Deep Siamese Neural Network Vs Random Forest for Myanmar Language Paraphrase Classification

Myint Myint Htay, Ye Kyaw Thu, Hnin Aye Thant, and Thepchai Supnithi

***Abstract*— Generally, paraphrase detection or semantic similarity of necessity is to understand the sentence as a whole sentence, but not just finding synonyms of the words. It is an important research area in natural language processing that plays a significant role in many applications such as question answering, summarization, information retrieval, and extraction. To our best knowledge, no studies have been conducted on Burmese (Myanmar language) paraphrase or not paraphrase detection and classification. In this research paper, we proposed the comparison of the results of Burmese paraphrase classification with the Deep Siamese Neural Network with MaLSTM (Manhattan LSTM) and Random Forest Classification with 21 features. More specifically, the contribution of this paper is the development of the human-annotated combination of Burmese paraphrase and non-paraphrase corpus that contained 40,461 sentence pairs and open-test data with 1,000 sentence pairs. According to the comparison of our implementation, the Random Forest Classifier is more accurate and useful for Burmese paraphrase classification than Deep Siamese Neural Network even with limited data.**

***Index Terms*—Semantic Text Similarity, Burmese (Myanmar Language), Deep Siamese Neural Network, Random Forest Modeling, Manhattan LSTM (MaLSTM), Harry Tool.**

I. Introduction

PARAPHRASING is exhibiting an input text in different ways but keeps on its original message. Many Natural Language Generation (NLG) tasks can be viewed as generating paraphrases. Paraphrase generation and detection is an important task in NLP, which is the main technology in many applications such as retrieval based question and answering system, semantic analyzing in NLP task, query expression in web searching, data summarization, data increasing for dialogue system such as chatbot system. However, due to the simplicity of natural language and as well as one of the under-resourced languages Burmese (Myanmar Language), automatically generating paraphrase and detection for paraphrase or not is still very challenging.

In research paper [1], measuring Semantic Textual Similarity (STS) is the task of calculating the similarity between a pair of texts. It is using both direct and indirect relationships between them. Text Similarity is very important in many natural language processing (NLP) applications such as question-answering systems, summarization, information retrieval, and extraction. In [2], it presented a survey on different methods of textual similarity and it also reported about the availability of different software and tools those are useful for Measuring Semantic Textual Similarity (STS). In translation memory, retrieval and matching also used text similarity model as shown in [3]. The traditional machine learning is involved in heavy feature engineering for early period [4]. The progress of word embeddings, and as a result of the success, neural networks have achieved in other fields, most of the methods proposed in recent years rely on neural network architectures [5]. Neural networks are preferred over traditional machine learning models as they generally perform better than traditional machine learning models.

In under-resourced languages such as Burmese (Myanmar Language), Thai, and Khmer, the complexity of natural language and automatic detection of paraphrasing or not is very complex and very challenging as well.

In this paper, we applied the NNs(Neural Networks) called Siamese neural networks for Burmese paraphrase text detection. These networks contain two or more identical subnetworks. These networks are identical and it has the same form with the same parameters and weights. Besides, parameter updating is reflected across these subnetworks. Siamese networks are famous among the tasks. It involves finding similarities or relationship between two corresponding things. These are successful in tasks like signature verification [6], image similarity [7] and have been recently used strongly in sentence similarity [8].

Siamese networks are good in these similarities tasks. It makes the similarities model to process similarities inputs. Therefore, these networks have design for vectors with same definition. Making the easier way is to compare couple of sentences. These weights are shared along with the sub-networks. This is the less parameters for training. It means that they need the fewer training data and little bias to overfit.

Because of the lack of information, the system can not detect the data as paraphrase or not, the information retrieval system has many problems. To avoid the weakness of information retrieval for Burmese, our research can support to detect the searching information in network. Moreover, there is no paraphrase detection in Myanmar language using semantic similarity measure. Therefore,

Myint Myint Htay is with the Faculty of Information Science, University of Technology (Yatanarpon Cyber City), Pyin Oo Lwin, Myanmar, Corresponding author email: myintmyinthtay22@googlemail.com

Ye Kyaw Thu is with LST Lab., NECTEC, Thailand and IDRI, CADT, Cambodia, Corresponding author email: yktnlp@gmail.com.

Manuscript received Feb 21, 2022; accepted Jul 17, 2022; revised Oct 10, 2022, published on 31 Oct 2022.

this paper introduces the first paraphrase classification system for Myanmar language using Siamese Neural Network with MaLSTM distance and Random Forest Modeling.

A. Related Work

The related task with the Sentence Textual Similarity is using linguistic resources such as pre-trained word embeddings and WordNet embeddings. It is a deep collection of learning algorithmic program such as Support Vector Regression (SVR). In the regression and neural networks, the various techniques are used for selection of multiple features of sentences to determine the similarities scores.

In research paper [5], it proposed an Siamese Neural Networks incorporated with language autonomous features. Moreover, it executed short text semantic calculation in multiple languages and domains which used three different corpora. It also changed the activation functions sigmoid to ReLU. The Pearson correlation (PC) was performed the evaluation. The Mean Squared Error (MSE) between the models' predicted values and the gold standard of corpora.

