Enhance Medical Sentiment Vectors through Document Embedding using Recurrent Neural Network

Rami N. M. Yousef[1], Sabrina Tiun[2], Nazlia Omar[3], Eissa M. Alshari[4]
Palestine Technical University (PTUK), Tulkarm, Palestine[1]
Center for Artificial Intelligence Technology (CAIT), Faculty of Information Science and Technology[2, 3]
Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor, Malaysia[2, 3]
Ibb University, Ibb, Yemen[4]

***Abstract*—Adverse Drug Reaction (ADR) extraction is the process of identifying drug implications mentioned in social posts. Handling medical text for the identification of ADR is vital to research in terms of configuring the side effect and other medical-related entities within any medical text. However, investigating the role of such effect in the context of positive and negative is the responsibility of sentiment classification task where every medical review document would be categorized into its polarity, this is known as Medical Sentiment Analysis (MSA). Several studies have presented various techniques for MSA. Most of the recent studies have concentrated on architectures such as the Convolutional Neural Network (CNN) to get the document embedding. Yet, such architecture focuses only on the input without considering the previous or latter input. This might lead to weaker embedding for the document where some terms would not be considered. Hence, this paper proposes a new document embedding approach based on the Recurrent Neural Network (RNN) to improve the sentiment classification. Using a benchmark dataset of medical sentiments, the proposed method showed greater performance of sentiment classification accuracy. Such finding proves the effectiveness of RNN in producing document embedding.**

Keywords*—Adverse drug reaction; medical sentiment analysis; recurrent neural network; support vector machine; logistic regression

I. Introduction

The exponential growth of medical-related text within social networks nowadays has posed a significant need for text mining [1]. The analysis of such medical text has been divided into two main tasks. The first task is relatively similar to the Named Entity Recognition (NER) where the medical-related concepts are being identified [2-4]. In particular, it concentrates on specific medical entity which is the drug implications or side-effects, this task is known as Adverse Drug Reaction (ADR) extraction [5-8]. While the second task belongs to the field of sentiment analysis in which the medical reviews are being processed to identify its polarity (whether positive or negative), this task is known as medical sentiment analysis [9]. The first task is important for identifying the types of side effects and drug-related entities. Whereas, the second task is important to clarify the impact of such a side effect.

For the task of medical sentiment analysis, the literature has shown greater interest in using the word embedding technique [10-12]. But since this task is based on the document analysis where the whole document is supposed to be classified into positive or negative. Therefore, the word embedding showed specific shortcoming which is represented by generating a holistic embedding for the whole document. Word embedding is supposed to produce a continuous vector for each term and since each document has a different number of words thus, it is necessary to articulate a holistic embedding for the document. For this purpose, Ghassemi et al. [13] and Chen & Sokolova [14] have proposed an averaging embedding based on either summation or Principle Component Analysis (PCA). In this regard, every term's embedding inside a given document will be summed or averaged. However, the averaging mechanism suffers from individual consideration of each word's embedding in which the single term would have a constant embedding for whatever document is being examined. Giving the same word, but with a different context, an identical embedding would not be an effective embedding especially if the word has been incorporated with other terms to form a generic and holistic embedding for a particular document.

For this purpose, Yadav et al. [12], Mahata et al. [15], and Liu & Lee [16] utilized CNN for more accurate document embedding. CNN can process both terms' vectors along with the document's vectors to produced sophisticated embedding. Yet, according to Yin et al. [17] CNN focuses only on the input without considering the previous or latter input. This might lead to weaker embedding for the document where some terms would not be considered. In this regard, there is a demand for proposing a method for document embedding that can consider the series of documents. Since most of the state of the art are relying on CNN architecture thus, they might suffer from weak document embedding which had led to weak classification accuracy of sentiment categorization.

