set is shifted either to the left extreme or right extreme.

The membership of the output Type-1 Fuzzy set obtained after application of our proposed Fuzzy Modifier Interpolation Rule is given by (3) for both left and right edge cases.

$$\mu_B(v) = f(\mu_{B_i}(v) + c) \tag{3}$$

In (3), the constant c defines the value of horizontal shift of fuzzy set, which is chosen after careful analysis of the domain. Figure 3 (Left). depicts the application of the proposed Fuzzy Modifier Interpolation rule for the right edge case with *relatively* modifier. Figure 3 (Right) depicts the resultant graph obtained after application of Fuzzy Modifier Interpolation rule. It shows the ability of our proposed rule to handle edge cases when fuzzy set $EXCESS$ are shifted to the right extreme and the ability to infer modified consequent fuzzy set.

3.2.2 Non edge case

This case deals where the equation (2) is not satisfied that is , $\overline{\mathrm{A}}$ is a non-edge case fuzzy set. When a modifier m is applied, the resulting output membership is given by the following results:

$$\mu_B = sup_j(m_j \cap B_j) \tag{4}$$

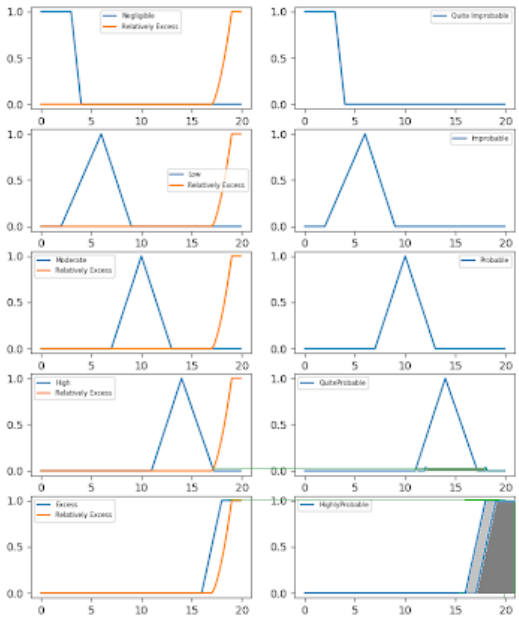


Figure 3: Fuzzy Modifier Interpolation Rule for Edge Case In Type-1 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Type-1 Fuzzy edge case
Right: Results on Application of Fuzzy Modifier Interpolation Rule

$$m_j = sup(A_j \cap A_i) \tag{5}$$

where $j = i - 1\, to\, i + 1$ and $A = mA_i$

Figure 4 (Left) depicts the application of proposed Fuzzy Modifier Interpolation rule for non edge case with extremely modifier. Figure 4 (Right) depicts the resultant graph obtained after application of Fuzzy Modifier Interpolation rule. It shows the ability of our proposed rule to deal with the non-edge case when applied to the $MODERATE$ fuzzy set. Slight changes in results are observed when compared with inferences drawn Fuzzy interpolation rule for the non-edge case.

3.3 Proposed fuzzy modifier interpolation rule on interval type-2 fuzzy set

This section extends the Fuzzy Modifier Interpolation rule to Interval Type-2 Fuzzy sets. Figure 5 shows the Modifiers mentioned in Table 1, i.e., $Extremely$ and $Somewhat$ applied to Interval Type-2 Fuzzy set 'Low'. The $f(x)$ is x^3 in case of Extremely and $x^{1/3}$ in case of Somewhat. Similar to Type-1 Fuzzy sets, the membership curve has shifted to-

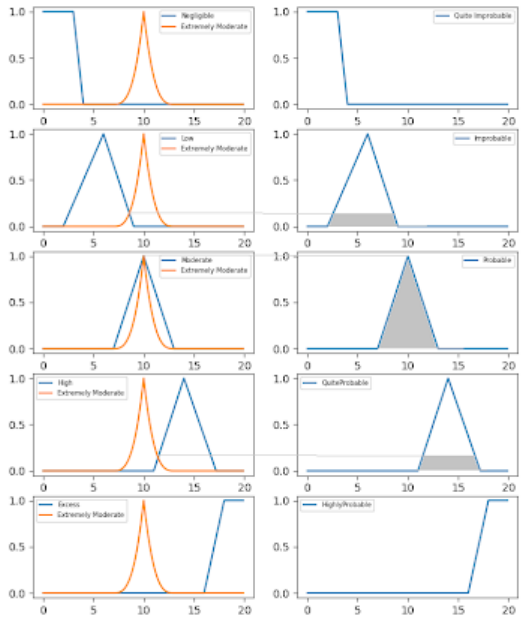


Figure 4: Fuzzy Modifier Interpolation Rule for Non-Edge Case In Type-1 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Type-1 Fuzzy non-edge case
Right: Results on Application of Fuzzy Modifier Interpolation Rule

wards the left for $extremely$ modifier and towards the right for the $somewhat$ modifier. This is in accordance with the context of the application of modifiers.
Similar to the Fuzzy Modifier Interpolation rule for Type-1 Fuzzy sets, the Fuzzy Modifier Interpolation rule in Interval Type-2 sets contains two cases as explained in Section 3.2:

1. Edge Case
2. Non-Edge Case

3.3.1 Edge case

Edge case is applied when (2) is satisfied by Interval Type-2 Fuzzy set $\overline{\mathrm{A}}$. Eq (6) depicts the application of modifier to Interval Type-2 Fuzzy sets and Eq (7) gives the membership of the output edge for Interval Type-2 Fuzzy set when l^{th} rule is fired is given:

$$\mu_G^l(y) = f(\mu_G^l(y) + c) \tag{6}$$

$$\mu_B^l(y)^{edge} = \mu_G^l(y) \cap F_{edge}^l \qquad y \in Y \tag{7}$$

where F_{edge}^l is given by Eq (8) and Eq (9) :

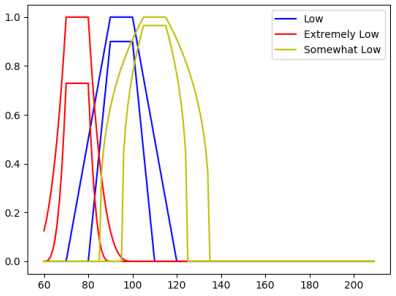


Figure 5: Modifier Applied to Interval Type-2 Fuzzy Set

$$\overline{\mathrm{F}}_{edge}^l = sup \int_{x_1 \in X_1} \dots \int_{x_p \in X_p} [\overline{\mu}_X(x1) \star \overline{\mu}_{F^l}(x_1)]/x_1 \star \dots \star [\overline{\mu}_X(x_p) \star \overline{\mu}_{F^l}(x_p)]/x_p \tag{8}$$

$$\underline{\mathrm{F}}_{edge}^l = sup \int_{x_1 \in X_1} \dots \int_{x_p \in X_p} [\underline{\mu}_X(x1) \star \underline{\mu}_{F^l}(x_1)]/x_1 \star \dots \star [\underline{\mu}_X(x_p) \star \underline{\mu}_{F^l}(x_p)]/x_p \tag{9}$$

3.3.2 Non-edge case

This case is applied when the condition (2) is not satisfied that is , $\overline{\mathrm{A}}$ is a non-edge case. Similar to edge case for Interval Type-2 Fuzzy sets, $\mu_G^l(y)$ is modified upon application of modifier as given in Eq (6) and Eq (10) gives Membership of the output non-edge Interval Type-2 Fuzzy set when l^{th} rule is fired is given :

$$\mu_B^l(y)_{non-edge} = \mu_G^l(y) \cap F_{non-edge}^l \qquad y \in Y \tag{10}$$

where $F_{non-edge}^l$ is given by Eq (11) and Eq (12):

$$\overline{\mathrm{F}}_{non-edge}^l = sup \int_{x_2 \in X_2} \dots \int_{x_{p-1} \in X_{p-1}} [\overline{\mu}_X(x2) \star \overline{\mu}_{F^l}(x_2)]/x_2 \star \dots \star [\overline{\mu}_X(x_{p-1}) \star \overline{\mu}_{F^l}(x_{p-1})]/x_{p-1} \tag{11}$$

$$\underline{\mathrm{F}}_{non-edge}^l = sup \int_{x_2 \in X_2} \dots \int_{x_{p-1} \in X_{p-1}} [\underline{\mu}_X(x2) \star \underline{\mu}_{F^l}(x_2)]/x_2 \star \dots \star [\underline{\mu}_X(x_{p-1}) \star \underline{\mu}_{F^l}(x_{p-1})]/x_{p-1} \tag{12}$$

For l^{th} rule , The membership of the output Interval Type-2 Fuzzy set is union of membership of outputs from edge and non-edge cases in Interval Type-2 Fuzzy set as given in Eq (13).

$$\mu_B^l(y) = \mu_B^l(y)_{edge} \sqcup \mu_B^l(y)_{non-edge} \tag{13}$$

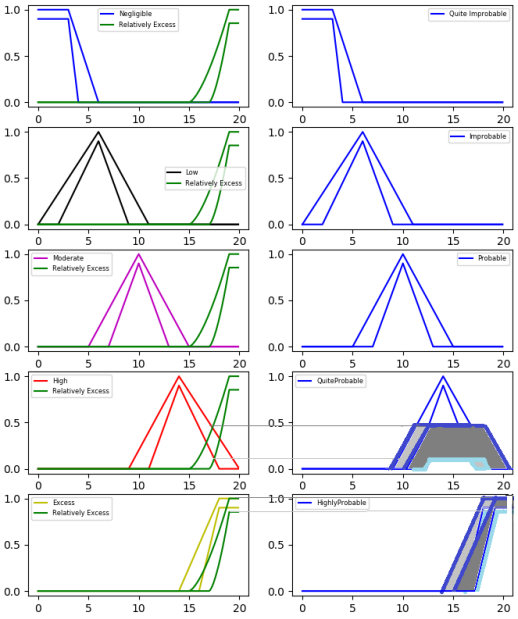


Figure 6: Example of Fuzzy Modifier Interpolation rule for edge case in Interval Type-2 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Interval Type-2 Fuzzy edge case
Right: Results obtained after application of Fuzzy Modifier Rule on Interval Type-2 Fuzzy sets for edge case

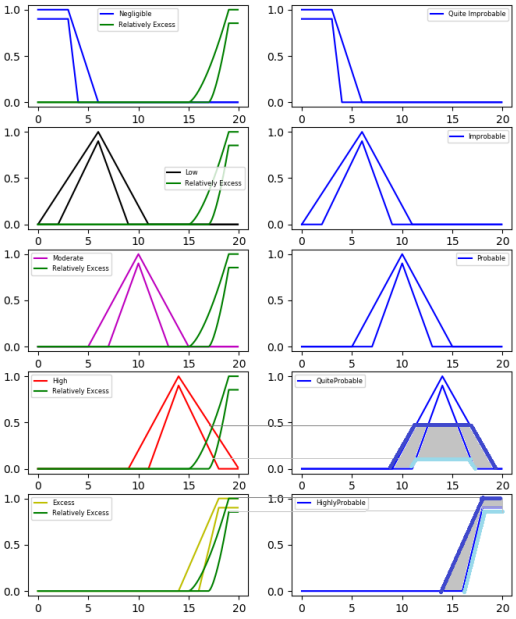


Figure 7: Example of Fuzzy Modifier Interpolation rule for non-edge case in Interval Type-2 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Interval Type-2 Fuzzy non-edge case
Right: Results obtained after application of Fuzzy Modifier Rule on Interval Type-2 Fuzzy sets for non-edge case

Membership of the output Interval Type-2 Fuzzy set when N rules are fired is :

$$\mu_B(y) = \sqcup_{l=1}^{N} \mu_B(y) \qquad y \in Y \tag{14}$$

Figure 7 (Left) depicts an example of application of Fuzzy Modifier Interpolation Rule applied on modified Interval Type-2 Fuzzy set $EXCESS$ with $RELATIVELY$ modifier and Figure 7(Right) shows the results obtained after application of our proposed interpolation rule on Interval Type-2 Fuzzy sets for non-edge cases. Similar to Type-1 Fuzzy sets for non-edge cases, slight changes in results are observed compared with inferences drawn from the Fuzzy interpolation rule.

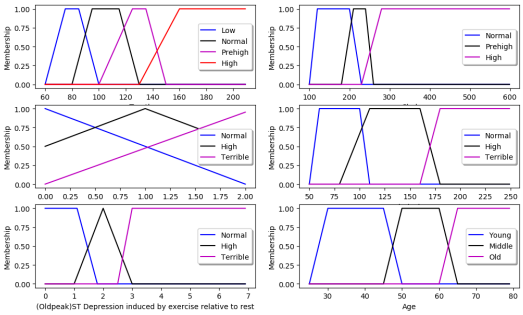
Fig 7 (Left) depicts an example of application of Fuzzy Modifier Interpolation Rule applied on modified Interval Type-2 Fuzzy set $EXCESS$ with $RELATIVELY$ modifier and Fig 7 (Right) shows the results obtained after application of our proposed interpolation rule on Interval Type-2 Fuzzy sets for edge cases. Similar to Type-1 Fuzzy sets, our proposed rule takes care of inference from fuzzy sets shifted towards extreme right and extreme left upon application of modifiers.

4 Experimentation

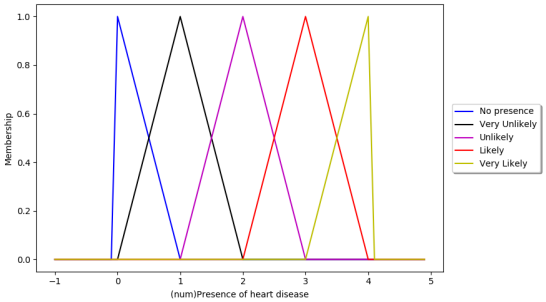
The proposed Fuzzy Modifier Interpolation for Fuzzy Sets is applied on the UCLA Cleveland dataset for heart disease for different cases. The dataset contains 76 attributes, but we are using six attributes (resting blood pressure or tresbps, serum cholesterol or chol, maximum heart rate achieved or thalach, ST depression induced by exercise relative to rest or oldpeak, resting electrocardiographic results or restecg and age) to predict the presence of heart disease.

Careful domain analysis on factors affecting heart diseases was done before formulating the selected attributes' fuzzy membership functions. Once the fuzzy sets of the selected attributes were formulated, the fuzzy rule was obtained by using Fuzzy Decision Trees Induction Method [28]. This method given by Yufei Yuan, Michael J. Shaw is based on reducing classification ambiguity with fuzzy evidence. To reduce the ambiguity while partitioning, we use a significance level of 0.45.

The experiments on Fuzzy Modifier Interpolation Rule were conducted in two phases with Phase 1 focusing on Type-1 Fuzzy sets and Phase 2 focusing on Interval Type-2 Fuzzy sets.

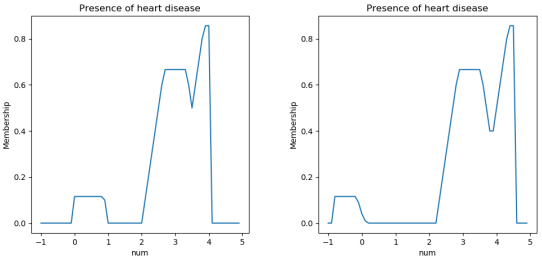


(a) Membership of Antecedents



(b) Membership of Consequent

Figure 8: Type-1 Fuzzy Sets for different factors affecting Heart disease



(a) Output Type-1 Fuzzy set after application of Fuzzy Interpolation Rule

(b) Output Type-1 Fuzzy set after application of Fuzzy Modifier Interpolation Rule

Figure 9: Comparison between graphs obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Type-1 Fuzzy sets

Input Variable	**Value**
Tresbps	Extremely High
Chols	Rather High
Restecg	Relatively Terrible
Thalach	Relatively Terrible
Oldpeak	Relatively Terrible
Age	Extremely Old

Table 2: Input Query

4.1 Experiments and analysis for type-1 fuzzy sets

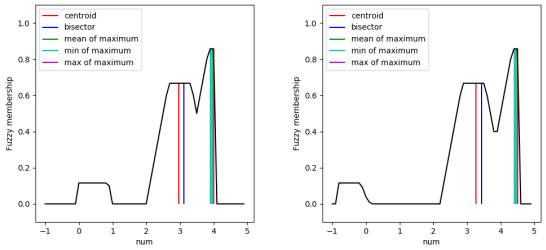
The first half of the experiment was focused on Type-1 Fuzzy sets. It was conducted for three cases - right edge case, left edge case, and non-edge cases. The antecedents and consequents are represented using Type-1 Fuzzy sets. Figure 8 (A). depicts membership of antecedents, and Figure 8 (B) depicts membership of consequent.

Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation Rule using *min T-norm* is applied to the query for right edge case and output curve obtained is depicted in Figure 9 (A) and Figure 9 (B) respectively for fuzzy interpolation rule and fuzzy modifier interpolation rule. We observe the graph obtained in Figure 9 (A) is shifted to the left and right taking care of the extreme left and right cases obtained on application of modifiers.

Input Query used for the right edge case is given in Table 2:

The obtained graphs for the edge cases were defuzzified to understand better the results obtained after applying the Fuzzy Modifier Interpolation Rule. For the left edge case, we used the least of the maximum for defuzzification. Similarly, for the right case, we used the largest of the maximum for defuzzification and centroid for the non-edge case. Figure 10 shows the comparison between the results obtained for the Fuzzy Interpolation rule and the Fuzzy Modifier Interpolation rule on defuzzification for the right edge case. In Figure 10 (B), the defuzzified values are shifted towards the right for the mean of maximum, min of maximum, and max of maximum criteria. This corroborates the ability of our proposed interpolation rule to infer from edge cases efficiently. The choice of the defuzzification technique is case-dependent and based on the shift of the fuzzy sets expected in each case.

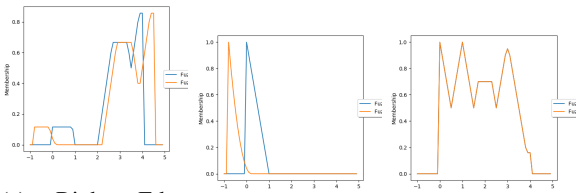
A comparison between the Fuzzy Modifier Interpolation Rule and Fuzzy Interpolation Rule for all three cases- left edge case, non-edge case, and right edge case is shown for the Type-1 Fuzzy set. The blue curve in Figure 11 is obtained after the application of Fuzzy Interpolation Rule to Type-1 Fuzzy sets, and the Orange curve is obtained after the application of Fuzzy Modifier Interpolation Rule to Type-1 Fuzzy sets. In Figure 11 (A), The modifier is applied to the right valued Type-1 Fuzzy input, and after the application of Fuzzy Modifier Interpolation Rule, the output curve is shifted to the right when compared with Fuzzy Interpolation Rule. Similarly, for Figure 11 (B), the output curve for Fuzzy Modifier interpolation Rule is shifted towards the left when a modifier is applied to the left valued Type-1 fuzzy set input. A very slight variation is observed in Figure 11 (C) when a modifier is applied to non-edge case valued Type-1 Fuzzy set input.



(a) A. Results obtained after defuzzification of the graphs obtained after application of Fuzzy IR

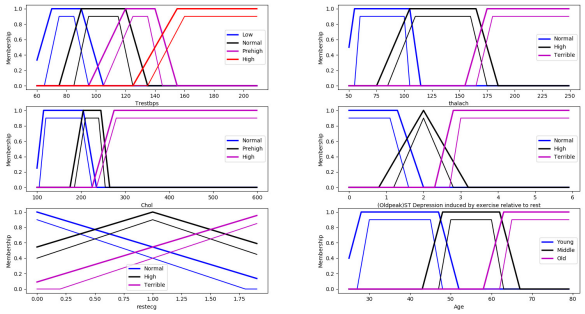
(b) B. Results obtained after defuzzification of the graphs obtained after application of Fuzzy Modifier IR

Figure 10: Comparison between Defuzzified results obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Type-1 Fuzzy sets

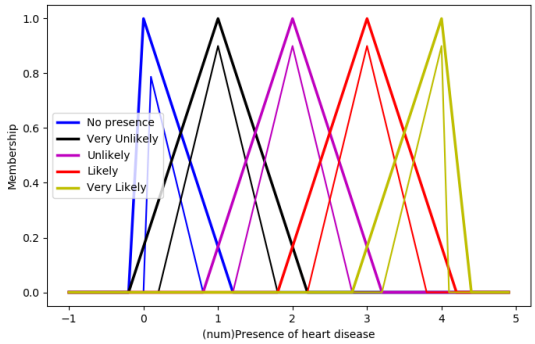


(a) Right Edge Case (b) Left Edge Case (c) Non-Edge Case

Figure 11: Comparison between Fuzzy Interpolation rule (blue) and Fuzzy Modifier Interpolation Rule (Orange) for Type-1 Fuzzy sets



(a) Membership of Antecedents



(b) Membership of Consequent

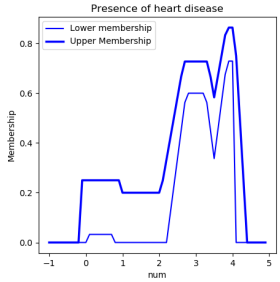
Figure 12: Type-2 Fuzzy Sets for different factors affecting Heart disease

4.2 Experiments and analysis for interval type-2 fuzzy sets

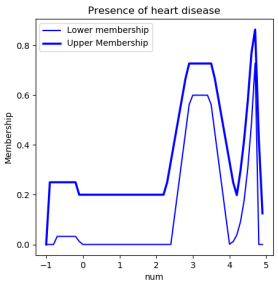
In the second half of the experiment, antecedents and consequent are represented using Interval Type-2 Fuzzy sets with triangular and trapezoidal memberships as shown in Figure 12. Similar to the experiments conducted on Type-1 Fuzzy sets, experiments on Interval Type-2 Fuzzy sets were conducted for three cases- right edge case, left edge case, and non-edge cases.

Similar to Type-1 Fuzzy sets, Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation Rule using min T-norm is applied to the query for right edge case as given in Table 2 and output curve obtained is depicted in Figure 13(A) and Figure 13(B) respectively for fuzzy interpolation rule and fuzzy modifier interpolation rule. We observe the graph obtained in Figure 13(B) is shifted to the left and right, taking care of the extreme left and right cases obtained on application of modifiers which is consistent with results obtained for Type-1 Fuzzy sets.

For type reduction of Interval Type-2 Fuzzy sets, the N-T algorithm was used, and for defuzzification, a similar methodology as discussed in the case of Type-1 fuzzy sets was used. Figure 14 shows the comparison between the results obtained for the Fuzzy Interpolation rule and Fuzzy Modifier Interpolation rule on defuzzification for the

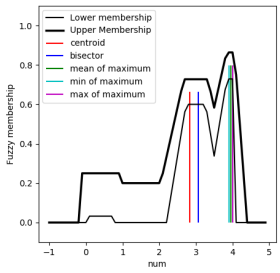


(a) Output Interval Type-2 Fuzzy set after application of Fuzzy Interpolation Rule

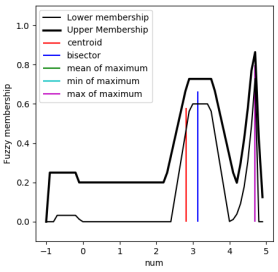


(b) Output Interval Type-2 Fuzzy set after application of Fuzzy Modifier Interpolation Rule

Figure 13: Comparison between graphs obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Interval Type-2 Fuzzy sets



(a) Output Interval Type-2 Fuzzy set after application of Fuzzy Interpolation Rule

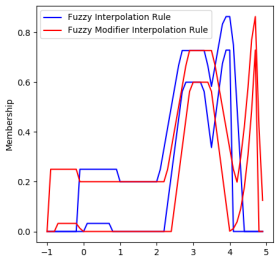


(b) Output Interval Type-2 Fuzzy set after application of Fuzzy Modifier Interpolation Rule

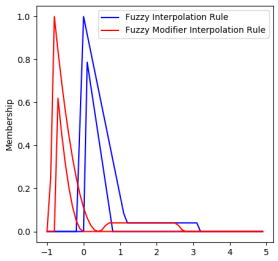
Figure 14: Comparison between defuzzified results obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Interval Type-2 Fuzzy sets

right edge case. In Figure 14 (B), the defuzzified values are shifted towards right for the mean of maximum, min of maximum, and max of maximum criteria. This corroborates the ability of our proposed interpolation rule to infer from edge cases efficiently.

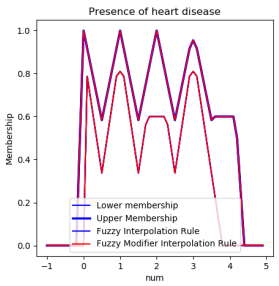
Similar to experiments performed on Type-1 Fuzzy sets, A comparison between Fuzzy Modifier Interpolation Rule and Fuzzy Interpolation Rule for all three cases- left edge case, non-edge case, and right edge case is shown for Interval Type-2 Fuzzy set. The blue curve in Figure 15 is obtained after the application of Fuzzy Interpolation Rule to Interval Type-2 Fuzzy sets, and Red curve is obtained after application of Fuzzy Modifier Interpolation Rule to Interval Type-2 Fuzzy sets. In Figure 15 (A), The modifier is applied to the right valued Interval Type-2 Fuzzy set input, and after application of Fuzzy Modifier Interpolation Rule the output curve is shifted to the right when compared with Fuzzy Interpolation Rule. Similarly, for Figure 15 (B),



(a) Right Edge Case



(b) Left Edge Case



(c) Non-Edge Case

Figure 15: Comparison between Fuzzy Interpolation rule (blue) and Fuzzy Modifier Interpolation Rule (red) for Interval Type-2 Fuzzy sets

the output curve for Fuzzy Modifier interpolation Rule is shifted towards the left when a modifier is applied to the left valued Interval Type-2 fuzzy set input. A very slight variation is observed in Figure 15 (C) when a modifier is applied to non-edge case valued Interval Type-2 Fuzzy set input.

5 Discussion

Table 3 shows the value of predicted attribute obtained after defuzzification of the output curve obtained as a result of the application of Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation Rule on both Type-1 Fuzzy set and Interval Type-2 Fuzzy set. No change in the values was obtained for the Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation rule for the non edge case. Our proposed inference interpolation rule can effectively infer results from left edge case and right edge case, where the fuzzy sets are shifted towards extreme left and extreme right respectively on the application of modifiers, as shown in Table 3. Table 4 shows the final results obtained from the decision-making system after fuzzification of the results obtained from defuzzification.

The results obtained in table 4 depict the ability of our proposed inference rule to handle the left and right edge case effectively when modifiers are applied to Fuzzy set. Thus, Fuzzy Modifier Interpolation inference results in better modeling of variables used in a decision-making system and improved output curves and results on defuzzification.

	Right Edge Case	Left Edge Case	Non-Edge Case
Type-1 Fuzzy Set With Fuzzy Interpolation Rule	3.99999999999	-2.220446049250	1.808727539420
Type-1 Fuzzy Set with Fuzzy Modifier Interpolation Rule (Proposed Approach)	4.49999999999	-0.8	1.808727539420
IT-2 Fuzzy Set With Fuzzy Interpolation Rule	3.99999999999	0.0999999999999	1.92260915424
IT-2 Fuzzy Set With Fuzzy Modifier Interpolation Rule (Proposed Approach)	4.69999999999	-0.700000000000	1.92260915424

Table 3: Defuzzified Results

	Right Edge Case	Left Edge Case	Non-Edge Case
Type-1 Fuzzy Set With Fuzzy Interpolation Rule	Very Likely	Very Likely	Unlikely
Type-1 Fuzzy Set with Fuzzy Modifier Interpolation Rule (Proposed Approach)	Relatively Very Likely	Relatively Very Likely	Unlikely
IT-2 Fuzzy Set With Fuzzy Interpolation Rule	No Presence	No Presence	Unlikely
IT-2 Fuzzy Set With Fuzzy Modifier Interpolation Rule (Proposed Approach)	Rather No Presence	Rather No Presence	Unlikely

Table 4: Final Results Obtained

6 Advantage and future use

Fuzzy Interpolation Rule is quite a general rule that cannot deal with modifiers in Computing with words engine effectively. Until now, no efforts have been made to deal with modifiers separately in Type-1 Fuzzy sets and Interval Type-2 Fuzzy sets.

The Fuzzy Modifier Interpolation rule in Fuzzy Logic Inference engine provides the capability to introduce result as an extreme end fuzzy set that Fuzzy Interpolation rule cannot. $RELATIVELY\ HIGHLY\ EXCESS$, which is an extreme end fuzzy set that can be obtained on the application of this rule.

Given in Table 5 is the comparison between results drawn from the above experimentation on the UCLA dataset, which depict the improvement introduced by the application of Fuzzy Modifier Interpolation Rule.

Approach	Value
Fuzzy Modifier Interpolation Rule	Relatively Very Unlikely
Fuzzy Interpolation Rule	Very Unlikely

Table 5: Comparison Between Results Obtained

Fuzzy Modifier Interpolation Rule gives $RELATIVELY\ VERY\ UNLIKELY$ whereas Fuzzy Interpolation would have given only $VERY\ UNLIKELY$, This allows us to not only take care of edge cases in the modified fuzzy set but also get modified consequent as results. The use of modifiers on consequent gives the flexibility to express results beyond the pre-defined linguistic values for a fuzzy set.

Future use includes:

1. Incorporation of Fuzzy Modifier Interpolation Rule to Computing with Words inference engine to get better results.
2. Extending this rule to Fuzzy Weighted Average used in different decision making problem [24].
3. Incorporating this rule in fuzzy logic control that is used to regulate the movement of robots [25].
4. Applications in Medicine Diagnostic Systems when modifiers are to be applied [8].

7 Limitations

The Fuzzy Modifier Interpolation Rule produces precise and accurate results in most cases, but it fails when modifiers fail to satisfy the condition that the shift of Fuzzy sets on the application of modifier should be to its adjacent Fuzzy set only. This rule does not work for NOT modifier.

Proposed Fuzzy Modifier Interpolation rule for Type-1 and Interval Type-2 Fuzzy sets is applicable where only a single modifier is applied to the input given to Inference Engine. It fails when multiple modifiers are applied to the input. For example, X is $RELATIVELY\ SMALL$ is valid input; however, X is $RELATIVELY$ and $SOMEWHAT\ SMALL$ is invalid.

The Fuzzy Modifier Interpolation rule also fails when multiple connectives are used in the proposition like $X\ is\ RELATIVELY\ VERY\ SMALL$ or $Y\ is\ VERY\ VERY\ LARGE$. In the former proposition, modifier $RELATIVELY\ VERY$ is applied to the

fuzzy set $SMALL$. In the latter proposition, the modifier $VERY\,VERY$ is applied to the fuzzy set $LARGE$. The shift in $SMALL$ and $LARGE$ curves is expected to extend beyond the adjacent fuzzy sets. As a result, the Fuzzy Modifier Interpolation Rule fails with multiple connectives. In future works, the Fuzzy Modifier Interpolation rule can be extended to include the application of multiple connectives as modifiers.

8 Conclusion

Computing with Words is gaining importance and becoming increasingly popular. It has many applications in science, and varied industries [29][30][31]. The inference engine or different techniques of inferencing results like the Fuzzy Interpolation rule in CW play a significant role in producing results. However, it has limitations when applied to modified Fuzzy sets. Therefore, if some modifications to the inference rules can give better outputs when dealing with modified Fuzzy sets, then they have a significant impact on all the fields where CW is being implemented. On a concluding note, the proposed Fuzzy Modifier Interpolation rule for Type-1 Fuzzy Sets and Interval Type-2 is an important milestone because of its ability to take care of left and right extreme cases when dealing with modified fuzzy sets. This results in improvement of the output curves obtained and better results on defuzzification. Its inclusion in the Inference rules would be beneficial for every user who applies the CW model.

Acknowledgement

We want to extend our sincere thanks to Aashi Jain, Divya Gupta, and Garima Gupta for their inputs during idea conception and our anonymous reviewers for their invaluable feedback.

References

[1] L.A. Zadeh , Fuzzy logic= computing with words, IEEE transactions on fuzzy systems 4.2, 103-11(1996)

[2] L.A. Zadeh, Key roles of information granulation and fuzzy logic in human reasoning, Concept formulation and computing with words.Proceedings of the Fifth IEEE International Conference , Vol. 1.(1996)

[3] J. Mendel, J. Lawry,L.A. Zadeh. Foreword to the special section on computing with words, IEEE Transactions on Fuzzy Systems, 18.3,437-440 (2010)

[4] L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning—I, Inform. Sci., 8,pp. 199–249 (1975)

[5] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning—II, Information sciences, 8.4, 301-357 (1975)

[6] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning-III, Information sciences, 9.1, 43-80 (1975)

[7] J.M. Mendel, Computing with words, when words can mean different things to different people, ICSC Symposium on Fuzzy Logic and Applications, pp. 158–164 (June 1999)

[8] C. Schuh, Fuzzy set and their application in medicine, NAFIPS Annual Meeting of the North American Fuzzy Information Processing Society (2005)

[9] L.A. Zadeh, A new direction in AI: Toward a computational theory of perceptions, AI magazine, 22.1, 73 (2001)

[10] L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy sets and systems, 1.1, 3-28 (1978)

[11] L.A. Zadeh, Fuzzy logic, Computer, 21.4, 83-93 (1988)

[12] Zadeh, Lotfi A. "A fuzzy-set-theoretic interpretation of linguistic hedges." (1972): 4-34.

[13] Hellendoorn, Johannes. “Reasoning with fuzzy logic.” (1990).

[14] De Cock, Martine, and Etienne E. Kerre. "Fuzzy modifiers based on fuzzy relations." Information Sciences 160.1-4 (2004): 173-199.

[15] De Cock, Martine, and Etienne E. Kerre. "A context-based approach to linguistic hedges." International Journal of Applied Mathematics and Computer Science 12 (2002): 371-382.

[16] B. Bouchon-Meunier and C. Marsala, "Linguistic modifiers and measures of similarity or resemblance," Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569), 2001, pp. 2195-2199 vol.4, doi: 10.1109/NAFIPS.2001.944410.

[17] Bin-Da Liu, Chuen-Yau Chen and Ju-Ying Tsao, "Design of adaptive fuzzy logic controller based on linguistic-hedge concepts and genetic algorithms," in IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 31, no. 1, pp. 32-53, Feb 2001, doi: 10.1109/3477.907563.

[18] Ying, Mingsheng, and Bernadette Bouchon-Meunier. "Approximate reasoning with linguistic modifiers." International journal of intelligent systems 13.5 (1998): 403-418.