In research paper [11], it showed that Recursive AutoEncoder (RAE) and WordNet framework are used to make sentence embeddings. It combined the embeddings with a SVM classifier to calculate a semantic connection score. In research paper [12], LSTM improves RNNs to handle long-term dependencies. It used the Siamese LSTM to encrypt sentences using a pre-trained word embedding. To encode the sentences, Siamese LSTMs used the same weights for the input sentences to produce same sentence representations for similar sentences. And then, this networks predicted the nearness of pair of sentences using the Manhattan distance between the two sentence representations.

To measure the semantic similarity, two sentences were trained by a siamese neural network architecture. It is the metric learning. In this [15], the solution of the result solution it shown in writing that is more efficient and demonstrative text. In research paper [9], it generated sentence embedding using a Siamese CNN architecture. And then it is used with various convolution and pooling operations to extract distinctive granularities of information. The convolution used filters that analyze entire word embeddings and each dimension of word embeddings with multiple window sizes.

In [21], it collected Burmese paraphrase word and sentences by using three Statistical machine translation(SMT) models such as Phrase-Based Statistical Machine Translation systems(PBSMT), Hierarchical Phrase-based Statistical Machine Translation(HPBSMT) and Operation Sequence Model(OSM) to generate the paraphrase words and sentences for input sentences. And then automatic measure the score with BLEU, RIBES and chrF^{++}. In paper [22], it also examined in Arabic and English for question retrieval in community question answering. In that research, propose for using word embeddings and it can support for semantic and syntactic information from contexts. In order to acquire longer sequences and questions are expanded with words having close word vectors. The embedding vectors are put into the Siamese LSTM model to consider the global context of questions. Using the Manhattan distance, to measure the similarity between the questions.

In order to improve the accuracy of recommendation, research paper [23] proposed a text matching model based on Siamese semantic network and MatchPyramid model. It is the algorithm that combines the innovative features of Siamese semantic network with MatchPyramid model. It also has the features for classification. In that paper, compare the proposed model and other models for the same datasets. The results show that their proposed model execute better than other models. In [24], it proposed an Attentive Siamese LSTM network for measuring semantic similarity. They also used raw sentence pairs and pre-trained word embedding which are used for input sentence. It also demonstrated with three corpora and three language tasks.

In research paper [10], it represented phrases using neural networks where inputs are word vectors learned independently from a large corpus. However, their purpose of learning interpretation explicitly represents the semantic similarity labels in question. It used neural networks to predict the similarity of word and sentences representations. Semantically ordered representation space should be trained in such a way that basic metrics are adequate to catch semantic sentences.

B. Methodologies

In this section, it will describe the methodologies of paraphrase sentence classification processes used in the experiments of this paper.

1) Word2Vec

Word2vec is a word embedding algorithm that is individually using for all of the NLP task such as similarity measure and other tasks. It is converting from word or sentences to their relating vectors format [27]. Moreover, the vectors can be used successfully for different forms of Natural Language Processing tasks. The size and form of vector is varied because of the converting state of its size. In this vector, consisting of words is a function of deep learning architecture.

2) FastText Embedding

fastText is another word embedding method that is an extension of the word2vec model. Instead of learning vectors for words directly, fastText represents each word as an n-gram of characters [28].

3) Character Embedding

Character level embedding uses one-dimensional convolutional neural network (1D-CNN) to find numeric representation of words by looking at their character-level compositions [29].

4) Harry: A Tool for Measuring String Similarity

In this article, we used Harry, a small tool specifically designed for measuring the similarity of strings. Harry implements 21 similarity measures, including common string distances and string kernels and string coefficients. The tool has been designed with efficiency in mind and is allowed for multi-threaded as well as distributed computing, enabling the analysis of large data sets of strings. Harry supports common data formats and thus can interface with analysis environments, such as Matlab, Pylab and Weka [25].

The current version of Harry supports the similarity measures listed in Table I. In this research, we used and measured all of the Harry String Similarity tool.

5) Random Forest Modeling

Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. Ensemble learning method is a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model. The trees run in parallel with no interaction amongst them. A Random Forest operates by constructing several decision trees during training time and outputting the mean of the classes as the prediction of all the trees.

1. Pick at random k data points from the training set.

2. Build a decision tree associated to these k data points.

3. Choose the number N of trees that is wanted to build and repeat steps 1 and 2.

4. For a new data point, make each one of N-tree trees predict the value of y for the data point in question and assign the new data point to the average across all of the predicted y values.

A Random Forest Regression model is powerful and accurate. It usually performs great on many problems, including features with non-linear relationships. Disadvantages, however, include the following: there is no interpretability, overfitting may easily occur, must be chosen the number of trees to include in the model. [26]

Several measures are available for feature importance in Random Forests:

6) Mean Decrease in Impurity (MDI)

Gini Importance or Mean Decrease in Impurity (MDI) calculates each feature importance as the sum over the number of splits (accross all tress) that include the feature, proportionaly to the number of samples it splits.

$$\textbf{Gini impurity} = \textbf{1} - \textbf{Gini} \tag{1}$$

7) Feature Permutation

Permutation Importance or Mean Decrease in Accuracy (MDA) is assessed for each feature by removing the association between that feature and the target. This is achieved by randomly permuting the values of the feature and measuring the resulting increase in error. The influence of the correlated features is also removed.