Therefore, this study aims to utilize the Recurrent Neural Network (RNN) for doing the document embedding. RNN has a unique characteristic represented by an additional layer that considered to be a memory that saves information about previous input [17]. This might improve the embedding for the document. A benchmark dataset of medical sentiments has been used. Besides, four machine learning classifiers have been

trained on the proposed RNN document embedding including Multi-layer Perceptron (MLP), Support Vector Machine, Logistic Regression, Radial Basis Function Neural Network (RBFNN).

The structure of the paper is organized as: Section II highlights the latest studies that have addressed MSA, Section III illustrates in detail the explanation of the proposed method, and Section IV shows the experimental results of the proposed method.

II. Related Work

The state of the art in medical sentiment analysis relies on word embedding techniques where various neural network architectures are being utilized for producing distinct vector for each term, sentence, or even document. Ghassemi et al. [13] have proposed a document embedding approach for clinical sentiment analysis. The authors have utilized the conventional word embedding technique for providing the vectors for each word. Consequentially, the authors have proposed the Principle Component Analysis (PCA) to give an averaged embedding for the document.

Chen & Sokolova [14] have discussed the impact of the word2vec model in terms of extracting medical-related entities within medical reviews. In particular, tasks such as ADR extraction enables the use of conventional word embedding yet, further tasks such as sentiment classification requires a holistic embedding for a whole document. Therefore, the authors have examined the averaging schemes of embedding where the words' embedding inside a document is being summed or averaged.

Yadav et al. [12] have proposed a document embedding based on CNN for medical sentiment analysis. For accommodating the experiments, the authors have scraped the medical web reviews from multiple sites. Consequentially, CNN has been trained on such reviews to make document embedding along with the classification tasks of categorizing the reviews into positive and negative polarities.

Similarly, Mahata et al. [15] have proposed a document embedding approach based on CNN for medical sentiment analysis. Unlike the previous study, this study has utilized medical reviews from Twitter where CNN has been trained on such reviews to generate document embedding. Finally, CNN also has been examined in terms of classifying the reviews into positive and negative polarities.

Liu & Lee [16] have examined two types of classification methods for medical sentiment analysis. First, the authors have used the benchmark dataset of medical sentiment reviews introduced by [18]. Then, the authors have utilized CNN for generating the document embedding. After that, the CNN has been examined in terms of the classification along with other classifiers such as SVM, LR, Multi-layer Perceptron (MLP), and Radial Basis Function Neural Network (RBFNN).

From the literature review, one could conclude that document embedding has been examined through either an average-based word embedding or via CNN. The former method suffers from weak embedding due to the identical embedding of a single term regardless of its position in different documents. Whereas, the latter method suffers from the inconsideration of input sequencing where current document input is being treated regardless of its preceding document.

III. Proposed Method

The general framework of the proposed RNN document embedding contains three main phases. The first phase is intended to prepare the dataset for the experiments, as well as, performing preprocessing tasks such as sentence splitting, tokenization, and stopwords removal.

The second phase represents the core contribution of this paper in which the proposed RNN document embedding model is being initiated. For this purpose, the dataset will be divided into positive and negative documents. Then, the one-hot encoding will be initiated for both the words and their documents. The one-hot vectors then will be fed into an RNN architecture to produce the embedding for each document.

The third phase is intended to perform the classification for each document's embedding produced by the previous phase to give it a class whether positive or negative. For this purpose, two categories of classification algorithms are being used where the first category includes traditional classifiers such as SVM, LR, MLP, and RBFNN. While the second category contains the RNN itself for accommodating the classification. The reason behind selecting these classifiers is to be consistent with the baseline study of Liu & Lee [16] which has examined CNN as document embedding and classification along with the first category of classifiers. Fig. 1 describes the general framework of the proposed method along with its phases. The following sections will tackle each phase independently.

A. Preprocessing

First of all, since this paper examines the task of medical sentiment analysis, it is necessary to consider a suitable dataset for this purpose. Fortunately, the dataset of [18] was designed for both ADR extraction and sentiment classification tasks. This can be represented by the existence of both classes of ADR and polarity. In other words, every sentence of the medical review within such a dataset has been associated with a class label that refers to the existence or absence of ADR, as well as, a class label that refers to the polarity of the sentence whether positive or negative. Table I shows a sample from the dataset.