[19] Bernadette Bouchon-Meunier. 1992. Fuzzy logic and knowledge representation using linguistic modifiers. Fuzzy logic for the management of uncertainty. John Wiley & Sons, Inc., USA, 399–414.

[20] Cordón, Oscar, María José del Jesus, and Francisco Herrera. "Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods." International Journal of Intelligent Systems 13.10-11 (1998): 1025-1053.

[21] S. Medasani and R. Krishnapuram, "A fuzzy approach to content-based image retrieval," FUZZ-IEEE'99. 1999 IEEE International Fuzzy Systems. Conference Proceedings (Cat. No.99CH36315), 1999, pp. 1251-1260 vol.3, doi: 10.1109/FUZZY.1999.790081.

[22] H. Nakajima, T. Sogoh and M. Arao, "Fuzzy database language and library-fuzzy extension to SQL," [Proceedings 1993] Second IEEE International Conference on Fuzzy Systems, 1993, pp. 477-482 vol.1, doi: 10.1109/FUZZY.1993.327514.

[23] P. Bosc and O. Pivert. 1994. Fuzzy queries and relational databases. In Proceedings of the 1994 ACM symposium on Applied computing (SAC '94). Association for Computing Machinery, New York, NY, USA, 170–174. DOI:https://doi.org/10.1145/326619.326690

[24] F. Liu , J. Mendel, Aggregation using The Fuzzy Weighted Average as computed using Kernel-Mendel Algorithm. IEEE transactions on fuzzy systems, vol. 16 (february 2008)

[25] Aguirre, Eugenio, and Antonio González, Fuzzy behaviors for mobile robot navigation: design, coordination and fusion, International Journal of Approximate Reasoning, 25.3 ,255-289 (2000)

[26] L.A. Zadeh, Fuzzy Sets, information and control, 8, 338–353 (1965)

[27] David Aha, Heart Disease Data Set

[28] Y. Yuan, M.J. Shaw, Induction of fuzzy decision trees,Fuzzy Sets and Systems, Vol 6, Pages 125-139 (1995)

[29] L.A. Zadeh, Some reflections on information granulation and its centrility in granular computing, computing with words, the computational theory of perceptions and precisiated natural language, Studies in Fuzziness and Soft Computing,95

[30] L.A. Zadeh, A summary and update of "fuzzy logic", Granular Computing (GrC), 2010 IEEE International Conference

[31] A. Jain, Applications of computing with words in medicine: Promises and potential, Fuzzy Systems (FUZZ-IEEE), 2017 IEEE International Conference

Evaluating Public Sentiments of Covid-19 Vaccine Tweets Using Machine Learning Techniques

Samuel Kofi Akpatsa, Hang Lei, Xiaoyu Li and Victor-Hillary Kofi Setornyo Obeng
E-mail: samoah15@yahoo.com, hlei@uestc.edu.cn, xiaoyu33521@163.com, hillary2gh@gmail.com
School of Information and Software Engineering, University of Electronic Science and Technology of China
Chengdu, Sichuan Province, 610054, P.R. China

Keywords: Natural Language Processing (NLP), Covid-19, twitter sentiment analysis, machine learning

Received: March 30, 2021

The quest to create a vaccine for covid-19 has rekindled hope for most people worldwide, with the anticipation that a vaccine breakthrough would be one step closer to the end of the deadly Covid-19. The pandemic has had a bearing on the use of Twitter as a communication medium to reach a wider audience. This study examines Covid-19 vaccine-related discussions, concerns, and Twitter-emerged sentiments about the Covid-19 vaccine rollout program. Natural Language Processing (NLP) techniques were applied to analyze Covid-19 vaccine-related tweets. Our analysis identified popular n-grams and salient themes such as "vaccine health information," "vaccine distribution and administration," "vaccine doses required for immunity," and "vaccine availability." We apply machine learning algorithms and evaluate their performance using the standard metrics, namely accuracy, precision, recall, and f1-score. Support Vector Machine (SVM) classifier proves to be the best fit on the dataset with 84.32% accuracy. The research demonstrates how Twitter data and machine learning methods can study the evolving public discussions and sentiments concerning the Covid-19 vaccine rollout program.

Povzetek: Narejena je analiza čivkov na Twitterju glede cepiv za COVID-19.

1 Introduction

The Covid-19 outbreak in late 2019 has led to tens of millions of confirmed cases and millions of deaths worldwide. The economic and social disruption of the pandemic has altered the way of life for many around the globe as public health protocols such as social distancing, wearing of masks, and travel restrictions were introduced. Although adherence to these protocols has effectively controlled the spread of the virus, there has been a global effort to develop a Covid vaccine to fight the virus head-on and help get immunity against it. Studies have indicated that at least 70% of the world's population is expected to be vaccinated to achieve herd immunity [1]. Consequently, some major pharmaceutical companies and research institutions across the globe have announced the development of Covid vaccines that promise to help ease restrictions and return the world to pre-pandemic routines.

While these developments inspire hope and optimism, other obstacles threaten the fight to eradicate the deadly virus. A significant proportion of people are unsure of the safety of the Covid vaccine, as skepticism on social media has led to the vaccine rollout exercise faced with fears, hesitancy, and opposition [2]. Meanwhile, public opinions and support for the Covid-19 vaccine are essential as this may affect whether vaccinations can be administered to large populations to achieve herd immunity.

As the Covid-19 pandemic continues to spread globally, Covid-related issues received increasing attention from the research community. Although some studies have highlighted the socio-economic impact of the pandemic [3]–[9], analysis of Covid-19 vaccine-related issues is rare. As a social media choice for many, Twitter plays a vital role in disseminating health information in the fight against Covid-19. There is an urgent need to analyze how issues related to Covid-19 vaccine have been discussed on Twitter to understand better public perceptions, concerns, and issues that may affect their willingness to get vaccinated. Besides, identifying popular themes in tweets related to Covid-19 vaccines can play a vital role in guiding vaccine education and communication. This study examines general sentiments and opinions related to the Covid-19 vaccine rollout program by analyzing English tweets collected between January 21, 2021, and January 31, 2021. The study also identifies algorithms with suitable metrics to evaluate the performance of supervised Machine Learning classifiers on the Covid-19 vaccine tweets. The findings will be handy in assisting governments and other public health policymakers to understand trends in social media data related to the Covid-19 vaccine and make timely adjustments to vaccine education to boost public confidence in the vaccination exercise.

2 Related works

Studies on Twitter sentiment analysis provide valuable insights into real-world events and people's perceptions of these events. A review of existing literature indicates that various studies related to Twitter sentiment analysis use

popular machine learning algorithms such as Logistic Regression, Support Vector Machines, Naive Bayes to predict sentiment from tweets [10]–[14]. These algorithms essentially give better accuracy with less computational resources and are regarded as the baseline learning methods in sentiment analysis of Twitter data [15].

Recent studies demonstrate that deep learning models allow sentiment analysis systems to capture complex linguistic features and read context within a text, achieving better accuracy and performance [16]. Researchers have used these techniques to analyze unstructured data from social media posts such as Twitter [17], [18]. A related study implemented a quantum-inspired sentiment representative framework that can model semantic and the sentiment information of subjective natural language text [19]. Experimental results demonstrate the effectiveness of the framework as it significantly outperforms most state-of-the-art baselines.

Following the successful applications of these algorithms on Twitter data, an increasing number of studies have used similar approaches to analyze and understand the public response and discussions on Twitter concerning Covid-19. For example, a study analyzed the global sentiments of tweets related to Covid-19 to understand how people's sentiment in different countries has changed over time [20]. Two types of analysis were performed concerning the positive and negative sentiments, fear, and trust emotions exhibited in tweets related to *Work from Home* and *Online Learning*. The first was exploratory data analysis to provide insight into the number of daily confirmed cases. The second aspect evaluated different deep learning methods for sentiment classification on the dataset. The results showed that the general positive sentiments towards *Work from Home* and *Online Learning* have been consistently higher than negative sentiments.

A similar analysis was performed on social media posts to increase understanding of public awareness of COVID-19 pandemic trends. The research uncovers meaningful themes of concern posted by Twitter users in English during the pandemic [21]. The analyses included frequency of keywords, sentiment analysis, and topic modeling to identify and explore discussion topics over time. The results indicate that people have a negative outlook toward COVID-19.

A related study applied machine learning techniques to investigate the psychological reactions of Twitter users to Covid-19 [22]. Several salient topics were identified and categorized into themes, including "confirmed cases," "Covid-19 related death," "early signs of the outbreak," "economic impact of the pandemic," and "Preventive measures." The analysis shows that fear for the unknown nature of the coronavirus is dominant in all topics. Other successful examples of studies analyzing Twitter sentiments to determine the impact of Covid-19 on the daily aspect of life are well documented [6], [7], [9], [12], [23], [24]. These studies proved essential in assisting governments in making informed choices on managing the Covid-19 pandemic situation.

However, research on Twitter emerged sentiments on Covid-19 vaccine rollout program remain less explored in literature. In relation to the related works, this study explores public reactions and discussions on Twitter concerning the Covid-19 vaccine rollout program. The performance comparison of different machine learning classifiers on Covid-19 vaccine Twitter dataset is also be evaluated.

3 Research methods

3.1 Research design

A purposive sampling technique was adopted in gathering Covid-19 Twitter data, published between 21st January to 31st January 2021. Our data analysis is divided into two broad parts. The first part dives into an exploratory analysis of tweets, data visualizations, and a description of the key characteristics of Covid-19 vaccine Twitter data. This approach aims to present insight and help understand public reactions and discussions on Twitter concerning Covid-19 vaccine rollout programs worldwide. The second part deals with the sentiment classification of tweets using supervised machine learning algorithms. We chose a supervised machine learning approach because our data is well labeled. The supervised learning technique allows us to measure the chosen classifiers' accuracy scores while performing sentiment classification.

3.2 Data collection

Twitter offers a variety of APIs to provide Twitter data access, including reading tweets and accessing user profiles. This study uses Twitter API and a Python script to access Covid-19 vaccine Twitter comments. A query for a hashtag (#Covid19vaccine) was run daily to collect a large number of tweet samples from around the globe. The study excluded tweets written in languages other than English. Our approach of getting the dataset is based on its availability and accessibility and how well the research community accepts the approach.

3.3 Data labeling

The labeling process aims to assign positive or negative labels to tweets. This study used human annotation to assign the value 1 (positive class) to text with positive sentiment and the value 0 (negative class) to text with negative sentiment. Some of the extracted tweets were duplicated, while others gave contradictory interpretations and proved difficult to label. As a result, some data points were removed from the dataset. The final dataset contains 15239 unique tweets, with 10519 labeled as positive and 4720 labeled as negative. The dataset contains two

	text	label
0	We have to change this The only chance we have to get through the pandemic is the COVID19vaccine ThisIsOurShot	1
1	In my personal opinion its a civic responsibility to get the COVID19vaccine This is about not only protectin	1
2	Lack of access to patient portals could affect COVID19 vaccine for adults ages 50 and older covid19 coronavirus	0
3	They may have retired but they returned We are incredibly grateful for the many nurses that came out of retirement to help with vaccination	1
4	World Health Organization recommends using the COVID19vaccine developed by the AstraZenecaOxford even in countri	1
5	World Scientists still dont know exactly how long this immunity lasts or how strong it is though recent researc	0
6	Inaccurate information surrounding the safety of the CovidVaccines have left many women fearing the vaccinations	0
7	Essential to COVID19Vaccine We need to encourage vaccination ethically so people feel comfortable and not coerce	1
8	Science really is like magic 2nd dose of my PfizerVaccine is in COVID19vaccine UofSCPharm forever grateful	1
9	Whoop Whoop I am now 96 immune to SARSCoV2 the virus that causes COVD19 thanks to the vaccine What are you waiting for	1

Figure 1: Sample tweets from the Twitter dataset.

columns (text and labels). The text column contains the text to which a label applies. These texts are transformed into features used by the model during training and prediction. The label column contains either 1 or 0, representing the sentiments of the tweet being classified. An example of the tweet dataset obtained can be seen in Figure 1.

3.4 Data preprocessing

Analyzing sentiment in tweets generally requires some fundamental cleaning and preprocessing steps to improve the quality of the dataset [25]. It includes cleaning and formatting the data before feeding it into a machine learning algorithm. Twitter datasets are often noisy with many irregularities such as punctuation marks, symbols, @links, stop words, and other special characters irrelevant for the sentiment analysis. The collected tweets are filtered using a python script to preprocess and clean the dataset to increase precision. Background noises such as white space, punctuation, hashtags, urls, special characters, hyperlinks, and stop words were removed. Also, tokenization using n-grams was applied to segment the text data and create a new document with the set n-grams, while lemmatization was applied to determine the base forms of words.

3.5 Feature extraction

This represents the extraction of lexical features such as n-grams and transforming them into a feature set that is usable by a machine learning classifier. It plays a crucial role in text classification and directly influences the text classification model [26]. Term Frequency-Inverse Document Frequency (TF-IDF) is a popular feature extraction method commonly used in text classification and sentiment analysis. TF-IDF evaluates how important a word is to a document in a dataset by converting textual representation of information into a vector space. This study applied a TF-IDF Vectorizer Python module of Scikit-learn to extract TF-IDF. First, we trained our classifier using unigram and bigram as the feature set to represent context in the Twitter data. The features are tested with the TF-IDF and trained on the classifier. After this, the trained classifier is used in predicting the test data.

3.6 Sentiment classification using machine learning

The next step after the feature extraction is to feed the feature vectors into the machine learning classifiers to perform sentiment classification. We classified the vaccine tweets using Logistic Regression (LR), Support Vector Machine (SVM), Naïve Bayes (NB), and Random Forest (RF) machine learning models, and their performances were compared. Scikit-learn python library, an open-source machine learning package that provides access to machine learning classification algorithms, was used. In each experiment, the training set is used to optimize and train the machine learning algorithms, while the test set is used to evaluate the performance of the models.

3.7 Performance evaluation of machine learning classifiers

The test data was evaluated to understand better how well the classifiers performed after training. The standard metrics used to evaluate the models include accuracy, precision, recall, and f1-score. The metrics were calculated in terms of positives and negatives. The classification accuracy presents the sum of true positives and true negatives divided by the sum of all data points in the test set. Precision is the number of correctly classified positive examples divided by the total number of examples that are classified as positive. The recall measures the number of correctly classified positive examples divided by the total number of actual positive examples in the test set. The f1-score finds a balance between Precision and Recall and tell how precise and robust the classifiers were. The mathematical representation of the metrics is presented below.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (3)$$

where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives

4 Results and discussions

4.1 Principal result

We performed data analysis from the collected tweets to identify public sentiments, keyword associations, and social media trends related to the Covid-19 vaccine rollout program. We search for insights using descriptive text analysis and data visualization such as word clouds and n-gram representations. Below are brief descriptions of the data analyses on the processed Twitter dataset.

4.1.1 Word cloud representation of tweets

Word cloud was used to visualize how words are distributed across the dataset. The most recurring words provide us insight into how user sentiments about the vaccine rollout program evolved on Twitter over the study period (Figure 2).

The main goal is to examine what trend can be inferred from the word frequency in our Twitter data. The illustration from the word cloud shows that along with the search word 'covid19vaccine', words such as 'vaccine', 'dose', 'first', 'second' had many mentions. These words emphasize the awareness of the number of vaccine doses required to be fully inoculated. Names of the initially approved Covid-19 vaccines (Pfizer, Moderna, AstraZeneca) also dominated Twitter during the study period. Again, some Twitter users highlighted the crucial roles governments, healthcare professionals, and other relevant state institutions played in the vaccination

exercise as words like 'government', 'state', 'doctor', and 'hospital' were among the most frequently used words.

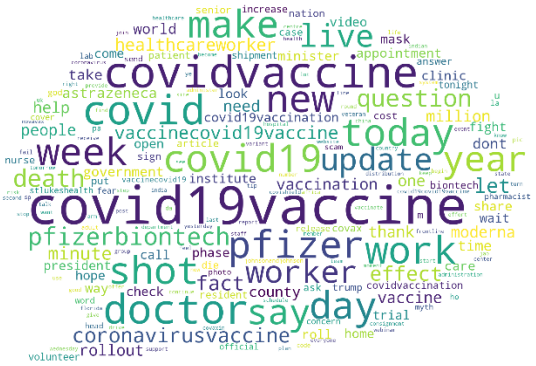


Figure 2: Word cloud representation of Covid-19 vaccine tweets.

	unigram	count	bigram	count	trigram	count
0	vaccine	4048	(first, dose)	243	(received, first, dose)	59
1	httpst	1877	(side, effects)	209	(first, dose, vaccine)	46
2	get	1205	(get, vaccine)	176	(vaccine, immunity, duration)	27
3	first	1124	(vaccination, drive)	138	(vaccine, side, effects)	26
4	vaccination	992	(second, dose)	135	(worlds, largest, vaccination)	22
5	vaccines	860	(moderna, vaccine)	133	(moderna, pfizer, vaccines)	22
6	people	774	(received, first)	117	(largest, vaccination, drive)	22
7	dose	734	(get, vaccinated)	110	(dose, moderna, vaccine)	21
8	health	691	(million, doses)	102	(first, consignment, covishield)	21
9	doses	620	(pfizer, vaccine)	98	(one, step, closer)	19
10	moderna	555	(2nd, dose)	97	(received, second, dose)	18
11	pfizer	528	(johnson, johnson)	97	(first, batch, vaccine)	18
12	vaccinated	523	(healthcare, workers)	93	(received, 2nd, dose)	17
13	getting	499	(dose, vaccine)	85	(health, care, workers)	17
14	one	478	(covishield, vaccine)	81	(virtual, town, hall)	17
15	received	470	(health, workers)	79	(johnson, johnson, pauses)	16
16	new	440	(getting, vaccine)	76	(first, dose, moderna)	14
17	dr	426	(vaccine, rollout)	66	(first, dose, pfizer)	14
18	astrazeneca	351	(astrazeneca, vaccine)	66	(dose, pfizer, vaccine)	14
19	news	350	(1st, dose)	64	(second, dose, vaccine)	14

Figure 3: Top20 n-grams of Covid-19 vaccine tweets.

4.1.2 N-gram representation of tweets

N-grams are a set of consecutive words or a sequence of words in a textual document. To identify the most popular n-grams, we built a list of unique words in our Twitter data and counted each word's occurrences in a corpus. Since the Twitter dataset for this study is about Covid-19 vaccine, Covid-19 related keywords such as 'covid19vaccine', 'covidvaccine', 'covid19', 'covid', and 'coronavirus' were excluded so they do not skew our word frequency analysis. We chose uni-grams ($n=1$), bi-grams ($n=2$), and tri-grams ($n=3$) for further analysis to understand which words were used the most separately and in combination regardless of the grammar structure and semantic meaning. Figure 3 shows the most popular n-grams related to the Covid-19 vaccine tweets. These n-grams highlight how vaccine-related themes such as 'vaccine distribution,' 'vaccine administration,' and 'health engagements' dominate Twitter discussions during the study period.

From the bi-gram, phrases such as the 'first dose', 'second dose', 'receive first' imply some people have already received their first or second dose of the vaccine. The phrase 'side effects' also gained significant recognition among Twitter users over the period. This discussion signifies the fear of potential vaccine side effects that could put the vaccination program at risk. From the tri-gram, phrases such as the 'get first dose', 'received second dose', and 'one step closer', indicate how well people have embraced the vaccination process. Also, phrases such as 'first consignment covishield', 'largest vaccination drive', and 'world largest vaccination', emphasize the global perspective of the fight against the Covid-19 pandemic. The phrase 'vaccine immunity duration' raises concern about how long Covid-19 vaccine-induced immunity will last.

4.1.3 Sentiment classification

We performed sentiment classification with four different machine learning classifiers: LR, RF, SVM, and NB, on the dataset and their performances were evaluated. To ensure that the models were learning the patterns in the data and not fitting to the noise, we implemented k-fold cross-validation technique to determine the efficiency of the classifiers. In k-fold cross-validation, the dataset is divided into k subsets which are repeated k times. For each iteration, the model is trained using k subset as the training sample and the resulting model validated on the remaining part of the data. All four different classifiers were cross-validated five (5) times and the experimental result is illustrated in Table 1.

Table 1: Cross-validation.

Runs	LR	RF	SVM	NB
1	80.75	81.38	82.79	77.01
2	81.54	82.49	81.24	78.46
3	81.25	83.26	80.23	78.90
4	81.86	81.17	82.81	77.65
5	82.33	82.41	83.74	78.14

Table 2: Evaluation of the ML Models.

Model	Accuracy	Precision	Recall	F1-score
LR	82.74	85.04	82.74	80.68
RF	83.05	84.56	83.05	81.31
SVM	84.32	84.32	84.32	83.50
NB	77.41	82.83	77.41	72.54

The result shows that the SVM classifier reaches the highest accuracy mark of 83.74 while the Naïve Bayes classifier has the lowest accuracy of 78.90 among all the classifiers (Table 1). Similarly, the predictive accuracy of the classifiers is determined to find out which model perform best in classifying the Covid-19 vaccine tweets (Table 2). The illustration in Figure 4 clearly identified SVM as the best-fit machine learning classifier on the Covid-19 vaccine Twitter dataset.

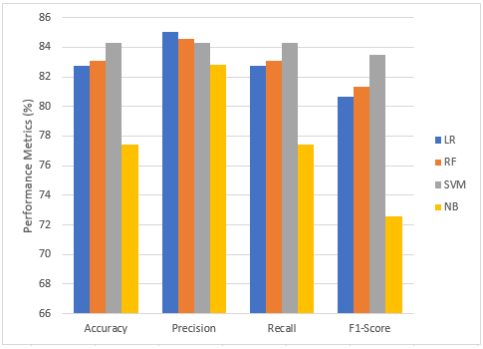


Figure 4: Performance of different ML classifiers on Covid-19 vaccine Twitter data.

4.2 Comparison to prior works

Our findings are consistent with studies using social media data to assess public responses to Covid-19. Compared with a study that implemented a Naïve Bayes model to analyze Twitter sentiments concerning Covid-19 [12], our study demonstrates that machine learning algorithms could be leveraged to study the evolving public discourse and sentiments during the Covid-19 vaccine rollout program.

Some prior Covid-related studies have described themes such as mask-wearing, social distancing, regular washing of hands, and the need for Covid vaccine as the most effective measures to stop further spread of the Covid-19 virus [4], [6]. Our study identifies a new trend of Covid-19-related discussions on Twitter during the study period. These discussions mainly focused on: (1) health information about the vaccine, (2) vaccine distribution and administration, (3) the number of vaccine doses required for immunity, and (4) questions about vaccine availability. Together with other vaccine education efforts, these themes are essential for the overall success of the vaccination program. Our n-grams were consistent with Covid-19-related studies [6], [24] that examine discussions and sentiments that emerged on Twitter and concerns about the safety measures to adopt when reopening from lockdown.

4.3 Practical implications

This study set out to examine trends that can be inferred from the targeted Twitter dataset. The study demonstrates that most of the collected tweets represent positive sentiments, which indicates the public's overall confidence in reaction to the Covid-19 vaccine rollout program. However, the n-gram representation results also suggest that a significant proportion of the public expresses negative sentiments about the Covid-19 vaccine on Twitter. There were questions about the vaccine's safety and efficacy, as the potential side effects dominate discussions over the period. Additionally, there were concerns about the vaccines' long-term protection against Covid-19. As the Covid-19 vaccine rollout program continues, more efforts from governments and relevant authorities are required to answer ongoing questions regarding vaccine choices, vaccine hesitancy, vaccine side-effects, and the durability of the immunity response to Covid-19 vaccines.

Twitter remains a great source of information for many and can be used to explore the levels of public awareness and sentiments about the Covid-19 and its related themes. During a pandemic where people may be confined to their homes, perceptions of people about the vaccine are more likely to be inferred from social media and information online [27]. Due to the need for a worldwide Covid-19 vaccination program, understanding the threat of vaccine misinformation on social media and its negative influence on the general public's vaccine uptake is important. To encourage a positive vaccine attitude, it is suggested that social media firms devise schemes on how to promote accurate vaccine information while removing vaccine misinformation from their platforms. Besides, governments worldwide should engage their citizens on key, accurate, and timely health information regarding the vaccination program.

As evidence is still evolving, understanding the worldwide Covid-19 vaccination program's potential challenges is crucial in helping governments and relevant institutions develop schemes that will allay citizens' apprehensions about Covid-19 inoculations.

While machine learning classifiers perform relatively well on the dataset, our analysis was limited to a small collection of tweets expressed in English. Our findings may not be a true reflection of public sentiments on the vaccination program due to the risk of missing out on vital information available from tweets generated in other languages. Future work might consider the evaluation of large-scale Covid-19 vaccine Twitter datasets using deep learning models.

5 Conclusion

Our work focused on examining public discourse and reactions on Twitter concerning the Covid-19 vaccine rollout program. Popular machine learning algorithms were applied to predict sentiments from the collected tweets. Most Twitter users were optimistic during the study period, although some negative sentiments threatened the overall success of the vaccine implementation program. Also, we identified a new Covid-related discussion trend that focuses on vaccine distribution and administration, the number of vaccine doses required for immunity, and other health information about the vaccine. These findings can be a handy tool to help policymakers and relevant authorities anticipate the appropriate measures that can be taken to mitigate any potential challenges to the vaccine rollout program. The pressing need to achieve herd immunity against Covid-19 requires timely reactions to address the concerns of the general public to boost trust and confidence in the vaccination program.

Data availability

The model, code, and the dataset are available in the GitHub repository: https://github.com/askasnr/Covid-19-vaccine-tweets-Dataset

Acknowledgements

This study was supported by the National Key R\&D Program of China, Grant No. 2018YFA0306703.

Conflict of interest

The authors declare that the research was conducted without any commercial or financial relationships that could be interpreted as a potential conflict of interest.

References

[1] R. Aguas, R. M. Corder, J. G. King, G. Goncalves, M. U. Ferreira, and M. G. M. Gomes, "Herd immunity thresholds for SARS-CoV-2 estimated from unfolding epidemics," medRxiv, 2020. https://doi.org/10.1101/2020.07.23.20160762

[2] S. Chawla and M. Mehrotra, "Impact of emotions in social media content diffusion," Informatica, vol. 45, no. 6, 2021. https://doi.org/10.31449/inf.v45i6.3575

[3] M. Cinelli et al., "The covid-19 social media infodemic," Sci. Rep., vol. 10, no. 1, pp. 1–10, 2020. Cross-reference

[4] W.-Y. S. Chou and A. Budenz, "Considering Emotion in COVID-19 vaccine communication: addressing vaccine hesitancy and fostering vaccine confidence," Health Commun., vol. 35, no. 14, pp. 1718–1722, 2020. https://doi.org/10.1080/10410236.2020.1838096

[5] E. M. Martey, H. Lei, X. Li, and O. Appiah, "Effective Image Representation using Double Colour Histograms for Content-Based Image Retrieval," Informatica, vol. 45, no. 7, 2021. https://doi.org/10.31449/inf.v45i7.3715

[6] J. Xue et al., "Twitter Discussions and Emotions About the COVID-19 Pandemic: Machine Learning Approach," J. Med. Internet Res., vol. 22, no. 11, p. e20550, 2020. https://doi.org/10.2196/20550

[7] J. Samuel, G. G. Ali, M. Rahman, E. Esawi, Y. Samuel, and others, "Covid-19 public sentiment insights and machine learning for tweets classification," Information, vol. 11, no. 6, p. 314, 2020. https://doi.org/10.3390/info11060314

[8] Addo Prince Clement et al., "COVID-19 and Actor Well-being: A Serial Mediated Moderation of Mask Usage and Personal Health Engagement," Asian J. Immunol., vol. 5, pp. 44–53, 2021. https://doi.org/10.1080/21645515.2021.2008729

[9] A. D. Dubey, "Twitter Sentiment Analysis during COVID19 Outbreak," Available SSRN 3572023, 2020. http://dx.doi.org/10.2139/ssrn.3572023

[10] F. H. Rachman and others, "Twitter Sentiment Analysis of Covid-19 Using Term Weighting TF-IDF And Logistic Regresion," in 2020 6th Information Technology International Seminar (ITIS), 2020, pp. 238–242. https://doi.org/10.1109/itis50118.2020.9320958

[11] S. Naz, A. Sharan, and N. Malik, "Sentiment classification on twitter data using support vector machine," in 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), 2018, pp. 676–679. https://doi.org/10.1109/wi.2018.00-13

[12] K. H. Manguri, R. N. Ramadhan, and P. R. M. Amin, "Twitter Sentiment Analysis on Worldwide COVID-19 Outbreaks," Kurdistan J. Appl. Res., pp. 54–65, 2020. Cross-reference

[13] D. Suleiman, W. Etaiwi, and A. Awajan, "Recurrent Neural Network Techniques: Emphasis on Use in Neural Machine Translation," Informatica, vol. 45, no. 7, 2021. https://doi.org/10.31449/inf.v45i7.3743

[14] W. Etaiwi, D. Suleiman, and A. Awajan, "Deep Learning Based Techniques for Sentiment Analysis: A Survey," Informatica, vol. 45, no. 7, 2021. https://doi.org/10.31449/inf.v45i7.3674

[15] S. Elbagir and J. Yang, "Sentiment Analysis of Twitter Data Using Machine Learning Techniques and Scikit-learn," in Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence, 2018, pp. 1–5. https://doi.org/10.1145/3302425.3302492

[16] S. K. Akpatsa, X. Li, and H. Lei, "A Survey and Future Perspectives of Hybrid Deep Learning Models for Text Classification," in International Conference on Artificial Intelligence and Security, 2021, pp. 358–369. https://doi.org/10.1007/978-3-030-78609-0_31

[17] H. Jelodar, Y. Wang, R. Orji, and H. Huang, "Deep sentiment classification and topic discovery on novel coronavirus or covid-19 online discussions: Nlp using lstm recurrent neural network approach," arXiv Prepr. arXiv2004.11695, 2020. https://doi.org/10.1109/JBHI.2020.3001216

[18] A. Roy and M. Ojha, "Twitter sentiment analysis using deep learning models," in 2020 IEEE 17th India Council International Conference (INDICON), 2020, pp. 1–6. https://doi.org/10.1109/indicon49873.2020.9342279

[19] Y. Zhang, D. Song, P. Zhang, X. Li, and P. Wang, "A quantum-inspired sentiment representation model for twitter sentiment analysis," Appl. Intell., vol. 49, no. 8, pp. 3093–3108, 2019. Cross-reference

[20] M. Mansoor, K. Gurumurthy, V. R. Prasad, and others, "Global Sentiment Analysis Of COVID-19 Tweets Over Time," arXiv Prepr. arXiv2010.14234, 2020. https://doi.org/10.48550/arXiv.2010.14234

[21] S. Boon-Itt and Y. Skunkan, "Public perception of the COVID-19 pandemic on Twitter: sentiment analysis and topic modeling study," JMIR Public Heal. Surveill., vol. 6, no. 4, p. e21978, 2020. https://doi.org/10.2196/21978

[22] J. Xue, J. Chen, C. Chen, C. Zheng, S. Li, and T. Zhu, "Public discourse and sentiment during the COVID 19 pandemic: Using Latent Dirichlet Allocation for topic modeling on Twitter," PLoS One, vol. 15, no. 9, p. e0239441, 2020. https://doi.org/10.1371/journal.pone.0239441

[23] P. C. Addo, F. Jiaming, N. B. Kulbo, and L. Liangqiang, "COVID-19: fear appeal favoring purchase behavior towards personal protective equipment," Serv. Ind. J., vol. 40, no. 7–8, pp. 471–490, Jun. 2020. https://doi.org/10.1080/02642069.2020.1751823

[24] M. E. Ahmed, M. R. I. Rabin, and F. N. Chowdhury, "COVID-19: Social media sentiment analysis on reopening," arXiv Prepr. arXiv2006.00804, 2020. https://doi.org/10.48550/arXiv.2006.00804

[25] Y. HaCohen-Kerner, D. Miller, and Y. Yigal, "The influence of preprocessing on text classification using a bag-of-words representation," PLoS One, vol. 15, no. 5, p. e0232525, 2020. https://doi.org/10.1371/journal.pone.0232525

[26] V. Singh, B. Kumar, and T. Patnaik, "Feature extraction techniques for handwritten text in various scripts: a survey," Int. J. Soft Comput. Eng., vol. 3, no. 1, pp. 238–241, 2013. Cross-reference

[27] W. H. Organization and others, "Behavioural considerations for acceptance and uptake of COVID-19 vaccines: WHO technical advisory group on behavioural insights and sciences for health, meeting report, 15 October 2020," 2020. https://iris.who.int/handle/10665/337335

Personalized Health Framework for Visually Impaired

Megha Rathi, Shruti Sahu, Ankit Goel and Pramit Gupta
E-mail: megha.rathi@jiit.ac.in, shrutisahu1196@gmail.com, goelankit1995@gmail.com, pramitgupta22@gmail.com
Department of Computer Science & IT, Jaypee Institute of Information Technology, Noida, India

Keywords: android application, computer vision, deep learning, object recognition, region-based convolutional neural networks, disease prediction, voice assistant

Received: August 23, 2019

Vision is one of the most essential human sense. The life of a visually impaired person can be transformed from a dependent individual to a productive and functional member of the society with the help of modern assistive technologies that use the concepts of deep learning and computer vision, the science that aims to mimic and automate human vision to provide a similar, if not better, capability to a computer. However, the different solutions and technologies available today have limited outreach and end users cannot fully realize their benefits. This research work discusses an easily-operable and affordable android application designed to aid the visually impaired in healthcare management. It also aims to resolve the challenges faced due to visual impairment in daily life and uses the concepts of computer vision and deep learning. Broadly, the application consists of the following modules: object recognition in immediate surroundings using region-based convolutional neural networks, disease prediction with the help of symptoms, monitoring of health issues and voice assistant for in-app interaction and navigation.

Povzetek: Razvita je androidna aplikacija za pomoč ljudem z okvarami vida.

1 Introduction

In 2014, the World Health Organization estimated 285 million people to be visually impaired worldwide, out of which 39 million are blind and 246 have low vision [1].About 90% of this population lives in low-income settings. Visually impaired people face several difficulties in their daily lives. From reading to navigation, be it in familiar or unfamiliar environments, every task is a new challenge [2]. Computer Vision is the science that aims to mimic and automate human vision and provide a similar, if not better, capability to a machine or computer [3]. With the combination of machine learning and computer vision, various technologies have been developed to help substitute for visual impairment in some manner or the other and enable people to live more independently. However, the problem of lack of accessible and affordable solutions to ease the routine of the visually impaired still persists. Expensive wearable's, usually powered by artificial intelligence, are the current advancements in computer vision. Aiming towards affordability, a simpler approach is adapted. Android platform is used to develop an application and make it accessible in hand-held mobile android devices which are essential in developing an aid for the visually impaired [4].