8) Siamese Recurrent Neural Networks

Input layer of Siamese neural network(SNN) changes over each vector of indexes accepted from sentences preprocessing into a word distributed demonstration. The capability of its semantic properties of the words fixes well in result representation. Skip-Gram model used the pre-trained sentences on an external corpus. Thus, this approach does not calculate on a extremity feature extraction process to correspond input words with efficiently. The SNN is also the neural networks that includes two similar sub networks and it produces the joint output. and it is widely used for the text similarity between two patterns, words and sentences such as paraphrase detection. The exchanged weights are across to the sub-networks. It also reduces the number of training parameters. And then the model produces the semantic. This LSTM used pre-trained corpus to read in vector-shaped terms. It defines each word vector and uses presentation which was past processed by the former request. In addition, the representations of similarity between sentences are often used as measure of semantic similarity [16]. The similarity of Siamese is a term for detection using the MaLSTM model. MaLSTM and Siamese are interrelated to use for the original weighted for each sentence to be recognized. Then it would do the machine learning activity by performing it using the weight of MaLSTM. Siamese similarities to MaLSTM models would recognize sentences. It have been often weighted and prepared. Extraction uses word2vec to get a vector output for each word. The vector value will be entered in the weighting phase for MaLSTM. The weight of each side of the LSTM will be graded using Siamese Similarities. Manhattan LSTM provides a reasonably simple solution to basic sentence similarity concerns. Since it is a joint network, and it's simpler to train. Because it can swap the weights on sides of both. Siamese networks have two or three of the same sub-networks on their networks. This network fits best for semantic sentence comparisons [16]. The results of MaLSTM are more easy in the preprocessing. It provided that the LSTM networks shares with their weight of each word in the sentence on the both sides. MaLSTM is also widely used to process text, sentences or phrases. Therefore, this MaLSTM is very suitable and useful for this research.

C. Word Segmentation

Word segmentation is the very important method for the text analysis level. The under resource languages such as Burmese text are not usually separated with white space between words. The white spaces are often used to distinguish sentences for easier reading. We used myWord Segmentation Tool [14] for Burmese word segmentation process. The myWord supports syllable, sub_word, word and phrase segmentation. But, word segmentation for this research is only used.

D. Building Burmese Paraphrase Corpus

In Burmese, various words and various conversation styles for the same performance are expressed in daily conversation and writing sentences. Some of the paraphrase sentences are different only one word in that sentence and some sentences are quite different for the whole sentences.

TABLE I: Similarity measures for strings supported by Harry (version 0.4.1)

10 String Similarity Distances			
Bag distance Compression distance Jaro-Winkler distance	Hamming distance Jaro distance Levenshtein distance	Kernel-substitution distance Lee distance	String alignment distance Damerau-Levenshtein distance
4 String Kernels			
Distance-substitution kernel	Spectrum kernel	Subsequence kernel	Weighted-degree kernel
7 String Coefficients			
Braun-Blanquet coefficient Jaccard coefficient	Kulczynski coefficient Otsuka coefficient	Simpson coefficient Soerensen-Dice coefficient	Sokal-Sneath coefficient

