Machine Learning based Paraphrase Identification System using Lexical Syntactic Features

Rutal S. Mahajan[1] Mukesh A. Zaveri[2]
Computer Engineering Department
S.V. National Institute of technology, Surat, India
[1]rutal.mahajan@gmail.com, [2]mazaveri@coed.svnit.ac.in

Abstract—**During the natural language communication, meaning understanding is the complex task that humans learn from their childhood but to automate this process of meaning understanding for computers has great real world applications. Simple text processing tasks are not enough to uncover the meaning from given unstructured natural language text. Our current research focuses on the issues pertaining to the same. Paraphrase identification is such important task of identifying the meaning similarity between two text segments in natural language understanding system. Proposed a machine learning system uses lexical features and dependency based features for sentence level paraphrase identification. The performance of proposed system is evaluated by conducting experiment on standard Microsoft paraphrase corpus. Moreover, a comparative study of current system with other machine learning based systems on Microsoft paraphrase corpus for paraphrase identification is carried out. The proposed system achieves competitive results compare to other state-of-the art machine learning systems by using simple linguistic features. The system using SVM classifier achieves 81.41% f-score by using simple lexical features only. Voting based classifier scores 80.97% with lexical features. Results with dependency features are highly sensitive to minor syntactic change.**

Keywords— Paraphrase identification, machine learning, support vector machine, naïve bayes, neural network, lexical features, syntactic features, dependency based features.

I. Introduction

Human being are capable of learning and performing complex meaning identification tasks from their early childhood but to automate this process of meaning understanding for computers and machine has great real world applications. When one sentence or concept in a natural language can be represented uniquely at various linguistic levels like lexical, syntactic, semantic, pragmatic or its combination, it is called paraphrasing [1]. In literature, various tasks on paraphrase are categorized into: paraphrase generation, paraphrase extraction and paraphrase identification. Paraphrase identification is the one of the important task of natural language understanding systems (NLU) that can be useful to improve many other related tasks such as machine translation evaluation [2], text summarization, question answering system , plagiarism detection and many more where we need to work with meaning of text [3]. The QA system given in [4], uses paraphrase identification to increase the chances of finding best answer to the user question. Paraphrase identification system recognizes the sentences which are paraphrases of others sentences to detect the plagiarism [3].

The reminder of the paper is structured as: Section II provides the background of earlier work in machine learning approaches for paraphrase identification. Section III describes our paraphrase identification system, including details on features used. Section IV describes details about set up of experiments, datasets and evaluation measures. In section V, results and error analysis is done which is followed by conclusion in Section VI.

II. Related Work

In literature, various solution approaches proposed to paraphrase identification task like rule-based, graph based, logical inference based and machine learning. Among those, the machine learning approaches to automatically learn training samples for paraphrase identification task is explored here. Machine learning based systems use different types of features such as lexical match features[1] such as word overlap, n-gram match, WordNet based similarity features. Microsoft paraphrase (MSR) corpus developed by [5] in 2004 is considered as the standard corpus for testing paraphrase identification system in English. Systems[6] [2] have used machine translation evaluation metrics for finding out similarity between two sentences. In [6], different machine translation evaluation metrics such as word error rate (WER), BLEU score and NIST score are used to identify paraphrase. Classifier used for learning paraphrase identification is SVM. They have tokenized and POS tagged sentences in preprocessing stage, while stemming is performed on noun and verbs. In[2], 8 different machine translation evaluation features have been used for identifying paraphrase like BLEU score, NIST score, METEOR, SEPIA,BADGER, MAXSIM,TER and TER. Authors in [2] have conducted experiments on MSR paraphrase corpus as well as on paraphrase detection corpus with classifier combination approach of three classifier and achieved 84.1% f-score for MSR paraphrase corpus. Simple distributional semantic space by [7] has been achieved good f-score 82.3% for paraphrase identification on MSR paraphrase corpus with different compositionality operators. Deep learning based systems [8] [9] and [10] also have better performance. In [9], linguistically motivated word embeddings

978-1-5090-0612-0/16/$31.00 ©2016 IEEE

are used with deep compositional distributional model. It addresses the problem of lexical ambiguity by introducing fix n senses per word. Word representations and compositional models have been pre-trained on British National Corpus (BNC) of 6 million sentences. The systems[10] and [9] have been used different variants of deep neural networks such as convolution neural networks and recursive neural networks respectively. The bird eye view of different paraphrase identification approaches and their results can be found on Wikipedia which are updated frequently with description provided [11].

