UTM UNIVERSITI TEKNOLOGI MALAYSIA
RESEARCH UNIVERSITY

International Journal
of
Innovative Computing

Journal Homepage: http://se.cs.utm.my/ijic

Feed-Forward Network Model for Multi-Document Relation Classification

Yogan Jaya Kumar
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
yogan@utem.edu.my

Naomie Salim
Faculty of Computing
Universiti Teknologi Malaysia
Johor, Malaysia
naomie@utm.my

***Abstract*—Using feed-forward artificial neural network to classify multi-document relation is the subject of this paper. Sentences across topically related documents can often be linked by means of relations that exist between them. In this study, we aim to identify four types of relations, namely, *Identity*, *Subsumption*, *Description* and *Overlap*. We propose to use neural network learning model for the task of classification; multi-class classification, in this case. The performance of our proposed approach was measured using Precision, Recall and F-measure. The experimental findings demonstrate that better results can be obtained by using the proposed approach when compared with the widely used SVM classifier.**

Keywords — cross-document structure theory (CST), multi document, supervised machine learning, neural network, support vector machine

I. Introduction

The study on multi-document relations was pioneered by Radev [1]. Radev introduced the CST model (Cross-document Structure Theory). The general schema of CST is shown in Fig. 1. Its fundamental idea is that documents which are related to the same topic usually contain semantically related textual units. These textual units can be words, phrases, sentences, or the documents itself. In our work, we investigate only the semantic relations between sentences. Four types of multi-document relations or CST relations are considered in this paper. These include *Identity*, *Subsumption*, *Description* and *Overlap*; as described in Table 1. The full descriptions of CST relations are given in [1].

The ability to automatically identify the CST relations from un-annotated text could be useful for applications related to multi document analysis. For instance, a number of works have addressed the benefits of CST for summarization task [2, 3]. However these works relies on text documents which were already annotated with CST relations. Thus the need for automation is deemed necessary. In this work, we present a learning model which is based on feed-forward neural network to identify the existence of CST relations in multi document texts.

The rest of this paper is organized as follows: Section 2 presents the related works. Section 3 outlines the proposed approach which is based on feed-forward neural network learning model to identify the CST relations. The experimental setting and results are given in Section 4. We finally end with conclusions in Section 5.

II. Related Works

A number of research works have attempted to learn the CST relations in texts. However only two works are known related to English texts [4, 5]. The authors used boosting, a classification algorithm to identify the presence of CST relationships between sentences. It is an adaptive algorithm which works by iteratively learning previous weak classifiers and adding them to a final strong classifier. The authors experimented with CST annotated articles collected from the CSTBank corpus. However their classifier showed poor performance in classifying most of the CST relations; obtaining average values of 45% precision, 31% recall, and 35% f-measure.

In another related work, the identification of CST relations on Japanese texts was investigated using SVM classifier [6].

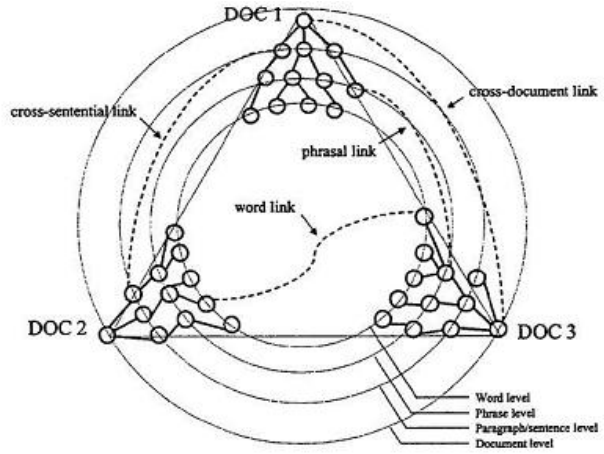

Fig. 1. CST general schema. (Radev, 2000)

TABLE I. CST RELATIONS USED IN THIS WORK

CST Relation	Description
Identity	The same text appears in more than one location
Subsumption	S1 contains all information in S2, plus additional information not in S2
Description	S1 describes an entity mentioned in S2
Overlap	S1 provides facts X and Y while S2 provides facts X and Z; X, Y, and Z should all be non-trivial.