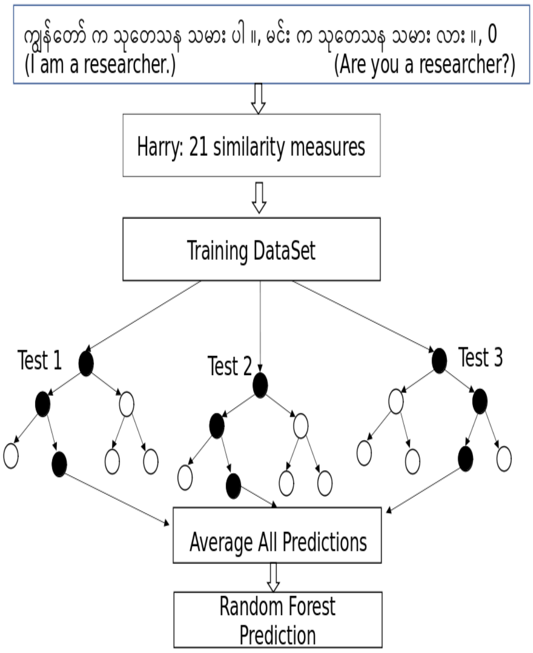

Fig. 1: Random Forest Modeling

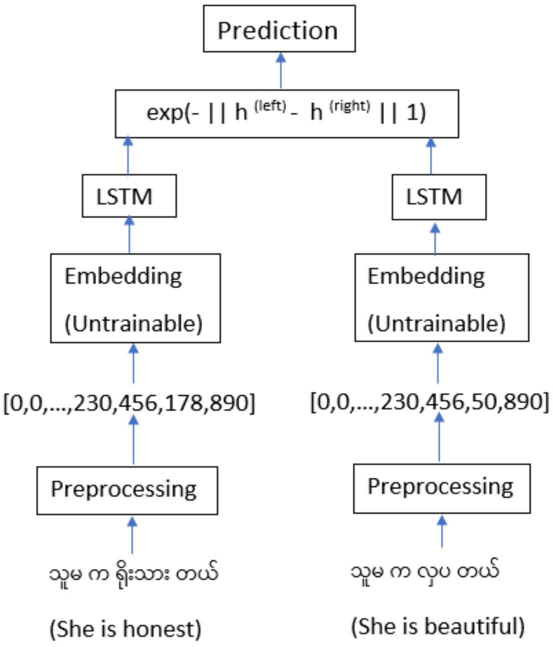

Fig. 2: Manhattan LSTM diagram

Some of the sentences are collected from social media (Facebook comments) and the comments are collected from the famous Myanmar news websites by extracting the Facepager Tool (version 4.2.7) [19]. Moreover, all of the paraphrase words are collected from the Burmese Wiktionary [20] site and extraction with Web Scraper tool. Using these words, more of the paraphrase sentences for this research are built manually.

Based on [21], paraphrase sentences (15,640 sentence pairs) and non paraphrase sentences (24,821 sentence pairs) with the total of 40,461 sentence pairs are selected. In this research, these total sentence pairs are trained and evaluated for three times. All of the training with 40,461 sentences pairs and the first evaluation data with 1,000 sentences pairs of open-test data that are not participate in the training data. In second evaluation with 1,000 of open-test data. In this time, we mixed the open-test data with training data and we shuffled all of the sentences pairs and among them we collected and used 1,000 sentences pairs as evaluation data. At the third times, we shuffled again of all data and we collected and used 1,000 sentences pairs as evaluation data.The Burmese corpus is a UTF-8 plain text file. If the sentences pair is paraphrase and it is denoted as similar and marked with "1" and if the sentences pair is not paraphrase and it is denoted as not similar and marked with "0". After labeling, paraphrase or not paraphrase datasets are as shown in Table II clearly.

E. Experimental Setup

In this experiment, two main factors are proposed. Firstly, the input corpus is segemented with syllable, manual and word segmentation (myWord). And then all of the input are converted as vector with character embedding, word2vec and fastText embedding. In implementation stage, two networks are proposed and the results of two structures are comprised. The structure shows two input layers and declare to transfer the vector representation for the first input sentence and the second sentence pairing to the embedding layers. Moreover, the results in the embedding of sentences and the embedding vectors are put to the LSTM networks. Next, the layer of LSTM generates a vector representation of two sentences in the input sentences pair. These layers combined two input representations into a single vector representation, which is then used for the final detection for Burmese paraphrase or not.

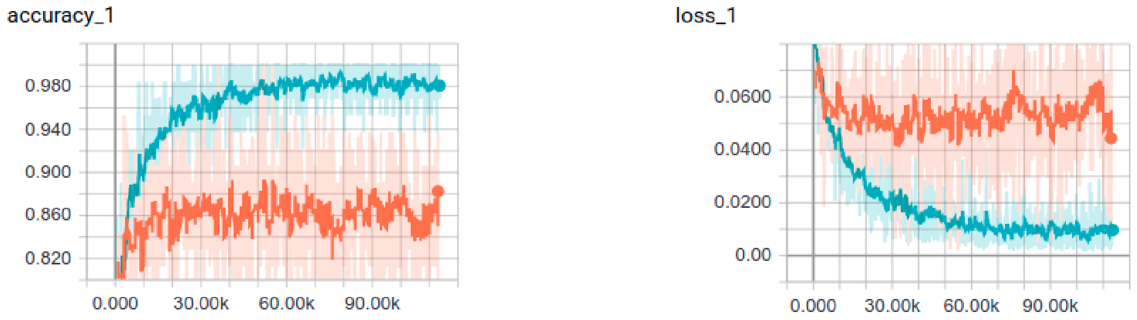

Fig. 3: Training and Validation Result with Manual-Word, word2vec, 200 epoch. Left: Accuracy, Right: Loss

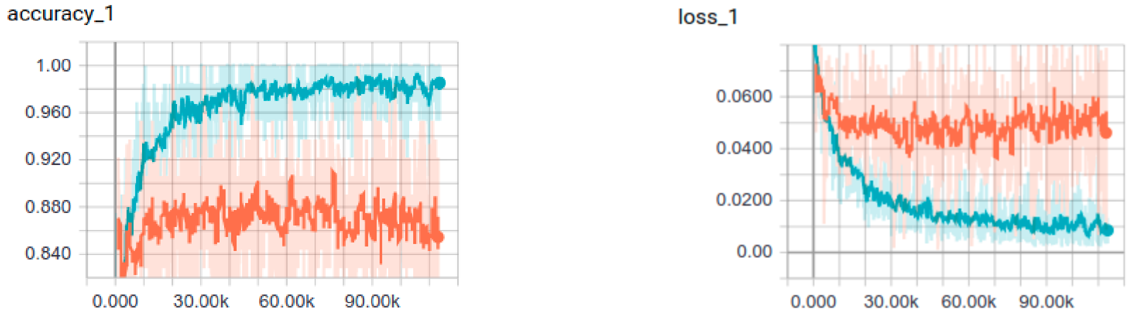

Fig. 4: Training and Validation Result with Syllable Unit, word2vec, 200 epoch. Left: Accuracy, Right: Loss