As semantics of text is important to identify meaning of text, various systems for finding semantic text similarity also have been studied for paraphrasing task. While working with semantic of text, mainly local information, i.e. meaning of lexical units, and structural information, i.e. syntactic and semantic structure, should be handled [12]. In [12] structural alignment feature has been used to identify paraphrase, which make use of syntactic information as well as the semantic. In [13] various syntactic features for handling textual entailment are discussed, which can be useful in paraphrase identification task. As paraphrase is a bidirectional relation, all features must capture sentence similarity in both direction [1]. As per our literature survey, features to identify the similarity between two sentences plays vital role in any system, which have been used to identify paraphrase occurring at different linguistic level. After analyzing existing systems, features can be classified into mainly two categories: statistical features and linguistic features. In proposed system, combination of different features with various classifiers is experimented to check their effectiveness on paraphrase identification task.

III. Paraphrase Identification System

Paraphrase can be defined as "two sentences expressed in different words having same meaning". Same meaning of sentence is expressed in another sentence using different words. Paraphrase identification is a task of classifying given pair of sentences as paraphrase (P) or not paraphrase (NP). This paraphrase identification system consists of four modules namely, pre-processing, feature extraction, machine learning and evaluation. To evaluate the system, standard Microsoft Paraphrase corpus [5] is used as dataset. Various external resources such as, WordNet, Stanford parser and weka . Fig.1. shows the overall architecture of paraphrase identification system.

A. Pre-processing Module

Pre-processing module takes sentence pair as input and gives part-of-speech tagged output sentences. In this stage name entities are also tagged in both pair of sentences using Stanford NEtagger. Similarly Stanford Parse is used for extracting dependency based features of both the sentences. Tokenization, stemming and stop word removal are also carried out as a pre processing step.

B. Feature Extraction Module

Feature extraction is the most important module of the machine learning based classification system. In this module, features are extracted from the datasets in a format supported by machine learning algorithm. The performance of any machine learning classifier is highly dependent on set of features chosen. In this work, total 11 features are used for paraphrase identification, which can be categorized into two types: lexical features and syntactic or dependency based features. First six features are lexical features, next five are syntactic or dependency based features and last is semantic feature. Dependency based features are not just dependency label comparison. It also incorporates the structural alignment as in[12], where matching is possible only when two dependency arcs have same dependency labels.

In this system, score for all the lexical features are calculated using Jaccard similarity. It can be defined for two sentences S_1 and S_2 using following equation:

$$Jaccard\ \_\ Similarity\ (S_1, S_2) = \frac{S_1 \cap S_2}{S_1 \cup S_2} \qquad (1)$$

It is the ratio of total numbers of unigrams or bigrams matched in sentences S_1 and S_2 and total numbers of unique unigrams or bigrams.

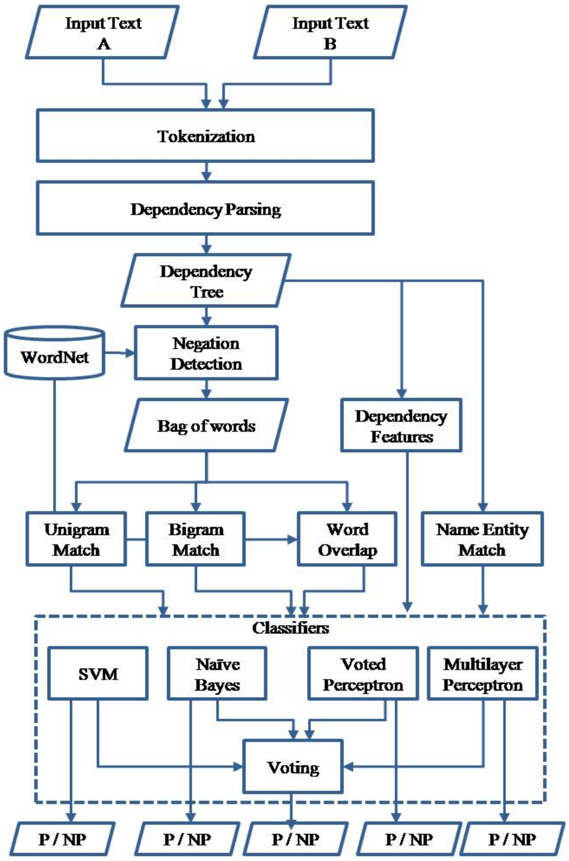

Fig. 1. System architecture of paraphrase identification system.

1) Word Overlap: This feature takes the ratio of overlapping words between two sentences to the all the words (non repeating) in sentence 1 and sentence 2 after removal of stop words. Each word in sentneces are also expanded with its synonym from WordNet synset and then matched with the words of other sentence. common name entities contributes well in paraphrasing. For example, in Table I sentence pair from MSR paraphrase corpus [5] is given for which word overlap is counted.