The objective of the proposed application significantly in facilitating an independent & healthy life include easy operability i.e. voice assistant for in health routine, object recognition in images captured via device camera, diagnosis of common diseases via symptoms spoken by the user. The modern digital era has revolutionized data storage. Huge volumes of data are available today that can be beneficially utilized for processing and automation an essential feature of health management is incorporated in the android application, along with health monitoring measures like body mass index (BMR), calorie intake, and daily steps. Before the breakthrough of deep learning in 2000s, PASCAL and conventional computer vision techniques like example-based learning, discriminatively trained part were used for object recognition structure as early deep learning based algorithms (for instance, R relevant regions, labeling each proposed region and cumulating outcomes from all output for the image [5] Figure 1 illustrates the classification error in top 5 models of the object recognition & image classification task of ImageNet Large Scale Visual Recognition [6]. It can be observed that deep learning even surpassing human error in 2015. A deep learning Region-based Convolutional Networks (R 'Methodology') is used to detect and identify objects from the images captured by the android device [7]. Figure 1. Classification Error of top 5 models of ILSVRC application is to provide an affordable solution significantly in facilitating an independent & healthy lifestyle for the visually-impaired voice assistant for in-app interaction and navigation, management of daily health routine, object recognition in images captured via device camera, diagnosis of common diseases The modern digital era has revolutionized data storage. Huge volumes of data are available today that can be beneficially utilized for processing and automation [8]. Automation of disease prediction is an essential feature of health management is incorporated in the android application, along with health mass index (BMI), body fat percentage (BFP) and basal metabolic rate intake, and daily steps. Before the breakthrough of deep learning in

2000s, PASCAL and conventional computer vision based learning, discriminatively trained part-based models and selective search [9]. Various algorithms shared similar structure as early deep learning based algorithms (for instance, R-CNN) i.e. identification and proposal of each proposed region and cumulating outcomes from all regions to produce. Illustrates the classification error in top 5 models of the object recognition & image classification Large Scale Visual Recognition Competition (ILSVRC) between 2010 and 2016 . It can be observed that deep learning-based models fare much better than others, even surpassing human error in 2015. Salient features navigation, management of daily health routine, object recognition in images captured via device camera, diagnosis of common diseases the modern digital era has revolutionized data storage. Huge volumes of data are available today that can. Automation of disease prediction is an essential feature of health management is incorporated in the android application, along with health and basal metabolic rate before the breakthrough of deep learning in 2000s, PASCAL and conventional computer vision based models and selective search. Various algorithms shared similar CNN) i.e. identification and proposal of regions to produce illustrates the classification error in top 5 models of the object recognition & image classification even 2010 and 2016 based models fare much better than others, (elaborated under 'Methodology') is used to detect and identify objects from the images captured by the android device .

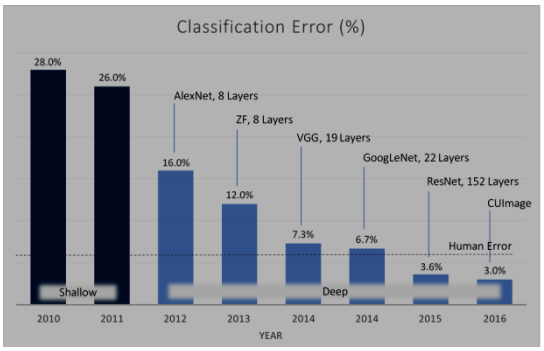


Figure 1: Classification error of top 5 models of ILSVRC.

2 Background study

Significant research has been carried out in the domain of developing android based apps for visually impaired patients. In recent research, authors have presented a novel technique for visually impaired user assistance using guidance mode activity [10]. A proposed system is comprised of gathering sensors data held or worn by a visually impaired user, these sensors collect data and further redirect it to the server for processing. The server contains a processor which is built using advanced artificial intelligence techniques. Basis functionality of the server is to send data extracted from sensors to an agent device, this device further gives data for vision on an agent interface. Finally, an agent can assist the low vision user in real-time navigation through audio signals or other feedback.

In another research, a system named "Smartvision" is developed for the navigational activity of blind users. With the recent development in advanced computational technologies, one can create models which can assist blind users in their daily routine tasks. The main emphasis of the proposed model is to assist no vision users so that they can easily navigate into strange environments indoor or outdoor [11]. A user-friendly interface is developed for the development of "Smartvision" app for blind users. The proposed model utilizes the concepts of computer vision and machine learning to achieve the objective.

In yet another novel research work, an android healthcare app known as "mHealth" is developed to assist blind users in their health tasks. Android-based technologies are gaining popularity in healthcare these days and could serve as a boon for visually impaired patients [12]. Smartphones allow extracting medical data from health sensors and then using the extracted data for further health analysis of the patient. Visually impaired people have bad health conditions than normal people because of poor accessibility of medical data and if a compatible app is present they can regularly monitor their health status and take preventive action accordingly. "mHealth" device is proven to be a boon as it captures patient health condition via sensors i.e. it can monitor blood pressure, diabetes level, etc. and further suggest medical action accordingly. A smartphone is the simplest way to access health sensors, and android developers can create IoT-based android applications to make medical sensors fully accessible on mobile devices.

A study conducted in another significant paper developed a novel technique for door detection for the strange environment to no vision user [13]. Earlier algorithms were designed to detect door-like objects to known environments only where particular characteristics of the door are fed as an input. In this study, the author presented a novel image-based door detection technique whose main objective is to find outdoors based on consistent characteristics like edges and corners. Animated door structure is created for finding out the doors by merging features of edges and corners. This algorithm is also able to distinguish other door-like objects from the door like it can distinguish bookshelf or cabinet from the door. The proposed technique is validated under different unknown situations over multiple ranges of door shapes, colors, textures, illumination, and views.

Another contribution presented recent advances in advanced homecare technologies for blind people. Assistive systems are required for blind users to provide information and allow them to safely move and complete their daily routine and health tasks and explore unknown environment [14]. For achieving this objective various new IOT based and other computational technologies have been experimented with to provide the solution to basic major problems of a blind user. For blind people, accessibility and self-determination are elementary

requirements so it is desirable to provide them insight regarding skills to lead a happy normal life like other human creatures. For blind self-determination and accessibility means they can seek employment, get a good education, get normal health routines, and social life. So, this study focuses on providing details about various smart home care solutions for blind users.

An Integrated mobile-based healthcare framework is used these days by medical professionals for healthcare tasks [15]. The usage of smartphones is expanding day by day and in the future, it incorporates every single clinical task. An easy-to-use interface makes healthcare apps be used by even illiterate persons. Effective utilization of advanced computational techniques, proper verification, and validation are substantially required to ensure a good standard of quality, security, and privacy for using these mobile-based healthcare applications. With the execution of all such quality standards in mobile-based healthcare tools, the main emphasis is on providing correct, relevant, and appropriate information to the user for achieving the objective of healthcare outcomes.

Within this paper extra emphasis has been put on enhancing the assistive technologies used for visually impaired people [16]. Assistive technology is used by many researchers to assist blind users but academicians are not paying attention to deriving new applications by amalgamating assistive technologies with computational intelligence for creating hybrid applications to assist the blind user. The socio-psychological feature is the main obstacle in the adoption of assistive technology for further research. Finding and pointing out those features can enhance the adoption of assistive technology by academicians. Visually impaired patients are heavily relying on these technologies for survival, this research focuses on finding out the socio-psychological feature that impacts assistive technology.

Another significant contribution provided in the study, in which authors generate particle libraries for drug discovery using recurrent neural networks [17]. The recurrent neural network can be proven to be very effective for creating systems for molecular structures, features of molecular structures coordinate positively with the features used to train the model. The proposed work proposed a model with a tiny molecular dataset that is dynamic in opposition to the target. The proposed model can outline the method for producing a huge set of molecules for discovering drugs.

Research conducted the study over a survey of Wearable Obstacle Avoidance Electronic Travel Aids for Blind [18]. Numerous wearable devices are developed to assist blind people in navigation in a known and strange environment. Broadly these navigational devices are classified into three categories: automatic inclination aid, electronic progression aid, and posture location tools. This research work presents the survey of portable navigational devices for visually impaired patients. This can help researchers in gaining insights to present assistive technologies for blind people and further amendments in these technologies.

Recent developments in computer vision include expensive wearables like Aira, MyEye, eSight, and BrainPort. Inspired by Google Glasses, Aira, developed is a pair of spectacles fitted with a camera that transfers the current field of view to a visually-abled person i.e. provides visual interpreting services [19]. Unlike Aira, artificial intelligence is used in MyEye, launched by which interprets visual data from a small camera into an audio earpiece. eSight, designed by Conrad [20] for the partially blind, uses a high-resolution camera to enlarge images and project them on an OLED screen in front of the wearer's eyes.

Object recognition and image processing are major functionality of computer vision systems. Deep neural networks, an arrangement following deep learning are networks with multiple hidden layers. However, as the depth of the network grows and it begins to converge, accuracy becomes constant and then reduces rapidly. A residual learning framework, ResNet, to easily train substantially deeper neural networks was presented by He et.al. [21]. Consider two neural network layers x and y. Instead of direct mapping $y=F(x)$, ResNet adapts a residual function $F(x)$ to map $y=F(x)+x$. The logic is that in the case of identical layers $y=x$, it is simpler to obtain $F(x)=0$ than $F(x)=x$. Hence, layers are framed by learning residual functions.

The author proposed a combination of convolutional neural networks with region proposals, called Region-based Convolutional Networks (R-CNN) [22]. The model inputs an image proposes bounding boxes or region proposals using selective search and checks if each proposal is an object or not. A Support Vector Machine (SVM) is used for the classification of region proposals or bounding boxes which are tightened using linear regression. The architecture has significantly better results than earlier CNN-based architectures on datasets like ImageNet.

Proposed by Girshick et.al. [23], Fast R-CNN is a simplified version of R-CNN. It clubbed all the components of R-CNN into a single network by adding a softmax layer and a linear regression layer parallel to the output layer of CNN. Softmax acts as a classifier in place of SVM and linear regression tightens the bounding boxes. In the same year, Faster R-CNN was proposed by Ren et.al. [24] to accelerate the region proposal process. The model trained a single CNN to implement region proposals and classification by adding a fully convolutional network, named Regional Proposal Network (RPN) on top of CNN. Anchor boxes are some common aspect ratios that are randomly generated to fit objects in an image. RPN slides a window over the CNN feature map outputs a bounding box per anchor and probability if it contains an object. These boxes are then passed to Fast R-CNN for classification.

A feature pyramid is a fundamental component in recognition systems for detecting objects at different scales which is troublesome for tiny objects. The proposed work [25] used the framework of convolutional neural networks to construct feature pyramids. This architecture, called a Feature Pyramid Network (FPN), comprising of a bottom-up (downscale) and a top-down

Year	R-CNN	FastR-CNN	Faster R-CNN
Year of Conception	2014	2015	2016
Input	Image	Image with region proposals	Image (region proposals not needed)
Output	Bounding boxes and labels for each object in the image.	Object classification of each region with more constricted bounding boxes.	Classifications and bounding box coordinates of objects in the images
Components	CNN (feature extractor), SVM (classifier), linear regressor (tighten bounding boxes).	Single CNN (feature extractor) having softmax layer (classifier) with linear regression layer (output bounding boxes) in parallel.	CNN (feature extractor), RPN (output bounding boxes per anchor and probability), Fast R-CNN (classifier, tighten bounding boxes)
Pooling	Max Pooling	RoI (Region of Interest) Pooling	--
Region Proposal	Selective Search	Selective Search	Region Proposal Network

Table 1: Comparison of existing approaches for object recognition.

(upscale) pathway. The bottom-up pathway uses ResNet for construction. Upsampling in the top-down pathway is done by convolutional filters. FPN shows significant improvement as a feature extractor in several applications.

In another recent study, for object detection deep convolutional neural networks (CNNs) are trained as N-way classifiers [26]. HMod Fast R_CNN is implemented for upgrading the overall computational power of R-CNN. In yet another novel work in the domain of disease prediction, authors have implemented machine learning techniques for the prediction of Dengue [27]. From the results it is concluded that LogitBoost ensemble technique is the most accurate one with accuracy equals to 92%. In the work [28] analysis on diabetes data is performed for the comparison of various machine learning techniques. In another significant study in the domain of disease prediction, authors are forecasting the casual effect association between Coronary Obstructive Pulmonary Disease and Cardiovascular Diseases [29].

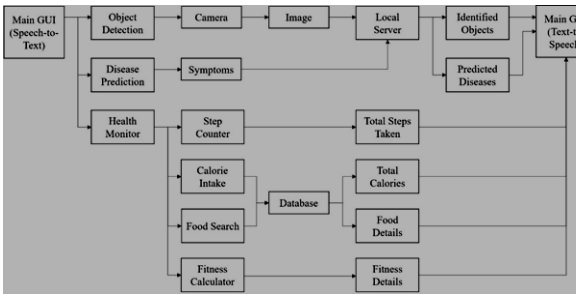


Figure 2: Control flow of the proposed android application.

3 System architecture

Visual impairment may be the result of an injury, disease, or some other condition. Visually impaired face problems in self-navigating outside known environments. They need to memorize details about their home environment. Furniture and other large obstacles must remain in one location to prevent injury. Individuals with low vision may find browsing websites problematic due to small fonts etc. They might also need to enlarge a screen significantly. Operating gadgets like mobile phones and tablets is also a challenge. The life of a visually impaired person can be transformed from a dependent individual to a productive and functional member of society who can read and write, use mobile phones, computers, and other gadgets efficiently with the help of modern assistive technologies.

Today, different solutions and technologies are present which have the potential to bring substantial change and improvement in the lives of people with visual impairment, especially the aging population. However, they have limited outreach, and end-users cannot fully realize their benefits. They must acquire essential awareness and know-how of using these technologies, as well as have the resources for obtaining them. Moreover, amidst abundant health management applications available today, applications catering to the visually impaired section of society are absent.

So, the need of the hour is to design affordable and accessible solutions to ease and significantly improve the daily routine and health management of the visually impaired. The main objective is to develop an application that would assist significantly in facilitating an independent & healthy lifestyle for the visually impaired.

- Object Recognition - A real-time object recognition module via the device camera is proposed. Objects will be identified and communicated to the user when they are in the camera's field of vision. Applying the concepts of computer vision, a deep learning-based architecture (Feature Pyramid Network with region Proposal Network) is used to identify and classify objects from the incoming stream of images.
- Disease Prediction - The user will be able to tell their symptoms in case he/she feels unwell and the same

will be checked against the data of various diseases. A graph database is implemented on Neo4j. The voice input of the user is queried and matched against the database to return one or multiple symptoms.

- Monitoring of Daily Health Routine - Daily health routine is monitored by calculating steps are taken, calorie intake, BMI, body fat percentage, and basal metabolic rate. Calorie intake for various food items is available to plan diet accordingly.
- Voice Assistant - For in-app interaction and navigation, a voice assistant is available. Google Speech-to-Text API is used to develop a text-to-speech module to communicate to the user and a speech-to-text module with natural language processing to process the user's input.

Figure 2 illustrates the flow of control of the android application. Initially, the user can select from the three modules - Object Recognition, Disease Prediction, and Health Monitor. Object Recognition is followed by capturing an image, which is sent to the local server hosting the deep learning model and then the identified objects are returned to the application and then to the user in the voice format. Disease Prediction is followed by taking voice input for symptoms, which are converted into text, sent to the local server for prediction of possible diseases, and returned to the user. A health monitor is comprised of keeping fitness records and the nutrition values of the food intake by the user.

4 Methodology

The main objective of the proposed health-based framework is to develop an android based app for visually impaired patients. The main modules of the proposed healthcare management tool for visually impaired patients are: 1) Object Detection, 2) Disease Prediction, 3) Real-time Monitoring of health Issues and 4) Voice assistant for In-App communication and Navigation. All these modules along with Dataset description are discussed in detail in this section.

4.1 Dataset description

A disease-symptom dataset was gathered from "WebMD" [30] and "MedicineNet"[31] . WebMD, founded in 1996, is one of the most-visited healthcare websites and contains data about various diseases, their corresponding symptoms, drug information, etc. MedicineNet is another site owned and operated by the WebMD consumer network. US Board-certified physicians and healthcare professionals maintain up-to-date information regarding diseases, symptoms, drugs, and remedies to the general masses in an easily understandable language. A graph database is created from the crawled data and the graph contains 149 disease nodes, 404 symptoms nodes, and 2126 "may cause" relationships i.e. un weighted directed edges from symptoms to corresponding disease. Also, the model was trained till 33 epochs on the Microsoft Common Objects in Context dataset [32] which contains 330 K images, 1.5 million object instances and 91 object categories.

4.2 Disease prediction

The user will be able to tell their symptoms in case he/she feels unwell and same will be checked against the data of various diseases. A graph database is implemented in Neo4j.Neo4j is a graph database management system that contains nodes with directed edges or relationships between them. These nodes and edges can have labels and any number of attributes. Labels can be used to narrow searches. Querying is done using Cypher Query Language.

Neo4j is scalable and supports replication. It also supports ACID (Atomicity, Consistency, Isolation, and Durability) rules. Atomicity means the database considers all transaction operations as one whole unit or atom. Consistency guarantees that a transaction never leaves your database in a half-finished state. Isolation keeps transactions separated from each other until they're finished. Durability guarantees that the database will keep track of pending changes in such a way that the server can recover from an abnormal termination.

A graph database on Neo4j was implemented on the dataset to eliminate sparsity. The graph contains 149 disease nodes, 404 symptom nodes, and 2126 "(: Symptom)-[: MAY_CAUSE]->(: Disease)" relationships i.e. unweighted directed edges from symptoms to the corresponding disease. Weighted relationships between symptoms and diseases could not be obtained due to the unavailability of proprietary datasets.

The Neo4j database was linked with Python using Py2neo driver and queried using Cypher Query Language (CQL). Symptoms spoken by the user were matched with corresponding diseases and returned. The Python code was hosted on a Flask server to send information back and forth between the android application and the database.

4.3 Object recognition

The proposed model consists of a 51-layer Residual Network (ResNet-51) as the backbone for the Feature Pyramid Network (FPN). All 4 convolutional blocks of the Residual Network are used as a base for FPN. It has multiple prediction and upsampling layers. It has a lateral connection between the bottom-up pyramid and the top-down pyramid. It applies a 1x1 convolution layer before adding each layer. A 3x3 Conv layer is then applied and the result is used as a feature map by upper layers.

The pyramid of convolutional activation maps generated by FPN is passed to Region Proposal Network (RPN), eliminating the bottleneck of hardcoded algorithms like EdgeBoxes to get region proposals. RPN works as a first pass on the image and makes the binary decision if a region contains an object or not. It also outputs the confidence it has over the proposal.

The proposed regions are sent to the RoI Align layer which maps the proposed regions in the image to the convolutional features maps of FPN. RoI Align layer, unlike RoI Pooling layers, does not quantize the input

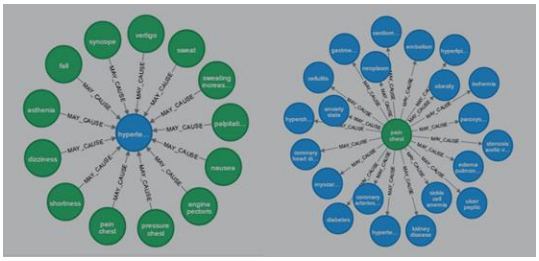


Figure 4: Relationships of “hypertensive disease” disease node and “pain chest” symptom node.

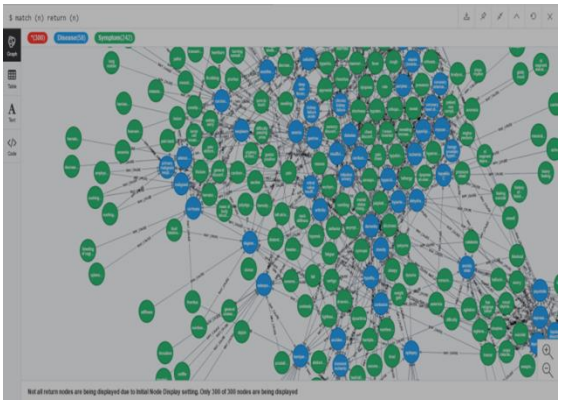


Figure 5: Neo4j graph database containing diseases, symptoms and their relationships.

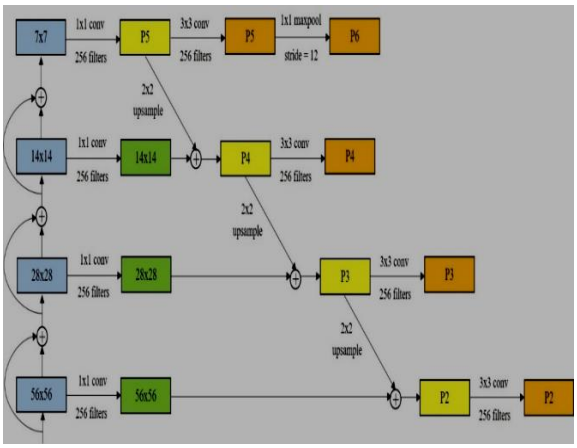


Figure 6: Feature pyramid network.

space by not rounding off to floor value i.e. it uses a/16 operation instead of [a/16] operation. This fixes the location misalignment issue.

The feature maps of candidate regions are then used by classification, bounding box, and mask predicting heads. Time distributed Keras layer is used to pass every feature map of the FPN to the heads. The entire heads share the first 2 fully connected layers which are implemented as convolutional layers; the first one has a dropout of 0.5 for regularization. The second convolutional layer has average pooling across the depth of the activation map making it function as fully connected layers.

Classification head: contains a fully connected layer that outputs logits for the proposal belonging to every

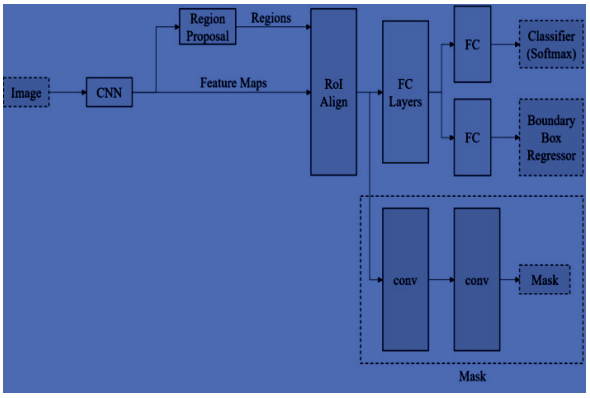


Figure 3: Architecture of object recognition model.

class or background class. The number of outputs is equal to a number of classes + 1 (background class). The softmax layer is used to calculate unnormalized log probabilities of the input belonging to every class and background class.

Bounding Box Regressor Head: Outputs bounding box deltas over the proposed bounding box by RPN. It consists of fully connected layers.

Mask Predicting head: The mask branch generates a mask of dimension 24 x 24 for each class for each proposed region. The total output is of size (number of classes + 1) * number of proposals. As the model tries to learn a mask for each class, masks are generated for every class. A mask is just a 24 x 24 binary grid which signifies the presence and absence of object instance for that pixel. It is implemented as 4 convolutional layers with batch normalization and ReLU activation followed by a fractionally stridden convolutional layer that upsamples the input. That is fed to another convolutional layer which has sigmoid activation to squash the input to -1 to 1. Loss is calculated using the mask associated with the ground truth class.

Monitoring of Daily Health Routine: Daily health routine is monitored by calculating steps are taken, calorie intake, BMI, body fat percentage, and basal metabolic rate. Calorie intake for various food items is available to plan diet accordingly. The Body Mass Index (BMI) is calculated using the height and weight of a person. It is defined as the body mass divided by the square of the body height and is universally expressed in units of kg/m2, resulting from mass in kilograms and height in meters. The value of BMI is used to categorize individuals as underweight, normal weight, or overweight. The body fat percentage (BFP) of an individual is the total mass of body fat divided by total body mass, multiplied by 100; body fat includes essential body fat and storage body fat. The body fat percentage is a measure of fitness level since it is the only body measurement that directly calculates a person's relative body composition without regard to height or weight. Metabolism comprises the processes that the body needs to function. Basal metabolic rate (BMR) is the amount of energy per unit of time that a person needs to keep the body functioning at rest.

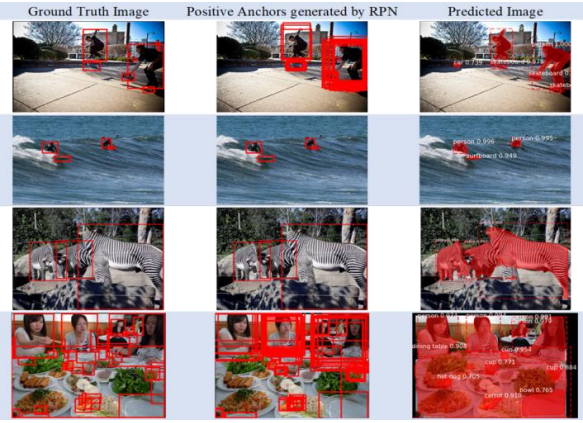


Figure 7: Results of object recognition model.

Voice Assistant: For in-app interaction and navigation, a voice assistant is available. The voice command that is input by the user in the English language is converted into an expression using Google Speech-to-Text API. WordNet is a large lexical database of the English language. Natural Language Toolkit is used to implement Part-of-Speech tagging on the expression. The similarities of the expression with the WordNet database are compared and the resulting command is executed or results of the operation are conveyed to the user using Google Text-to-Speech API.

5 Results and findings

The object recognition model was trained to 33 epochs with 1000 steps each epoch and 50 validation steps on the Microsoft Common Objects in Context dataset which contains 330K images, 1.5 million object instances, and 91 object categories. Mean image subtraction was done to center align data. The system used for training was expanded by an NVidia 1050 Ti OC Graphic Card for higher processing capability.

Object Recognition Model Hyper parameters

- No of epochs trained = 33
- Steps per epoch = 1000
- Validation steps = 50
- Threshold ratio of positive to negative RoI for training = 33
- Minimum probability value to accept a detected instance = 0.7
- Non-Maximum Suppression threshold = 0.3s
- Optimiser: Stochastic Gradient Descent with Momentum
  - Learning Rate = 0.002
  - Learning Momentum = 0.9
  - Weight Decay Regularization Strength = 0.0001

Mean Average Precision (MAP) is the standard single-number performance measure for comparing search algorithms. The MAP (mean Average Precision) score for the proposed model (ResNet-51-FPN) was 51.8%. Generation of feature maps in the proposed model using Feature Pyramid Networks is evaluated against existing models.

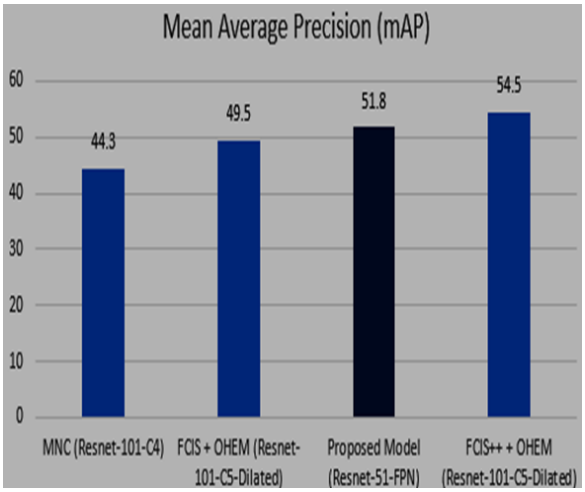


Figure 8: Performance comparison of proposed model with other object recognition models using mean average precision (mAP) score.

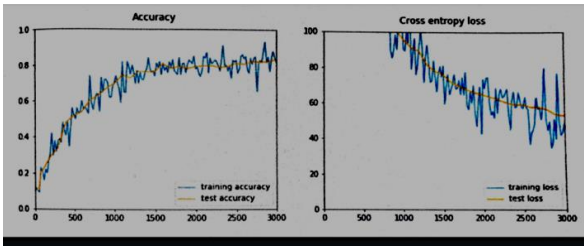


Figure 9: ANN-accuracy and cross entropy loss.

- Multi-task Network Cascades (MNC) [Dai, J., et al. 2015] consists of three networks, respectively differentiating instances, estimating masks, and categorizing objects. These networks form a cascaded structure and are designed to share their convolutional features.
- Fully Convolutional Instance-aware Semantic Segmentation (FCIS)[Li, Y., et al. 2016] proposed semantic segmentation and instance mask proposal. It detects and segments the object instances jointly and simultaneously.

The use of Feature Pyramid Networks to generate feature maps in the model fared better than existing models like MNC and FCIS+OHEM but worse than FCIS+++OHEM due to the lesser number of layers in the Residual Network backbone of the architecture.

5.1 Comparative Analysis between ANN and CNN

In our experiments, ANN of 4 hidden layers of 20 neurons each with Batch Normalization gives 93% accurate results. Adding dropout technique in the model reduces the efficiency as it works for network which can afford to lose neurons. On the other hand, CNN even without over fitting countermeasures achieves accuracy of 98%, and with Batch Normalization and Dropout applied it gives accuracy approximately equals to 99.3%. CNN have repetitive blocks of neurons that are applied across space. At the training time, the weight gradients

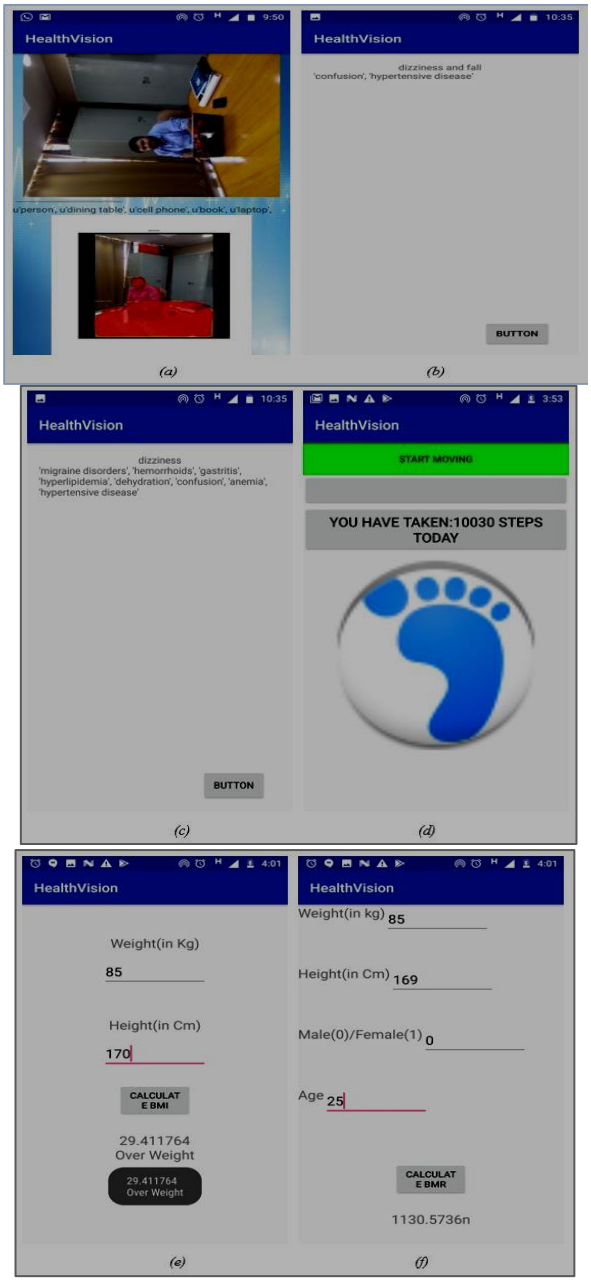


Figure 10: voice automated android application - (a) real-time object recognition (b)disease prediction when symptoms is dizziness (c) disease prediction when symptoms are dizziness and fall (d) no of steps taken daily (e) calculation of BMI (f) calculatio

learned over various image patches are averaged which exploits the spatial or temporal invariance in object recognition.

For object recognition, the use of Feature Pyramid Network (FPN) provide valid proposals for most of the objects in the image while without FPN fails to provide valid proposals and misses out on a majority of objects. The model which has been trained till 40 epochs accurately captures even the small ground truth objects. That is why, FPN is used to generate feature maps for our recognition model. The test accuracy and test losses are visualized in the figures shown below.

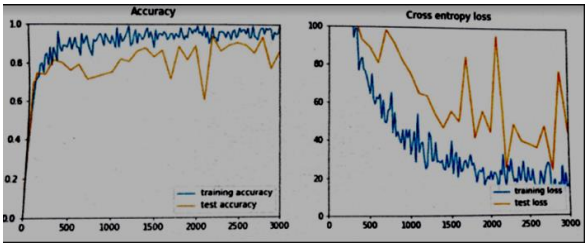


Figure 11: ANN + batch normalization-accuracy and cross entropy loss.

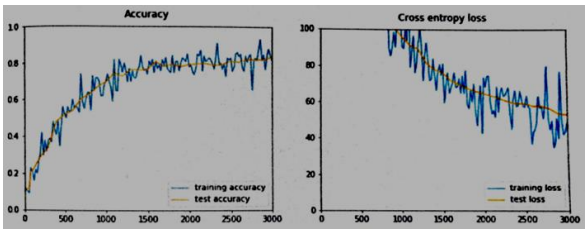


Figure 12: ANN + batch normalization+ dropout-accuracy and cross entropy loss.

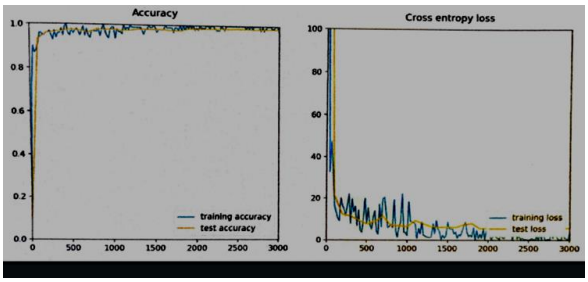


Figure 13: ANN- accuracy and cross entropy loss.

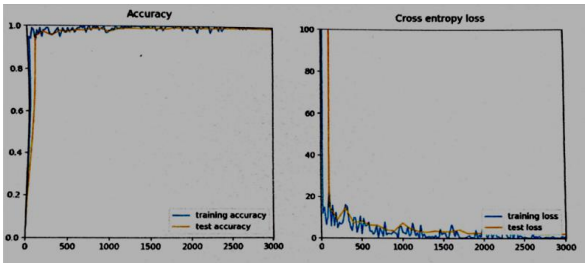


Figure 14: CNN + batch normalization- accuracy and cross entropy loss.