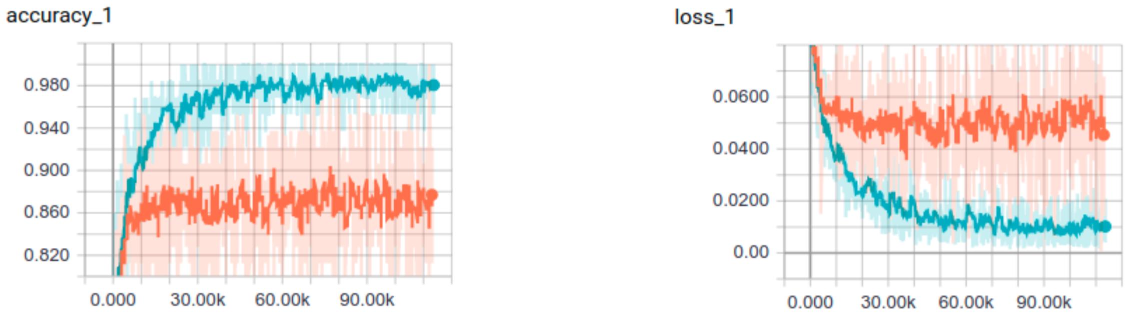

Fig. 5: Training and Validation Result with ”Word Unit Segmented with myWord”, word2vec, 200 epoch. Left: Accuracy, Right: Loss

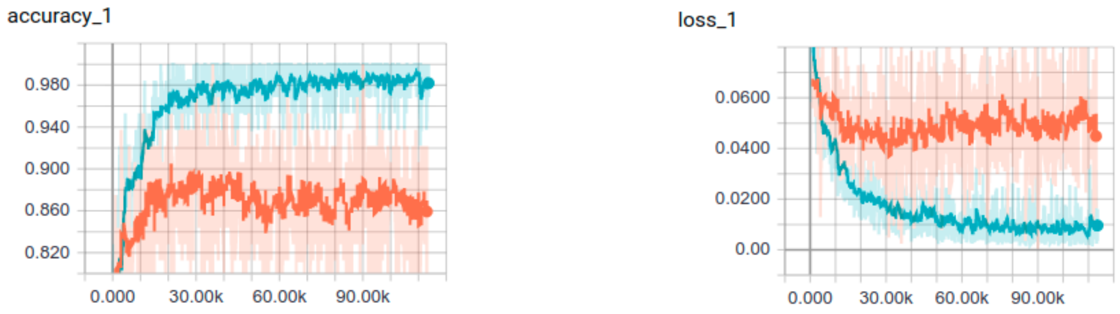

Fig. 6: Training and Validation Result with ”Manual-Word”, fasttext, 200 epoch. Left: Accuracy, Right: Loss

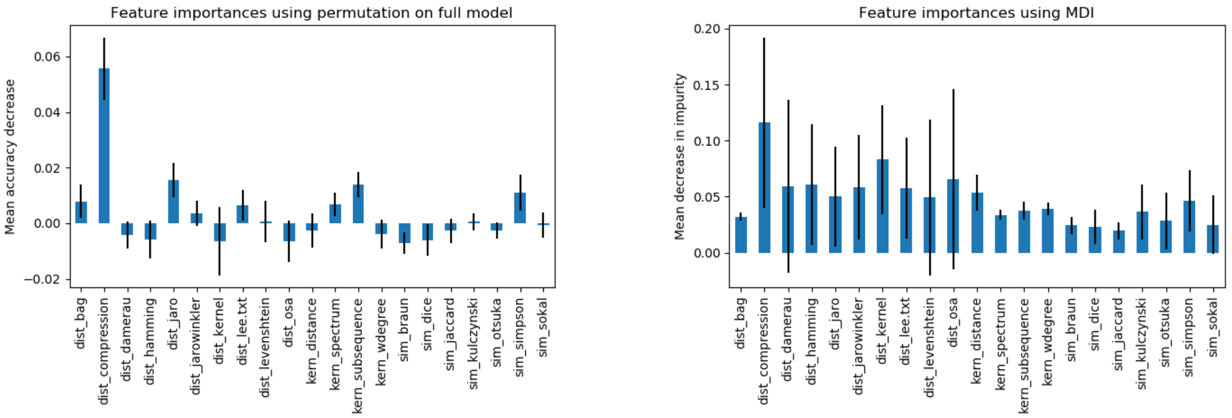

Fig. 7: Important feature graph of 21 string similarity measures for Random-Forest (train: train1, eval with test1). Left: with Feature Permutation, Right: with MDI