TABLE I. WORD OVERLAP

	Sentences	Score
Sentence 1	She first went to a specialist for initial tests last Monday, feeling tired and unwell.	$= \frac{5}{16}$ $= 0.312$
Sentence 2	The star, who plays schoolgirl Nina Tucker in Neighbours, went to a specialist on June 30 feeling tired and unwell.	

2) Unigram match: Various unigrams in sentence 1 is matched with its presence in sentence 2. Score of matching is calculated using Jaccard Similarity equation as given in (1). Similarly stemming is applied on unigrams of both sentences, prior to the matching.

3) Bi-gram matching feature: Each bi-gram in Sentence 1 is searched for a match in Sentence 2. Similarly stemming is applied on it prior to matching. Bi-gram matching score is calculated here by (1).

4) Negation detection and replacement: Each word in sentence 1 and sentence 2 from [5] preceded by negation token is labeled as negated word. Negation tokens are not, n't, nor, null, neither, never, nothin,without and so on. Negated words are used to model dissimilarity between negated and non-negated words. Stanfor dependency parser also detects negation word and its modifier. If match is found between negation modifier word (or its synonym from wordnet) and word in other sentence; it is marked as 0. For example, in Sentence 1 of Table II , neg (didn't, detail) are identified as negative modifier.

TABLE II. NEGATION DETECTION

	Sentences
Sentence 1	The company **didn't detail** the costs of the replacement and repairs.
Sentence 2	But company officials expect the costs of the replacement work to run into the millions of dollars.

5) Subject-Subject comparision feature: When subject in sentence 1 and dependency label nsub or nsubpass match togather with same in sentence 2. E.g. sentence 1: X is driving a car. Sentence 2: A car is driving X. Here X as nsubj in sentence1 should be matched with X as nsubj or nsubjpass in sentence2. Here it will not match, as in sentence 2 x is as dobj of drive.

6) Subject-verb comparision feature: When subject and verb in sentence1 togather with their dependency label matched with subject and verb in sentence2 togather with their dependecy label, match is considered. If subjects in both sentences matched but verbs do not match then, the wordnet distance of both the verbs are calculated [13]. If it is less than 0.5 then match is considered and score is assigned.

7) Object-verb comparision feature: When object and verb in sentence1 togather with their dependency label matched with oubject and verb in sentence2 togather with their dependecy label, match is considered. If objects in both sentences matched but verbs do not match then, the wordnet distance of both verbs are calculated[13]. If it is less than 0.5 then match is considered and score is assigned.

8) Noun comparision feature: The system compares noun in sentence 1 with noun in sentence 2 by depepndency relation nn. If match found, score is assigned.

9) Name Entity match: The system compares name entity in sentence 1 with name entity in sentence 2 using stanfor name entity recogniser. If match found, score is assigned.

C. Machine Learnig Module

This module composed of different numbers of classifiers. We used WEKA for implementing different classifiers for paraphrase identification task. Based on processing time and performance on training data four machine learning classifiers are used in this work: Support Vector Machine, Naïve Bayes Classifier, Voted Perceptron and multilayer Perceptron neural network.

Support Vector Machine SVM are known to work best with binary classification problem. Among different kernels of SVM, linear kernel is used here with default parameter setting. Naïve Bayes is the simplest classifier which assumes the independence of attributes. Due to its faster processing of data and simplest nature we have used it here. Another two classifiers are voted perceptron and multilayer perceptron neural networks, which are online algorithms. So they are useful for working with streaming data. These algorithms also minimizes the error from misclassified points.

IV. EXPERIMENTAL SETUP AND EVALUATION

The goal of this research is to build the paraphrase identification system using simple machine learning algorithms and different set of features such that it yields competitive results compare to state of the art systems. Thus system performance is evaluated by conducting experiments on the standard Microsoft paraphrase corpus developed and provided by Micorsoft [5]. It consists total 5801 sentences, which are divided into training and testing data in the form of pair of texts sentences and its class value. The class values are given as paraphrase (P) or not paraphrase (NP). Training data includes total 4076 sentence pairs, 2753 of those are positive (67.5%). Test data consists of total 1725 sentence pairs among which 1147 are positive (66.5%).