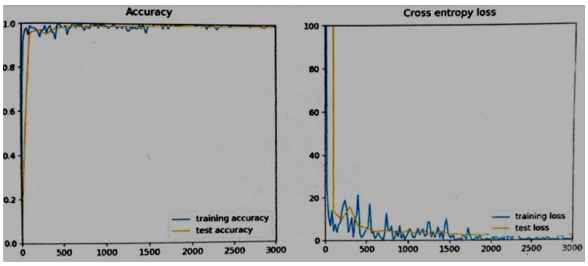


Figure 15: CNN + batch normalization+ dropout-accuracy and cross entropy loss.

5.2 Android interface

An Android application was developed for visually impaired with the following modules object recognition,

disease prediction, real time health monitoring, and voice assistant. Figure 10 (a)-(f) presents the screenshots of developed application interface.

6 Conclusion

The proposed application can open a new door to the possibilities of creating affordable and accessible solutions for improving the lifestyle and healthcare management of the visually impaired. The life of a visually impaired person can be transformed from a dependent individual to a productive and functional member of the society who is able to use mobile phones and other gadgets efficiently, detect objects around him and track his health with the help of such assistive technologies. A similar approach can also be extended to people suffering from other types of impairment.

The android application was successfully implemented with the following modules: object recognition, disease prediction, health monitor, and voice assistant. For object recognition, the use of Feature Pyramid Networks to generate feature maps in the model fared better than existing models like MNC and FCIS+OHEM but worse than FCIS+++OHEM due to the lesser number of layers in the Residual Network backbone. The graph database implementation of the disease-symptoms dataset gave the required results. However, graph algorithms could not be efficiently applied due to a lack of weighted edges i.e. probability that a symptom will cause a disease. This was due to the unavailability of appropriate non-proprietary datasets for general diseases.

7 Future research directions

The authors aim to improve the performance of the object recognition model by implementing aggregated residual transformations i.e. ResNeXt instead of ResNet for the backbone of Feature Pyramid Network. This model can also be extended for real-time depth realization to calculate the distance of the objects recognized. They also intend to enhance the disease prediction model by obtaining a more suitable weighted disease-symptom dataset, applying various graph algorithms, and adding drug suggestions. Lastly, they propose to upgrade the solution for healthcare management for the visually impaired from a voice-automated android application to a voice-automated hand-held device and add features like automatically alerting selected contacts in case of emergency.

References

[1] Jonas et al., "Visual Impairment and Blindness Due to Macular Diseases Globally: A Systematic Review and Meta-Analysis", American Journal of Ophthalmology, vol. 158, no. 4, pp. 808-815, 2014. DOI: 10.1016/j.ajo.2014.06.012.

[2] A. Gordois et al., “An estimation of the worldwide economic and health burden of visual impairment,” Glob. Public Health, vol. 7, no. 5, pp. 465–481, 2012. DOI: 10.1080/17441692.2011.634815

[3] Bradski, G. R. “Computer vision face tracking for use in a perceptual user interface”,1998.

[4] R. Velázquez, “Wearable Assistive Devices for the Blind,” in Wearable and Autonomous Biomedical Devices and Systems for Smart Environment, Springer, 2010, pp. 331–349.

[5] [A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Advances in Neural Information Processing Systems, vol. 25, 2012, [Online].DOI : https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

[6] O. Russakovsky et al., “ImageNet Large Scale Visual Recognition Challenge,” International Journal of Computer Vision, vol. 115, no. 3, pp. 211–252, Apr. 2015, doi: 10.1007/s11263-015-0816-y.

[7] C. Szegedy, A. Toshev, and D. Erhan, “Deep Neural Networks for Object Detection,” Neural Information Processing Systems, 2013. https://proceedings.neurips.cc/paper/2013/hash/f7cade80b7cc92b991cf4d2806d6bd78-Abstract.html (accessed Mar. 11, 2022).

[8] S.A.S, “Intelligent Heart Disease Prediction System Using Data Mining Techniques,” nternational Journal of healthcare & biomedical Research, Volume: 1, Issue: 3, April 2013, pp. 94-101. DOI: https://ijhbr.com/pdf/94-101.pdf.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1627-1645, Sept. 2010, doi: 10.1109/TPAMI.2009.167.

[10] Kanuganti, S., Chang, Y., & Bock, L. U.S. Patent No. 9,836,996. Washington, DC: U.S. Patent and Trademark Office, 2017.

[11] H. Fernandes, P. Costa, V. Filipe, L. Hadjileontiadis and J. Barroso, "Stereo vision in blind navigation assistance,"World Automation Congress, 2010, pp. 1-6.

[12] Milne LR, Bennett CL, Ladner RE. “The accessibility of mobile health sensors for blind users”. In International Technology and Persons with Disabilities Conference Scientific/Research Proceedings (CSUN 2014), Dec 2014 ,pp. 166-175.

[13] Y. Tian, X. Yang, and A. Arditi, “Computer Vision-Based Door Detection for Accessibility of Unfamiliar Environments to Blind Persons,” Lecture Notes in Computer Science, pp. 263–270, 2010, doi: 10.1007/978-3-642-14100-3_39.

[14] B. Ando, C. O. Lombardo, and V. Marletta, “Smart homecare technologies for the visually impaired:

recent advances," Smart Homecare Technology and TeleHealth, p. 9, Dec. 2014, doi:10.2147/shtt.s56167.

[15] Y. Ren, R. Werner, N. Pazzi and A. Boukerche, "Monitoring patients via a secure and mobile healthcare system," in IEEE Wireless Communications, vol. 17, no. 1, pp. 59-65, February 2010, doi: 10.1109/MWC.2010.5416351.

[16] Sachdeva N, Suomi R. Assistive technology for totally blind− barriers to adoption. SOURCE IRIS: Selected Papers of the Information Systems Research Semina. 2013;47.

[17] M. H. S. Segler, T. Kogej, C. Tyrchan, and M. P. Waller, "Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks," ACS Central Science, vol. 4, no. 1, pp. 120–131, Dec. 2017, doi: 10.1021/acscentsci.7b00512.

[18] D. Dakopoulos and N. G. Bourbakis, "Wearable Obstacle Avoidance Electronic Travel Aids for Blind: A Survey," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 40, no. 1, pp. 25-35, Jan. 2010, doi: 10.1109/TSMCC.2009.2021255.

[19] Wexler, Y., &Shashua, A.. U.S. Patent No. 9,025,016. Washington, DC: U.S. Patent and Trademark Office,2015.

[20] Lewis, C. W., Mathers, D. R., Hilkes, R. G., Munger, R. J., & Colbeck, R. P.. U.S. Patent No. 8,135,227. Washington, DC: U.S. Patent and Trademark Office,2012.

[21] He K, Zhang X, Ren S, Sun J." Deep residual learning for image recognition". In Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770-778).

[22] Girshick R. Fast r-cnn. InProceedings of the IEEE international conference on computer vision, 2015, pp. 1440-1448.

[23] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 1, pp. 142-158, 1 Jan. 2016, doi: 10.1109/TPAMI.2015.2437384.

[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," Neural Information Processing Systems, 2015.

[25] T.-Y. Lin et al., "Microsoft COCO: Common Objects in Context," Computer Vision – ECCV 2014, pp. 740–755, 2014, doi: 10.1007/978-3-319-10602-1_48.

[26] A. Chaudhuri, "Hierarchical Modified Fast R-CNN for Object Detection," Informatica, vol. 45, no. 7, Dec. 2021, doi: 10.31449/inf.v45i7.3732.

[27] N. Iqbal and M. Islam, "Machine learning for dengue outbreak prediction: A performance evaluation of different prominent classifiers," Informatica, vol. 43, no. 3, Sep. 2019, doi: 10.31449/inf.v43i3.1548.

[28] A. A. Abaker and F. A. Saeed, "A Comparative Analysis of Machine Learning Algorithms to Build a Predictive Model for Detecting Diabetes Complications," Informatica, vol. 45, no. 1, Mar. 2021, doi: 10.31449/inf.v45i1.3111.

[29] D. Panda, S. R. Dash, R. Ray, and S. Parida, "Predicting the Causal Effect Relationship Between COPD and Cardio Vascular Diseases," Informatica, vol. 44, no. 4, Dec. 2020, doi:10.31449/inf.v44i4.3088.

[30] WebMD, L. L. C. (2010). WebMD.

[31] Shiel Jr, W. C. (2009). MedicineNet. com.

[32] http://www.har-dataset.org/

https://doi.org/10.31449/inf.v46i1.3775 Informatica **46** (2022) 87–93 **87**

Network Security Situational Level Prediction Based on a Double-Feedback Elman Model

Jinbao He and Jie Yang
E-mail: jinbhe@yeah.net
Qian'an College, North China University of Science and Technology, Tangshan, Hebei 064400, China

Student paper

Keywords: network security, neural network, situational level, back-propagation neural network, situational prediction

Received: October 8, 2021

Network Security Situational Awareness (NSSA) is an important element in network security research. Predicting network security situational level can help grasp the network security situation. This study mainly focuses on the double-feedback Elman model. Firstly, NSSA was briefly introduced. Then, relevant indicators were selected to establish a security situational indicator system. A back-propagation neural network (BPNN) model was designed to evaluate the situational value. A dual-feedback Elman model was used to predict the future situational level. The actual network environment was built to conduct experiments. The results showed that the evaluation results of only three samples obtained by the BPNN model did not match the actual situation, with an accuracy of 90%, and the prediction results of only four samples obtained by the dual-feedback Elman model did not match the actual situation, with an accuracy of 96.67%. The experimental results verify the reliability of the network security situational level prediction method designed in this study. The NSSA method can be promoted and applied in practice.

Povzetek: S pomočjo globokih nevronskih mrež so razvili metodo za napovedovanje glede varnosti v omrežjih.

1 Introduction

As the network develops rapidly [1], people's living standards have gradually improved [2], and the growing scale of the network and the more and more complex network environment has led to a higher frequency of security problems, which greatly challenges the integrity and confidentiality of network data [3]. Although there are many security measures, such as firewalls [4] and anomaly detection [5], none of them can provide a systematic and holistic perception of the network, discover the problems and solve them, which leads to the emergence of Network Security Situational Awareness (NSSA) [6]. NSSA can fuse all available information to assess the situation of the network, provide network administrators with relevant decision bases, and minimize possible risks and losses, which is important for improving the monitoring capability and responsiveness of the network. NSSA has currently become a hot topic. Zhao and Liu [7] used a parallel reduction algorithm to reduce all data, optimized the wavelet neural network by the particle swarm algorithm, and performed situational awareness. The simulation experiments showed that the method had a higher convergence speed and better fitting effect. Zhang et al. [8] proposed a stochastic game-based method, quantified the situational value of the network with the utility of both sides of the game, and predicted the attack behavior by Nash equilibrium. The experiments found that the method could well reflect the change of network situation and predict the attack behavior. Li et al. [9] used Glowworm Swarm Optimization (GSO) to optimize the Gaussian process and performed situational prediction. They found through experiments that the method had better convergence and smaller prediction error. Hu et al. [10] used MapReduce for distributed training using a support vector machine (SVM) and built a situation prediction model. The experiment found that the method effectively improved accuracy and reduced time cost compared with the traditional method. In this study, NSSA was analyzed based on neural networks, the back-propagation neural network (BPNN) model was used for situational assessment, the dual-feedback Elman model was used for situational prediction, and an experimental analysis was conducted to verify the effectiveness of the method. This work provides a stronger guarantee for network security.

2 Network security situational awareness

Network security refers to protect computer systems from damage and also to avoid interruption of computer services. With the widespread use of computers, the Internet and WiFi in life, and the promotion of smart terminals and small smart devices, network security has become more important.

Situational awareness originated in military thinking and was first applied in aviation [11]. Then, it was also

well applied in medicine, electric power, and networks [12]. It mainly includes three stages. The first stage is understanding the situation, i.e., acquiring raw data through sensors, network monitoring, etc., and pre-processing them to extract useful security elements. The second stage is evaluating the situation, i.e., processing the extract security elements with fusion and association to evaluate the current network state. The third stage is predicting the situation, i.e., finding out the underlying laws through some prediction methods according to the historical situational value.

In the big data environment, the network behaves like a more massive and complex system, and at the same time, it also faces more security risks. The massive amount of data has certain errors and redundancy and stronger correlation and uncertainty [13] and changes faster, so the difficulty of data processing is further enhanced, which brings more difficulties to NSSA. Improving the efficiency of NSSA and the timeliness and accuracy of prediction to better achieve network security is an important issue at present. More and more methods have been applied in NSSA, such as machine learning, immune systems, game analysis and visualization techniques [14]. This study focuses on the neural network algorithm.

3 The double-feedback Elman model-based prediction model

3.1 Network security situational indicator system

Network security situational assessment is the basis of NSSA. The network data collected needs to be processed scientifically to guide administrators' decision-making. Firstly, relevant indicators should be selected to establish an indicator system that can comprehensively, objectively and scientifically reflect the network security situation. This study divided the security situation into four independent level 1 indicators and selected level 2 indicators to describe, as shown in Table 1.

Due to the complexity of the network environment, some of the indicators collected in Table 1 were nominal data that could not be directly input into the model for calculation; therefore, these indicators were quantified. Based on the Common Vulnerability Scoring System (CVSS) 3.0, these indicators were calculated as follows.

(1) Network vulnerability level: This indicator is mainly affected by the number and type of vulnerabilities, and its calculation formula is:

$$g = \frac{\sum_{i=1}^{n}\sum_{j=1}^{M} w_{ij}Q_i D_{ij}}{N},$$

$$Q_i = \frac{I_i}{\sum_{i=1}^{n} I_i},$$

$$I_i = \begin{cases} 1.0, & confidential \\ 0.7, & important \\ 0.4, & ordinary \end{cases},$$

where n refers to the number of hosts, M refers to the number of vulnerability categories, N refers to the total number of vulnerabilities, w_{ij} refers to the grade factor of the j-th vulnerability category in the i-th host, which can be obtained according to CVSS, D_{ij} refers to the number of the j-th vulnerability category in the i-th host, Q_i refers to the importance indicator of the i-th device, and I_i refers to the importance score of information scored by the host.

Level 1 indicator	Level 2 indicator
Stability indicators	Mean free error time
	Rate of change in flow
	Total data flow
	Number of surviving critical devices in the network
Threat indicators	Number of alarms
	Bandwidth utilization
	Data inflow
	Historical frequency of security incidents
Vulnerability indicators	Number of network vulnerabilities
	Network vulnerability level
	Total number of open ports for critical devices
	Network topology
Disaster tolerance indicators	Network bandwidth
	Number of safety equipment
	Host operating system
	Number of concurrent threads supported by the primary server

Table 1: Situational assessment indicator system.

Operating system	Versions	Score
Windows	XP	1
	7	2
	8	3
	10	4
	Server 2016	5
Ubuntu	14.4	4
	16.4	5
	Web Server 12	5
	Web Server 16	6
Mac OS	10	8

Table 2: Corresponding scores for operating systems.

(2) Network topology: Topology refers to the layout of security devices, and its calculation formula is:

$$g = \sum_{i=1}^{n} T_i,$$

$$T_i = \begin{cases} 1.0, m \in [0,3) \\ 0.5, m \in [3,5) \\ 0.1, m \in [5,+\infty) \end{cases},$$

where T_i stands for the security score of the i-th topology and m stands for the number of nodes in the topology.

(3) Host operating system: the higher the version of the operating system is, the smoother the operation of the network is. The points corresponding to different operating systems [15] are shown in Table 2, and the overall calculation formula is:

$$g = \sum_{i}^{N} \mathrm{OSTypeScore}_i,$$

where $OSTypeScore_i$ refers to the score of the operating system of the i-th device.

At the same time, the indicators were normalized, and the values were in $[0, 1]$. The corresponding formula is:

$$x_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}.$$

Referring to the National Computer Network Emergency Response Technical Team/Coordination Center of China, the security situation was divided into five different levels, as shown in Table 3.

The hierarchical matrices between different level 1 indicators and the security situation are shown in Table 4.

The security levels of level 1 indicators are shown in Table 5.

Security indicator value	Dangerous level
0.0-0.2	Safe
0.2-0.4	Mildly dangerous
0.4 - 0.75	Generally dangerous
0.75-0.9	Moderately dangerous
0.9-1.0	Highly dangerous

Table 3: Network security situation levels.

	Safe	Mildly dangerous	Generally dangerous	Generally dangerous	Highly dangerous
Stability indicators	High	High	Medium	Low	Low
Threat indicators	Low	Medium	High/ medium	High	High
Vulnerability indicators	Low	Low/ medium	High/ medium	Medium/high	High
Disaster tolerance indicators	High	High/ medium	Medium	Medium/low	Low

Table 4: The hierarchical matrices of level 1 indicators.

	Low	Medium	High
Stability indicators	0-0.3	0.3-0.7	0.7-1.0
Threat indicators	0-0.4	0.4-0.7	0.7-1.0
Vulnerability indicators	0-0.3	0.3-0.6	0.6-1.0
Disaster tolerance indicators	0-0.4	0.4-0.8	0.8-1.0

Table 5: The security level table of level 1 indicators.

3.2 Assessment methodology of the network security situation

For the situation assessment, BPNN was chosen as the model in this study [16]. BPNN has strong adaptive, self-organizing and learning abilities, and it has a good effect on the uncertainty of security elements. It has a high learning speed and a relatively simple modeling process. Therefore, it is well suited for evaluating security situations.

Level 2 indicators in Table 1 were used as the input of BPNN, and level 1 indicators were used as the output of BPNN. Then, the situation of the whole network was evaluated. $A_i(i = 1,2,\cdots,m)$ denotes the i-th level one indicator, $B_{ij}(i = 1,2,\cdots,m, j = 1,2,\cdots,n)$ denotes the j-th level two indicator corresponding to the i-th level one indicator, and W and V denote the weight from the input layer to the hidden layer and the weight from the hidden layer to the output layer. The BPNN-based evaluation model is written as:

$$A = f(B, W, V).$$

Situational level L is a function of m level one indicators, which can be written as:

$$L = f(A_1, A_2, \cdots, A_m).$$

The BPNN model has a three-layer structure. The number of nodes in the input layer was the number of level two indicators. The number of nodes in the output layer was the number of level 1 indicators, i.e., 1. The number of nodes in the hidden layer was determined by an empirical formula [17]:

$$l = \sqrt{n + m} + a,$$

where n, m, and l are the number of nodes in the output, output, and hidden layer, respectively, and a is a regulation constant between 1 and 10. After determining the range of the nodes in the hidden layer, the final number of nodes in the hidden layer was determined by training networks containing different number of nodes in the hidden layer.

The model used the Sigmoid function as the activation function. The learning rate was 0.1. The maximum number of training was 1000.

	CPU	Memory	Hard disk	Network card	Operating system	Main services
Experimental machine 1	Inter(R)Core(TM) i5	4G	1T	100 M	Windows 7	DHCP, FTP, TELNET, HTTP
Experimental machine 2	Inter(R)Core(TM) i3	2G	1T	100 M	Windows XP	FTP, RPC, DNS
Experimental machine 3	Inter(R)Core(TM) i5	4G	1T	100 M	Windows 8	DHCP, FTP, TELNET, HTTP, Browser
Experimental machine 4	Inter(R)Core(TM) i5	4G	1T	100 M	Windows 10	DHCP, NLA, COM+, Event System, DCOM
Experimental machine 5	Inter(R)Atom(TM) N450	4G	1T	Gigabit	Ubuntu14	Sever, ICS, FTP

Table 6: Equipment configuration.

Sample number	The input of BPNN				Expert score	Evaluation result
1	0.59	0.67	0.81	0.83	0.78	High
2	0.24	0.26	0.28	0.25	0.25	Low
3	0.78	0.86	0.79	0.64	0.81	High
4	0.56	0.52	0.61	0.63	0.64	Medium
5	0.45	0.57	0.58	0.64	0.59	Medium
...	...	...	...	...	...	...
100	0.11	0.27	0.18	0.16	0.21	Low

Table 7: Experimental data for stability assessment.

3.3 Prediction method of network security situational level

Based on the security situational values obtained from the above evaluation, the future situation can be predicted.

The Elman neural network has the nonlinear dynamic characteristic and is is sensitive to time series data; therefore, the prediction of the network security situational level with the ELman neural network is a dynamic time series problem. Therefore, the Elman neural network [18] was improved to obtain a dual-feedback Elman neural network model for predicting security situation. Compared with the original Elman neural network, the dual-feedback Elman neural network achieved the timely correction of errors by increasing the feedback, which improved the learning speed of the model. The relevant equations are:

$$\mathrm{x(k)} = \mathrm{f}\left(\mathrm{W^{I1}x_c(k)} + \mathrm{W^{I2}u(k-1)} + \mathrm{W^{I4}y_c(k-1)}\right),$$

$$x_c(k) = a\left(x_c(k-1) + x(k-1)\right),$$

$$y_c(k) = \gamma\left(y_c(k-1) + y(k-1)\right),$$

$$\mathrm{y(k)} = \mathrm{g}\left(\mathrm{W^{I3}x(k)}\right),$$

where x(k) represents the output of the hidden layer, $\mathrm{x_c(k)}$ represents the output of unit 1 in the succession layer, $\mathrm{y_c(k)}$ represents the output of unit 2 in the succession layer, y(k) represents the output of the output layer, a and γ are adjustment factors, $\mathrm{W^{I1}}$, $\mathrm{W^{I2}}$, $\mathrm{W^{I3}}$ and $\mathrm{W^{I4}}$ are connection weights, and f(x) represents the transfer function, $\mathrm{f(x)} = \frac{1}{1+\mathrm{e^{-x}}}$.

Through the gradient descent algorithm, the partial derivatives of all the connection weights were calculated based on error E. It was assumed that the expected output was $\mathrm{y_d(k)}$, then its error function, i.e., the objective function, is written as:

$$\mathrm{E(k)} = \frac{1}{2}\left(\mathrm{y_d(k) - y(k)^T}\left(\mathrm{y_d(k) - y(k)}\right)\right).$$

4 Experimental analysis

Two desktop computers, two laptops and one cloud server were prepared to build the experimental environment. Two actual websites in the campus network were used as data sources. One of the desktops was used as the client, and the other four computers were used as the attackers. The data was MySQL5.1. The configurations of different devices are shown in Table 6.

The data were collected in the laboratory for ten consecutive days. Expert opinions were obtained anonymously. One hundred samples were generated for every level one indicator in Table 1. Among the 100 samples, 70 samples were randomly selected as the training set and 30 as the test set. Taking the stability evaluation as an example, the samples obtained are shown in Table 7.

Sample number	Model evaluation results	Actual results
1	Low	Medium
2	Medium	Medium
3	Medium	Medium
4	Medium	Medium
5	High	High
6	High	High
7	High	High
8	Medium	Medium
9	Medium	Medium
10	Medium	Medium
11	Low	Low
12	Medium	Medium
13	Low	Medium
14	High	High
15	High	High
16	Medium	Medium
17	Low	Low
18	High	High
19	Low	Low
20	Low	Medium
21	Medium	Medium
22	High	High
23	Low	Low
24	High	High
25	Medium	Medium
26	Medium	Medium
27	Low	Low
28	High	High
29	Medium	Medium
30	Medium	Medium

Table 8: Comparison of stability assessment.

Experiments were conducted on 30 test samples using the trained BPNN model, and the obtained results are shown in Figure 1.

It was seen from Figure 1 that the output of the BPNN model had a good match with the actual situation, indicating that the BPNN model could make a good evaluation of the situational values. The stability level corresponding to the output of the BPNN model was compared with the stability level corresponding to the actual situation, and the results are shown in Table 8.

It was found from Table 8 that only 3 out of the 30 test samples had inconsistent evaluation results with the actual situation. The evaluation result of sample 1 was "low", but its actual result was "medium", and the same was true for samples 13 and 20. The accuracy of the BPNN model for the assessment of stability situation reached 90%, which verified the reliability of the BPNN model. In conclusion, the BPNN model could make a more accurate assessment of the network security situation, which was conducive to the next situational prediction.

Five hundred samples were selected from the situation database for the experiment of situation prediction. The first five samples were used to predict the fifth sample, and so on. Five samples were regarded as one group, and there were 100 groups. Seventy groups were randomly selected as training samples, and 30 groups were used as test samples. The inputs and outputs of the model are shown in Table 9.

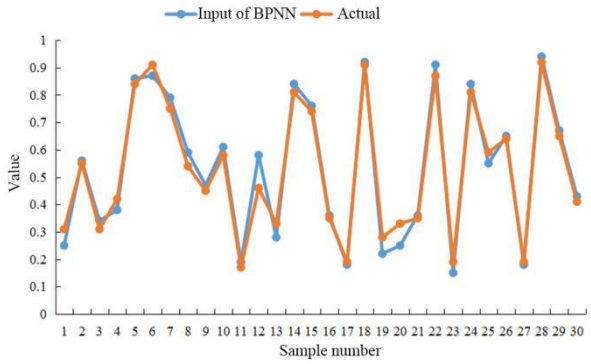


Figure 1: Results of the situational assessment.

Sample number	The first sample value	The second sample value	The third sample value	The fourth sample value	Sample Output
1	0.75	0.26	0.37	0.21	0.26
2	0.46	0.52	0.34	0.29	0.19
3	0.78	0.12	0.61	0.35	0.42
4	0.64	0.25	0.36	0.81	0.28
5	0.77	0.14	0.62	0.37	0.75
....	...	...	...	...	...
100	0.66	0.25	0.84	0.16	0.82

Table 9: Inputs and outputs of the model.

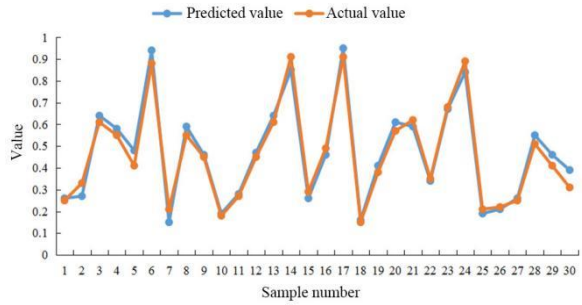


Figure 2: Prediction results of the situation.

The situation was predicted using the trained dual-feedback Elman model, and the results are shown in Figure 2.

It was seen from Figure 2 that the prediction results of the dual-feedback Elman model were closer to the actual values, and the predicted situational values matched well with the actual situation, indicating that the dual-feedback Elman model could make more accurate predictions of the situational values and the predicted results had high accuracy. Then, the predicted security situational levels were compared with the actual situation, and the results are shown in Table 10.

It was seen from Table 10 that four samples were predicted wrongly. The prediction result of sample 1 was highly dangerous, while it was moderately dangerous in fact. The prediction result of sample 14 was moderately dangerous, while it was highly dangerous in fact. The prediction result of sample 19 was generally dangerous, while it was mildly dangerous in fact. The prediction result of sample 25 was safe, while it was mildly dangerous in fact. In general, the accuracy of the prediction reached 86.67%, which verified that the dual-feedback Elman model was effective for situational prediction and worth further promotion and application in practice.

5 Conclusion

This study analyzed the prediction of network security situational level based on neural networks. The BPNN model was used for situational assessment, and the dual-feedback Elman model was used for situational prediction, and the network situational data were collected through the actual built network environment for experimental analysis. The results demonstrated that the accuracy of the BPNN model in predicting situations was 90%, and the dual-feedback Elman model had an accuracy of 86.67% in predicting the situational level. The two models have high stability and can effectively predict the network security situational level to achieve network security.

References

[1] Xhafa F, Li J, Kolici V (2015). SPECIAL ISSUE: Advances in Secure Data Streaming Systems. Informatica, 39, pp. 337-337.

[2] Zhang YN, Zhang MQ, Wang XA, Niu K, Liu J (2016). A Novel Video Steganography Algorithm Based on Trailing Coecients for H.264/AVC. Informatica, 40, pp. 63-70.

	Predicted results	Actual situation
1	Safe	Safe
2	Mildly dangerous	Mildly dangerous
3	Generally dangerous	Generally dangerous
4	Generally dangerous	Generally dangerous
5	Generally dangerous	Generally dangerous
6	Highly dangerous	Moderately dangerous
7	Safe	Safe
8	Generally dangerous	Generally dangerous
9	Generally dangerous	Generally dangerous
10	Safe	Safe
11	Mildly dangerous	Mildly dangerous
12	Generally dangerous	Generally dangerous
13	Generally dangerous	Generally dangerous
14	Moderately dangerous	Highly dangerous
15	Mildly dangerous	Mildly dangerous
16	Generally dangerous	Generally dangerous
17	Highly dangerous	Highly dangerous
18	Safe	Safe
19	Generally dangerous	Mildly dangerous
20	Generally dangerous	Generally dangerous
21	Generally dangerous	Generally dangerous
22	Mildly dangerous	Mildly dangerous
23	Generally dangerous	Generally dangerous
24	Moderately dangerous	Moderately dangerous
25	Safe	Mildly dangerous
26	Mildly dangerous	Mildly dangerous
27	Mildly dangerous	Mildly dangerous
28	Generally dangerous	Generally dangerous
29	Generally dangerous	Generally dangerous
30	Mildly dangerous	Mildly dangerous

Table 10: Comparative results of security situational levels.

[3] Ponnuramu V, Tamilselvan L (2016). Secured Storage for Dynamic Data in Cloud, Informatica, 40, pp. 53-61.

[4] Kayashima M, Terada M, Fujiyama T, Koizumi M, Katou E (2015). VPN construction method for multiple firewall environment. Systems & Computers in Japan, 31, pp. 57-63. https://doi.org/10.1002/1520-684X(200012)31:14<57::AID-SCJ7>3.0.CO;2-M

[5] Gogoi P, Borah B, Bhattacharyya DK (2013). Network Anomaly Identification using Supervised Classifier. Informatica, 37, pp. 93-105.

[6] Xu G, Cao Y, Ren Y, Li X, Feng Z (2017). Network Security Situation Awareness Based on Semantic Ontology and User-defined Rules for Internet of Things. IEEE Access, pp. 1-1. https://doi.org/10.1109/ACCESS.2017.2734681

[7] Zhao DM, Liu JX (2018). Study on Network Security Situation Awareness based on Particle Swarm Optimization Algorithm. Computers & Industrial Engineering, pp. S036083521830007X. https://doi.org/10.1016/j.cie.2018.01.006

[8] Zhang H, Yi Y, Wang J, Cao N, Duan Q (2018). Network Security Situation Awareness Framework based on Threat Intelligence. Computers, Materials & Continua, 56, pp. 381-399. https://doi.org/10.3970/cmc.2018.03787

[9] Li JZ, Meng XR, Wen XX, Kang QY (2015). Network security situation prediction based on Gaussian process optimized by glowworm swarm optimization. Systems Engineering & Electronics, 37, pp. 1887-1893. https://doi.org/10.3969/j.issn. 1001-506X.2015.08.26

[10] Hu J, Ma D, Liu C, Shi Z, Yan H, Hu C (2019). Network security situation prediction based on MR-SVM. IEEE Access, PP, pp. 1-1. https://doi.org/10.1109/ACCESS.2019.2939490

[11] Muehlethaler C M, Knecht C P (2016). Situation Awareness Training for General Aviation Pilots using Eye Tracking. IFAC Papersonline, 49, pp. 66-71. https://doi.org/10.1016/j.ifacol.2016.10.463

[12] Holsopple J, Yang S J, Sudit M (2015). Mission Impact Assessment for Cyber Warfare. Studies in Computational Intelligence, 563, pp. 239-266. https://doi.org/10.1007/978-3-319-08624-8_11

[13] Zhang H, Huang Q, Li F, Zhu J (2016). A network security situation prediction model based on wavelet neural network with optimized parameters. Digital Communications and Networks, pp. 139-144. https://doi.org/10.1016/j.dcan.2016.06.003

[14] Li F W, Zhang X Y, Zhu J, Huang Q (2016). Network security situation prediction based on APDE-RBF neural network. Systems Engineering and Electronics, 38, pp. 2869-2875. https://doi.org/ 10.3969/j.issn.1001-506X.2016.12.28

[15] Wang ZP (2010). Research on network security situation assessment based on index system. National University of Defense Technology.

[16] Liu S, Ren T, Mu H, Yan G (2016). Temperature prediction of the molten salt collector tube using BP neural network. IET Renewable Power Generation, 10, pp. 212-220. https://doi.org/10.1049/iet-rpg.2015.0065

[17] Liu W, Liu CK, Zhuang DM, Liu ZQ, Yuan XG. (2012). Study on Pilot Performance Model. Advanced Materials Research, 383-390, pp. 2545-2549. https://doi.org/10.4028/www.scientific.net/AMR.383-390.2545

[18] Liu H, Tian HQ, Liang XF, Li YF (2015). Wind speed forecasting approach using secondary decomposition algorithm and Elman neural networks. Applied Energy, 157, pp. 183-194. https://doi.org/10.1016/j.apenergy.2015.08.014

Intelligent Course Recommendation Based on Neural Network for Innovation and Entrepreneurship Education of College Students

Jinding Zou
E-mail: 00194026@wzu.edu.cn
School of Design and Art, Wenzhou University of Technology, Wenzhou, Zhejiang 325000, China

Student paper

Keywords: neural network, innovation and entrepreneurship education, recommendation technique, intelligence course, collaborative filtering

Received: October 9, 2021

This paper focuses on intelligence course recommendations for college students' innovation and entrepreneurship education. Firstly, the traditional collaborative filtering algorithm was introduced, and then a new recommendation technique was designed based on an artificial neural network (ANN). The experimental data were collected through a crawler framework. The two methods were compared and analyzed. It was found that the training time of collaborative filtering and ANN was 16.78 s and 12.36 s, the testing time was 2.64 s and 2.12 s, the hit rate (HR) were 0.6078 and 0.6264, and the normalized discounted cumulative gain (NDCG) values were 0.2948 and 0.3356, respectively. The results reveal that ANN was more efficient in computation and better in recommendations. The results demonstrate the effectiveness of the ANN method for intelligent course recommendations. The method can be applied to the selection of innovation and entrepreneurship education courses for college students.

Povzetek: Opisana je metoda nevronskih mrež za izbiro primernih študijskih predmetov.