After describing the dataset, it is necessary to highlight the preprocessing tasks performed on such data. First, sentence splitting has been used to split the text into a series of sentences by identifying sentence boundaries. Besides, tokenization has been used to split the text stream into a series of tokens. Lastly, stopwords removal has been conducted to get rid of insignificant words.

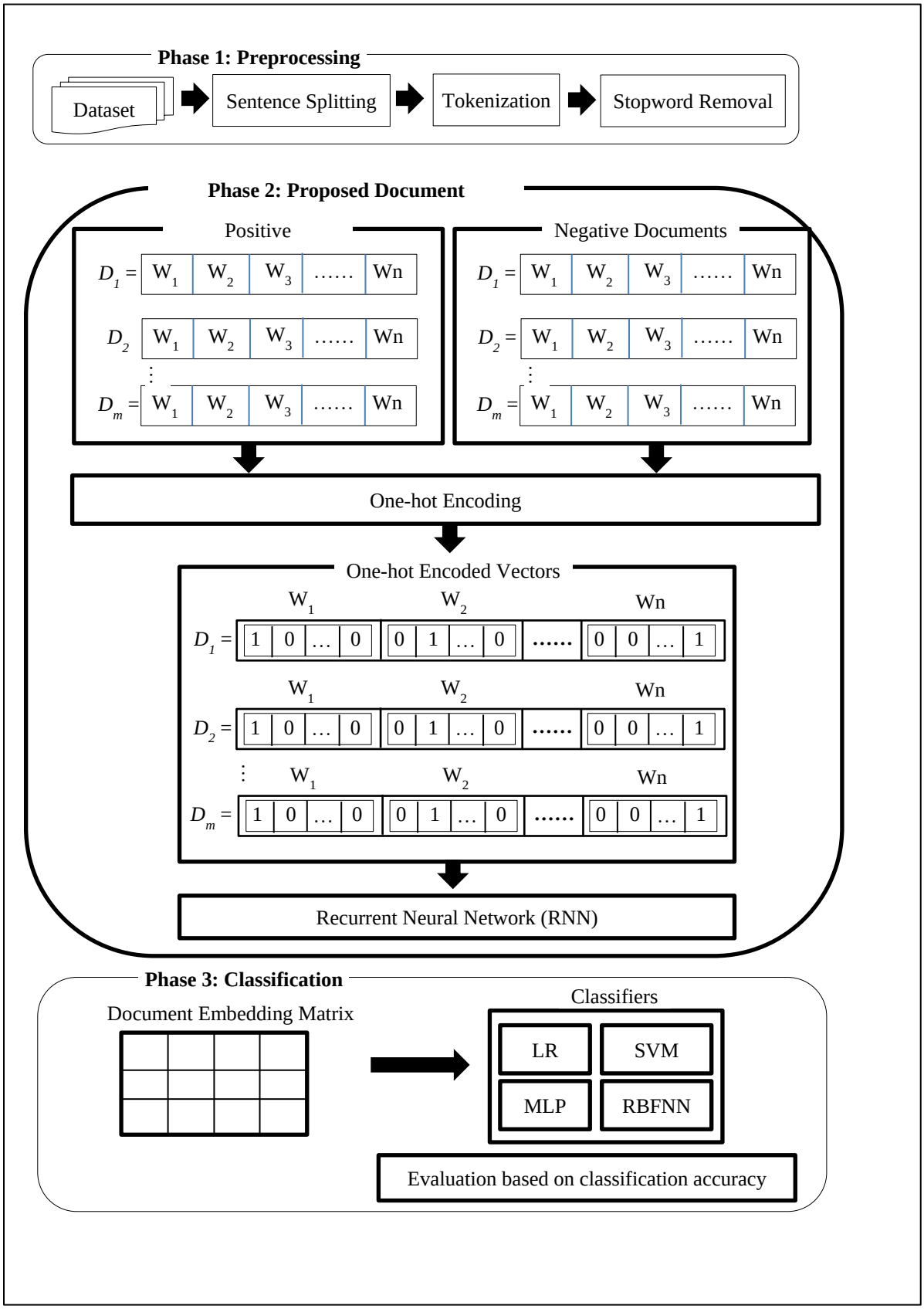

Fig. 1. The General Framework of the Proposed Method.

