Research Journal of Applied Sciences, Engineering and Technology 4(22): 4894-4904, 2012
ISSN: 2040-7467
© Maxwell Scientific Organization, 2012
Submitted: May 22, 2012 Accepted: June 08, 2012 Published: November 15, 2012

Paraphrase Identification using Semantic Heuristic Features

[1]Zia Ul-Qayyum and [2]Wasif Altaf
[1]Department of Computing and Technology, IQRA University, Islamabad, Pakistan
[2]Department of Computer Science and Engineering, University of Engineering and Technology, Lahore, Pakistan

Abstract: Paraphrase Identification (PI) problem is to classify that whether or not two sentences are close enough in meaning to be termed as paraphrases. PI is an important research dimension with practical applications in Information Extraction (IE), Machine Translation, Information Retrieval, Automatic Identification of Copyright Infringement, Question Answering Systems and Intelligent Tutoring Systems, to name a few. This study presents a novel approach of paraphrase identification using semantic heuristic features envisaging improving the accuracy compared to state-of-the-art PI systems. Finally, a comprehensive critical analysis of misclassifications is carried out to provide insightful evidence about the proposed approach and the corpora used in the experiments.

Keywords: Natural language generation, natural language processing, paraphrase detection

INTRODUCTION

Language and speech processing, also referred to as Natural Language Processing (NLP) or Natural Language Understanding (NLU) is regarded as automation or mechanization of human languages. Humans use language in everyday life, in various forms, such as writing, reading, listening or speaking and is the most preferred mode of communication and interaction probably. Although there exist many approaches to NLP, but there are two main branches of NLP, namely Natural Language Analysis (NLA) and Natural Language Generation (NLG). NLA is mainly concerned with lexical, syntactic, semantic, pragmatic and morphological analysis of text. Lexical analysis is concerned with the study of lexemes and their relationships. Syntax puts text into a structure more convenient for semantic or literal meaning analysis. Moreover, semantics is the study and analysis of literal meanings of text. On the other hand, pragmatics is analysis of utterances or text with reference to context, while morphology is the study of how root words and affixes are composed to form words. As opposed to NLA, NLG is concerned mainly with generation of fluent/eloquent multi-sentential or multi-paragraph response in natural language (Dale, 2010).

The domain of NLP includes research challenges in multifarious dimensions like, semantics and pragmatics, NLG, textual entailment, knowledge representation of Quran, text summarization, sentiment analysis and paraphrasing. Each of the these research issue may comprise of sub-categories like paraphrasing has at least three categories, namely paraphrase generation, paraphrase acquisition and paraphrase identification. We envisage to address paraphrase identification problem specifically as it has potential applications in question answering, paraphragiarism detection and natural language generation.

Paraphrase identification: Before describing paraphrasing and its categories, we first look at definitions of term "paraphrase". Some definitions have been given in Table 1 and more can be found in Lintean *et al*. (2010). Most definitions include expressions such as, different words, own words, clearer or shorter way. So it is evident that, a wholesome concept of paraphrase is believed to maintain the same idea or semantic meaning in a clearer or often shorter way.

Paraphrasing can be done at various levels e.g., word, sentence, paragraph or discourse level. However, from NLP point of view, research issues related to paraphrasing are paraphrase generation, paraphrase acquisition and paraphrase identification. Paraphrase Generation (PG) is the task of automatically paraphrasing text at any of afore stated levels, as in Wubben *et al*. (2010). PG may also be enumerated as a task much related to NLG. Paraphrase acquisition or paraphrase extraction is the task of extracting paraphrases or candidate paraphrases from text corpora automatically, as in Bhagat *et al*. (2009). On the other hand, paraphrase identification is the task of classifying that whether two or more texts at any of the afore stated levels, are in paraphrase relationship or not. Paraphrase Recognition (PR) and Paraphrase Detection (PD) are the other terms used for Paraphrase Identification (PI).

Although PI is an active field of research and has possible applications in Information Extraction (IE), Machine Translation (MT), Information Retrieval (IR),

Corresponding Author: Zia Ul-Qayyum, Department of Computing and Technology, Iqra University, Islamabad, Pakistan

Table 1: Definitions of paraphrase identification

Definitions of paraphrase
• A *rewording* of something spoken or written, usually for the purpose of making its meaning *clear* (Michael, 1999).
• To express in a *shorter*; *clearer* or *different* way than someone has said or written (Longman, 2004).
• To express what someone else has said or written using *different* words especially in order to make it *shorter* or *clearer* (Macmillan, 2006).
• To repeat something written or spoken using *different* words, often in a humorous form or in a similar or *shorter* form that makes the original meaning *clearer* (Cambridge, Year).