The authors used a Japanese corpus annotated with CST relationships. The authors propose to use the detected "Equivalence" relations to address the task of "Transition" identification. They obtained f-measure of 75.50% for equivalence and 45.64% for transition. However, their approach is only limited to the two aforementioned relations.

CST parsing has also been experimented on Brazilian Portuguese texts [7]. The authors investigated three types of classifiers, namely, the multi-class, hierarchical, and binary classifiers to perform the CST classification over the Brazilian Portuguese CSTNews corpus. From their experiments (on unbalanced data), they obtained a general accuracy of 41.58%, 61.50% and 70.51% for the respective classifiers.

III. METHODS

The basic idea of machine learning is to learn or make decisions from existing data (or usually called training data); giving the ability to produce a useful output in new cases. If the training examples are given with correct input output pairs, then the learning is called supervised. Neural network is an example of supervised machine learning approach which is based on a network of many simple processors (called neurons). It learns by comparing the network output and target output and makes adjustments on the weights (connections between neurons) in order to move the network outputs closer to the targets. This process is depicted in Fig. 2. The network trains until the network output matches the target or achieves error below certain threshold value. A more comprehensive foundation of neural network can be found in [8].

In our CST relationship identification problem, we applied the Feed-forward neural networks. Feed-forward neural networks are usually trained by the back propagation algorithm [9]. In our work, we have used the Levenberg-Marquardt backpropagation (a variant of back propagation algorithm). All our training data are reprensented as feature vectors (input) of sentence pairs with its corresponding CST relation (output). We used the following features for each sentences pair (S_1, S_2):

Cosine similarity – cosine similarity is used to measure how similar two sentences are. Here the sentences are represented as word vectors having words with tf-idf as its element value:

$$\cos(S_1, S_2) = \frac{\sum S_{1,i} \cdot S_{2,i}}{\sqrt{\sum (S_{1,i})^2} \cdot \sqrt{\sum (S_{2,i})^2}} \quad (1)$$

Word overlap – this feature represents the measure on the numbers of words overlap in the two sentences (after stemming process). This measure is not sensitive to the word order in the sentences:

$$overlap(S_1, S_2) = \frac{\#commonwords(S_1, S_2)}{\#words(S_1) + \#words(S_2)} \quad (2)$$

Length type of S_1 – this feature gives the length type of the first sentence when the lengths of two sentences are compared:

$$\begin{aligned} lengtype(S_1) = 1 \quad & \text{if} \quad length(S_1) > length(S_2), \\ -1 \quad & \text{if} \quad length(S_1) < length(S_2), \\ 0 \quad & \text{if} \quad length(S_1) = length(S_2) \end{aligned} \quad (3)$$

NP similarity – this feature represents the noun phrase (NP) similarity between two sentences. The similarity between the NPs was calculated according to Jaccard coefficient as defined in the following equation:

$$NP(S_1, S_2) = \frac{NP(S_1) \cap NP(S_2)}{NP(S_1) \cup NP(S_2)} \quad (4)$$

VP similarity – this feature represents the verb phrase (VP) similarity between two sentences. The similarity between the VPs was calculated according to Jaccard coefficient as defined in the following equation:

$$VP(S_1, S_2) = \frac{VP(S_1) \cap VP(S_2)}{VP(S_1) \cup VP(S_2)} \quad (5)$$

Our network model is shown in Fig. 3. The numbers in the figure indicates 5 inputs features (cosine similarity, word overlap, length type, NP similarity and VP similarity), 12 hidden neuron, 1 output neuron dan 1 final output (CST relation type). When a neuron receives the input, it multiplies its strength by weigth *w*. This weighted input is then added with bias *b* which is much like a weight, having constant value

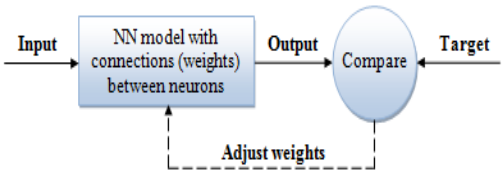

Fig. 2. Neural network's general process paradigm.