To examine the effect of different features, we have experimented with different machine learning algorithms including Support Vector Machine (SVM), Naïve Bayes (NB), Multilayer Perceptron (MLP) and Voted Perceptron (VP) using WEKA tool. The effect of using combination of classifiers with given all features is studied using voting of each classifier for ensemble as in [1]. In this experimental setup, generated outputs of SVM, NB, VP and MLP are examined. In that, sentence pairs whose classes are found common by majority of classifiers are assigned majority class. For the sentence pairs where two classifiers disagree, the class of classifier with highest performance is adopted.

For evaluation purpose traditional evaluation measures precision, recall and f-measure is used here. Precision can be given by (3) as follow. Higher precision values indicate fewer occurrences of False Positive (FP) values.

$$precision = \frac{TP}{TP + FP} \tag{3}$$

Recall can be given by (4) as follow. Higher recall values indicate fewer occurrences of False Negative (FN) values. It is also known as True Positive Rate (TPR).

$$recall = \frac{TP}{TP + FN} \tag{4}$$

Precision and recall are not independent measures. System recall can be easily increased by labeling class values at the cost of precision and vice versa. Thus F-score is used which considers both the precision and recall to balance. It can be given by (5) as follow.

$$f - score = 2 * \frac{precision * recall}{precision + recall} \tag{5}$$

V. RESULTS AND ERROR ANALYSIS

Table III presents the results of system with individual classifier on set of different linguistic features. The system performance is observed clearly better with combination of all features for all classifiers. As paraphrase occurs at different linguistic levels, combination of features can be useful to identify paraphrases efficiently than single type of feature.

SVM performs the best in terms of recall as well as f-measure. As its recall is 1 no false positive is noted in this experiment. But lesser precision indicates the possibility of higher false positive values. This error analysis is done in table V. All classifiers perform better than other state of the art machine learning systems even with only lexical features. f-score of SVM , VP and MLP are 81.41% , 81.12% and 81.13% respectively, which are competitive with [2], [7]–[9].

Table IV presents the results of voting based classifier with different linguistic features. Voting is performed by SVM, Naïve Bayes, Multilayer Perceptron and Voted Perceptron classifiers.

After conducting experiments on different settings, we have performed error analysis on misclassification of sentence pairs that our system made on test dataset. Here error analysis is done for combination of features setting for all the classifiers in individual and in combination. Table V contains data for error analysis for each classifier and their combination.

TABLE III. PARAPHRASE IDENTIFICATION RESULTS WITH LEXICAL AND SYNTACTIC FEATURES ON INDIVIDUAL CLASSIFIERS

Evaluation Measure	***Feature type***	**Classifiers**			
		SVM	***NB***	***VP***	***MLP***
Precision	Lexical	75.57	80.81	74.4	74.78
	Syntactic	66.49	72.07	70.68	69.41
	All	66.49	77.11	69.86	69.39
Recall	Lexical	88.23	60.59	89.18	88.66
	Syntactic	100	77.16	91.63	94.16
	All	100	65.21	90.15	88.14
F- Measure	Lexical	**81.41**	69.25	**81.12**	**81.13**
	Syntactic	79.87	74.53	79.80	**79.91**
	All	**79.87**	70.67	78.72	77.65

TABLE IV. PARAPHRASE IDENTIFICATION RESULTS WITH COMBINED FEATURES ON COMBINATION OF CLASSIFIERS

Evaluation Measure	**SVM+ NB+VP+ MLP (Voting based)**		
	Lexical	Syntactic	All
Precision	75.59	70.13	69.89
Recall	87.18	92.50	90.67
F-Measure	**80.97**	79.77	78.94

TABLE V. ERROR ANALYSIS OF DIFFERENT CLASSIFIERS

Classifier	**TP**	**TN**	**FP**	**FN**
SVM	1147	0	578	0
Naïve Bayes	748	356	222	399
Voted Perceptron	1034	132	446	113
Multilayer Perceptron	1011	132	446	136
SVM+ NB + VP + MP (Voting based)	1040	130	448	107

SVM has given best performance at predicting positive class values, i.e., here “paraphrase (P) class. But predicted 0 negative class values, i.e. not paraphrase (NP) correctly. All the test data values are predicted to P class by SVM. So it has misclassified maximum false positives. Voted Perceptron and multilayer Perceptron performed well for correctly classifying positive values. From 1147 true positive value they have classified 1034 and 1011 respectively. Voted Perceptron has misclassified 113 cases and multilayer Perceptron has misclassified 136 cases as false negative, which should be classified as positive. Moreover from observation it is seen that Naïve Bayes classifier performed better in correct classification of negative class values.

Few examples from experiment for false positive and false negative are identified in result and listed down in Table VI. Here the examples for class P and class NP denotes false positive and false negative respectively. To identify paraphrase for these sentence pairs, deeper level of features need to be used.