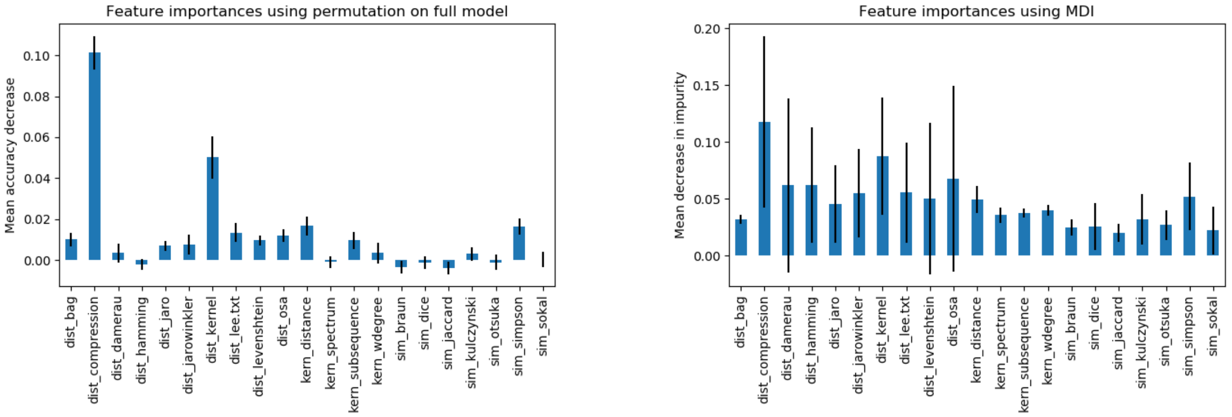

Fig. 8: Important feature graph of 21 string similarity measures for Random-Forest (train: train2, eval with test2). Left: with Feature Permutation, Right: with MDI

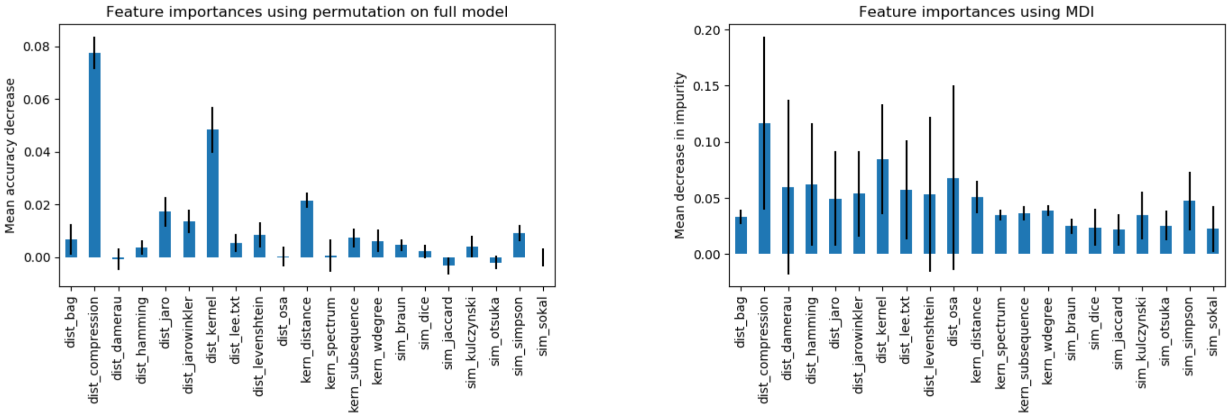

Fig. 9: Important feature graph of 21 string similarity measures for Random-Forest (train: train3, eval with test3). Left: with Feature Permutation, Right: with MDI

TABLE II: Some examples of paraphrase and non-paraphrase sentences

Id	Sentence1	Sentence2	Paraphrase?
1	ဆက် ကြိုးစား ကြပါ (Keep trying)	ဆက် ပြီး ကြိုးစား ပေး ပါ (Keep trying)	1
2	လေးစား တယ် အား လည်း ကျ မိ တယ် (I respect and envy)	မလေးမစား မ လုပ် နဲ့ အတုယူ ပါ (Do not disrespect and then imitate)	0
3	သူမ က သူ့ ကို အပြစ်တင် တယ် (She blames him)	သူမ က သူ့ ကို အပြစ်မတင် ခဲ့ ပါ ဘူး (She did not blame him)	0
4	ကောင်း သော ညချမ်း လေး ပါ (Good evening)	ပျော်ရွှင် စရာ ညချမ်း လေး ပါ (Have a nice evening)	1
5	ဒီ ဖလင် ကို ကွန်ပျူတာ သုံး ပြီး ပြင် မှာ လား ။ (Will this film be computer-generated?)	ဒီ ဖလင် ကို ဘယ် ကွန်ပျူတာ မှာ ပြင် ထား တာ လဲ ။ (On which computer was this film edited?)	0

TABLE III: Deep Siamese Neural Network Training/Evaluation with Test Data

Deep Siamese Neural Network Training/Evaluation with Test Data		
Segmentation/Method	**Closed-test**	**Open-test**
Manual (word2vec)	**0.97**	0.45
Syllable (word2vec)	0.96	0.44
Word (word2vec)	0.96	0.44
Manual (char-embedding)	0.94	0.44
Syllable(char-embedding)	0.92	0.43
Word (char-embedding)	0.93	0.44
Manual (fasttext embedding)	0.97	0.44
Syllable (fasttext embedding)	0.94	0.46
Word (fasttext embedding)	0.94	**0.49**

TABLE IV: Random-Forest Training/Evaluation with Manual Open Test Data

Random-Forest Training/Evaluation with Manual Open Test Data	
Accuracy on the training: 1	0.99
Accuracy on the testing 1	0.61
Accuracy on the training: 2	0.99
Accuracy on the testing 2	0.85
Accuracy on the training: 3	0.99
Accuracy on the testing 3	0.85

For Random Forest Classification, Harry tool and extracted 21 features are used. And then that features are input in the Random Forest Modeling.