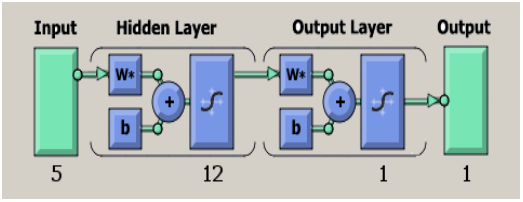

Fig. 3. Feed-forward network model.

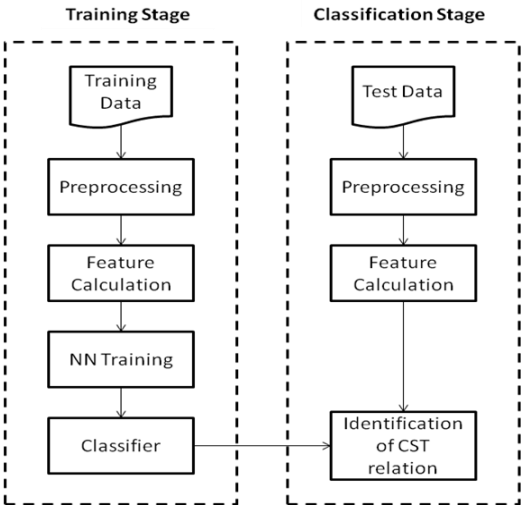

Fig. 4. The training and classification processes.

of 1. Figure 4 shows the training and classification stages. We first preprocess the text by stopword filtering and word stemming. After computing each of the feature value for every sentence pair from the training set, we input them for the training of NN. Once the training is completed, the resulting classifier model will be tested with test data to measure its performance.

IV. Experiments And Results

In this study, we used the dataset obtained from CSTBank [10] – a corpus consisting clusters of English news articles annotated with CST relationships. Our training and testing set consist of sentence pairs with its corresponding CST type label. We selected 476 sentence pairs for training and 206 sentence pairs for testing. These include a sample of 100 pairs of sentences that have no CST relations. We trained the data using the neural network tool on MATLAB. We use the multilayer feed-forward network with the default tan-sigmoid transfer function in the hidden layer and linear transfer function in the output layer. The number of hidden nodes H_i is initially set to 1. The accuracy of the network is then recorded for H_i after training it. Then H_i is incremented and the process continues. The process ends when the result of H_i is better than H_{i+1} and H_{i+2}. After determining the best H, we fixed it as the number of hidden node in the network hidden layer.

Table 2 and Fig. 5 show the precision, recall, and f-measure of NN classification. We also tested the performance of support vector machine (SVM) classifier; for comparison use. The SVM model best parameters were chosen after applying 5-fold cross validation. Figure 6 shows the f-measure comparison between these two techniques. We can see that both techniques give good performance (i.e. > 90%) for the relationship type "Identity" and (> 80%) for "No Relation". This is most likely due to the characteristics of "Identity" type sentences which have high similarity in terms of words and length while "No Relation" has the complete opposite characteristics. Overall, the classification accuracies obtained by NN and SVM are 80.09% and 78.64% respectively. NN was able to outperform SVM. This could probably related to number of features used as SVM normally performs well with high dimensionality datasets; which plausibly explain why SVM could not well differentiate between the different classes of relations.

TABLE II. Precision , Recall And F-Measure Of NN Classification

CST Type	Precision	Recall	F-Measure
No Relation	1	0.8	0.8888889
Identity	1	0.966667	0.9830508
Subsumption	0.754717	0.8	0.776699
Description	0.711111	0.842105	0.7710843
Overlap	0.727273	0.689655	0.7079646

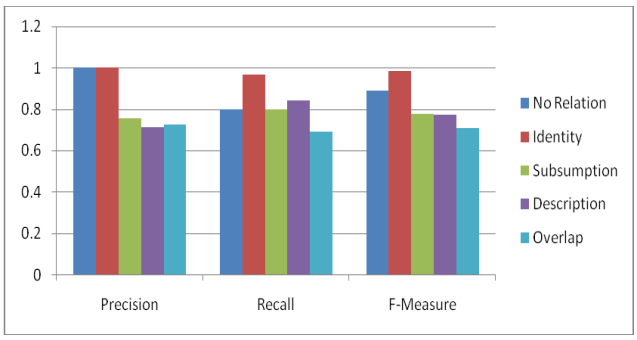

Fig. 5. Performance of NN classification.