TABLE VI. FALSE NEGATIVES AND FLASE POSITIVES

pID	Sentence 1	Sentence 2	Class
4	A tropical storm rapidly developed in the Gulf of Mexico Sunday and was expected to hit somewhere along the Texas or Louisiana coasts by Monday night.	A tropical storm rapidly developed in the Gulf of Mexico on Sunday and could have hurricane-force winds when it hits land somewhere along the Louisiana coast Monday night.	NP
243	Wal-Mart estimates more than 100 million Americans visit their stores every week.	Each week 138 million shoppers visit Wal-Mart's 4,750 stores.	P
1429	Kodak expects earnings of 5 cents to 25 cents a share in the quarter.	Analysts surveyed by Thomson First Call had expected Kodak to earn 68 cents a share for the quarter.	P
1695	Bush turned out a statement yesterday thanking the commission for its work, and said, "Our journey into space will go on."	Mr. Bush did not discuss this when he issued a brief statement yesterday thanking the commission for its work, and saying, "Our journey into space will go on."	NP

VI. CONCLUSION

In this work we have proposed paraphrase identification system using machine learning classifiers with only 9 different linguistic features and achieve the competitive results with the state of-the-art systems. Paraphrase occurs at different linguistic level so only set of single type of linguistic feature cannot be very useful in paraphrase identification. We have used structural alignment in syntactic features instead of simple matching. As the MSR corpus was built using key word based search of news events on web, it highly supports lexical matching. Thus using lexical features on this corpus showed good results. Experimental results have cleared our way to explore the work for features capturing loose as well as precise paraphrase in MSR corpus for identifying meaning of the text. Selection of better syntactic features as per the MSR corpus design is also required the attention.

REFERENCES

[1] Z. Kozareva and A. Montoyo, "Paraphrase Identification on the Basis of Supervised Machine Learning Techniques," in *Proceedings of 5th International Conference on NLP, FinTAL Turku, Finland, August 23-25*, 2006, pp. 524–533.

[2] N. Madnani, J. Tetreault, and M. Chodorow, "Re-examining Machine Translation Metrics for Paraphrase Identification," in *Proceedings of 2012 Conference of the North American Chapter of the Association for Computational Linguistics NAACL*,2012, pp. 182–190.

[3] M. Potthast, B. Stein, A. Barrón-Cedeño, and P. Rosso, "An Evaluation Framework for Plagiarism Detection," in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, 2010, pp. 997–1005.

[4] F. Rinaldi, J. Dowdall, K. Kaljurand, M. Hess, and D. Mollá, "Exploiting paraphrases in a Question Answering system," in *Proceedings of Second International Workshop on Paraphrasing*, 2003, pp. 25–32.

[5] B. Dolan, C. Quirk, and C. Brockett, "Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources," in *Proceedings of the 20th International Conference on Computational Linguistics*, 2004.

[6] A. Finch, Y.-S. Hwang, and E. Sumita, "Using Machine Translation Evaluation Techniques to Determine Sentence-level Semantic Equivalence," in *Proceedings of the Third International Workshop on Paraphrasing*, 2004, pp. 17–24.

[7] W. Blacoe and M. Lapata, "A Comparison of Vector-based Representations for Semantic Composition," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2012, pp. 546–556.

[8] R. Socher, E. H. Huang, J. Pennington, A. Y. Ng, and C. D. Manning, "Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection," in *Proceedings of the 24th International Conference on Neural Information Processing Systems*, 2011, pp. 801–809.

[9] J. Cheng and D. Kartsaklis, "Syntax-Aware Multi-Sense Word Embeddings for Deep Compositional Models of Meaning," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, no. 17–21 September, pp. 1531–1542.

[10] H. He, K. Gimpel, and J. Lin, "Multi-Perspective Sentence Similarity Modeling with Convolutional Neural Networks," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 17–21 September, 2015, pp. 1576–1586.

[11] "Paraphrase Identification (State of the art)," *ACL wiki*. [Online]. Available: https://aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art). [Accessed: 06-Dec-2016].

[12] C. Liang, P. Paritosh, V. Rajendran, and K. D. Forbus, "Learning Paraphrase Identification with Structural Alignment," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence Learning*, 2016, pp. 2859–2865.

[13] P. Pakray, S. Bandyopadhyay, and A. Gelbukh, "A Hybrid Textual Entailment System using Lexical and Syntactic Features," in *proceedings of 9th IEEE International Conference on Cognitive Informatics (ICCI'10)*, 2010, pp. 291–296.