Training results can then be entered into the learning process with the Siamese Similarity RNN to evaluate the similarity of the context of the statement used in the equation 2.

$$exp(-\|h^{(left)} - h^{(right)}\|1)] \in [0,1] \tag{2}$$

The difference of the left side of the network h$^{(\text{left})}$ is where the exp is the right side of the network h$^{(\text{right})}$. Since the values from left to right are changed using joint or twin network characters to evaluate the significant increase between the two networks.

F. Result And Discussion

In this research, two networks are used and proposed, then the results are comprised.

1) Deep Siamese Neural Network's Results

The experiment is mainly emphasize in training data and test data with manual, syllable and word segmentaion and all of the data with character embedding, word2vec and fasttext embedding. According to the experiment, the accuracy and loss are shown in figures. In the results, manual segmentation with word2vec embedding with Closed-Test data is the highest score of **0.97** and for Open-test data with Word segmentation with (myWord) and fasttext embedding is the highest score of **0.49** as shown in Table III.

In Figure 3, it shows the accuracy and loss of training and validation with manual word segmentation and then using with word2vec embedding and the epoch set up with 200.

In Figure 4 it also shows the accuracy and loss of training and validation result of segmentation with syllable unit and the embedding unit with word2vec and 200 epoch.

Also in Figure 5, it expresses the training and validation result with word unit segmented with myWord and the embedding method is word2vec and 200 epoch.

In Figure 6, it shows the training and validation result with manual-word unit with myWord segmented and the embedding with fasttext and 200 epoch.

2) Random Forest Modelling Results

As the results are shown in Table IV, it is proposed and tested with three training times and three testing times. According to the experiment, accuracy on all of the training is 0.99 and testing 2 and 3 are the same results with 0.85. Thus, as the results of experiment, Random Forest Modeling is more accurate and classifies the Burmese paraphrase sentences than Deep Siamese Neural Network.

In Figure 7, it shows feature permutation and MDI of the important feature graph of 21 string similarity measures for Random-Forest with training with train1, evaluation with test1.

Also in Figure 8, it shows feature permutation and MDI of the important feature graph of 21 string similarity measures for Random-Forest with training with train2, evaluation with test2.

As in Figure 9, it shows feature permutation and MDI of the important feature graph of 21 string similarity measures for Random-Forest with training with train3, evaluation with test3.

II. Conclusion

The sentences can be rendered after deep learning using the Siamese similarity RNN and Random Forest Modeling. The accuracy improved and loss reduced in the testing with Random Forest Modeling. In this paper, a fast and generic Burmese paraphrase classification model based on MaLSTM and Random Forest Classification are proposed. It achieved promising results on our developing Burmese paraphrase corpus. Results also determined that the machine learning process has been replicated from random input data and data replication. For future work, it will be arranged to study how these methods work on longer Burmese sentences and paragraph level.

References

[1] V. Rus, M.C. Lintean, R. Banjade, N.B. Niraula, and D. Stefanescu. 2013. Seminar: “The semantic similarity toolkit“. In ACL.

[2] G. Majumder, P. Pakray, A.F. Gelbukh, and D. Pinto “Semantic textual similarity methods, tools, and applications“ A survey. Computación y Sistemas, 2016.

[3] R. Gupta, H. Bechara, and C. Orasan. “Intelligent translation memory matching and retrieval metric exploiting linguistic technology.“ Proceedings of Translating and the Computer, 36: pp. 86–89.

[4] H. Béchara, C. Orasan, H. Costa, S. Taslimipoor, R. Gupta, G.C. Pastor, and R. Mitkov. Miniexperts: “An svm approach for measuring semantic textual similarity“. In SemEval@NAACL-HLT. 2015.

[5] J. V. A. Souza, L. E. S. E. Oliveira, Y. B. Gumiel, D. R. Carvalho, and C. M. C. Moro “Exploiting Siamese Neural Networks on Short Text Similarity Tasks for Multiple Domains and Languages“ P. Quaresma et al. (Eds.): PROPOR 2020, LNAI 12037, pp. 357–367, 2020.

[6] J. Bromley, Y. LeCun, I. Guyon, E. Säckinger, and R. Shah. “Signature verification using a siamese time delay neural network“. IJPRAI, 7: pp. 669–688 1993.

[7] G. Koch, R. Zemel, and R. Salakhutdinov. 2015. “Siamese neural networks for one-shot image recognition“. In ICML Deep Learning Workshop, volume 2.

[8] P. Neculoiu, M. Versteegh, and M. Rotaru. 2016. “Learning text similarity with siamese recurrent networks“. In ACL 2016.

[9] H. He , K. Gimpel & J.J. LIN (2015). “Multi-perspective sentence similarity modeling with convolutional neural networks“. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015, p. 1576–1586.

[10] J. Mueller & A. Thyagarajan (2016). “Siamese recurrent architectures for learning sentence similarity“. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI’16, p. 2786–2792: AAAI Press.

[11] B. Rychalska, K. Pakulska, C.K. Hodorowska, W. Walczak & A. Ndruszkiewiczp, Samsung poland nlp team at semeval-2016 task 1: Necessity for diversity; combining recursive autoencoders, wordnet and ensemble methods to measure semantic similarity. In SemEval@ NAACL-HLT.