1 Introduction

With the popularization of computers, the course selection system for college students has also changed. The original academic year system, in which all the courses were arranged by the school and completed according to the school's course plan, has developed to a credit system, in which students choose courses independently based on minimum credits for graduation. With the development of network technology, the method of online course selection has been popularized, but the independent selection of courses often leads to unsatisfactory final grades and is not conducive to smooth learning because the selected courses are not strongly related to the courses they have already taken [1]. If the recommendation of courses can be achieved through intelligent methods [2] to help students choose suitable and interesting courses, it will be a good contribution to the education of college students. The recommendation technique refers to a method of targeted recommendation of information based on individual interests and behaviors, which has been widely used in e-commerce [3], social entertainment [4], etc. Liu [5] designed a method based on an improved clustering algorithm to find users' nearest neighbors before making recommendations by clustering users and items separately and found through theoretical analysis and experiments that the recommendation accuracy of the method has been significantly improved. He et al. [6] studied the tweet recommendation of SINA micro-blog, extracted user-focused topics by k-cores analysis, used RS and linear regression to determine the parameters, and established a recommendation model based on semantic network. The experiment found that the method could make a timely and personalized recommendations for SINA micro-blog tweets. Tan et al. [7] designed a recommendation model based on probability matrix decomposition, which combined the social relationship and content information of items to improve the accuracy of recommendations. The experiment showed that the method was scalable and had a good recommendation effect. Due to the difference between course recommendation and commodity recommendation, recommendation techniques in course selection have been less studied. Zhang et al. [8] proposed an improved algorithm based on the Apriori algorithm, which preprocessed the data through Hadoop and then mined association rules. Through a series of experiments, they found that the method had good efficiency and was more applicable to the current Massive Open Online Courses (MOOC) platform. Based on neural networks, this paper analyzed the course recommendation method for college students' innovation and entrepreneurship education and verified the effectiveness of the method through experimental analysis. This work makes some contributions to promote the intelligent development of courses and also helps to promote the further development of college education.

2 Neural network-based recommendation technique

2.1 Innovation and entrepreneurship education for college students

Innovation and entrepreneurship education is a new education mode that combines various educational concepts such as innovation, entrepreneurship and quality, and its main content is to strengthen the innovation and entrepreneurship consciousness of college students, shape the innovation and entrepreneurship spirit, improve the innovation and entrepreneurship ability, and cultivate the innovation and entrepreneurship morality, so that college students can achieve flexible employment and independent entrepreneurship. Its main purposes are as follows.

(1) In line with the trend of the times: With the development of society, encouraging entrepreneurship, supporting entrepreneurship and establishing an innovative country have become key content. The employment pressure has become more and more prominent since the enrollment expansion of colleges. In order to ensure that college students can be successfully employed, it is necessary to carry out innovation and entrepreneurship education and create a better employment environment and more employment opportunities to improve the country's innovation ability.

(2) Cultivating innovative and entrepreneurial talents: With the progress of society, the competition among countries is also reflected in the competition of innovative and entrepreneurial talents. Innovation and entrepreneurship education is conducive to implementing the strategy of developing the country through science and education and promoting the progress of the country and society.

(3) Promoting higher education reform: Conducting innovation and entrepreneurship education can bring new vitality and content to the education reform and is also a good way to improve the employability and psychological quality of college students, which can support the current higher education reform.

(4) Promoting the overall development of college students: Innovation and entrepreneurship education can encourage students to actively innovate, help them better understand the laws of the market, seize the opportunity to develop from passive employment to active employment and from active employment to active entrepreneurship, and maximize the development of personal creativity.

In the current curriculum of college students, different innovation and entrepreneurship education courses are arranged, and students can choose the corresponding courses to study according to their majors and needs. In order to better play the role of innovation and entrepreneurship education courses, this paper adopts a neural network-based method to recommend courses intelligently to students.

2.2 Collaborative filtering technology

The collaborative filtering technique is a relatively mature and widely used method in recommendation technology [9], which can recommend items to users according to their needs [10]. Its principle is as follows. For user A and the set of items that the user is interested in, the set of items is analyzed to find user B that is most similar to user A. Then, the items in the set of items of user B that have not been selected by user A are recommended to A. This method is applied to the intelligent course recommendation for college students' innovation and entrepreneurship education. The process can be described as follows.

(1) Establishing a matrix for student course evaluation: The matrix is established by students' evaluation of a course. For student 1, his evaluation of course1 is C_{11}; then, student M's evaluation of course N is C_{mn}. The established matrix is shown in Table 1.

	Course 1	Course 2	.....	Course N
Student 1	C_{11}	C_{12}	……	C_{1n}
Student 2	C_{21}	C_{22}	……	C_{2n}
.....	……	……		……
Student M	C_{m1}	C_{m2}	……	C_{mn}

Table 1: Student course evaluation matrix.

(2) Selecting students' nearest neighbor: The similarity between students is calculated based on the above matrix. The corresponding formula is:

$$\mathrm{Sim}(u_a, u_b) = \frac{\sum_{i \in I_{a.b}} (C_{a,i} - \overline{C_a})(C_{b,i} - \overline{C_b})}{\sqrt{\sum_{i \in I_{a.b}} (C_{a,i} - \overline{C_a})^2} \sqrt{\sum_{i \in I_{a.b}} (C_{b,i} - \overline{C_b})^2}},$$

where i refers to the courses evaluated by students a and b, $C_{a,i}$ and $C_{b,i}$ refer to the common evaluation of course i by students a and B, $\overline{C_a}$ and $\overline{C_b}$ refer to the average score of the evaluation. By calculating and ranking this similarity, k students with the most similarity are selected to establish students' nearest neighbor, U_s .

(3) Recommending courses intelligently: After getting the nearest neighbor U_s, the scores of the courses are predicted, and the formula is:

$$P_{u,j} = \overline{C_u} + \frac{\sum_{u_k \in U_s} Sim(u,u_k)\left(C_{u_k,i} - \overline{C_{u_k}}\right)}{\sum_{u_k \in U_s} (|Sim(u,u_k)|)},$$

where $\overline{C_u}$ refers to the average score of all courses given by student u, $C_{u_k,i}$ refers to the score of course i, and $\overline{C_{u_k}}$ refers to the average score of all courses.

Through calculating and sorting $P_{u,j}$, the first N courses are recommended to students as a set to complete the intelligence course recommendation for college students' innovation and entrepreneurship education.

2.3 Neural network-based recommendation technique

In this paper, an intelligent course recommendation technique for college students' innovation and entrepreneurship education was designed based on artificial neural networks (ANN). ANN is a method that simulates how the human brain nerve processes information to solve problems [11]. ANN is characterized by a strong self-learning capability that enables the algorithm to achieve higher accuracy [12]. It has an extensive application in fields such as face recognition [13]. The algorithm first performs weighted summation on all the signals before processing the previous input, and the calculation formula is:

$$u_i = \sum w_{ij} x_{j\theta}.$$

When the input u_i of the neuron > 0, it indicates that the neuron is in an excited state; otherwise, it is in an inhibited state. Then, the output of the neuron is obtained by performing an activation operation on the input, and the calculation formula is:

$$y_i = K(u_i),$$

where K$(*)$ refers to the activation function.

The idea of ANN is applied to intelligent course recommendations for innovation and entrepreneurship education. The weight of college students' course is calculated according to the idea of ANN. Suppose that college students have n innovation and entrepreneurship education courses to choose from, and the weight of each course can be written as: $W_i(i = 1,2,\cdots,n)$, each course has m$(\mathrm{m} = 0,1,\cdots,\mathrm{m})$ leading courses, and the influence coefficient of every leading course is $u_i = \frac{1}{m}$. If the student has not browsed the course once, it makes the weight of the course +1. Each time a student views the course, the weight of the course is increased by 1. If a student has taken t leading courses of some course, then the influence weight of t leading courses is added to the weight of the course is:

$$\mathrm{W_I} = \sum_{\mathrm{j}-1}^{\mathrm{t}} \mathrm{u_j w_{pre}},$$

where w_{pre} refers to the weight of the leading course.

After calculation, k courses that rank high in terms of weight are recommended to students as a set.

3 Experimental analysis

Since there is a lack of public datasets about course recommendations, this paper crawled the data needed for the experiment from the MOOC platform of Chinese universities through a crawler. In the MOOC, many colleges offer courses about innovation and entrepreneurship education, and some of them are shown in Table 2.

It was seen from Table 2 that the innovation and entrepreneurship education courses offered on MOOC were different in terms of target audience and professional focus. Some courses were more inclined to cultivate innovation consciousness, while some were more inclined to improve entrepreneurial ability; therefore, intelligent course recommendation was needed to help college students better choose the appropriate course.

The Uniform Resource Locator (URL) address of the detail page of every innovation and entrepreneurship education course was crawled through the Scrapy crawler framework. The crawled content included user identity, course scores, etc. The courses without scores and users who scored less than ten courses were excluded. The crawled data were cleaned. 70% of the data were randomly selected as the training set, and 30% were as the test set. The data sets are shown in Table 3.

The experimental device was a MacBook Pro with Intel Core i7 2.2GHz processor, 16 G memory, 251 G hard disk, and Python 2.7 software environment. 70% of the obtained dataset was randomly selected as the training set, and the remaining 30% was used as the test set.

The evaluation indicators of the algorithm are as follows.

(1) Hit rate (HR): it was used to describe whether the recommendation results of the algorithm were in the top k recommendation lists, and its expression is:

$$\mathrm{HitRatio@K} = \frac{\mathrm{Number\ of\ Hits@K}}{|\mathrm{GT}|} \times 100\%,$$

where $|GT|$ refers to the number of test sets and Number of Hits@K is the sum of the number of the recommendation results that are consistent with the predicted results of the test set.

(2) Normalized discounted cumulative gain (NDCG) [14]: it was used to describe the difference between the predicted and true results, and its expression is:

$$\begin{cases} CG_k = \sum_{i=1}^{k} reli \\ DCG_k = \sum_{i=1}^{k} \frac{2^{reli}-1}{log_2^{(i+1)}} \\ IDCG_k = \sum_{i=1}^{k} \frac{1}{log_2^{1+i}} \\ NDCG_k = \frac{DCG_k}{IDCG_k} \end{cases},$$

where $reli$ refers to the relevance of the recommended results (if $reli = 1$, then there was a hit; if $reli = 0$, then there was no hit), CG_k is the cumulative gain, DCG_k is the discount cumulative gain, and $IDCG_k$ is the ideal discount cumulative gain. The larger the value of NDCG was, the better the recommendation was, i.e., the higher the overlap between the recommendation list given by the model and the real list.

The running time of the two algorithms was first compared, and the results are shown in Figure 1.

It was seen from Figure 1 that the training time of the ANN algorithm was 26.34% less than that of the collaborative filtering algorithm (16.78 s vs. 12.36 s) and the test time of the ANN algorithm was 19.7% less than that of the collaborative filtering algorithm (2.64 s vs. 2.12 s). The results revealed that the ANN algorithm showed better computational efficiency in both training and test and had better availability in the recommendation of innovation and entrepreneurship education courses.

Course name	Institutions	Course introduction
Chinese medicine innovation and entrepreneurship	Chengdu University of Traditional Chinese Medicine	Innovation and entrepreneurship are combined with traditional Chinese medicine health. The main content is how to grasp the good opportunity of entrepreneurship in developing the traditional Chinese medicine health industry.
Innovation and entrepreneurship practice	Shandong Transport Vocational College	Based on the new trend of entrepreneurship and the characteristics of the college student group, the course focuses on improving the quality of college students' entrepreneurship and solving common problems in different stages of the entrepreneurial process, which helps students to use innovative thinking to solve problems encountered in the process of future entrepreneurship and improve the comprehensive quality and ability to start and manage enterprises.
Design thinking and innovative design	Zhejiang University	The research results in the field of innovative design in China are combined with cutting-edge achievements in design thinking in China and abroad. The course aims to effectively enhance product innovation and entrepreneurial practice.
Innovation and entrepreneurship management	Nanjing University of Posts and Telecommunications	The course helps students understand the rules and characteristics of entrepreneurship, recognize the problems they may encounter in the process of starting a business, draw lessons from them, and improve competitiveness.
Innovation management	Zhejiang University	The course helps students to develop their creative abilities and acquire relevant theoretical knowledge and practical skills.
Foundations of college student entrepreneurship	Wenzhou University	The course helps students master the basic theory and process of entrepreneurship, establish a correct view of innovation and entrepreneurship, and guide them from theory to practice.

Table 2: Examples of innovative entrepreneurship education courses.

	Training set	Test set
Number of users	3650	1564
Number of courses	1300	557
Number of ratings	319398	136885

Table 3Experimental data set.

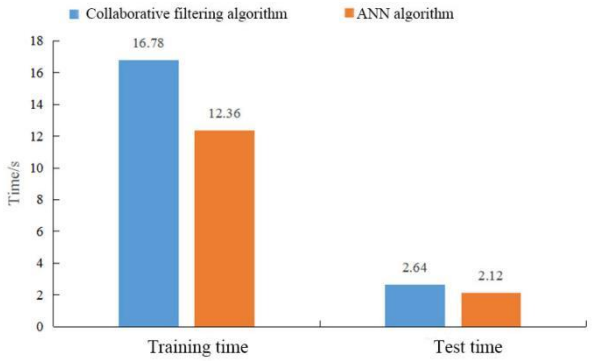


Figure 1: Running time comparison.

Then, the HR and NDCG of the two algorithms were compared, and the results are shown in Figure 2.

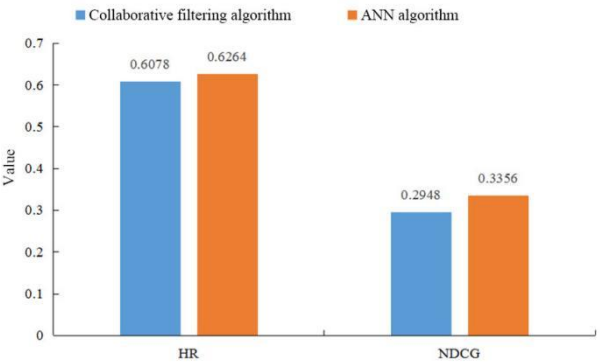


Figure 2: Performance comparison between algorithms.

It was seen from Figure 2 that the HR of the ANN algorithm was 3.06% higher than that of the collaborative filtering algorithm (0.6078 vs. 0.6264) and the NCDG value of the ANN algorithm was 13.84% higher than that of the collaborative filtering algorithm (0.2948 vs. 0.3356). The results revealed that the ANN algorithm showed better performance in both HR and NDCG values; thus, it had better reliability in recommending innovation and entrepreneurship education courses for college students.

4 Discussion

The Internet has become more and more widely used in people's lives. In order to effectively obtain the required information from the increasing network information and resources, "information retrieval" and "information filtering" technologies emerged. The former refers to retrieve the required information with keywords and other methods [15], and the latter refers to filter the information on the network with users' personal information to recommend information to users. For example,

recommendation systems [16] solves the problem of identifying relevant resources from a large number of available choices [17], and the collaborative filtering algorithm is the most commonly used one [18]; however, during courses selection, due to the small differences between users, i.e., students, in age and major and the fact that the number of courses is not as huge as the e-commerce recommendation system, it is not applicable to intelligent course recommendation.

Innovation is the inexhaustible power of development. Strengthening innovation and entrepreneurship education for college students has become a key content in the college curriculum [19], which is also a necessary way to cultivate innovative and entrepreneurial talents needed by society. With the development of network technology and the popularity of online learning, innovation and entrepreneurship education courses can also be learned online, which has a positive effect on improving the innovation consciousness and innovation ability of college students [20]. Therefore, this paper mainly analyzed the intelligent course recommendation for innovation and entrepreneurship education of college students.

The comparison between the traditional collaborative filtering algorithm and the neural network-based recommendation technique suggested that the ANN algorithm had better performance in terms of both the running time and the recommendation performance. Figures 2 and 3 showed that the training time and test time of the ANN algorithm were 12.36 s and 2.12 s on the same data set, i.e., its computational efficiency was higher than that of the traditional collaborative filtering algorithm, and the HR and NDCG values of the ANN algorithm were 0.6264 and 0.3356 respectively, i.e., the performance of the ANN algorithm was also superior to that of the traditional collaborative filtering algorithm. In conclusion, the ANN algorithm was effective.

Although some results have been achieved in the study of intelligent course recommendations, there are still some shortcomings. For example, the data used in the experiments were offline data, which did consider the real-time nature of the recommendation; the selection of data and the determination of weights were also subjective. These problems need to be addressed in future work.

5 Conclusion

This paper analyzed the intelligent course recommendation for college students' innovation and entrepreneurship education based on neural networks and compared the performance of the traditional collaborative filtering algorithm and the ANN algorithm on course recommendation. The experiments on the same data set found that the ANN algorithm not only could train and test data faster but also had better results in recommendation; the HR and NDCG values of the ANN algorithm were significantly higher than those of the traditional collaborative filtering algorithm, which verified the reliability of the method. The ANN algorithm can be further promoted and applied in practice.

References

[1] Upendran D, Chatterjee S, Sindhumol S, Bijlani K (2016). Application of Predictive Analytics in Intelligent Course Recommendation. *Procedia Computer Science*, 93, pp. 917-923. https://doi.org/10.1016/j.procs.2016.07.267

[2] Zhu YF, Lu H, Qiu P, Shi KZ, Chambua J, Niu ZD (2020). Heterogeneous teaching evaluation network based offline course recommendation with graph learning and tensor factorization - ScienceDirect. *Neurocomputing*, 415, pp. 84-95. https://doi.org/10.1016/j.neucom.2020.07.064

[3] Kumar T S, Pandey S (2015). Customization of Recommendation System Using Collaborative Filtering Algorithm on Cloud Using Mahout. *Advances in Intelligent Systems and Computing*, 321, pp. 39-43. https://doi.org/10.15623/ijret.2014.0319008

[4] Zhang Z J, Liu H (2015). Research on context-awareness mobile SNS recommendation algorithm. *Pattern Recognition and Artificial Intelligence*, 28, pp. 404-410. https://doi.org/10.16451/j.cnki.issn1003-6059.201505003

[5] Liu X (2017). An improved clustering-based collaborative filtering recommendation algorithm. *Cluster Computing*, 20, pp. 1281-1288.

[6] He Y, Tan J (2015). Study on SINA micro-blog personalized recommendation based on semantic network. *Expert Systems with Applications*, 42, pp. 4797-4804. https://doi.org/10.1016/j.eswa.2015.01.045

[7] Tan F, Li L, Zhang ZY, Guo YL (2016). A multi-attribute probabilistic matrix factorization model for personalized recommendation. *Pattern Analysis & Applications*, 19, pp. 857-866. https://doi.org/10.1007/978-3-319-21042-1_57

[8] Zhang H, Huang T, Lv Z, Liu S, Zhou Z (2018). MCRS: A course recommendation system for MOOCs. *Multimedia Tools & Applications*, 77, pp. 7051-7069. https://doi.org/10.1007/s11042-017-4620-2

[9] Li C, He K (2017). CBMR: An optimized MapReduce for item-based collaborative filtering recommendation algorithm with empirical analysis. *Concurrency and Computation: Practice and Experience*, 29, pp. e4092. https://doi.org/10.1002/cpe.4092

[10] Tan YL, Zhao HJ, Wang YR, Qiu M (2018). Probability matrix decomposition based collaborative filtering recommendation algorithm. *Informatica*, 42, 265-271.

[11] Koprinkovahristova P, Mladenov V, Kasabov N K (2015). Artificial Neural Networks. *European Urology*, 40, pp. 245. https://doi.org/10.1002/1097-

0142(20010415)91:8%2B<1615::AID-CNCR1175>3.0.CO;2-L

[12] Jain A K, Mao J, Mohiuddin K M (2015). Artificial neural networks: a tutorial. *Computer*, 29, pp. 31-44. https://doi.org/10.1109/2.485891

[13] Dalila C, Badis EAO, Saddek B, Naït-ali A (2020). Feature level fusion of face and voice biometrics systems using artificial neural network for personal recognition. *Informatica*, 44, 85-96. https://doi.org/10.31449/inf.v44i1.2596

[14] Jayashree R, Christy A (2015). Improving the Enhanced Recommended System Using Bayesian Approximation Method and Normalized Discounted Cumulative Gain. *Procedia Computer Science*, 50, pp. 216-222. https://doi.org/10.1016/j.procs.2015.04.057

[15] Huy HNL, Minh HH, Van TN, Van HN (2021). Keyphrase extraction model: a new design and application on tourism information. *Informatica*, 45, 563-569.

[16] Balasingam U, Palaniswamy G (2019). Feature augmentation based hybrid collaborative filtering using tree boosted ensemble. *Informatica*, 43, 477-483. https://doi.org/10.31449/inf.v43i4.2141

[17] Alhamid M F, Rawashdeh M, Dong H, Hossain MA, Saddik AE (2017). Exploring Latent Preferences for Context-Aware Personalized Recommendation Systems. *IEEE Transactions on Human-Machine Systems*, 46, pp. 1-9. https://doi.org/10.1109/THMS.2015.2509965

[18] Zhang J, Lin Y, Lin M, Liu J (2016). An effective collaborative filtering algorithm based on user preference clustering. *Applied Intelligence*, 45, pp. 230-240. https://doi.org/10.1007/s10489-015-0756-9

[19] Gao H, Qiu Z, Liu Z, Huang L, San Y (2016). Research and Practice on College Students' Innovation and Entrepreneurship Education. *International Conference of Pioneering Computer Scientists, Engineers and Educators*, pp. 14-15. https://doi.org/10.1007/978-981-10-2098-8_6

[20] Zhou Z (2016). Research on Model Construction of Innovation and Entrepreneurship Education in Domestic Colleges. *Creative Education*, 7, pp. 655-659. https://doi.org/10.4236/ce.2016.74068

Innovative Application of Recombinant Traditional Visual Elements in Graphic Design

Jing Lu
E-mail: lutong3912344133@163.com
Xi'an International University, Xi'an, Shaanxi 710077, China

Student paper

Keywords: traditional visual element, graphic design, reorganization, convolutional neural network

Received: November 23, 2021

Excellent graphic design can enhance the attractiveness of promotional products. This paper briefly introduces traditional visual elements and their main application methods. Three graphic designs used for promoting Fengxiang painted clay sculpture were analyzed using the analytic hierarchy process method, and a convolutional neural network (CNN) was used for auxiliary analysis. The results showed that the performance of the trained CNN was initially verified in the training set. The comparison between the evaluation results of the three graphic designs and the manual evaluation results further verified the performance of the CNN in aiding the evaluation of the graphic design. All three graphic designs proposed in this paper effectively deconstructed and recombine the traditional visual element, "painted clay sculpture", enhancing its attractiveness.

Povzetek: Za iskanje atraktivnih grafičnih predstavitev so uporabili metodo konvolucijskih globokih mrež.

1 Introduction

In our daily life, we often come into contact with things related to graphic design, such as product packaging, magazine covers, posters and advertisements [1]. The visual experience of these things is the result of careful design, and the ultimate goal is to attract the attention of the public and to convey the designer's intention contained in the graphic design. Through the expression of their intentions to empathize with the audience, new things or concepts can be widely accepted. Graphic design is the combination of text and images in a two-dimensional space according to certain rules, i.e., typography, and then the results of the combination are shown through printing and other means [2]. With the development of technology and design fields, the elements that can be filled in graphic design become more and more diverse. As a form of artistic expression that focuses on visual communication, visual elements are the focus of graphic design. The ultimate goal of graphic design is to make something compelling, so the design process needs to guarantee that the work is unique, especially the visual elements that are directly related to visual communication [3]. China's traditional culture has a long history, and various traditional arts contain visual elements worthy of reference. These visual elements not only provide a new development direction for graphic design, but also facilitate the creation of graphic design works with national characteristics. Shen et al. [4] proposed a method that can measure color harmony and verified the effectiveness of the method through experiments. Yuan et al. [5] proposed a method based on Kansei engineering (KE) and interactive genetic algorithm (IGA), and the results of example analysis showed that the method could obtain a satisfactory color design. Carlos Velasco et al. [6] proposed an experimental method capable of assessing the response of customers to changes in the orientation of various design elements on product packaging, such as food images. The results highlighted the complex relationship between preferences and willingness to pay and raised some questions about the role of orientation in visual aesthetics, preferences and perceived value. The paper briefly introduced traditional visual elements and the main ways they are used and analyzed three graphic designs used for promoting the Fengxiang painted clay sculpture.

2 Traditional visual elements and reorganization

2.1 Visual elements

Visual elements are an important part of visual communication in graphic design. Visual elements have various types, but they can be generally categorized into three types: composition and pattern, color, and text [7]. Composition and pattern reflect the style of graphic design. Graphic designs will use abstract means to express the specific product content [8]. Especially now that computer technology is commonly used, it is possible to build abstract visual effects through various irregular geometric texture patterns to visualize the connotation of goods that are difficult to describe with words. In addition, graphic design will also use painting, cartoon and exaggerated deformation to enhance the expressiveness of the pattern [9]. Color matching highlights the spirit of a graphic design program. When a viewer appreciates or

stumbles upon a graphic design, the color of the graphic design will first come into the scope of observation. Reasonable color matching can first attract the attention of the audience [10]. Text description directly reflects the essential content of a graphic design. Texts can play a decorative role. Chinese calligraphy, in particular, is inherently artistic and can be used to decorate graphic design through a variety of calligraphic variants, thus resonating with the audience's imagination [11].

2.2 Application of traditional visual elements

The application of visual elements in graphic design is closely related to market competition. The uniqueness of visual element design can effectively distinguish one's products from other products. Traditional culture with a long history provides a new direction for selecting visual elements in graphic design, adding a unique national character to graphic design [12].

Traditional visual elements are visual elements extracted from traditional culture and have distinctive characteristics of Chinese culture. Traditional visual elements with specific images include dragon, Kylin, Chinese painting, facial makeup, pottery, etc. In addition, there are also many visual elements without specific images but with distinct Chinese cultural contexts. The traditional visual elements with specific images often have been processed before applications [13].

The application method of traditional visual elements is shown in Figure 1, including the simplification, deconstruction and reorganization, replacement and isomorphism of traditional visual elements. The simplification of traditional visual elements is a common means in graphic design. The simplified traditional visual elements can reflect the designer's ideology more intuitively, which is not only in line with the eye-catching purpose of modern graphic design but also can be more compatible with other elements [14]. The deconstruction and reorganization of traditional visual elements are to rearrange and combine different visual elements to obtain new visual elements. Since the deconstruction and reorganization are based on the original elements, they give the audience a new visual impact while preserving the traditional cultural context. The replacement of traditional visual elements means replacing the original visual elements with other visual elements in nature according to some connection in the structure of graphic design [15]. Isomorphism of traditional visual elements and modern visual elements is to connect similar traditional visual elements and modern visual elements to obtain a more logical imagination effect [16].

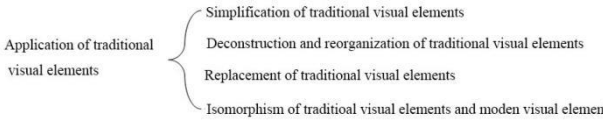


Figure 1: The way of applying traditional visual elements.

Figure 2: Three graphic designs.

3 Example analysis

3.1 Object of analysis

Fengxiang painted clay sculpture is a traditional folk art in Fengxiang District, Baoji City, Shaanxi Province, also known as clay goods by the locals. The craft of clay sculpture can be traced back to the Spring and Autumn Period and the Warring States Period when clay sculptures were used as funerary objects. Fengxiang painted clay sculptures are mainly produced in the Sixth Battalion Village of Chengguan Town. A story is going around that soldiers from the sixth battalion of Zhu Yuanzhang's army in the Ming Dynasty were stationed in a place and turned to be residents and they were engaged in the craft of pottery making and developed pottery as a local specialty. Fengxiang painted clay sculpture has become a unique fine art of folk art in modern times. Fengxiang clay sculptures are exaggerated in shape and colorful.The ornamentation of Fengxiang clay sculpture draws on stone carving, paper cutting and other arts. It was included in the Chinese intangible cultural heritage list on May 20, 2006. In order to better promote this folk craft and better inherit and protect it, it is necessary to promote Fengxiang clay sculpture accordingly. Figure 2 shows the three graphic designs of the brochure and poster for Fengxiang clay sculptures.

3.2 Evaluation methodology

The evaluation of graphic design is a rather subjective personal feeling. Even the same work may give different feelings to different people at different times. In addition, when facing a large number of designs, it is inefficient to rely on one person alone to evaluate and select, but the participation of multiple people will reduce the credibility of the evaluation due to the subjectivity of personal feelings. Intelligent algorithms have high processing efficiency and can imitate human thinking patterns. The imitation can be regarded as a single individual's evaluation of a large number of designs.

This paper uses the CNN algorithm that has a wide application in image recognition to evaluate graphic designs. The basic structure of CNN includes the input layer, the convolutional layer, the pooling layer and the output layer [17], as shown in Figure 3. The graphic design was input into the input layer. The convolutional layer convoluted the graphic design with convolution kernels to extract characteristics of graphic images. The pooling layer compressed the convolutional feature map by pooling to reduce the number of operations. After

repeating the convolution and pooling operations several times, the evaluation results were output in the output layer.

Before evaluating graphic designs with the CNN algorithm, the algorithm was trained with training samples. The flow of the algorithm in the training was almost the same as described in the last paragraph, and the difference between them was that the output results were compared with the expected results of the training samples. The gap between the output results and expected results determined whether to stop the training or reversely adjust the algorithm parameters [18]: when the gap converged to within the set range, the training was stopped; when the gap exceeded the set range, the algorithm parameters were adjusted reversely in accordance with the gap, and the stochastic gradient descent method was used in this paper.

The relevant parameters of CNN are as follows. Its structure was two convolutional layers, one pooling layer, two convolutional layers and one pooling layer. There were 16 convolution kernels in a size of 2 × 2 in the convolutional layers. There were mean-pooling boxes in a size of 2 × 2 in the pooling layers. Sigmoid was used as the activation function. The size of the image input to the input layer was 800 × 600. The training samples and test samples were in the format of jpg.

Among the samples used for training and testing, in addition to the graphic design, every design had a corresponding evaluation label. These labels were manually labeled. Graphic designs within the samples used for training and testing came from various promotional schemes for local painted clay sculptures in the market. One thousand samples were collected, of which 70% were used as training samples and 30% as testing samples. The formula for measuring the performance of the algorithm using the test samples is:

$$\begin{cases} P = \dfrac{TP}{TP + FP} \\ R = \dfrac{TP}{TP + FN} \\ F = \dfrac{2PR}{P + R} \end{cases}, \quad (3)$$

where P is the precision rate, R is the recall rate, F is the average accuracy after blending the precision rate and recall rate, TP denotes the number of positive samples that are judged as positive by the algorithm, FP denotes the number of negative samples that are judged as positive by the algorithm, FN denotes the number of positive samples that are judged as negative by the algorithm, TN denotes the number of negative samples that are judged as negative by the algorithm.

After training and testing, the three graphic designs proposed in this paper were evaluated, and the evaluation results were compared with the manual evaluation results. The evaluation labels of the training and testing samples were labeled in the same way as the manual evaluation, i.e., using the analytic hierarchy process (AHP) method [19]. The hierarchical structure of the evaluation of the reorganization effects of traditional visual elements in graphic designs is shown in Figure 4. The highest layer was the reorganization effect of traditional visual elements

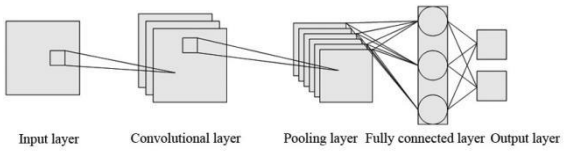


Figure 3: The basic structure of CNN.

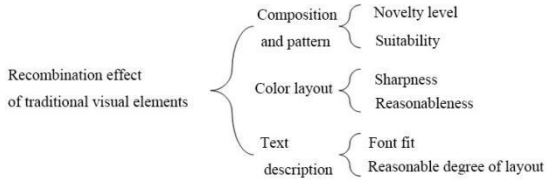


Figure 4: Hierarchy for evaluating the effect of reorganization of traditional visual elements.

in graphic design, i.e., the final evaluation result. The intermediate layers were composition and pattern, color layout and text description, and the target layers under the intermediate layers are shown in Figure 4. In the AHP method, in addition to collecting the scores of the target layer items through questionnaires, it was more important to set the weight of the layers. This paper constructed hierarchy weights by information entropy, and the corresponding formulas are as follows:

$$\begin{cases} E_j = -\dfrac{\sum_{i=1}^{n} p_{ij} \ln p_{ij}}{\ln n} \\ p_{ij} = \dfrac{x_{ij}}{\sum_{i=1}^{n} x_{ij}} \\ \omega_j = \dfrac{1 - E_j}{m - \sum E_j} \end{cases}, \quad (2)$$

where E_j stands for the information entropy of indicator j, x_{ij} stands for the standardized data of the j-th indicator of sample i, p_{ij} stands for the proportion of sample x_{ij} in all the data of indicator j, n is the number of samples, ω_j is the weight of indicator j, and m is the number of indicators. The above indicators are six indicators in the target layer in the AHP method.

After the weights of the layers were determined, the survey content of the questionnaire was designed according to the items of the target layers. Ten experts who had been engaged in graphic design for more than five years rated the graphic designs using a ten-point scale. The average score was used as the final result of the evaluation.

3.3 Analysis results

The test set was tested using the trained CNN algorithm, and the final evaluation performance is shown in Figure 5. The CNN algorithm had a precision rate of 97.6%, a recall

rate of 97.3%, and an F-value of 97.5% for scoring the graphic designs. The analysis of the precision rate, recall rate and F-value of the CNN algorithm in Figure 5 showed that the CNN algorithm had good accuracy, i.e., a high precision rate and a recall rate for evaluating graphic designs, and the high F-value further illustrated the stable comprehensive performance of the CNN algorithm for evaluating graphic designs.

The scores of the three graphic designs for promoting the Fengxiang painted clay sculpture were collected from the ten experts. The average scores are shown in Table 1, which also includes the scoring results of the CNN algorithm. The final decision layer of the AHP method was the final score of the graphic packaging design. The final score of design (1) given by manual evaluation was 7.18, and the final score of design (1) given by the CNN algorithm was 7.02. The final score of design (2) given by manual evaluation was 8.25, and the final score of design (1) given by the CNN algorithm was 8.09. The final score of design (2) given by manual evaluation was 7.58, and the final score of design (1) given by the CNN algorithm was 7.73. The reason for the small difference in the final scores between the three graphic designs is that they were all designed to promote the local painted clay sculptures, i.e., the difference in the overall composition was small. In addition, the comparison of scores between the manual evaluation and the CNN algorithm showed that the CNN algorithm only differed from the manual evaluation in one aspect, and the final score given by the CNN algorithm was similar to the actual score of the manual evaluation, which further verified the effectiveness of the CNN algorithm for graphic design evaluation.