TABLE I. SAMPLE OF THE DATASET

Sentence	ADR Class	Sentiment Class
I slowly started to experience dizzy spells while taking this medicine for 3 weeks.	1	N
I hope others can benefit from this drug because it really does work!	0	P
this stuff is fantastic never breathed so good	1	P
It worked for me. I felt relief immediately and had no side effects.	0	P
I'm using it intermittently on high pollen days (or in anticipation of them) Works great!	0	P

B. Positive and Negative Document Division

Usually, most of the studies in document embedding for the task of sentiment analysis are examining both positive and negative reviews randomly which leads to an embedding that might mix the contexts of different words. Therefore, this study proposes a division in which the positive documents and negative documents are being treated separately within the training. This would make the embedding focus on similar terms in different contexts but within a unified polarity. In this regard, the embedding will be oriented toward positive or negative contexts. Table II shows a sample of positive and negative reviews.

As shown in Table II, the word 'pain' has been mentioned in both positive and negative reviews. In this regard, the random blend among positive and negative would not be effective in which the word 'pain' would have the same embedding and thus, misleading the classification.

C. One-hot Encoding

The one-hot encoding is one of the primary steps for getting an embedding for any word. Let consider two small documents from the negative reviews in Table II for simplicity. Table III depicts these documents.

Similar to the application of one-hot encoding will focus on the unique terms within the two documents in Table III excluding the stopwords as in Table IV.

Additional to the term-to-term one-hot encoding, the document embedding requires examining the document-to-term one-hot encoding as in Table V.

Now both the term-to-term and document-to-term one-hot encoding will be processed via the RNN architecture to produce the document embedding.

D. Recurrent Neural Network (RNN)

Similar to any neural network architecture, RNN contains three main layers including the input, hidden, and output layers. However, the architecture of RNN has a unique layer known as the context layer or memory layer [19, 20]. Such a layer saves information about the current input in which the next input can use such information.

Now, to get the embedding of the last documents, the one-hot vectors of the terms and the one-hot vector of their document will be processed via the RNN as shown in Fig. 2.

As shown in Fig. 2, the first document (i.e. D1) has been processed via the RNN. This can be represented by bringing the vectors of terms inside D1 from Table IV alongside the vector of D1 from Table V.

Similar to word2Vec, the context words will be considered as input where the aim is to predict the target word. But additionally, the RNN architecture has considered also the document vector along with the words' vectors. On the other hand, RNN contains a context or memory layer which saves the information of the current document.

The training of RNN will be conducted in which the weights are being initiated and multiplied by the neurons to get the output terms' vectors. It is obvious that the beginning epochs would show differences between the target output and the predicted output. Therefore, the Backpropagation has been used to change the weighted to reduce the error rate. Once the error is reduced to zero where the target output would correspond to the predicted output, the weights between the document vector and the hidden layer will be considered as the document embedding as shown in Fig. 3.

TABLE II. SAMPLE OF REVIEW DIVISION BETWEEN POSITIVE AND NEGATIVE

Positive Reviews	Negative Reviews
I have been having great test results from my CT scans, bone scan, etc.	The post-surgical pain was excruciating
to this point so I don't have any comparisons to rate against.	I already have joint pain
For the treatment of acute pain due to kidney, I would say that Demerol is effective	I can't find proper pain medication without having allergic reactions

TABLE III. SAMPLE OF NEGATIVE DOCUMENTS

D_1=	*"The post-surgical pain was excruciating"*
D_2=	*"I already have joint pain"*