[12] Y. KIM “Convolutional neural networks for sentence classification“. In Proceedings of EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, p. 1746–1751.

[13] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu & P. Kuksa “Natural language processing (almost) from scratch“ 2493–2537.

[14] Ye Kyaw Thu,“myWord: Syllable, Word and Phrase Segmenter for Burmese, Sept 2021, GitHub Link: https://github.com/ye-kyaw-thu/myWord“.

[15] A. Y. Ichida, F. Meneguzzi, D. D. Ruiz, “Measuring Semantic Similarity Between Sentences Using a Siamese Neural Network“.

[16] Z. Chen, H. Zhang, X. Zhang, and L. Zhao, “Quora Question Pairs“, pp. 1–7, 2017.

[17] E. L. Pontes, S. Huet, A. C. Linhares, J. Manuel, T. Moreno, “Predicting the Semantic Textual Similarity with Siamese CNN and LSTM“

[18] T. Mikolov, M. Karafiát, L. Burget, and S. Khudanpur, “Recurrent neural network based language model,“ Proceedings of 11th Annual Conference on the International Speech Communication Association, pp. 1045–1048, Sept, 2010.

[19] https://github.com/strohne/Facepager/releases/

[20] https://my.wiktionary.org/wiki/

[21] M.M. Htay, Y.K. Thu, H.A Thant, T. Supnithi “Statistical Machine Translation for Myanmar Language Paraphrase Generation“, Proceedings of 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP) pp. 255-260, 2020.

[22] N. Othman, R. Faiz, and K. Smaïli, “Manhattan Siamese LSTM for Question Retrieval in Community Question Answering“ Conference paper, 2019.

[23] Z. Tang, J. Li, “Jointly Considering Siamese Network and MatchPyramid Network for Text Semantic Matching“ SAMSE 2018.

[24] W. Bao, J. Du, Y. Yang and X. Zhao, “Attentive Siamese LSTM Network for Semantic Textual Similarity Measure“, 2018 International Conference on Asian Language Processing (IALP)

[25] Konrad Rieck and Christian Wressnegger,“Harry: A Tool for Measuring String Similarity“,Journal of Machine Learning Research,Vol:17, pp. 1-5,2016.

[26] Chaya Bakshi, “https://levelup.gitconnected.com/random-forest-regression-209c0f354c84“.

[27] https://wiki.pathmind.com/word2vec

[28] https://blogs.sap.com/2019/07/03/glove-and-fasttext-two-popular-word-vector-models-in-nlp/

[29] https://towardsdatascience.com/the-definitive-guide-to-bidaf-part-2-word-embedding-character-embedding-and-contextual-c151fc4f05bb

Myint Myint Htay is a Ph.D candidate at University of Technology (Yatanarpon Cyber City), Pyin Oo Lwin and Faculty of Computer Science(UCS(Monywa)) Myanmar. Her current doctoral thesis research focuses on Machine Translation of Burmese Paraphrase. She is interested in the research area of natural language processing (NLP), big data analysis, and deep learning.

Ye Kyaw Thu is a Visiting Professor of Language & Semantic Technology Research Team (LST), Artificial Intelligence Research Unit (AINRU), National Electronic & Computer Technology Center (NECTEC), Thailand and Affiliate Professor at Cambodia Academy of Digital Technology (CADT), Cambodia. He is also a founder of Language Understanding Lab., Myanmar. His research lie in the fields of artificial intelligence (AI), natural language processing (NLP) and human-computer interaction (HCI). He is actively co-supervising/supervising undergrad, masters' and doctoral students of several universities including Assumption University (AU), Kasetsart University (KU), King Mongkut's Institute of Technology Ladkrabang (KMITL) and Sirindhorn International Institute of Technology (SIIT).

Hnin Aye Thant She is currently working as a Professor and Head of Department of Information Science at the University of Technology (Yatanarpon Cyber City), Pyin Oo Lwin Township, Mandalay Division, Myanmar. Shegot Ph.D (IT) Degree from University of Computer Studies, Yangon, Myanmar in 2005. The current responsibilities are managing professional teachers, doing instructional designer of e-learning content development and teaching. She has 14 years teaching experiences in Information Technology specialized in Programming Languages (C,C++, Java and Assembly), Data Structure, Design and Analysis of Algorithms/Parallel Algorithms, Database Management System, Web Application Development, Operating System, Data Mining and Natural Language Processing. She is a member of research group in "Neural Network Machine Translation between Myanmar Sign Language to Myanmar Written Text" and Myanmar NLP Lab in UTYCC. She is also a Master Instructor and Coaching Expert of USAID COMET Mekong Learning Center. So, she has trained 190 Instructors from ten Technological Universities, twelve Computer Universities and UTYCC for Professional Development course to transform teacher-centered approach to learner-centered approach. This model is to reduce the skills gap between Universities and Industries and to fulfill the students' work-readiness skills.

Thepchai Supnithi received the B.S. degree in Mathematics from Chulalongkorn University in 1992. He received the M.S. and Ph.D. degrees in Engineering from the Osaka University in 1997 and 2001, respectively. He is currently head of language and semantic research team artificial intelligence research unit, NECTEC, Thailand.