3.4 Discussion

This paper describes the application of the reorganization of traditional visual elements in graphic design and presents an analysis of three graphic designs for promoting the Fengxiang painted clay sculpture. The AHP method was used in the example analysis. The recombinant traditional visual elements to be evaluated in the graphic design were divided into three intermediate layers: composition and pattern, color layout and text description. Every intermediate layer was divided into two target layers, and the target layers were scored using questionnaires. Finally, the evaluation scores were summarized. In addition, in order to improve the evaluation efficiency of graphic designs and reduce the consumption of manual evaluation, the CNN algorithm, an intelligent algorithm, was used to assist in the evaluation. The obtained results were compared with those of manual evaluation, and the final results have shown above. The CNN algorithm trained with the training set got good test results, which initially verified the evaluation effect of the CNN algorithm on recombinant traditional visual elements in graphic design. After that, the CNN algorithm was applied to evaluate three graphic designs, and the results were compared with the results of manual evaluation. The final results further verified the evaluation effectiveness of the CNN algorithm on recombinant traditional visual elements in graphic design.

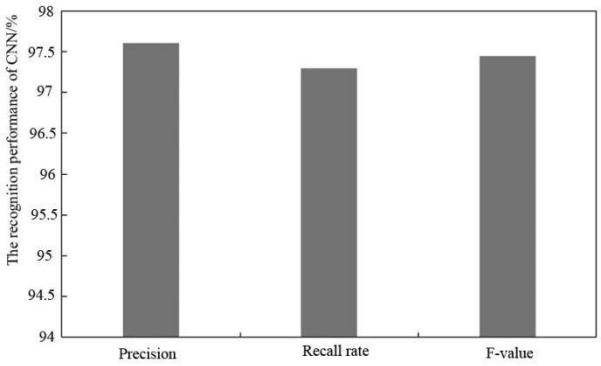


Figure 5: Performance of the CNN algorithm for evaluating graphic designs.

Middle layer	Composition and pattern		Color layout		Text description	
Weight	0.3		0.4		0.3	
Target layer	Novelty level	Suitability	Sharpness	Reasonableness	Font fit	Reasonable degree of layout
Weight	0.4	0.6	0.4	0.6	0.5	0.5
Design (1) (Manual)	7	8	7	7	8	6
Design (1) (CNN)	7	8	6	7	8	6
Design (2) (Manual)	8	8	9	9	9	6
Design (2) (CNN)	8	8	8	9	9	6
Design (3) (Manual)	9	8	8	7	8	6
Design (3) (CNN)	9	8	8	7	8	7

Table 1: Questionnaire results of three graphic designs.

According to the results of the questionnaire survey in the AHP method and the evaluation results of the CNN algorithm, the three graphic designs were analyzed in detail. Design (1) used red as the main color, which was quite eye-catching in tone. In terms of composition and pattern, the background of design (1) adopted the fish scale pattern of pottery decoration, which increased the sense of depth and strengthened the impression of "porcelain". The non-background composition appeared to be a mask at first glance, but it was actually an abstraction of the head front of a painted clay figure in the shape of a rooster. The main element, the head front of the

painted clay sculpture of a rooster, was retained. The flattened head front instead of the entire shape of a rooster was used as the main composition and filled with other clay sculpture-related elements. As to the text description, there were three text strings in a vertical layout, one was "Fengxiang painted clay sculpture" in "SimYou" font, one was the pinyin of "Fengxiang painted clay sculpture", and one was "Fengxiang clay sculpture" in Hui font, and a cloud was added to the right of the word "Xiang" as a decoration. The three text strings in vertical layout directly indicated the content of the graphic design, and "Fengxiang clay sculpture" in Hui font also served as an attractive decoration.

In terms of composition and pattern, design (2) also used the fish scale pattern in the background, which had the same effect as described above. The non-background composition also appeared to be a mask at first glance, but it was an abstraction of the head front of a "dog" in the painted clay sculpture. Similar to design (1) described above, the flattened head of a "dog" was used as the main composition and filled with some other elements. In terms of color, this design used the color of green tile as the main color, which was not as eye-catching as red, but this color gave people a sense of bluestone. In terms of textual description, it was consistent with design (1).

In design (3), the background also used the fish scale pattern. The non-background composition was not the same as the animal clay sculpture used in the above two designs, but a traditional visual element, "kite", was used and filled with the element of painted clay sculptures. In terms of color, the light cyan color was chosen to match the "kite" element. The textual description was the same as that of design (1).

Fengxiang painted clay sculptures have a variety of shapes, the most common of which are the twelve Chinese zodiac signs. Since they are "painted", they are colorful in appearance. The three graphic designs analyzed in this paper deconstructed and recombined the traditional visual element, "painted clay sculpture". Designs (1) and (2) deconstructed the frontal images of the head from rooster and dog sculptures and recombined them by filling some colors. The main image in design (3) did not apply the head shape of the clay sculpture, but adopted "kite" as the main composition and filled it with the color and composition elements of the painted clay sculpture, realizing the reorganization of two traditional visual elements, "kite" and "clay sculpture".

4 Conclusion

This paper briefly introduced the traditional visual elements and their main applications and analyzed three graphic designs used for promoting the Fengxiang painted clay sculpture. The AHP method was used, and the CNN algorithm was used for auxiliary analysis. The results are as follows. The trained CNN algorithm had a precision rate of 97.6%, a recall rate of 97.3% and an F-value of 97.5% in evaluating the samples in the test set. The difference between the scores obtained using the CNN algorithm and the AHP method was not significant, which verified the effectiveness of the CNN algorithm in assisting graphic design analysis. The analysis of the three graphic designs by manual evaluation and the CNN algorithm showed that the three graphic designs effectively deconstructed and recombined "painted clay sculpture", which improved the attractiveness of painted clay sculptures.

References

[1] Yoxall A, Gonzalez V, Best J, Rodriguez-Falcon EM, Rowson J (2019). As you like it: Understanding the relationship between packing design and accessibility. Packaging Technology and Science, 32. https://doi.org/10.1002/pts.2466

[2] Wang F (2015). Research on Design Factors and Packing Design Method of Preschool Children's Food Packing Security. Advance Journal of Food Science and Technology, 9, pp. 434-438. https://doi.org/10.19026/ajfst.9.1898

[3] Alexa A (2015). Books: graphic discourse. Metropolis, 34, pp. 168-168.

[4] Shen YC, Yuan WH, Hsu WH (2015). Color Selection in the Consideration of Color Harmony for Interior Design. Color Research & Application, 25, pp. 20-31.

[5] Yuan G, Xie Q, Pan W (2017). Color design based on Kansei engineering and interactive genetic algorithm. Academic Journal of Manufacturing Engineering, 15, pp. 12-18.

[6] Velasco C, Woods AT, Spence C (2015). Evaluating the orientation of design elements in product packaging using an online orientation task. Food Quality and Preference, 46, pp. 151-159. https://doi.org/10.1016/j.foodqual.2015.07.018

[7] Sirimamilla A, Ye H, Wu Y (2019). Phenomenological Modeling of Carpeted Surface for Drop Simulation of Portable Electronics. Journal of Electronic Packaging, 141, pp. 021006.1-021006.6.

[8] Ai XQ, Wu ZD, Guo T, Zhong JY, Hu N, Fu CL (2021). The effect of visual attention on stereoscopic lighting of museum ceramic exhibits: A virtual environment mixed with eye-tracking. Informatica, 45.

[9] Qian L, Xia Y, He X, Qian K, Wang J (2018). Electrical modeling and characterization of silicon-core coaxial through silicon vias in three-dimensional integration. IEEE Transactions on Components, Packaging, and Manufacturing Technology, PP, pp. 1-1. https://doi.org/10.1109/TCPMT.2018.2854829

[10] Wu ZD (2018). Empirical Study on the Optimization Strategy of Subject Metro Design Based on Virtual Reality. Informatica, 42.

[11] Liao K H (2015). The abilities of understanding spatial relations, spatial orientation, and spatial visualization affect 3D product design performance: using carton box design as an example. International

Journal of Technology & Design Education, 27, pp. 1-17. https://doi.org/10.1007/s10798-015-9330-3

[12] Chtioui I, Bossuyt F, de Kok J, Vanfleteren J, Bedoui MH (2016). Arbitrarily Shaped Rigid and Smart Objects Using Stretchable Interconnections. IEEE Transactions on Components Packaging & Manufacturing Technology, 6, pp. 533-544. https://doi.org/10.1109/TCPMT.2015.2511077

[13] Bhandari A K, Kumar A, Chaudhary S, Singh GK (2016). A novel color image multilevel thresholding based segmentation using nature inspired optimization algorithms. Expert Systems with Applications, 63, pp. 112-133. https://doi.org/10.1016/j.eswa.2016.06.044

[14] Lee D, Plataniotis K N (2016). Toward a No-Reference Image Quality Assessment Using Statistics of Perceptual Color Descriptors. IEEE Transactions on Image Processing, 25, pp. 3875 - 3889. https://doi.org/10.1109/TIP.2016.2579308

[15] Xie Q, Zhao Q, Xu Z, Meng D (2020). Color and direction-invariant nonlocal self-similarity prior and its application to color image denoising. Science China (Information Sciences), v.63, pp. 87-103. https://doi.org/10.1007/s11432-020-2880-3

[16] Min JK, Lee JH (2016). A Study on the Specialized Color Plan of Industrial Complex Reflecting Local Identity According to Intergrated Design Communication - Focused on Gimpo Goldvalley Industrial Complex. Journal of the Korean Society Design Culture, 22, pp. 169-182.

[17] Zhang QM, Luo J, Cengiz K (2021). An Optimized Deep Learning based Technique for Grading and Extraction of Diabetic Retinopathy Severities. Informatica, 45.

[18] Gaafar AS, Dahr JM, Hamoud AK (2022). Comparative Analysis of Performance of Deep Learning Classification Approach based on LSTM-RNN for Textual and Image Datasets. Informatica, 46.

[19] Li C, Guo J, Guo C (2017). Emerging from Water: Underwater Image Color Correction Based on Weakly Supervised Color Transfer. IEEE Signal Processing Letters, PP, pp. 1-1. https://doi.org/10.1109/LSP.2018.2792050

An Illustration of Rheumatoid Arthritis Disease Using Decision Tree Algorithm

Uma Ramasamy and Santhoshkumar Sundar
E-mail: seen.uma25@gmail.com, santhoshkumars@alagappauniversity.ac.in
Alagappa University, Karaikudi, Tamil Nadu, India

Student paper

Keywords: rheumatoid arthritis, decision tree, entropy, information gain, gain ratio

Received: November 13, 2020

The Data Mining domain integrates several partitions of the computer science and analytics field. Data mining focuses on mined data from a repository of the dataset to identify patterns, discover knowledge, additionally to predict probable outcomes. Decision tree belongs to classification techniques is a well-known method appropriate for medical diagnosis. Iterative Dichotomiser 3 (ID3) is the general significant algorithm to construct a decision tree. C4.5 is the successor of ID3 that handles dataset contains different numerical attributes. Although many studies have described and compared different decision tree algorithms, some studies have confined paper with analysis and comparison of the decision tree algorithm without the output of the decision tree. One of the inflammatory diseases is Rheumatoid Arthritis (RA) caused by specific autoantibodies with the destruction of synovial joint autoantibodies. Medical dataset applied to construct a decision tree as output has become seldom study. This study elucidates to explore the medical dataset with the decision tree approach and exhibit the derived decision tree output from the RA dataset. The objective of this paper is to construct a decision tree and display the prominent features that predict RA from the RA dataset using the decision tree algorithm.

Povzetek: Za predstavitev bolezni revmatoidnega artritisa so uporabili metodo za gradnjo odločitvenih dreves.

1 Introduction

Rheumatoid Arthritis (RA) is a rheumatic disease. The word 'Rheumatoid' implies 'rheumatism' relates to a musculoskeletal illness, 'arthr' means 'to joints,' and 'itis' denotes 'inflammation.' It is an inflammatory disorder that mostly impairs the joints, as well as other organs like the skin and lungs. Well-defined and reliable estimation of RA symptoms circumvents durable destruction to the patient's joints and bones if treated earlier, or else it affects the patient's quality of life. The research gap has found in the field of Rheumatoid Arthritis using data mining [1, 2].

A dataset is an indispensable component in the discussion of the classification algorithm. The dataset features or attributes are qualitative (nominal) and quantitative (numeric). Many researchers have applied various datasets [3-6] on different classification algorithms and have processed different results based on it. The dataset was utilized as a training set. From the training set decision tree is built.

'Playing tennis' is often used dataset in the decision tree illustration [7-10]. Preferably the next used dataset in the decision tree example is the student performance [11]. Similarly, dataset like 'a dog represents a risk for citizens [7, 12],' 'reservoir inflow forecasting [13],' 'PEP (Portfolio Evaluation Plan) [14],' 'rainfall forecasting [15],' and 'college scholarship evaluation [16],' are some illustrations in the classification algorithm that rarely handled by many research authors. A few authors only have examined and published medical datasets for the decision tree illustration.

The medical dataset created for this study is named the 'RA dataset.' The RA dataset was obtained from a new approach of the 2010 ACR/EULAR (American College of Rheumatology / European League Against Rheumatism) classification criteria of RA, which was formed, by two active groups of the ACR and the EULAR [17]. It contains qualitative attributes in a binary category (yes / no). This dataset aims to diagnose whether the patients have Rheumatoid Arthritis or do not have Rheumatoid Arthritis.

Most RA patients experience abhor pain on the joints of the hands, legs, hip, spine, and shoulder. It would be beneficial for medical practitioners to predict the prominent features responsible for RA disease. The feasible attributes to identify RA patients are displayed in Figure 3. Among these feasible attributes, the optimal attributes for the RA patient are predicted in this study using the RA dataset.

Information gain was determined to find the dominant attributes from the dataset to build the decision tree for the iterative dichotomiser 3 (ID3) algorithm. C4.5 is another algorithm to construct the decision tree by calculating the gain ratio. Decision tree algorithms such as ID3 and C4.5 (modified version of ID3) are popular and efficiently used classifiers for RA prediction from a RA dataset**.** Only a few authors practiced the decision tree illustration with

medical datasets [18,19]. Although many authors have described and compared the decision tree algorithms, some confined their papers without the relevant decision tree result.

2 Related study

Data mining is the method to classify models from massive databases, that being broad, applied to learn and analyze, and obtain information [20-23]. The decision tree algorithm falls under the type of supervised learning. It is the most familiar data mining technique used frequently to build the classification model. They are used to solve both regression and classification problems. All classification model, function with the classifier, which is a supervised learner that automatically perform the learning process for the training dataset, to predict its target attribute. Data mining techniques are widely used for classification and prediction of the healthcare domain so that it can be an aid for the doctors to identify complex diseases precisely and design a more reliable Decision Support System (DSS) [24].

The Electronic Health Record (EHR) of RA patients were studied for early prediction and diagnosis of the RA disease. Moreover, the comparative study made on several machine learning algorithms identifies which algorithm suites well for the prediction of RA disease [25, 26]. rheumatoid factor (RF), anti-cyclic citrullinated peptide (Anti-CCP), swollen joint count (SJC), and erythrocyte sedimentation rate (ESR) are four essential judging factors for rheumatoid arthritis [27]. Once a patient is diagnosed with RA, the probability of getting heart failure is higher compared to the non-RA patient [28-31]. Medical research and biological research are the ever-growing fields where many biological data are collected, classified, estimated, predicted, associated, clustered, and finally visualized through reports and patterns using data mining techniques [32, 33].

The application of data mining is always in the progress of continuous development. The ID3 algorithm has some issues to handle multi-valued attributes and requires a high amount of computational complexity. A novel approach has been introduced to split attributes in the ID3 algorithm [34]. In the field of bioinformatics, data mining has some challenges like sequencing technologies and data analysis skills. Under analysis estimation instruments, a review of data mining methods performs with the combination of examination tools suitable in research tasks. The literature review finalized the merits and demerits of data mining in bioinformatics [35].

After simulation analysis, ID3 decision tree classification accuracy was higher 6-7 percentage compared to other classifiers. The author proposed an optimized ID3 algorithm that constructs a tree with a minimum node so that it can improve the efficiency and reduce the error rate [36]. Using the Gaussian mixture model, the analysis done using different clinical and laboratory data displayed results with various distributions. The patient global assessment (PGA) and health assessment questionnaire (HAQ) collected after three months of RA diagnosis, SJC, and tender joint count (TJC) considered being the functional attribute for RA diagnosis [37]. Regarding Arthritis disease, women are affected at a higher rate when compared to men [38]. The RA prediction and the RA diagnosis development done by the machine learning approach, it is mandatory to diagnose the essential features for RA prediction [39, 40].

The earlier study practiced the decision tree computation technique to investigate the selection of the second-line drug DMARDs (Disease Modifying Antirheumatic Drug) by rheumatologists which depend on the factor of disease rigor to treat RA patients after the failure of Methotrexate [41]. A few years back the immune suppression effects of DMARDs are systematic and lead to various side effects. Medical experts improved autoimmune response produced from RA by customizing a good care plan and predicting the prognosis of the disease [42]. A recent study was made to support clinical RA treatments using the decision support system to predict a model that can support medical people to give suitable decisions in the early stage of RA disease [43].

The specific proteomic biomarkers have identified for RA diagnosis using matrix-assisted laser desorption/ionization time-of-flight mass spectrometry (MALDI-TOF-MS) combined with weak cationic exchange (WCX) magnetic beads. The classification tree model has been considered an innovative diagnostic tool for RA [44]. The combination of proteomic fingerprint technology and magnetic beads obtained efficient biomarkers and discovered the diagnostic patterns for RA. The biomarker C-C motif chemokine 24 (CCL24) has considered as a significant diagnostic indicator for RA [45].

The author states anti-citrullinated protein antibodies (ACPAs) are specific for RA and, RF was observed in health and elder people with other autoimmune diseases, which indicate immune response for RA development. The shared epitope alleles dwell in the major histocompatibility complex (MHC) class II region involved in a genetic risk factor for RA development. ACPA is the spectrum of autoantibodies that aims for posttranslational modification (PTM) [46].

The authors declare that in the future machine learning (ML) will support rheumatologists to analyze and predict the development of the disease and discover significant disease agents. Furthermore, the authors affirm ML will perform treatment propositions and evaluate their predicted outcome. The shared decision-making combines the patient's viewpoint, rheumatologist's suggestion, and also machine-learned evidence in the future [47].

The general methodologies applied to examine the intensity of RA are the clinical, laboratory, and physical examinations. The authors proposed a hybrid optimization strategy called rheumatoid arthritis disease using weighted decision tree approach (REACT), which combines the features of ID3 and particle swarm optimization (PSO) for feature selection and classification of RA to improve the efficiency and reliability of RA diagnosis [48].

It is necessary to develop therapies for RA patient's treatment at each stage of the disease progress using pathological mechanisms that urge the deterioration of RA progress in individuals. Several modern pharmacologic

therapies play a vital role in disease relief without joint deformity. The RA pathogenesis, disease-modifying drugs, and views on next-generation therapeutics for RA have been discussed in this review [49]. Though joint connection, serology, levels of acute-phase reactants, and the duration of the symptoms are marked to be the primary diagnosis classification criteria for RA, yet the diagnosis requires well trained specialists who can discern early symptoms of RA from additional pathology [50].

The paper [51] developed a model for the flare prediction on the RA patients, with reduced intake of biological disease-modifying anti-rheumatic drugs (bDMARDs) in sustained remission. This proposed model used nested cross-validation and optimal hyper-parameters for a suitable model selection approach with machine learning algorithms like Logistic Regression, k-Nearest Neighbors, Naïve Bayes and Random Forest. A dose reduction, feature was selected to be the predominant flare predictor attribute.

A new method [52] focused to promotes the treatment selection in RA patients using GUIDE (Generalized, Unbiased, Interaction Detection and Estimation) decision tree, which matches with predefined rules to predict treatment response to sarilumab and adalimumab. The result classified the presence of Anti-CCP and C-reactive protein (CRP) with a threshold greater than 12.3mg/l exposed as a biomarker pattern to predict response to sarilumab.

Since RA diagnosis is prominently challenging because of reliable biomarkers, the authors [53] identified nine hub genes namely CFL1 (Cofilin 1), COTL1 (Coactosin Like F-Actin Binding Protein 1), ACTG1 (Actin Gamma 1), PFN1 (Profilin-1), LCP1 (Lymphocyte Cytosolic Protein 1), LCK (lymphocyte-specific protein tyrosine kinase), HLA-E(Major Histocompatibility Complex, Class E), FYN (Proto-oncogene tyrosine-protein kinase), and HLA-DRA (Human Leukocyte Antigen – DR isotype) biomarkers that probably distinguished the RA samples out of 52 differentially expressed genes (DEGs) from 112 RA patients. Further, Machine Learning models namely logistic regression and random forest were applied based on the identified genes.

This paper [54] presents a review that summarized the healing treatment for RA. The objective was to highlight, polypeptides, small intermediate or end products of metabolism, and epigenetics regulators as the new targets for healing RA. And prominent molecular targets for medication design were identified, which lessen the early RA and determine nonresponses followed by the partial responses and severe effects for modern DMARDs.

Algorithm Pipeline Development and Validation Study were conducted on this paper [55] using EHR to identify patients with RA. Patients' records who had their first visits were suggested as input from EHRs, and Natural Language Processing (NLP) text processing was applied from randomly selected EHRs. Moreover, Six Machine Learning Methods were utilized in the training and 10- fold cross-validation dataset to identify patients with rheumatoid arthritis from format-free text fields of EHRs.

In this paper [56] dataset taken from The Korean College of Rheumatology Biology (KOBIO) Registry, nearly 1204 RA patients were treated with biologic disease-modifying anti-rheumatic drugs (bDMARDs). To predict remission machine learning techniques included Lasso, Ridge, SVM, Random Forest, and Xgboost and explainable artificial intelligence (XAI) were used to identify the essential clinical features correlated with remission. The accuracy and area under the receiver operating characteristic (AUROC) curve were analysed for prediction.

Treatment guideline for RA patients is given in this paper [57], many references and research work associated with vaccination were collected from precise literature reviews formed by ACR guidelines to deal with RA. These studies recommend services to assist the clinician and patient decision-making and relieve them from RA disease anxiety. In this study, let us analyze the RA dataset using the decision tree model and predict the efficient features that diagnose the disease.

3 About decision tree

A tree structure classifier is the decision tree with a decision node or internal node, a branch, and a leaf node. The test of the attribute has denoted by each internal node.Each leaf node predicts the target classification. Each branch corresponds to the attribute value. To classify training dataset using the decision tree, begin from the root node, follow the suitable decision branches corresponding to the attribute values, and finally reach a leaf-node predicted with the target class. The conjunction of attribute tests corresponds to each path from the root to the leaf. Further, as a whole, the disjunction of these conjunctions represents the tree [*58*]. The dominant attribute is the best attribute classifier from the training set. The internal node represents the dominant attribute that supports to build the decision tree. The dominant attribute is the attribute with the highest information gain and gain ratio, which is discussed in sections 4.2 and 4.3.

3.1 Algorithm

3.1.1 ID3

A set of training examples are processed to learn and construct the decision tree. Furthermore, with the learned classifier, the decision tree classifies the new training examples. The algorithm technique employed is from the basic top-down greedy approach. The fundamental algorithm to build the decision tree is the ID3 algorithm developed by Quinlan in 1973 based on the Concept Learning System (CLS) algorithm. ID3 finds the dominant attribute that classifies the training examples by applying a greedy search and never backtrack [*58*], [*59*] (p.55).

3.1.2 C4.5

ID3 cannot handle practical issues such as attributes with missing values in the training dataset and attributes with continuous values. Additional problems to handle are a small sample of data leads to overfitting, to select an

attribute for the decision node, one feature tested at the moment is time-consuming, and it is sensitive with a greater number of attribute values. Practical issues in ID3 overcome by the C4.5 algorithm, stated by Ross Quinlan, create the decision tree. C4.5 is a continuation of Quinlan's earlier ID3 algorithm [59] (p.55).

4 Metrics of ID3 and C4.5

Decision tree metrics are a set of measurement support to draw a decision tree with some parameters quantitative assessment derived from the dataset.

4.1 Entropy

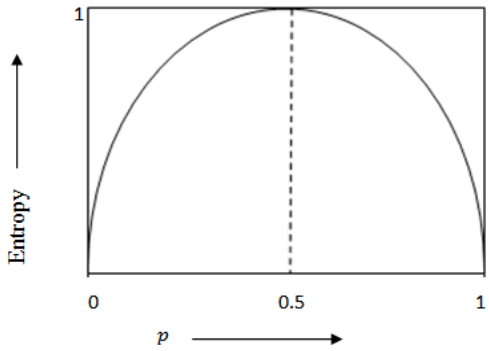


Figure 1: Entropy function relative to binary classification.

S is the sample of training examples (size =10). In the S dataset, positive proportion examples denoted as 'p,' and negative proportion examples denoted as 'n.' Entropy(S) is zero, if the proportion of positive examples (10+, 0-) is the same as the size of the training examples, similarly if the proportion of negative examples (0, 10-) is the same as the size of the training examples. Suppose, positive and negative examples are of equal size (5+, 5-), the impurity in the dataset S is maximum, i.e., entropy is one as shown in Figure 1. Therefore, it is distinct that the impurity of dataset S is measured by entropy.

Entropy(S) is the expected number of bits needed to encode class (true or false, + or -, yes or no, low or medium or high) of randomly drawn members of S. A novel way to assign $-\log_2 p$ bits to messages having probability 'p' introduced in the Information Theory concept of optimal length code [58]. So the expected numbers of bits to encode (yes or no, true or false, + or -) a random member of S is $-p\log_2 p - n\log_2 n$, where positive examples proportion denoted as 'p,' and negative examples proportion denoted as 'n.' Entropy characterizes the impurity of a collection and measures the information content from the sample of training examples. If the number of unique target feature values assigned as m, then the entropy of S w.r.t n-wise classification is equated as

$$Entropy(S) = -\sum_{i=1}^{n} p_i \log_2 p_i \quad (1)$$

Where,

$p_i -$ *proportion of S belonging to class* c_i

4.2 Information Gain

Let S be the sample of the training examples with $A_1, A_2, \ldots, A_n$ are the non-target attributes. All the features in the dataset calculated using the information gain formula as shown in Equation 2. Attribute with the highest information gain is the best classifier because the expected reduction is laid out by the information gain in entropy formed by partitioning the records of the dataset using the attribute. How effectively an attribute classifies the training examples according to their target classification has been defined in the information gain measure [59] (p.57-58). WA(A) defines the weighted sum of the information content of each subset of the examples partitioned by the possible values of the attribute. It measures the total disorder or in-homogeneity of the leaf nodes. The minimum WA (A) or maximum information gain(S, A) shows attribute A as the best attribute at a node [58]. The best attribute to select in growing the tree using each step of the ID3 algorithm, a precise measure is the information gain. The calculation of information gain is briefly described in Section 7.

$$Gain(S,A) = Entropy(S) - WA(A)$$
$$= Entropy(S) - \sum_{v \in Values(A)} \left(\frac{S_v}{S}\right) Entropy(S_v) \quad (2)$$

Where,

$S_v -$ *subset from S for which attribute A has value v*
$Value(A) -$ *set of all possible values for attribute A*

4.3 Gain ratio

The gain ratio is a ratio between information gain and the split information. Rather than considering the entropy(S) on the target attribute, entropy(S) is concerned about all possible values of the attribute A defined to be the split information [59] (p.73-74). Information Gain Ratio is the fundamental information from the required decrease in entropy. The purpose of Quinlan to introduce this was to overcome bias on multi-valued features by considering the count of branches when choosing an attribute [60-62]. Section 7 discussed to implement gain ratio with an example.

$$GR(S,A) = \frac{IG(S,A)}{IV(S,A)} \quad (3)$$

$$IV(S,A) = \sum_{i=1}^{c} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|} \quad (4)$$

Where,

$GR(S,A)$
$-$ *Information Gain Ratio after splitting set S on attribute A*
$IG(S,A)$
$-$ *Information Gain after splitting set S on attribute A*
$IV(S,A)$
$-$ *Intrinsic Value or Split Information value after splitting set S on attribute A, where* S_i *through* S_c *are the c subsets of examples resulting from partitioning S by the* $c -$ *valued attribute A*

5 Work flow model for proposed illustration

The proposed illustration workflow model consists of a tree algorithm for RA [17], which is further converted to

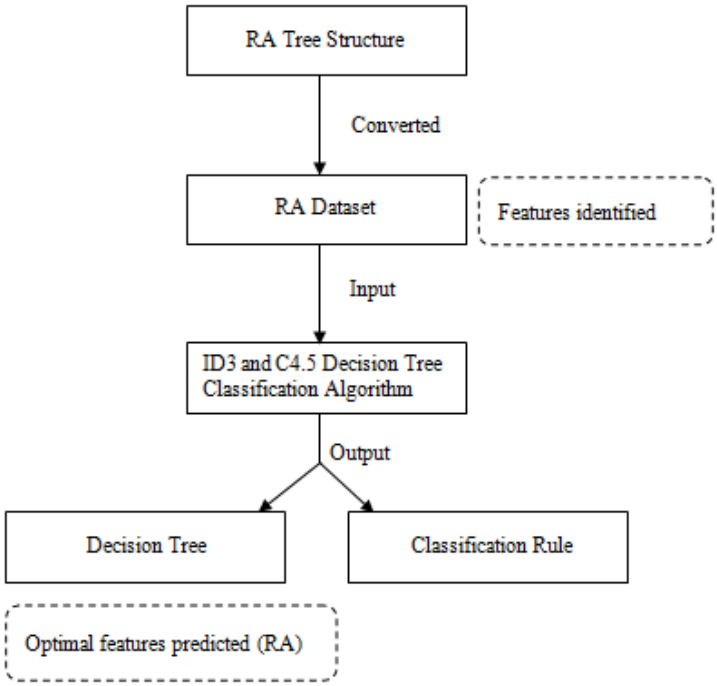


Figure 2: Work flow Model for Proposed Illustration.

a relational database as shown in Table 1. The resultant RA dataset is applied to computational techniques such as ID3 and C4.5 decision tree classifier to obtain decision tree and classification rule. The RA dataset contains all feasible features necessary to identify RA patients, whereas the final result of the decision tree predicts only the optimal features mandatory to predict RA patients.

6 About dataset

As mentioned in the workflow model (Figure 2), the conversion of RA Tree Structure (Figure 3) to RA dataset (Table 1) is done by following each path from the root node to the leaf node. The shape of the root node and the intermediate node is a rectangle, whereas the leaf node is in a circle (Figure 3). Each path represents each row in the RA dataset. There are 60 paths (in Figure 3), so the RA dataset consists of 60 rows. The root node in Figure 3 is '>10 joints (at least one small joint)', and the leaf nodes in the Figure 3 are 'RA' and 'crossed RA' (not RA). The root node and the intermediate node indicate the features/attributes, and the leaf node implies the class label of the RA dataset.

The aim of this dataset is to Classifying patients by diagnosis of Rheumatoid Arthritis or not Rheumatoid Arthritis. The source of our dataset is from the tree flowchart for classifying distinct Rheumatoid Arthritis (RA) given in the 2010 RA classification criteria. Two active groups of the ACR and the EULAR join together to form a new approach for the 2010 ACR/EULAR classification criteria of RA [17]. The number of instances (rows) of the RA dataset – 60. The number of features (columns) of the RA dataset – 9. Number of Classes (unique values of the target feature) – 2. Number of missing values – 0.

The attributes used to diagnosis RA are mixed of both phenotype and genotype. They are '> 10 joints (at least one small joint)', '4-10 small joints', '1-3 small joints', and '2 – 10 large (no small) joints' are four features of phenotype. 'Serology +' (low positive RF or low positive ACPA), 'Serology ++' (high positive RF or high positive ACPA), and 'APR (Acute phase reactants) Abnormal' (abnormal C-reactive protein (CRP) or abnormal ESR) are three features of genotype and the last attribute is 'Duration of symptoms >=6 weeks'. In Table 1, the features name is followed with a score value to classify RA patients. The cumulative score value of each attribute per record is less than 6 out of 10. Such a score is not classifiable to diagnose RA. Those scores status is yet to be evaluated, and the criteria might be later fulfilled [17].

7 Illustration of RA dataset with ID3 and C4.5 classifiers

RA [Rheumatoid Arthritis] dataset contains the data field of the qualitative binary asymmetric attribute. Binary data has two conditions such as, 'yes or no,' 'affected or unaffected,' ' true or false.' Asymmetric defines binary values are not equally important. Both the predictor (non-target attribute) and response (target attribute) variable in the RA dataset is binary and categorical. Two response variables 'ra' and 'no ra' suggest, diagnosis of rheumatoid arthritis and not rheumatoid arthritis.