TABLE IV. ONE-HOT ENCODING FOR THE TERMS

	post	surgical	pain	excruciating	joint
post	1	0	0	0	0
surgical	0	1	0	0	0
pain	0	0	1	0	0
excruciating	0	0	0	1	0
joint	0	0	0	0	1

TABLE V. ONE-HOT ENCODING FOR THE DOCUMENTS

	post	surgical	pain	excruciating	joint
D_1	1	1	1	1	0
D_2	0	0	1	0	1

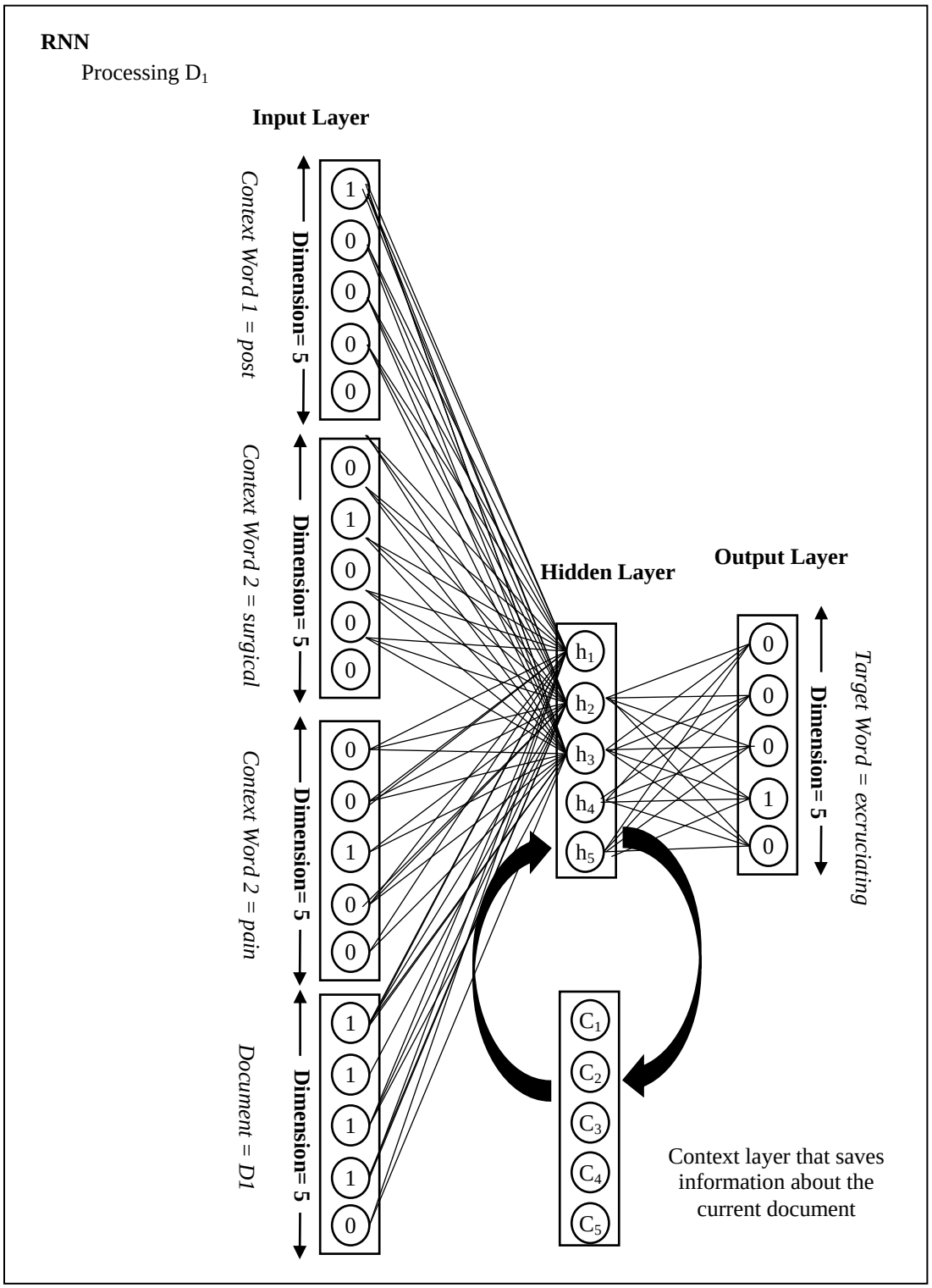

Fig. 2. Processing D1 via RNN.