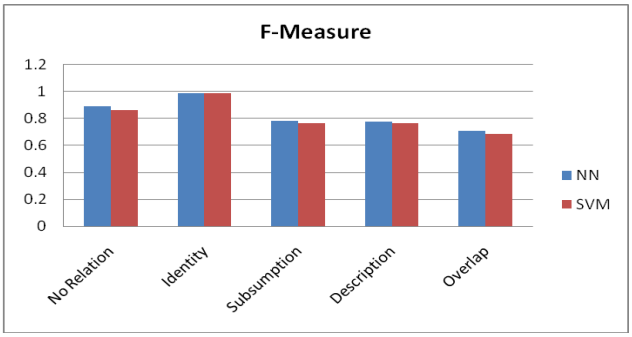

Fig. 6. F-measure comparison between NN and SVM.

V. Conclusion

This work provides the study on multi-document relation (CST relation) identification between sentences in topically related documents. The task of identifying relations between sentences is very important and significant to the field of multi-document analysis. However many applications still rely on human experts to perform such task. Although past works have attempted to automate this task, the obtained results were however not convincing. In this paper, we proposed a classification model based on supervised learning using feed-forward neural network (NN) to identify the relations between sentences. Four types of relations have been considered, namely *Identity*, *Overlap*, *Subsumption*, and *Description*. The multilayer network learns the features from the training examples, which represent the relationship type of each sentence pair. Tests were also carried out to determine the number of hidden nodes that best fit the network model. We experimented using the dataset obtained from CSTBank which comprises human annotated CST relations. Experimental results show that NN obtained a general accuracy of 80.09% and outperformed the SVM classifier which obtained 78.64% accuracy. Apart from improving the classification accuracy, we are currently working on how the identified relations can facilitate task related to multi document summarization. We regard this as our future work.

Acknowledgment

This research is supported by the Ministry of Higher Education (MOHE) and Universiti Teknologi Malaysia (UTM) under the Research University Grant.

References

[1] D.R. Radev, "A Common Theory of Information Fusion from Multiple Text Sources Step One: Cross-Document Structure," In Proceedings of the 1st ACL SIGDIAL Workshop on Discourse and Dialogue, 10, 2000, pp. 74-83.

[2] Z. Zhang, S. Blair-Goldensohn, and D.R. Radev, "Towards CST-Enhanced Summarization," In Proceedings of the 18th National Conference on Artificial Intelligence, 2002, pp. 439-446.

[3] M.L.C. Jorge, and T.S. Pardo, "Experiments with CST-based Multidocument Summarization," Workshop on Graph-based Methods for Natural Language Processing, ACL, 2010, pp. 74–82.

[4] Z. Zhang, J. Otterbacher, and D.R. Radev, "Learning cross-document structural relationships using boosting," In Proceedings of the 12th International Conference on Information and Knowledge Management, 2003, pp. 124-130.

[5] Z. Zhang, and D.R. Radev, "Combining Labeled and Unlabeled Data for Learning Cross-Document Structural Relationships," In Proceedings of IJCNLP, 2004, pp. 32-41.

[6] Y. Miyabe, H. Takamura, and M. Okumura, "Identifying cross-document relations between sentences," In Proceedings of the 3rd International Joint Conference on Natural Language Processing, 2008, pp. 141–148.

[7] E.G. Maziero, and T.A.S. Pardo, "Automatic Identification of Multi-document Relations," In: PROPOR 2012 PhD and MSc/MA Dissertation Contest, 2012, pp. 1-8.

[8] S.S. Haykin, Neural Networks: A Comprehensive Foundation, Macmillan, New York, 1994.

[9] Y. Chauvin and D.E. Rumelhart, Eds., Backpropagation: Theory, Architectures, and Applications, Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1995.

[10] D.R. Radev, and J. Otterbacher, CSTBank Phase I, 2003. http://tangra.si.umich.edu/clair/CSTBank/phase1.htm