7.1 Step-by-step illustration of ID/C4.5 algorithm using RA dataset

Step 1: Find the Entropy for the current RA dataset, S. In RA dataset 'ra' and 'no ra', two classes are present with the count of 26 and 34, total instances in the dataset are 60. The 'ra' target value informs the patient diagnosed with

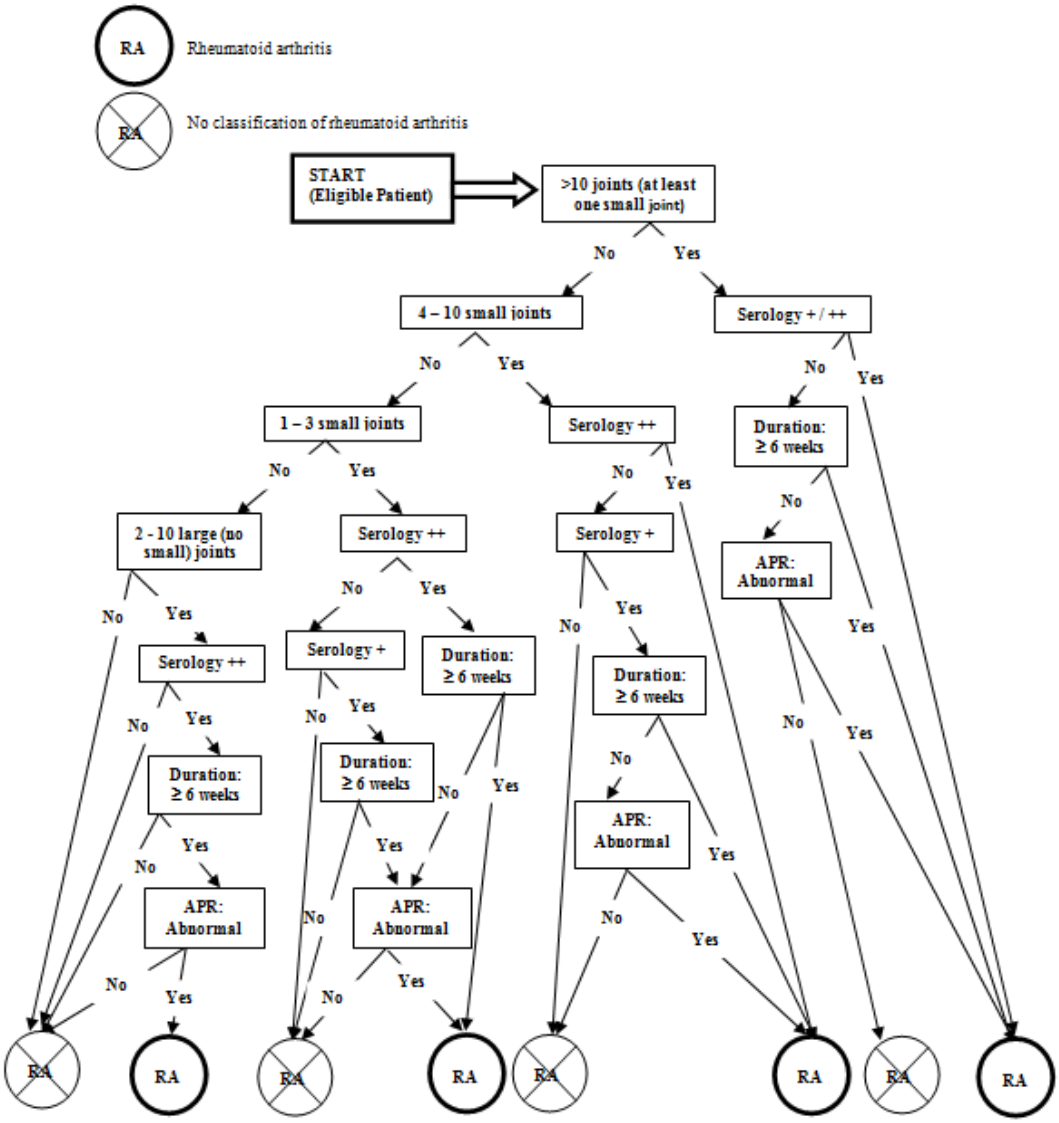


Figure 3 : Tree Algorithm for RA (Rheumatoid Arthritis).

Rheumatoid Arthritis, whereas the 'no ra' target value reveals the patient diagnosed with no Rheumatoid Arthritis. To draw the decision tree initial step is to measure the uncertainty for this dataset, i.e., the Entropy of dataset, S denoted as E(S). Calculate E(S) using Equation 1 discussed in Section 4.

$$\text{E(S)} = -\frac{26}{60}\log_2\frac{26}{60} - \frac{34}{60}\log_2\frac{34}{60} = 0.9871$$

Step 2: Find Information Gain by applying Equation 2 for each feature value in the RA dataset.

$$\begin{aligned} &Information\ Gain(S, > 10\ joints) \\ &= E(S) - \sum[p(S, > 10\ joints).E(S, > 10\ joints)] \\ &= E(S) - [p(S| > 10\ joints = yes) \\ &\quad * E(S, > 10\ joints = yes) \\ &\quad + p(S| > 10\ joints = no) \\ &\quad * E(S, > 10\ joints = no)] \end{aligned}$$

$$\begin{aligned} E(S, > 10\ joints = yes) &= -\frac{14}{16}\log_2\frac{14}{16} - \frac{2}{16}\log_2\frac{2}{16} \\ &= 0.5436 \end{aligned}$$

$$\begin{aligned} E(S, > 10\ joints = no) &= -\frac{12}{44}\log_2\frac{12}{44} - \frac{32}{44}\log_2\frac{32}{44} \\ &= 0.8454 \end{aligned}$$

$$\begin{aligned} &Information\ Gain(S, > 10\ joints) \\ &= 0.9871 - (\tfrac{16}{60} * 0.5436 + \tfrac{44}{60} * 0.8454) \\ &= 0.9871 - 0.7649 \\ &= 0.2222 \end{aligned}$$

Furthermore, obtain the information gain for enduring all feature values of the examples.

Step 3: Pick the feature which has the highest information gain. The attribute'> 10 joints' have the highest information gain, as shown in Table 2, '>10 joints' is the best classifier and determined as the root node as shown in Figure 4.

Calculate split information for each attribute using Equation 4.

$$SplitInformation(S, > 10\ joints)$$

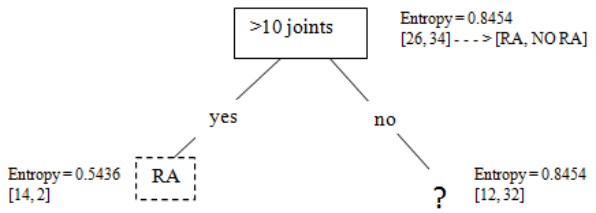


Figure 4 : Root node of the ID3/C4.5 decision tree using RA dataset.

$$= -\frac{16}{60}\log_2\frac{16}{60} - \frac{44}{60}\log_2\frac{44}{60} = 0.8366$$

Calculate Gain Ratio for each attribute using Equation 3.

$$GainRatio(S, > 10\ joints) = \frac{0.2222}{0.8366} = 0.2656$$

Now the decision tree node (root node) is the '>10 joints' attribute with a maximum of information gain (in the Table 2 it is represented as Info Gain). Since the RA dataset is categorical and not in continuous attribute, the decision tree built is the same for the ID3 and C4.5 algorithms. So, here the gain ratio measure is necessary to construct the decision tree using the C4.5 algorithm.

Step 4: Each branch from the attribute '>10 joints' partition the set S into subsets corresponds to the attribute value 'yes' and 'no.' From the root node '>10 joints', the 'yes' branch of the subset has 14 'RA' and 2 'NO RA' examples obtained. Though we can grow a tree further from the 'yes' branch, we have stopped with the target class RA, to avoid overfitting in the decision tree. This approach followed to stop growing the tree earlier before it attains the level to classify the training data perfectly [59] (p.68).

Now recurse (from step 2 to step 3) on the subset (from the root node '>10 joints', the 'no' branch of the subset has 12 'RA' and 32 'NO RA' mentioned as '?' in Figure 4) until the ID3 algorithm satisfies the stopping criteria [63] or by following the first-class approach to avoid overfitting [59] (p.68).

Step 5: The classification rule is generated from the decision tree.

7.2 Top-down generalization approach for the decision tree

Figure 5 illustrates the decision tree built from Table 1, which depicts the RA dataset, after applying the ID3 algorithm [58], [59] (p. 56).

The basic steps for the algorithm as follows:

- *Dmat ← dominant attribute for root (initial) / non-leaf node*
- *Set Dmat as dominant attribute for the node*
- *Every unique value of Dmat form new descendant*
- *Classify the dataset records to the leaf node corresponding to the dominant attribute value of the branch*
- *If complete dataset records are ideally classified (target feature has identical values) stop, else iterate over new leaf node*

The dominant attribute (decision attribute) is the best attribute classifier from the training set.

Table 1 : A Sample Dataset of RA [Rheumatoid Arthritis] Derived from Figure 3.

S.No.	>10 Joints (atleast 1 Small Joint) (5)	4 - 10 Small Joints (3)	1 - 3 Small Joints (2)	2 - 10 Large Joints (1)	Serology + (2)	Serology ++ (3)	APR: Abnormal (1)	Duration : >=6 Weeks (1)	Class Label
1	no	no	no	no	no	yes	yes	yes	no ra
2	no	no	no	no	yes	no	yes	yes	no ra
3	no	no	no	no	no	no	yes	yes	no ra
4	no	no	no	no	no	yes	no	yes	no ra
5	no	no	no	no	yes	no	no	yes	no ra
…	…	…	…	…	…	…	…	…	…
56	yes	no	no	no	no	no	yes	no	ra
57	yes	no	no	no	no	no	no	yes	ra
58	yes	no	no	no	no	no	yes	yes	ra
59	yes	no	no	no	no	no	no	yes	ra
60	yes	no	no	no	no	no	yes	yes	ra

Table 2: A sample of Information gain and gain ratio for RA dataset.

Features	Features Values	ra	no ra	Tot. freq. count	E(t)	p(t)*E(t)	Info Gain	Split Info	Gain Ratio
>10 Joints (atleast 1 Small Joint) (5)	yes	14	2	16	0.5436	0.7649	0.2222	0.8366	0.2656
	no	12	32	44	0.8454				
4 - 10 Small Joints (3)	yes	7	5	12	0.9799	0.9708	0.0163	0.7219	0.0226
	no	19	29	48	0.9685				
1 - 3 Small Joints (2)	yes	4	8	12	0.9183	0.9797	0.0074	0.7219	0.0103
	no	22	26	48	0.995				
2 - 10 Large Joints (1)	yes	1	7	8	0.5436	0.9382	0.0489	0.5665	0.0863
	no	25	27	52	0.9989				
Serology + (2)	yes	8	8	16	1	0.9824	0.0047	0.8366	0.0056
	no	18	26	44	0.976				
Serology ++ (3)	yes	12	8	20	0.971	0.9464	0.0407	0.9183	0.0443
	no	14	26	40	0.9341				
APR: Abnormal (1)	yes	16	14	30	0.9968	0.9576	0.0295	1	0.0295
	no	10	20	30	0.9183				
Duration: >=6 Weeks (1)	yes	16	14	30	0.9968	0.9576	0.0295	1	0.0295
	no	10	20	30	0.9183				

ID3 algorithm follows with following input and output.
Input: Datts ←A set of non-target attributes, R← target attribute and D← training examples.
Output: returns a decision tree.

ID3(Datts, R, D)
Step 1: If D is null, return a single node with value Failure
Step 2: If D holds the records of the same class, it returns a single leaf node with that value.
Step 3: If Datts is null, then return a node with the value of the most frequent value of R in D.
Step 4: Begin
4.1: Dmat← the attribute from Atts that best classifies D*
4.2: tree← a new decision tree with root test Dmat
4.3: for each value v_j of Dmat do
4.3.1: D_j ← subset of D with Dmat= v_j
4.3.2: subt ← ID3(Datts-Dmat, R, D_j)
4.3.3: Add a branch to the tree with label v_j and subtree subt
4.4: return tree.
** The highest information gain is the best attribute defined in Equation 2.*

7.3 Extracting classification rule from decision tree algorithm ID3 & C.5

- Classification rules outline the information in the pattern of IF-Then rules
- Single rule is built for each way starting from the root node to a leaf node
- Each attribute-value pair along a path makes an association
- The leaf node contains the predicted class [64]

Classification Rule extracted from Figure 5 decision tree:

Rule 1: If *> 10 joints ="yes"* then class = "RA"

***Rule 2:** If *>10 joints ="no"* AND *4-10 small joints = "yes"* AND *Serology++ ="yes"* then class="RA"

Rule 3: If *>10 joints ="no"* AND *4-10 small joints ="yes"* AND *Serology++ ="no"* AND *Serology+ ="yes"* then class="RA"

***Rule 4:** If *>10 joints ="no"* AND *4-10 small joints ="yes"* AND *Serology++ ="no"* AND *Serology+ ="no"* then class=" NO RA"

Rule 5: If *>10 joints ="no"* AND *4-10 small joints="no"* AND *1-3 small joints ="no"* then class="NO RA"

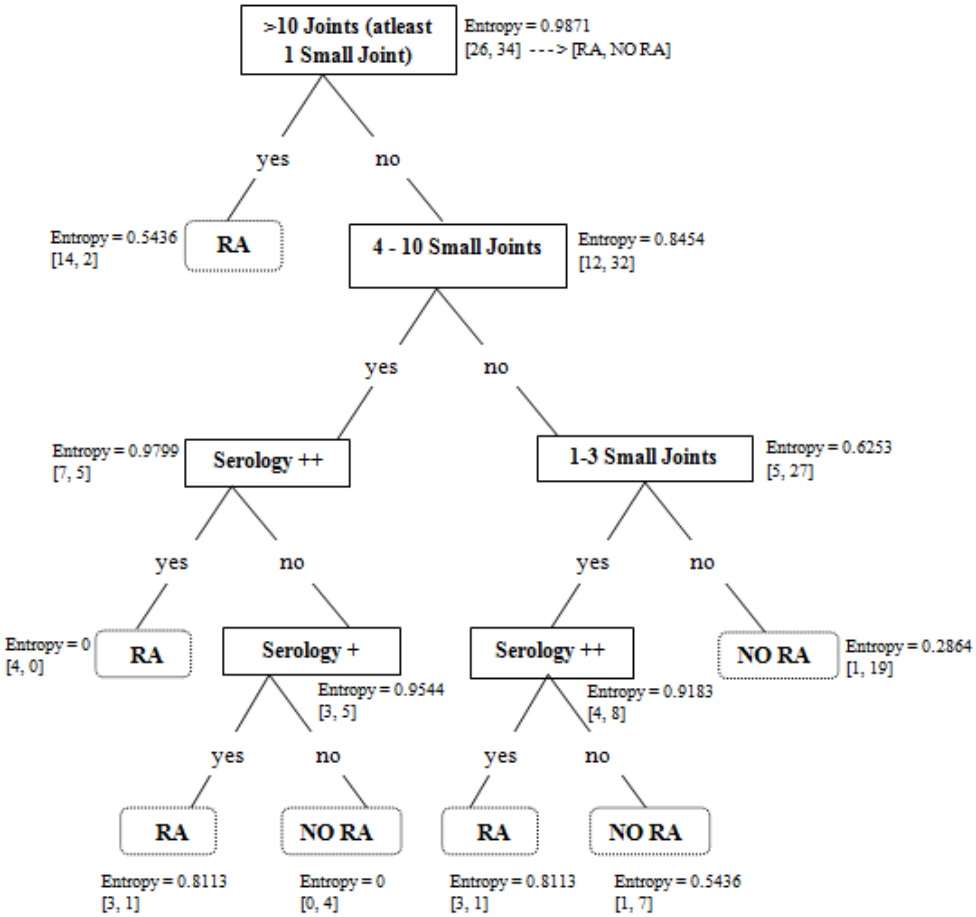


Figure 5: ID3 and C4.5 Final decision tree for RA dataset.

Rule 6: If *>10 joints ="no"* AND *4-10 small joints="no"* AND *1-3 small joints ="yes"* AND *Serology++ ="yes"* then class="RA"

Rule 7: If *>10 joints ="no"* AND *4-10 small joints="no"* AND *1-3 small joints ="yes"* AND *Serology++ ="no"* then class="NO RA"

* in rule denotes **Rule obtained from pure class**

Table 3 : Levelwise Leaf Node Membership for ID3 & C4.5 obtained from Figure 5

Level	Leaf Node Class Membership
Level 1	[14,2]
Level 2	-
Level 3	[4, 0]
	[1,19]
Level 4	[3,1]
	[0,4]
	[3,1]
	[1,7]

8 Illustration analysis report

The RA dataset consists of all possible feasible features from a RA patient. The predicted optimal features for RA disease are obtained using the classifier ID3 and C4.5. The Figure 5, describes the first predictor variable, '>10 joints' is achieved from level 1, the second predictor variable, ' 4-10 small joints' is identified from level 2, the third and fourth predictor variables namely *'serology ++' and '1-3 small joints'* exhibited from level 3 and finally, the fifth predictor variable, *'serology +'* is obtained from level 4. Therefore, five optimal features (predictor variables) are *'>10 joints', '4-10 small joints', 'serology ++', '1-3 small joints', and 'serology +' plays a vital role to predict RA patients.* The accuracy is 90% (54/60) for both ID3 and C4.5 decision tree. The performance is identical for both ID3 and C4.5 because the RA dataset contains categorical data. As shown in Table 2 (first level), for all the remaining levels in the decision tree, the information gain and gain ratio are simultaneously highest as displayed in Figure 6.

Table 4 : Performance of Class rulesets for ID3 & C4.5 in RA Dataset.

Class	Generalized Rule	Pure Rule	Instances covered	False Positive	False Negative
RA	4	1	24	0	4
NO RA	3	1	30	2	0

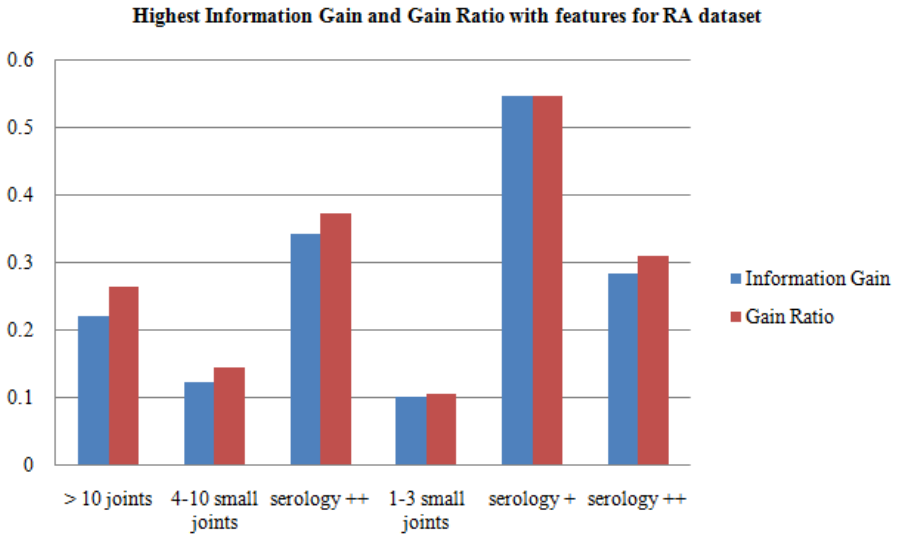


Figure 6 : Highest Information Gain and Gain Ratio for RA optimal features

9 Conclusion

The tree-structured data is converted to a relational database (RA dataset), to identify all feasible features for RA disease. Furthermore, the RA dataset is fed into the decision tree algorithm to obtain optimal features for RA disease. Therefore, we have explored the medical dataset to elucidate with the decision tree approach, and derived decision tree and classification rule as the output from the RA dataset. To summarize the work, ID3 and C4.5 decision tree algorithms construct the same decision tree with a classifier accuracy level of 90% for the RA dataset derived from the tree flowchart for diagnosing precise Rheumatoid Arthritis given in the 2010 RA classification criteria. ID3 and C4.5 classifiers result are equal in performances when considered with RA dataset.

Acknowledgment

This article has been published under AURF Start-up Grant 2018, Alagappa University, Karaikudi, Tamil Nadu, India, Dt. 23.03.2018.
This article has been published under RUSA Phase 2.0 grant sanctioned vide letter No. F. 24-51/2014-U, Policy (TN Multi-Gen), Dept. of Edn. Govt. of India, Dt. 09.10.2018.

References

[1] Zahra Shiezadeh, Hedieh Sajedi, and Elham Aflakie, “Diagnosis of Rheumatoid Arthritis using an Ensemble Learning Approach,” Computer Science and Information Technology. © CS & IT-CSCP 2015, pp. 139–148. https://doi.org/10.5121/csit.2015.51512

[2] Rohini Handa, Rao, U. R. K., Juliana F. M. Lewis, Gautam Rambhad, Susan Shiff, and Canna J. Ghia, “Literature review of rheumatoid arthritis in India International Journal of Rheumatic Diseases,” vol. 19, pp. 440-451, 2016. https://doi.org/10.1111/1756-185x.12621

[3] Jena, Monalisa, and Satchidananda Dehuri. "DecisionTree for Classification and Regression: A State-of-the Art Review." Informatica 44, no. 4 (2020). https://doi.org/10.31449/inf.v44i4.3023.

[4] Yang, Fen. "Decision tree algorithm-based university graduate employment trend prediction." Informatica 43, no. 4 (2019). https://doi.org/10.31449/inf.v43i4.3008.

[5] Kalyani, G., MVP Chandra Sekhara Rao, and B. Janakiramaiah. "Decision tree-based data reconstruction for privacy preserving classification rule mining." Informatica 41, no. 3 (2017). https://doi.org/10.1007/s13369-017-2834-2.

[6] Wang, Xi. "Research on Recognition and Classification of Folk Music Based on Feature Extraction Algorithm." Informatica 44, no. 4 (2020). https://doi.org/10.31449/inf.v45i4.3819.

[7] Halima ELAIDI, Zahra BENABBOU, Hassan ABBAR, A comparative study of algorithms constructing decision trees: ID3 and C4.5, LOPAL’18, May 2-5, 2018, 1-5, Rabat, Morocco © 2018 Association for Computing Machinery, https://doi.org/10.1145/3230905.3230916

[8] Badr HSSINA, Abdelkarim MERBOUHA, Hanane EZZIKOURI, Mohammed ERRITALI, A comparative study of decision tree ID3 and C4.5, International Journal of Advanced Computer Science and Applications, pp: 13-19, 2014. https://doi.org/10.14569/specialissue.2014.040203

[9] Sonia Singh, Manoj Giri, Comparative Study Id3, Cart and C4.5 Decision Tree Algorithm: A Survey, International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682 3(7): 47-52, July 2014, https://doi.org/10.15693/ijaist/2014.v3i7.47-52.

[10] Elaidi, Zahra Benabbou, Hassan Abbar, Using Game Theory to Handle Missing Data at Prediction Time of ID3 and C4.5 Algorithms, (IJACSA) International Journal of Advanced Computer Science and Applications, 9(12): 218-224, 2018. https://doi.org/10.14569/ijacsa.2018.091232

[11] Kalpesh Adhatrao, Aditya Gaykar, Amiraj Dhawan, Rohit Jha and Vipul Honrao, Predicting Students' Performance Using ID3 and C4.5 Classification Algorithms, International Journal of Data Mining & Knowledge Management Process (IJDKP) 3(5) , 39-52, September 2013. https://doi.org/10.5121/ijdkp.2013.3504

[12] Y.Wang, Y.Li, Y.Song, X.Rong, S.Zhang, Improvement of ID3 Algorithm Based on Simplified Information Entropy and Coordination Degree, Algorithms journal, 10 (4): 124, 2017. https://doi.org/10.3390/a10040124

[13] Pattama Charoenporn, Reservoir Inflow Forecasting Using ID3 and C4.5 Decision Tree Model, 2017 IEEE 3rd International Conference on Control Science and Systems Engineering, 978-1-5386-0484-7/17/$31.00 ©2017 IEEE, 698-701. https://doi.org/10.1109/ccsse.2017.8088023

[14] Sudrajat, I. Irianingsih, D. Krisnawan, Analysis of data mining classification by comparison of C4.5 and ID algorithms,, IOP Conf. Series: Materials Science and Engineering 166: 1-9, 2017. https://doi.org/10.1088/1757-899x/166/1/012031

[15] Joko Azhari Suyatno, Fhira Nhita, Aniq Atiqi Rohmawati, Rainfall Forecasting in Bandung Regency using C4.5 Algorithm, 2018 6th International Conference on Information and Communication Technology (ICoICT), ISBN: 978-1-5386-4571-0 (c) 2018 IEEE, 324-328. https://doi.org/10.1109/icoict.2018.8528725

[16] X. Wanga, C. Zhoua, X. Xub, Application of C4.5 decision tree for scholarship evaluations, The 10th International Conference on Ambient Systems, Networks and Technologies (ANT), The Authors. Published by Elsevier Ltd. ScienceDirect, Procedia Computer Science 151 (2019) 179–184. https://doi.org/10.1016/j.procs.2019.04.027

[17] Daniel Aletaha, Tuhina Neogi, Alan J. Silman, Julia Funovits, David T. Felson, Clifton O. Bingham., … Gillian Hawker(2010). "2010 Rheumatoid Arthritis Classification Criteria," Arthritis & Rheumatism, (62)9: 2569-2581, 2010. https://doi.org/10.1002/art.27584

[18] Lakshmi.B.N, Dr.Indumathi.T.S, Dr.Nandini Ravi, A study on C.5 Decision Tree Classification Algorithm for Risk Predictions during Pregnancy, International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST - 2015), Elsevier Ltd., ScienceDirect, Procedia Technology 24: 1542–1549, 2016. https://doi.org/10.1016/j.protcy.2016.05.128

[19] Yanwei Xing, Jie Wang and Zhihong Zhao, Yonghong Gao, Combination data mining methods with new medical data to predicting outcome of coronary heart disease, 2007 International Conference on Convergence Information Technology,204:868-872,2007. https://doi.org/10.1109/iccit.2007.204

[20] Begona Garcia-Zapirian, Yolanda Garcia-Chimeno, and Heather Rogers(2015), "Machine Learning Techniques for Automatic Classification of Patients with Fibromyalgia and Arthritis," International Journal of Computer Trends and Technology, 25(3):2231-2803,2015. https://doi.org/10.14445/22312803/ijctt-v25p129

[21] Begum Cigsar and Deniz Unal, Comparison of Data Mining Classification Algorithms Determining the Default Risk, Hindawi Scientific Programming Feb. 2019, Article ID 8706505.

[22] Nikita Jain, Vishal Srivastava, Data Mining Techniques: A Survey Paper, International Journal of Research in Engineering and Technology, 2(11), eISSN: 2319-1163, pISSN: 2321-7308, 2013. https://doi.org/10.15623/ijret.2013.0211019

[23] S. Umadevi, and K. S. Jeen Marseline, A Survey on Data Mining Classification algorithms, International Conference on Signal Processing and Communication (ICSPC' 17), July 2017. https://doi.org/10.1109/cspc.2017.8305851

[24] Shanmugam, S., & Preethi, J., "Design of Rheumatoid Arthritis Predictor Model Using Machine Learning Algorithms," SpringerBriefs in Applied Sciences and Technology, https://doi.org/10.1007/978-981-10-6698-6_7

[25] Vaishali S. Parsania, Krunal Kamani, and Gautam J Kamani, "Comparative Analysis of Data Mining Algorithms on EHR of Rheumatoid Arthritis of Multiple Systems of Medicine International," Journal of Engineering Research and General Science, 3 (1): 344-350, 2015.

[26] Beau Norgeot, MS; Benjamin S. Glicksberg, Laura Trupin, Dmytro Lituiev, Milena Gianfrancesco, Boris Oskotsky, Gabriela Schmajuk, Jinoos Yazdany, and Atul J. Butte, Assessment of a Deep Learning Model Based on Electronic Health Record Data to Forecast Clinical Outcomes in Patients with Rheumatoid Arthritis, JAMA Network Open. 2019,2(3). https://doi.org/10.1001/jamanetworkopen.2019.0606

[27] Jihyung Yoo, Mi Kyoung Lim, Chunhwa Ihm, Eun Soo Choi and Min Soo Kang (2017). A Study on Prediction of Rheumatoid Arthritis Using Machine Learning. International Journal of Applied Engineering Research, 12 (20). ISSN 0973-4562, pp. 9858-9862, 2017.

[28] Catia Sofia Tadeu Botas (2017), "Feature analysis to predict treatment outcome in Rheumatoid Arthritis," Instituto Superior Tecnico, Lisboa, Portugal. pp. 1-10, 2017.

[29] Elena Myasoedova, John M. Davis Ill, Eric L.Matteson, Sara J. Achenbach, Soko Setoguchi, Shannon M. Dunlay, Veronique L. Roger, Sherine E. Gabriel, and Cynthia S. Crowson, " Increased hospitalization rates following heart failure diagnosis in rheumatoid arthritis as compared to the general population," Seminars in Arthritis and Rheumatism, 50:25-29, 2020 © Elsevier Inc. https://doi.org/10.1016/j.semarthrit.2019.07.006

[30] Cynthia S Crowson, Katherine P Liao, John M Davis, Daniel H Solomon, Eric L Matteson, Keith L Knutson, Mark A Hlatky, and Sherine E Gabriel, Rheumatoid Arthritis and Cardiovascular Disease, NIH Public Access Am Heart J. 166(4):622–628, 2013. https://doi.org/10.1016/j.ahj.2013.07.010

[31] Usman Khalid, Alexander Egeberg, Ole Ahlehoff, Deirdre Lane, Gunnar H. Gislason, Gregory Y. H. Lip and Peter R. Hansen, Incident Heart Failure in Patients with Rheumatoid Arthritis: A Nationwide Cohort Study, Journal of the American Heart Association 2018. https://doi.org/10.1161/jaha.117.007227

[32] Khalid Raza, Application of Data Mining in Bioinformatics, Indian Journal of Computer Science and Engineering 1(2):114-118, 2012, ISSN: 0976-5166.

[33] Xing-Ming Zhao, Data Mining in Systems Biology, IEEE/ACM Transactions on Computational Biology and Bioinformatics 2016, vol 13(6), 1003-1003. https://doi.org/10.1109/tcbb.2016.2617698

[34] Zijing Wang, Yo Liu and Li Liu," A New Way to Choose Splitting Attribute in ID3 Algorithm," 978-1-5090-6414-4/17 ©2017 IEEE. https://doi.org/10.1109/itnec.2017.8284813

[35] Audu Musa Mabu, Rajesh Prasad, Raghav Yadav, Suleiman S Jauro," A Review of Data Mining Methods in Bioinformatics," Recent Advances of Engineering, Technology and Computational Sciences, 978-1-5386-1686-4/18 ©2018 IEEE. https://doi.org/10.1109/raetcs.2018.8443785

[36] He Zhang and Runjing Zhou," The Analysis and Optimization of Decision Tree Based on ID3 Algorithm," The 9th International Conference on Modeling, identification and Control, 2017. https://doi.org/10.1109/icmic.2017.8321588

[37] Jorn Lotsch, Lars Alfredsson, Jon Lampa, "Machine-Learning-based knowledge discovery in rheumatoid arthritis-related registry data to identify predictors of persistent pain," The International Association for the Study of Pain Research Paper – PAIN, 161:114-126, 2020. https://doi.org/10.1097/j.pain.0000000000001693

[38] Tiffany D. Pan, Beth A. Mueller, Carin E. Dugowson, Michael L. Richardson, and J. Lee Nelson, Disease progression in relation to pre-onset parity among women with rheumatoid arthritis, Seminars in Arthritis and Rheumatism 2019, 0049-0172/© 2019 Elsevier Inc. https://doi.org/10.1016/j.semarthrit.2019.06.011

[39] Ho Sharon, I. Elamvazuthi, CK. Lu, S. Parasuraman and Elango Natarajan, Classification of Rheumatoid Arthritis using Machine Learning Algorithms, IEEE Student Conference on Research and Development (SCOReD) :345-350, 2019. https://doi.org/10.1109/scored.2019.8896344

[40] Ho Sharon, Irraivan Elamvazuthi, Cheng-Kai Lu, S. Parasuraman and Elango Natarajan, Development of Rheumatoid Arthritis Classification From Electronic Image Sensor Using Ensemble Method, Sensors 2020, 20, 167. https://doi.org/10.3390/s20010167

[41] Fautrel, B. Guillemin, F. Meyer, O. Bant, M.D. et al., Choice of Second-Line Diases-Modifying Antirheumatic Drugs After Failure of Methotrexate Therapy for Rheumatoid Arthritis: A Decision Tree for Clinical Practice Based on Rheumatologists' Preferences. Arthritis Rheumatol. 61:425–434, 2009. https://doi.org/10.1002/art.24588

[42] Keyser, F.D. Choice of biologic therapy for patients with rheumatoid arthritis: The infection perspective. Curr. Rheumatol. Rev., 7: 77-87, 2011. https://doi.org/10.2174/157339711794474620

[43] Wu, C.-T.; Lo, C.-L.; Tung, C.-H.; Cheng, H.-L. Applying Data Mining Techniques for Predicting Prognosis in Patients with Rheumatoid Arthritis. Healthcare,8:85,2020. https://doi.org/10.3390/healthcare8020085

[44] Li Y, Sun X, Zhang X, Liu Y, Yang Y, Li R, Liu X, Jia R, Li Z. Establishment of a decision tree model for diagnosis of early rheumatoid arthritis by proteomic fingerprinting. Int J Rheum Dis. 18(8):835-41, 2015. PMID: 26249836. https://doi.org/10.1111/1756-185x.12595

[45] Ma Dan, Liang Nana, Zhang Liyun. Establishing Classification Tree Models in Rheumatoid Arthritis Using Combination of Matrix-Assisted Laser Desorption/Ionization Time-of-Flight Mass Spectrometry and Magnetic Beads. Frontiers in Medicine. 2021; 8:190. ISSN:2296-858X. https://doi.org/10.3389/fmed.2021.609773

[46] Hans Ulrich Scherer, Thomas Häupl, Gerd R. Burmester,The etiology of rheumatoid arthritis, Journal of Autoimmunity, Volume 110,2020,102400, ISSN 0896-8411, https://doi.org/10.1016/j.jaut.2019.102400

[47] Maria Hugle, Patrick Omoumi, Jacob M. van Laar, Joschka Boedecker and Thomas Hugle. Applied machine learning and artificial intelligence in

rheumatology.Rheumatology Advances in Practice 20;0:1–10,2020. https://doi.org/10.1093/rap/rkaa005

[48] Shanmugam, S., Preethi, J. Improved feature selection and classification for rheumatoid arthritis disease using weighted decision tree approach (REACT). J Supercomput 75, 5507–5519 (2019). https://doi.org/10.1007/s11227-019-02800-1

[49] Guo Q, Wang Y, Xu D, Nossent J, Pavlos NJ, Xu J. Rheumatoid arthritis: pathological mechanisms and modern pharmacologic therapies. Bone Res. 6:15, 2018. PMID: 29736302; PMCID: PMC5920070. https://doi.org/10.1038/s41413-018-0016-9

[50] Maria Kourilovitch, Claudio Galarza-Maldonado, Esteban Ortiz-Prado. Diagnosis and classification of rheumatoid arthritis. Journal of autoimmunity, 2014. https://doi.org/10.1016/j.jaut.2014.01.027

[51] Vodencarevic, A., Tascilar, K., Hartmann, F. et al. Advanced machine learning for predicting individual risk of flares in rheumatoid arthritis patients tapering biologic drugs, Arthritis Res Ther, 23, 67 (2021). https://doi.org/10.1186/s13075-021-02439-5

[52] Rehberg, M., Giegerich, C., Praestgaard, A. et al., Identification of a Rule to Predict Response to Sarilumab in Patients with Rheumatoid Arthritis Using Machine Learning and Clinical Trial Data, Rheumatol Ther 8, 1661–1675 (2021), https://doi.org/10.1007/s40744-021-00361-5

[53] Liu, J., Chen, N. A 9 mRNAs-based diagnostic signature for rheumatoid arthritis by integrating bioinformatic analysis and machine-learning, J Orthop Surg Res 16, 44 (2021). https://doi.org/10.1186/s13018-020-02180-w

[54] Huang Jie, Fu Xuekun, Chen Xinxin, Li Zheng, Huang Yuhong, Liang Chao, Promising Therapeutic Targets for Treatment of Rheumatoid Arthritis, Frontiers in Immunology 2021, Vol.12, ISSN:1664-3224, https://doi.org/10.3389/fimmu.2021.686155

[55] Maarseveen TD, Meinderink T, Reinders MJT, Knitza J, Huizinga TWJ, Kleyer A, Simon D, van den Akker EB, Knevel R, Machine Learning Electronic Health Record Identification of Patients with Rheumatoid Arthritis: Algorithm Pipeline Development and Validation Study, JMIR Med Inform, 8(11):2020, e23930, PMID: 33252349, PMCID: 7735897, https://doi.org/10.2196/23930

[56] Koo, B.S., Eun, S., Shin, K. et al. Machine learning model for identifying important clinical features for predicting remission in patients with rheumatoid arthritis treated with biologics. Arthritis Res Ther 23, 178, 2021. https://doi:10.1186/s13075-021-02567-y

[57] Fraenkel L, Bathon JM, England BR, et al. 2021 American College of Rheumatology Guideline for the Treatment of Rheumatoid Arthritis, Arthritis Rheumatol,73(7):1108-1123,2021. https://doi:10.1002/art.41752

[58] nptelhrd. (2008, October 16). Lecture – 35 Rule Induction and Decision Trees – I. Retrieved from https://www.youtube.com/watch?v=WfsRaLmh8js&t=547s.