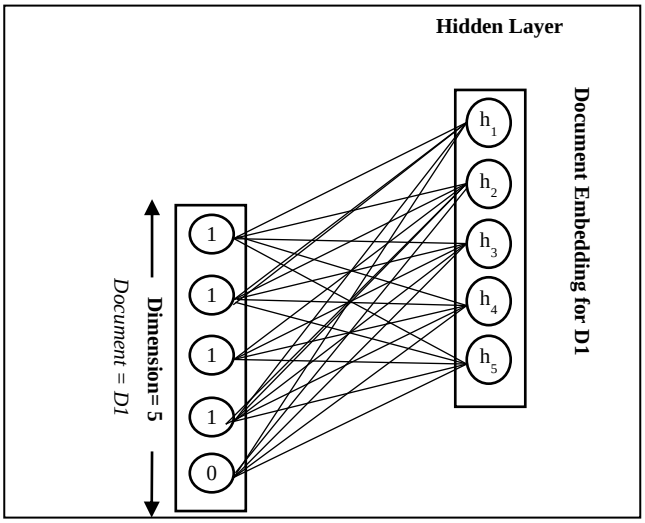

Fig. 3. Getting Document Embedding.

This process will be applied for D2 and other documents.

E. Classification

After acquiring the document embedding produced by the proposed RNN document embedding model, multiple classifiers will be used to classify the reviews document matrices into positive or negative. For this reason, four classifiers will be used including SVM, LR, MLP, and RBFNN. The reason behind using such classifiers lies in the consistency required to compare the results with the baseline study of Liu and Lee [16] where the same classifiers have been used to classify document embedding produced by CNN. Therefore, the core of the comparison will target both the document embedding produced by CNN and the proposed RNN.

To evaluate the classification accuracy, three common information retrieval metrics of Precision, Recall and F-measure will be considered. Precision is the ratio between corrected classified documents and the total number of documents, it can be computed as in the following equation:

$$\text{Precision} = TP / TP + FP \quad (1)$$

where TP is the number of correctly classified documents, and FP is the number of incorrectly classified documents. On the other hand, Recall is the ratio between correctly classified documents and total number of both classes (i.e. positive and negative), it can be computed as in the following equation:

$$\text{Recall} = TP / TP + TN \quad (2)$$

where TN is total number of either positive or negative classes. Finally, the harmony between Precision and Recall can be acquired through F-measure which can be computed as follow:

$$\text{F-measure} = 2 \times \text{Precision} \times \text{Recall} / \text{Precision} + \text{Recall} \quad (3)$$

IV. Experimental Results

In this section the results of applying traditional classifiers including SVM, LR, MLP, and RBFNN are being depicted. The evaluation has been done using precision, recall, and f-score. Note that, the results of classifiers will be depicted as based on the baseline study of document embedding using CNN against the proposed RNN. Table VI shows these results.

As shown in Table VI, the results of precision, recall, and f-score for all classifiers using the proposed RNN document embedding have outperformed the ones depicted by the baseline CNN document embedding. This can be represented where SVM has been improved from an f-score of 0.55 (using CNN) into 0.89 (using RNN). While the f-score of LR has raised from 0.54 (using CNN) into 0.90 (using RNN). As well as, for the MLP, the f-score was 0.58 using CNN and has been improved into 0.89 based on the proposed RNN. Finally, the RBFNN classifier showed an f-score of 0.58 when using CNN, while got a 0.87 when the proposed RNN document embedding has been used. All these results demonstrate the outperformance of RNN over CNN in terms of document embedding. The reason behind such superiority lies in the ability of RNN to consider the series of text, document, terms where the weights of current input can be shared with the next input. This has facilitated toward producing more sophisticated and accurate embedding for the document.