[59] Tom M. Mitchell. Machine Learning McGraw-Hill Science/Engineering/Math. 1997; pp. 52-76.

[60] Wikipedia Website. [Online]. Available: https://en.wikipedia.org/wiki/Information_gain_ratio

[61] Seema Sharma, Jitendra Agrawal, Sanjeev Sharma, Classification Through Machine Learning Technique: C4.5 Algorithm based on Various Entropies, International Journal of Computer Applications, 82(16):0975-8887, 2013. https://doi.org/10.5120/14249-2444

[62] R. Sudrajat, I. Irianingsih, D. Krisnawan, Analysis of data mining classification by comparison of C4.5 and ID algorithms, IOP Conf. Series: Materials Science and Engineering 166, 2017, 012031, https://doi.org/10.1088/1757-899x/166/1/012031

[63] Wikipedia Website. [Online]. Available: https://en.wikipedia.org/wiki/ID3_algorithm.

[64] Poonkuzhali. S, Saravanakumar. C. Data Warehousing & Data Mining. Charulatha Publications 2008. 1st Ed. pp: 6.12.

Using Semi-Supervised Learning and Wikipedia to Train an Event Argument Extraction System

Patrik Zajec and Dunja Mladenić
E-mail: patrik.zajec@ijs.si, dunja.mladenic@ijs.si
Jožef Stefan Institute and Jožef Stefan International Postgraduate School

Student paper

Keywords: event extraction, event argument extraction, semi-supervised learning, probabilistic soft logic

Received: June 2, 2021

The paper presents a methodology for training an event argument extraction system in a semi-supervised setting. We use Wikipedia and Wikidata to automatically obtain a small noisily labeled dataset and a large unlabeled dataset. The dataset consists of event clusters containing Wikipedia pages in multiple languages. The unlabeled data is iteratively labeled using semi-supervised learning combined with probabilistic soft logic to infer the pseudo-label of each example from the predictions of multiple base learners. The proposed methodology is applied to Wikipedia pages about earthquakes and terrorist attacks in a cross-lingual setting. Our experiments show improvement of the results when using the proposed methodology. The system achieves F1-score of 0.79 *when only the automatically labeled dataset is used, and F1-score of* 0.84 *when trained according to the methodology with semi-supervised learning combined with probabilistic soft logic.*

Povzetek: V prispevku predstavimo metodologijo za polnadzorovano učenje sistema, katerega naloga je ekstrakcija ključnih atributov dogodkov. Kot vir podatkov uporabimo prosto enciklopedijo Wikipedija in bazo znanja Wikidata. Iz obeh virov na avtomatski način pridobimo manjši del označenih podatkov ter večji del neoznačenih podatkov, sestavljenih iz gruč dokumentov, ki poročajo o posameznih dogodkih v več jezikih. Neoznačene podatki iterativno označimo s polnadzorovanim učenjem v kombinaciji z verjetnostno mehko logiko, ki napovedi za vsak primer iz večih predikcijskih modelov združi v eno samo psevdo-labelo. Predlagano metodologijo uporabimo na dogodkih iz Wikipedije na temo potresov in terorističnih napadov. Eksperimenti pokažejo izboljšanje rezultatov sistema, ki doseže F1 $0,79$*, ko je naučen samo na avtomatsko označenih podatkih, ter* $0,84$*, ko je naučen, z uporabo predlagane metodologije, s polnadzorovanim učenjem v kombinaciji z verjetnostno mehko logiko.*

1 Introduction

The event extraction task is usually divided into two subtasks, namely event type detection and event argument extraction. Event type detection aims to determine the type or topic of the event, while event argument extraction aims to extract arguments for some predefined argument roles associated with the topic of the event. In this paper, we focus on event argument extraction and assume that the topic of the event is known in advance. For example, consider an excerpt from a news article reporting on an earthquake:

*"On **April 22, 2019**, a **6.1** magnitude earthquake struck the island of Luzon in the Philippines, leaving at least **18** dead, 3 missing and injuring at least **256** others."* [1]

we are interested in extracting the arguments for the following roles: the magnitude of the earthquake, the time and date when the earthquake occurred, the number of victims, and the number of injured. The goal is to extract an argument for each role, which in our case is mentioned in the text. The output is a set of *(role, argument)* pairs for the given text, for example *(magnitude, **6.1**)*.

We formulate argument extraction as a supervised classification task, where the classifier is used to assign arguments into predefined roles. We assume that each document reports on a single event, where the topic of the event is either earthquake or terrorist attack and is known in advance. Since there is no dataset that contains the annotations for the roles we are interested in, one must be constructed to train the classification model. Manually constructing a labeled dataset is a resource-intensive process, so we instead develop a methodology to automate the labeling.

We start by automatically constructing a small, noisily labeled set of examples for each role by matching structured knowledge from Wikidata with text from Wikipedia pages. Then we use semi-supervised learning [1] with multiple base learners to increase the size of the labeled set. In

[1] Source: en.wikipedia.org/wiki/2019_Luzon_earthquake

each iteration, the most confident predictions for the examples from the unlabeled set are used to increase the training set by assigning pseudo-labels. We introduce an additional component that combines the confidence of multiple base learners for each example.

The contribution of this paper is a novel methodology for automatic construction of a multi-lingual labeled dataset and training of a classification system for event argument extraction in multiple languages.

2 Related work

Wikipedia is a commonly used source for dataset construction [2, 3, 4]. Most datasets are constructed, at least in part, under the supervision of a human annotator, since for most tasks the labels cannot be reliably generated automatically [5, 6]. The initial annotations are usually obtained using distant supervision [7], which is the automatic approach of matching knowledge from the knowledge base with free text, and later refined by the human annotator, while we use semi-supervised learning instead. As in our case, the common choice is to use Wikidata [8] as the knowledge source and Wikipedia [2] as the text source.

Wikidata often does not reflect all of the knowledge available in Wikipedia pages, and a considerable amount of work has been done on the automatic population of both Wikipedia infoboxes and Wikidata [9, 10, 11, 12]. Such approaches develop the models capable of extracting the structured knowledge from free text. However, since the task is limited to Wikipedia, they mostly rely on the structure of Wikipedia pages to achieve good performance. This is especially true for the language-independent models [11] and makes them not directly applicable to other domains, such as newswire data.

Event extraction, which includes both event detection and argument extraction, is an active research topic that is formulated at several levels (document and sentence level) and approached using different techniques [13, 14, 15]. The approaches are most commonly evaluated on the ACE 2005 corpus [16] and the MUC-4 corpus [17]. Recently, RAMS [18] and WikiEvents [13] corpora were introduced, both being manually annotated and containing only English documents. Further, there is no information about the documents that mention the same event which could be used to form the event clusters.

The use of event clusters as a source of redundancy has been explored previously [19, 20]. Approaches on cross-lingual argument classification [21] are mostly focused on transfer learning, where the model is first trained on the source language and later applied on the target language. None of the approaches build the event clusters using the documents from multiple languages and incorporates cross-linguality directly into the semi-supervised training.

[2] https://www.wikipedia.org/

The currently best-performing approaches for event argument extraction are based on machine reading comprehension [22] or conditional generation [13] and therefore do not train a specific classifier for each argument. We find them less suitable for the semi-supervised learning part of our approach as they require a kind of event templates that are usually manually constructed. We envision that such models can instead be trained or fine-tuned on the automatically labeled dataset that our methodology generates.

Pseudo-labeling can be considered one of the most intuitive and simple forms of semi-supervised learning, while still achieving competitive performance if approached correctly [23]. We have chosen a simple and extensible technique to combine the predictions of multiple base learners into a single pseudo-label based on [24].

3 Methodology

3.1 Problem definition

We are given a collection of events $\mathcal{E}$ and a collection of topics $\mathcal{T}$, where the topic of each event $e \in \mathcal{E}$ is exactly one topic $t \in \mathcal{T}$. An event e has occurred at a particular time and its event cluster $\mathcal{D}_e$ consists of documents in multiple languages, with a single document per language, reporting on it.

For each topic $t \in \mathcal{T}$ there is a set of argument roles $\mathcal{R}_t$ that are known in advance. For example, the members of $\mathcal{R}_{earthquake}$ are *magnitude*, *number of injured*, *number of deaths*. For each document d from the event cluster $\mathcal{D}_e$, there is a set of arguments that were already extracted from the text. The task is to assign at most one argument role from $\mathcal{R}_t$ to each extracted argument.

Note that there are multiple documents in the event cluster, each with its own set of arguments. For event e, a single argument role might have zero, one, or multiple different arguments assigned. It is possible, for example, that the earthquake caused no casualties, so there is no argument assigned to the role of *magnitude*. The documents from the same event cluster might have different arguments for the same role, since, for example, the reported number of injured could come from different sources. In addition, there may also be multiple arguments for the same role in the single document, since the assumption that each document specifically reports only on a single event is not always true in practice. Such example is when one magnitude refers to an actual earthquake that the event is about, while the other magnitude refers to a stronger earthquake that hit the same region years ago.

Choosing a single argument for each argument role is challenging and sometimes even impossible. The task is instead to assign all feasible arguments from the event cluster to a particular role, even if not all are directly related to the event.

3.2 Approach

There is no labeled dataset that would follow the required structure and contain the labels for argument roles. We develop a methodology to automatically build such dataset and train a classification model which is used to pseudo-label the dataset and once trained can be used to perform classification on new data.

First, we select the Wikidata entities that are instances of topics from $\mathcal{T}$ and class *occurence*. Each Wikidata entity links to Wikipedia pages in multiple languages that form an event cluster. We align the argument roles $\mathcal{R}_t$ with Wikidata properties and try to automatically match the value for each property with the text from the pages. The matching is performed automatically either by using anchor links or literal text matching for numerical values. This gives us the arguments from text assigned to particular argument roles.

As most of the Wikidata entities lack some property values and automatic matching is frequently ambiguous, we obtain a small, nosily labeled set which we refer to as a *seed set*. We further use the named entities from the pages as arguments and assign the ones that are not a part of the *seed set* to an *unlabeled set*. Each such argument is considered as an unlabeled example.

To obtain label more arguments we use semi-supervised learning with multiple base learners and pseudo-labeling. Each of the base learners is trained on the set of labeled arguments from the topic (or multiple topics) and the language assigned to it. The prediction probabilities for each of the unlabeled arguments are determined by combining the probabilities of all base learners. This is done either by averaging or by feeding the probabilities as approximations of the true labels into the component, which attempts to derive the true value for each argument and the error rates for each learner [24]. The arguments with probabilities above or below the specified thresholds are given a pseudo-label and added to the training set.

The entire workflow is repeated in each iteration until no new arguments are selected for pseudo-labeling. The result is an automatically labeled dataset that includes the given topics and labels for selected argument roles and a classification system that assigns the argument role to each argument.

3.2.1 Representing the arguments

Following the related work [25], we introduce two new special tokens, <*e*> and </*e*>. The context of each argument is converted to a sequence of tokens with the additional tokens that mark the beginning and end of the argument span in the context. For example, argument 6.1 from the sentence *"A 6.1 magnitude earthquake."* is represented with its context as *"A <e>6.1</e>magnitude earthquake."*. Such representation is feed through the pretrained version of the XLM Roberta model [26]. We use the implementation from the Transformers library [27] and use the last hidden state of <*e*> token as a representation of the argument. The XLM Roberta model remains fixed during the learning as we have observed that the representation from pretrained model is expressive enough for our purposes and it significantly speeds up the iterations.

3.2.2 Using multiple topics

Instead of grouping all arguments assigned to the same roles across topics we keep the information about the topic of the argument's event. Firstly because the way of expressing an argument role might be slightly different in different topics, secondly as shown by the experiments, such separation enables us additional supervision between the topics, and finally as we can use the all arguments from one topic as negative examples for some particular role in the other topic.

For two topics t and t' there is potentially a set of common roles and a set of distinct roles, appearing in only one of the topics. For role r, which appears in both topics, the base learner trained on t' can be used to make predictions for examples from t. By combining predictions from learners trained on t and t', we could get better estimates of the true labels of the examples. For the role r', which is specific to the topic t, all examples from the topic t' can be used as negative examples. Selecting reliable negative examples from the same topic is not easy, as we may inadvertently mislabel some of the positive examples.

3.2.3 Using multiple languages

In a sense, articles from different languages provide different views on the same event. The important arguments should appear in all the articles, as they are highly relevant to the event. The arguments for the roles such as location and time should be consistent across all articles, whereas this does not necessarily apply to other roles such as the number of injured or the number of casualties. Matching such arguments across the articles is therefore not a trivial task, and although a variant of soft matching can be performed, we leave it for future work and limit our focus only on the values that can be matched unambiguously. We can combine the predictions of several language-specific base learners into a single pseudo-label for examples where arguments can be matched across the articles.

3.2.4 Base learners

Each iteration starts with a set of labeled arguments X_l, a set of unlabeled arguments X_u and a set of base learners trained on X_l. Base learners are simple logistic regression classifiers that use the vector representations of arguments.

Each base learner $\bar{f}^r_{t,l}$ is a binary classifier trained on the labeled data for the role r from the topic t and the language l. Such base learners are *topic-specific* as they are trained on a single topic t. Base learners $\bar{f}^r_l$ are trained on the labeled data for the role r from the language l and all the topics with the role r. Such base learners are *shared* across topics, as they consider the arguments from all the topics as a single training set. We use the classification probability

of the positive class instead of hard labels, $\bar{f}^r_{t,l}(x), \bar{f}^r_l(x) \in [0, 1]$.

For each argument x from a news article with the language l reporting on the event e from the topic t we obtain the following predictions:

- $\bar{f}^r_{t',l}(x)$ for each $r \in \mathcal{R}_t$ and all such t' that $r \in \mathcal{R}_{t'}$, that is the probability that x is a argument for the role r, where r is a role from the topic t, using the *topic-specific* base learner trained on examples from the same language on the topic t' that also has the role r,
- $\bar{f}^r_{t,l'}(x)$ which equals $\bar{f}^r_{t,l'}(y)$ for each $r \in \mathcal{R}_t$ and for each language l' such that there is an article reporting about the same event e in that language and contains an argument y which is matched to x,
- $\bar{f}^r_l(x)$ for each $r \in \mathcal{R}_t$, using the *shared* base learner on arguments from all topics t' that have the role r.

3.2.5 Combining the predictions

Multiple base learners make predictions for each argument. We combine such predictions to a probability distribution over argument roles as a weighted average.

The weight of each base learner $\bar{f}$ is determined by its error rate $e(\bar{f})$ which is estimated using both unlabeled and labeled examples. We introduce the following logical rules (referred to as *ensemble rules* in [24]) for each of the base learners $\bar{f}^r$ predicting for x:

$$\begin{aligned} \bar{f}^r(x) \wedge \neg e(\bar{f}^r) &\rightarrow f^r(x) \\ \bar{f}^r(x) \wedge e(\bar{f}^r) &\rightarrow \neg f^r(x) \\ \neg\bar{f}^r(x) \wedge \neg e(\bar{f}^r) &\rightarrow \neg f^r(x) \\ \neg\bar{f}^r(x) \wedge e(\bar{f}^r) &\rightarrow f^r(x) \end{aligned}$$

Here, the truth values are not limited to Boolean values, but instead represent the probability that the corresponding ground predicate or rule is true. For a detailed explanation of the method, we refer the reader to [24].

We introduce a prior belief that the predictions of base learners are correct via the following two rules:

$$\bar{f}^r(x) \rightarrow f^r(x), \, and \, , \neg\bar{f}^r(x) \rightarrow \neg f^r(x).$$

Since each example can be a value for at most one role, we introduce a *mutual exclusion* rule:

$$\bar{f}^r(x) \wedge f^{r'}(x) \rightarrow e(\bar{f}^r).$$

The rules are written in the syntax of a Probabilistic soft logic [28] program, where each rule is assigned a weight. We assign a weight of 1 to all *ensemble rules*, a weight of 0.1 to all *prior belief* rules and a weight of 1 to all *mutual exclusion* rules. The inference is performed using the PSL framework [3]. As we obtain the approximations for all $x \in X_u$, we extend the set of positive examples for each role r with all x such that $f^r(x) >= T_p$ and the set of negative examples with all x such that $f^r(x) <= T_n$, for predefined thresholds T_p and T_n.

[3] https://psl.linqs.org/

4 Experiments

4.1 Dataset

To evaluate the proposed methodology, we have conducted experiments on two topics: *earthquakes* and *terrorist attacks*.

We have collected Wikipedia articles and Wikidata information of 913 earthquakes from 2000 to 2020 in 6 different languages, namely English, Spanish, German, French, Italian, and Dutch. We have manually annotated the arguments of 85 English articles using the argument roles *number of deaths*, *number of injured* and *magnitude*, which serve as a labeled test set and are not included in the training process. In addition, we have collected data from 315 terrorist attacks from 2000 to 2020 and articles from the same 6 languages.

4.2 Evaluation settings

The evaluation for each approach is performed on a labeled English dataset, where 76 arguments are labeled as number of deaths, 45 as number of injured and 125 as magnitude. The threshold values for the pseudo-labeling are set to $T_p = 0.6$ and $T_n = 0.05$. The approaches differ by the subset of base learners used to form the combined prediction and by the weighting of the predictions.

Single or multiple languages In a single language setting, only English articles are used to label the arguments and train the base learners. In a multi-language setting, all available articles are used and the arguments are matched across the articles from the same event.

Single or multiple topics In a single topic setting, only the arguments from the *earthquake* topic are used. In a multi-topic setting, the arguments from *terrorist attacks* are used as negative examples for *magnitude*, the base learners for the roles *number of deaths* and *number of injured* are combined as described in the section 3.2.4.

Uniform or estimated weights In the uniform setting all predictions of the base learners contribute equally, while in the estimated setting the weights of the base learners are estimated using the approach described in section 3.2.5.

4.3 Results

The results of all the experiments are summarized in Table 1. Since the test set is limited to the topic *earthquake* and English, only a subset of base learners was used to make the final predictions. We report the average value

Table 1: Results of all experiments. The column *Single iteration* reports the results of approaches where base learners were trained on the seed set only. Results, where base learners were trained in the semi-supervised setting with different weightings of the predictions, are reported in the columns *Uniform weights* and *Estimated weights*. The values of precision, recall, and F1 are averaged over all argument roles.

	Single iteration			Uniform weights			Estimated weights		
Model	P	R	F1	P	R	F1	P	R	F1
Single language, single topic	0.94	0.64	0.76	0.83	0.75	0.77	0.84	0.76	0.79
Multiple languages, single topic	0.94	0.64	0.76	0.82	0.74	0.76	0.83	0.75	0.77
Single language, multiple topics	0.91	0.76	0.83	0.83	0.83	0.83	0.86	0.83	0.84
Multiple languages, multiple topics	0.93	0.76	0.83	0.82	0.83	0.82	**0.84**	**0.84**	**0.84**

of precision, recall, and F1 across all argument roles. The probability threshold of 0.5 was used to determine the classification label.

Single iteration Approaches in which base learners are trained on the initial seed set for a single iteration achieve higher precision (0.94 compared to 0.84 achieved by *estimated weights* in *single language, single topic* setting) with the cost of a lower recall (0.64 compared to 0.76). We observe that they distinguish almost perfectly between the argument roles from the seed set and produce almost no false positives. Using one or more languages has almost no effect on the averaged scores when the number of topics is fixed. When using multiple topics, a higher recall is achieved without a significant decrease in precision. All incorrect classifications of the role *number on injured* are actually examples of the *number of missing* role that is not included in our set and likewise almost all incorrect classifications for the role *magnitude* are examples of the role *intensity on the Mercalli scale*. This could easily be solved by expanding the set of roles and shows how important it is to learn to classify multiple roles simultaneously.

Uniform and estimated weights Semi-supervised approaches in which base learners are trained iteratively trade precision in order to significantly improve recall (0.64 compared to 0.76 achieved by *estimated weights* in *single language, single topic* setting). Most of the loss of precision is due to misclassification between roles *number of deaths* and *number of injured*, similar to the example *"370 people were killed by the earthquake and related building collapses, including **228** in Mexico City, and more than 6,000 were injured."* where 228 was incorrectly classified as the number of injured and not the number of deaths. The use of multiple topics reduces misclassification between these roles and further improves recall as new contexts are discovered by the base learners trained on *terrorist attacks*.

Using the estimated error rates as weights for the predictions of base learners shows a slight improvement in performance. It may be beneficial to estimate multiple error rates for *topic-specific* base learners, as they tend to be more reliable in labeling arguments from the same topic. We believe more data and experiments are needed to properly evaluate this component. A major advantage is its flexibility, as we can easily incorporate prior knowledge about the roles or additional constraints on the predictions in the form of logical rules.

5 Conclusion

The proposed method avoids the need to manually annotate the data for event argument extraction and instead combines Wikipedia and Wikidata to obtain labeled data. Compared to the related work, the proposed methodology uses semi-supervised learning and integrates cross-lingual data into the learning process to enhance the pseudo-labeling supported by probabilistic soft logic. The resulting classification models, used for automatic labeling, can be readily used to extract the event arguments in new texts. It is also possible to train or fine-tune a stronger state-of-the-art model on the resulting dataset, extending it to new event arguments and languages beyond those included in the original training datasets. The experiments were performed on a relatively small dataset and show that the proposed direction seems promising. However, the more suitable test of our approach would be to apply it to a much larger number of topics and events, which we will do in the next step. Moreover, the current approach needs to be evaluated in more detail.

Acknowledgement

This work was supported by the Slovenian Research Agency under the project J2-1736 Causalify and cofinanced by the Republic of Slovenia and the European Union's H2020 research and innovation program under NAIADES EU project grant agreement H2020-SC5-820985.

References

[1] Jesper E. van Engelen and H. Hoos. A survey on semi-supervised learning. *Machine Learning*, 109:373–440, 2019. https://doi.org/10.1007/s10994-019-05855-6.

[2] Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James

Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online, June 2021. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.200.

[3] Daniel Hewlett, Alexandre Lacoste, Llion Jones, Illia Polosukhin, Andrew Fandrianto, Jay Han, Matthew Kelcey, and David Berthelot. WikiReading: A novel large-scale language understanding task over Wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1545, Berlin, Germany, August 2016. Association for Computational Linguistics. https://doi.org/10.18653/v1/P16-1145.

[4] Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 166–175, New York, NY, USA, 2019. Association for Computing Machinery. https://doi.org/10.1145/3292500.3330955.

[5] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada, August 2017. Association for Computational Linguistics. https://doi.org/10.18653/v1/K17-1034.

[6] Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. FewRel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-1514.

[7] Alessio Aprosio, Claudio Giuliano, and Alberto Lavelli. Extending the coverage of dbpedia properties using distant supervision over wikipedia. *CEUR Workshop Proceedings*, 1064, 01 2013. https://dl.acm.org/doi/10.5555/2874479.2874482.

[8] Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September 2014. https://doi.org/10.1145/2629489.

[9] Dustin Lange, Christoph Böhm, and Felix Naumann. Extracting structured information from wikipedia articles to populate infoboxes. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 1661–1664, New York, NY, USA, 2010. Association for Computing Machinery. https://doi.org/10.1145/1871437.1871698.

[10] Florian Schrage, Nicolas Heist, and Heiko Paulheim. Extracting literal assertions for dbpedia from wikipedia abstracts. In *SEMANTiCS*, pages 288–294, 11 2019. https://doi.org/10.1007/978-3-030-33220-4_21.

[11] Nicolas Heist, Sven Hertling, and Heiko Paulheim. Language-agnostic relation extraction from abstracts in wikis. *Information*, 9(4):75, 2018. https://doi.org/10.3390/info9040075.

[12] Boya Peng, Yejin Huh, Xiao Ling, and Michele Banko. Improving knowledge base construction from robust infobox extraction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*, pages 138–148, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-2018.

[13] Sha Li, Heng Ji, and Jiawei Han. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 894–908, Online, June 2021. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.69.

[14] Fayuan Li, Weihua Peng, Yuguang Chen, Quan Wang, Lu Pan, Yajuan Lyu, and Yong Zhu. Event extraction as multi-turn question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 829–838, Online, November 2020. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.73.

[15] Manling Li, Alireza Zareian, Ying Lin, Xiaoman Pan, Spencer Whitehead, Brian Chen, Bo Wu, Heng Ji, Shih-Fu Chang, Clare Voss, Daniel Napierski, and Marjorie Freedman. GAIA: A fine-grained multimedia knowledge extraction system. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 77–86, Online, July 2020. Association

for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-demos.11.

[16] George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, May 2004. European Language Resources Association (ELRA).

[17] Beth M. Sundheim. Overview of the fourth message understanding evaluation and conference. In *Proceedings of the 4th conference on Message understanding - MUC4 '92*. Association for Computational Linguistics, 1992. https://doi.org/10.3115/1072064.1072066.

[18] Seth Ebner, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. Multi-sentence argument linking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8057–8077, Online, July 2020. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.718.

[19] James Ferguson, Colin Lockard, Daniel Weld, and Hannaneh Hajishirzi. Semi-supervised event extraction with paraphrase clusters. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 359–364, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-2058.

[20] Xiao Liu, Heyan Huang, and Yue Zhang. Open domain event extraction using neural latent variable models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2860–2871, Florence, Italy, July 2019. Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1276.

[21] Ananya Subburathinam, Di Lu, Heng Ji, Jonathan May, Shih-Fu Chang, Avirup Sil, and Clare Voss. Cross-lingual structure transfer for relation and event extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 313–325, Hong Kong, China, November 2019. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1030.

[22] Jian Liu, Yufeng Chen, and Jinan Xu. Machine reading comprehension as data augmentation: A case study on implicit event argument extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2716–2725, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.214.

[23] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning, 2021.

[24] Emmanouil A. Platanios, Hoifung Poon, Tom M. Mitchell, and Eric Horvitz. Estimating accuracy from unlabeled data: A probabilistic logic approach. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4364–4373, Red Hook, NY, USA, 2017. Curran Associates Inc. https://dl.acm.org/doi/10.5555/3294996.3295190.

[25] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy, July 2019. Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1279.

[26] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.747.

[27] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-demos.6.

[28] Stephen H. Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. Hinge-loss markov random fields and probabilistic soft logic. *J. Mach. Learn. Res.*,

18(1):3846–3912, January 2017. https://dl.acm.org/doi/10.5555/3122009.3176853.

JOŽEF STEFAN INSTITUTE

Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovene parents, he obtained his Ph.D. at Vienna University, where he was later Director of the Physics Institute, Vice-President of the Vienna Academy of Sciences and a member of several scientific institutions in Europe. Stefan explored many areas in hydrodynamics, optics, acoustics, electricity, magnetism and the kinetic theory of gases. Among other things, he originated the law that the total radiation from a black body is proportional to the 4th power of its absolute temperature, known as the Stefan–Boltzmann law.

The Jožef Stefan Institute (JSI) is the leading independent scientific research institution in Slovenia, covering a broad spectrum of fundamental and applied research in the fields of physics, chemistry and biochemistry, electronics and information science, nuclear science technology, energy research and environmental science.

The Jožef Stefan Institute (JSI) is a research organisation for pure and applied research in the natural sciences and technology. Both are closely interconnected in research departments composed of different task teams. Emphasis in basic research is given to the development and education of young scientists, while applied research and development serve for the transfer of advanced knowledge, contributing to the development of the national economy and society in general.

At present the Institute, with a total of about 900 staff, has 700 researchers, about 250 of whom are postgraduates, around 500 of whom have doctorates (Ph.D.), and around 200 of whom have permanent professorships or temporary teaching assignments at the Universities.

In view of its activities and status, the JSI plays the role of a national institute, complementing the role of the universities and bridging the gap between basic science and applications.

Research at the JSI includes the following major fields: physics; chemistry; electronics, informatics and computer sciences; biochemistry; ecology; reactor technology; applied mathematics. Most of the activities are more or less closely connected to information sciences, in particular computer sciences, artificial intelligence, language and speech technologies, computer-aided design, computer architectures, biocybernetics and robotics, computer automation and control, professional electronics, digital communications and networks, and applied mathematics.

The Institute is located in Ljubljana, the capital of the independent state of S**love**nia (or S♡nia). The capital today is considered a crossroad between East, West and Mediterranean Europe, offering excellent productive capabilities and solid business opportunities, with strong international connections. Ljubljana is connected to important centers such as Prague, Budapest, Vienna, Zagreb, Milan, Rome, Monaco, Nice, Bern and Munich, all within a radius of 600 km.

From the Jožef Stefan Institute, the Technology park "Ljubljana" has been proposed as part of the national strategy for technological development to foster synergies between research and industry, to promote joint ventures between university bodies, research institutes and innovative industry, to act as an incubator for high-tech initiatives and to accelerate the development cycle of innovative products.

Part of the Institute was reorganized into several high-tech units supported by and connected within the Technology park at the Jožef Stefan Institute, established as the beginning of a regional Technology park "Ljubljana". The project was developed at a particularly historical moment, characterized by the process of state reorganisation, privatisation and private initiative. The national Technology Park is a shareholding company hosting an independent venture-capital institution.

The promoters and operational entities of the project are the Republic of Slovenia, Ministry of Higher Education, Science and Technology and the Jožef Stefan Institute. The framework of the operation also includes the University of Ljubljana, the National Institute of Chemistry, the Institute for Electronics and Vacuum Technology and the Institute for Materials and Construction Research among others. In addition, the project is supported by the Ministry of the Economy, the National Chamber of Economy and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
Tel.:+386 1 4773 900, Fax.:+386 1 251 93 85
WWW: http://www.ijs.si
E-mail: matjaz.gams@ijs.si
Public relations: Polona Strnad

INFORMATICA

AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

INVITATION, COOPERATION

Submissions and Refereeing

Please register as an author and submit a manuscript at: http://www.informatica.si. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible from typing errors to global philosophical disagreements. The chosen editor will send the author the obtained reviews. If the paper is accepted, the editor will also send an email to the managing editor. The executive board will inform the author that the paper has been accepted, and the author will send the paper to the managing editor. The paper will be published within one year of receipt of email with the text in Informatica MS Word format or Informatica LaTeX format and figures in .eps format. Style and examples of papers can be obtained from http://www.informatica.si. Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the managing editor.

SUBSCRIPTION

Please, complete the order form and send it to Dr. Drago Torkar, Informatica, Institut Jožef Stefan, Jamova 39, 1000 Ljubljana, Slovenia. E-mail: drago.torkar@ijs.si

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than twentyeight years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica web edition is free of charge and accessible at http://www.informatica.si.
Informatica print edition is free of charge for major scientific, educational and governmental institutions. Others should subscribe.

Informatica

An International Journal of Computing and Informatics

Web edition of Informatica may be accessed at: http://www.informatica.si.

Subscription Information Informatica (ISSN 0350-5596) is published four times a year in Spring, Summer, Autumn, and Winter (4 issues per year) by the Slovene Society Informatika, Litostrojska cesta 54, 1000 Ljubljana, Slovenia.
The subscription rate for 2022 (Volume 46) is
– 60 EUR for institutions,
– 30 EUR for individuals, and
– 15 EUR for students
Claims for missing issues will be honored free of charge within six months after the publication date of the issue.

Typesetting: Borut, Peter and Jaša Žnidar; borut.znidar@gmail.com.
Printing: ABO grafika d.o.o., Ob železnici 16, 1000 Ljubljana.

Orders may be placed by email (drago.torkar@ijs.si), telephone (+386 1 477 3900) or fax (+386 1 251 93 85). The payment should be made to our bank account no.: 02083-0013014662 at NLB d.d., 1520 Ljubljana, Trg republike 2, Slovenija, IBAN no.: SI56020830013014662, SWIFT Code: LJBASI2X.

Informatica is published by Slovene Society Informatika (president Niko Schlamberger) in cooperation with the following societies (and contact persons):
Slovene Society for Pattern Recognition (Vitomir Štruc)
Slovenian Artificial Intelligence Society (Sašo Džeroski)
Cognitive Science Society (Olga Markič)
Slovenian Society of Mathematicians, Physicists and Astronomers (Dragan Mihailović)
Automatic Control Society of Slovenia (Giovanni Godena)
Slovenian Association of Technical and Natural Sciences / Engineering Academy of Slovenia (Mark Pleško)
ACM Slovenia (Nikolaj Zimic)

Informatica is financially supported by the Slovenian research agency from the Call for co-financing of scientific periodical publications.

Informatica is surveyed by: ACM Digital Library, Citeseer, COBISS, Compendex, Computer & Information Systems Abstracts, Computer Database, Computer Science Index, Current Mathematical Publications, DBLP Computer Science Bibliography, Directory of Open Access Journals, InfoTrac OneFile, Inspec, Linguistic and Language Behaviour Abstracts, Mathematical Reviews, MatSciNet, MatSci on SilverPlatter, Scopus, Zentralblatt Math

Volume 46 Number 1 March 2022 **ISSN 0350-5596**

Informatica

An International Journal of Computing and Informatics

Informatica **46** (2022) Number 1, pp. 1–129