On the other hand, there is another observation that can be concluded from the results. Such observation lies in the best results among the classifiers themselves. In CNN, both MLP and RBFNN got the highest f-score values (i.e. 0.58) which make sense due to these classifiers are based on the neural network where numerous error-tuning process is being conducted. In contrast, using RNN, the superiority goes to LR where the f-score was 0.90. Such classifier showed fair results when the feature space is being improved and well-trained due to the classifier's ability to linearly learn the features. Although most of the classifiers based on RNN showed similar performance, the outperformance of LR indicates that the feature space has been well represented when the proposed RNN generates accurate embedding for the documents.

Generally speaking, the proposed embedding based on RNN has improved all the classifiers' abilities in terms of classifying the medical sentiments' polarities as shown in Fig. 4. Since it outperformed the baseline study (which has used CNN) thus, the effectiveness of using RNN for document embedding is demonstrated. Hence, the objective of this paper of providing an accurate document embedding is accomplished.

TABLE VI. Results of Traditional Classifiers

Classifier	Baseline (document embedding using CNN) [16]			Proposed (document embedding using RNN)		
	Precision	Recall	F1-score	Precision	Recall	F1-score
SVM	0.55	0.54	0.55	1.0	0.80	0.89
LR	0.54	0.54	0.54	0.90	0.90	**0.90**
MLP	0.58	0.58	0.58	0.95	0.86	0.89
RBFNN	0.59	0.56	0.58	0.87	0.87	0.87

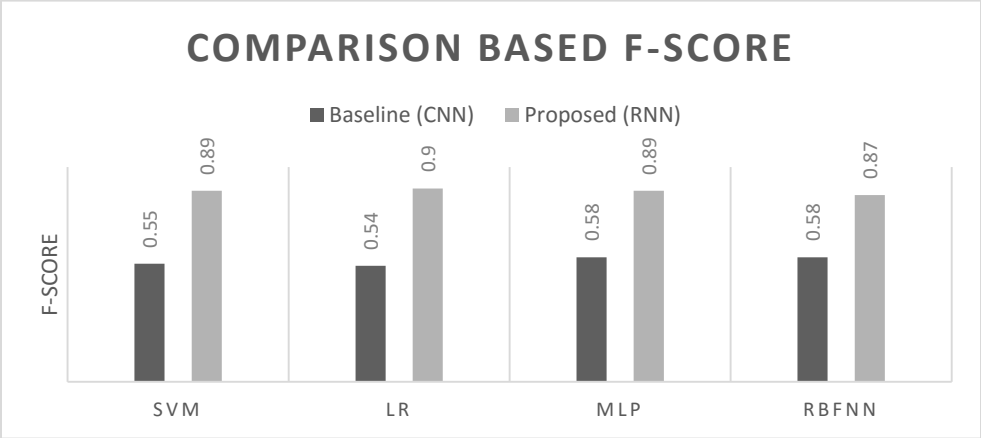

Fig. 4. Comparison between the Baseline (CNN) and the Proposed (RNN) based on F-Score.

V. Conclusion

This paper has proposed an effective document embedding method using RNN for MSA task. The novelty of the paper's contribution is represented by improving the classification of medical sentiments appear within social networks. The importance of enhancing such classification lies in the need of determining the exact effect of specific drugs or medications. In this regard, both the drug industry alongside the consumer would have the ability to see the feedback coming from people who were experiencing certain medications.

The main limitation of this study lies in considering a non-real time data of medical sentiments where a benchmark dataset has been considered for the comparison and validation purposes. Considering a real-time medical sentiments data for the process of classifying the polarity in future researches, would considerably contribute toward discovering new drug implications. In addition, further examination of deep learning architectures would probably provide promising performance in terms of improving vector representation in future researches.

Acknowledgments

This research has partially funded by the Faculty of Information Science & Technology and research code of GGPM-2017-025.

References

[1] M. Cohen, and W. R. Hersh, "A survey of current work in biomedical text mining," Briefings in bioinformatics, vol. 6, no. 1, pp. 57-71, 2005.

[2] B. Alshaikhdeeb, and K. Ahmad, "Biomedical Named Entity Recognition: A Review," International Journal on Advanced Science, Engineering and Information Technology, vol.6, no.6, pp. 889-895, 2016.

[3] B. Alshaikhdeeb, and K. Ahmad, "Feature selection for chemical compound extraction using wrapper approach with Naive Bayes classifier." pp. 1-6, 2017.

[4] B. Alshaikhdeeb, and K. Ahmad, "Comparative analysis of different data representations for the task of chemical compound extraction," International Journal on Advanced Science, Engineering and Information Technology, vol. 8, no. 5, pp. 2189-2195, 2018.

[5] R. N. Yousef, S. Tiun, and N. Omar, "Extended Trigger Terms for Extracting Adverse Drug Reactions in Social Media Texts," Journal of Computer Science, vol. 15, no. 6, pp. 873-879, 2019.

[6] C. Wu, F. Wu, Z. Yuan, J. Liu, Y. Huang, and X. Xie, "MSA: Jointly Detecting Drug Name and Adverse Drug Reaction Mentioning Tweets with Multi-Head Self-Attention." pp. 33-41, 2019.

[7] C. Shen, H. Lin, K. Guo, K. Xu, Z. Yang, and J. Wang, "Detecting adverse drug reactions from social media based on multi-channel convolutional neural networks," Neural Computing and Applications, vol. 31, no. 9, pp. 4799-4808, 2019.

[8] N. Kumar, B. Narendra, K. Upendra, and K. Rajesh, "A review on Adverse drug reactions monitoring and reporting," International Journal of Pharmacy Research & Technology, vol. 9, no. 2, pp. 12-15, 2019.

[9] R. Swaminathan, A. Sharma, and H. Yang, "Opinion mining for biomedical text data: Feature space design and feature selection," 2010.

[10] K. Denecke, and Y. Deng, "Sentiment analysis in medical settings: New opportunities and challenges," Artificial intelligence in medicine, vol. 64, no. 1, pp. 17-27, 2015.

[11] I. Korkontzelos, A. Nikfarjam, M. Shardlow, A. Sarker, S. Ananiadou, and G. H. Gonzalez, "Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts," Journal of biomedical informatics, vol. 62, pp. 148-158, 2016.

[12] S. Yadav, A. Ekbal, S. Saha, and P. Bhattacharyya, "Medical sentiment analysis using social media: towards building a patient assisted system," 2018.

[13] M. M. Ghassemi, R. G. Mark, and S. Nemati, "A visualization of evolving clinical sentiment using vector representations of clinical notes." pp. 629-632, 2015.

[14] Q. Chen, and M. Sokolova, "Word2Vec and Doc2Vec in Unsupervised Sentiment Analysis of Clinical Discharge Summaries," arXiv preprint arXiv:1805.00352, 2018.

[15] D. Mahata, J. Friedrichs, R. R. Shah, and J. Jiang, "Detecting personal intake of medicine from twitter," IEEE Intelligent Systems, vol. 33, no. 4, pp. 87-95, 2018.

[16] S. Liu, and I. Lee, "Sentiment Classification with Medical Word Embeddings and Sequence Representation for Drug Reviews." pp. 75-86, 2018.

[17] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative study of CNN and RNN for natural language processing," arXiv preprint arXiv:1702.01923, 2017.

[18] A. Yates, and N. Goharian, "ADRTrace: detecting expected and unexpected adverse drug reactions from user reviews on social media sites." pp. 816-819, 2013.

[19] P. Corbett, and J. Boyle, "Chemlistem–chemical named entity recognition using recurrent neural networks." pp. 61-68, 2017.

[20] A. Cocos, A. G. Fiks, and A. J. Masino, "Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in Twitter posts," Journal of the American Medical Informatics Association, vol. 24, no. 4, pp. 813-821, 2017.