Table 2: True and false paraphrase instances

Sentence ID	Sentence pair	Quality
1390995	The settling companies would also assign their possible claims against the underwriters to the investor plaintiffs, he added.	1
1391183	Under the agreement, the settling companies will also assign their potential claims against the underwriters to the investors, he added.	
1430402	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.	0
1430329	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.	

Automatic Identification of Copyright Infringement, Question Answering (QA), NLG, Modelling Language Perception in an Intelligent Agent (Nirenburg *et al*., 2008) and Intelligent Tutoring Systems (ITS). Still, to our knowledge, there is not any application using it practically, while Malakasiotis (2009) envisaged applying it in a semi-supervised environment.

To illustrate the PI task, consider the following simple pairs of sentences given in Table 2. It can be observed that pair of sentences {1390995, 1391183} is in paraphrase relationship with each other although both sentences have lexical differences but content delivered is same at a higher level. While sentence pair {1430402, 1430329} is not in paraphrase relationship because sentences differ in details (or perception) about the event. So, PI is a binary classification problem. In the following table, a true paraphrase pair has been assigned pair quality "1", while a false paraphrase pair has been assigned pair quality "0" and the same convention has been used throughout this study.

Although PI is an active field of research and has possible applications in Information Extraction (IE), Machine Translation (MT), Information Retrieval (IR), Automatic Identification of Copyright Infringement, Question Answering (QA), NLG, Modelling Language Perception in an Intelligent Agent (Nirenburg *et al*., 2008) and Intelligent Tutoring Systems (ITS). Still, to our knowledge, there is not any application using it practically, while Malakasiotis (2009) envisaged to apply it in a semi-supervised environment.

This study is aimed at developing a PI system using semantic heuristic features. It is envisaged that the proposed system will have improved paraphrase identification accuracy compared to other state of the art systems in this domain. This objective will be achieved by improving upon the pre-processing techniques being employed in such systems and by using an enhanced feature set. Moreover, a detailed misclassification analysis has been carried out to provide an insight into the syntactic structure of corpus causing misclassifications. The envisaged improvement in pre-processing phase is sought through comparative analysis of various PI systems based on cosine similarity measure while improvement in feature set is proposed by the introduction of enhanced text based features and different standard similarity measures.

LITERATURE REVIEW

Paraphrase identification has been approached previously by various lexical, syntactic, semantic and hybrid techniques. As usage of supervised machine learning has been common to most of the techniques, the following section, therefore presents supervised machine learning based PI techniques.

Supervised learning based approaches: Corley and Mihalcea (2005) used knowledge-based methods for measuring the semantic similarity of texts and showed that their approach performs significantly better than lexical matching techniques. They tokenized and POS tagged strings and then inserted words into their respective word class sets, verb, noun, adjective, adverb and cardinal for number entities. They used bag of words model for similarity analysis of respective word classes, where word-to-word semantic similarity was measured only for verbs and nouns, using WordNet Similarity package. While lexical matching was performed for adverbs, adjectives and cardinals. They used directional similarity of strings, where word specificity based on Inverse Document Frequency (IDF) which is inherently a corpus-based measure was used. Final similarity score was obtained as mean of similarity scores in both directions, as given by Eq. (1) and (2) below:

$$sim(T_i, T_j)_{T_i} = \frac{\sum_{pos}(\sum_{w_k \in \{W\,S_{pos}\}}(maxSim(w_k) * idf_{w_k}))}{\sum_{w_k \in \{T_{i_{pos}}\}} idf_{w_k}} \quad (1)$$

$$sim(T_i, T_j) = \frac{sim(T_i, T_j)_{T_i} + sim(T_i, T_j)_{T_j}}{2} \quad (2)$$

For evaluation purposes they used standard dataset MSRPC where a similarity threshold of 0.5 and supervised learning based on voted perceptron algorithm was used. They reported optimal accuracy of 0.715. They also reported that bag-of-words model ignores many important relationships present within a string.

Finch *et al*. (2005) used Word Error Rate (WER), Position-independent word Error Rate (PER), BLEU score, NIST score and POS enhanced PER, all based on Bag of words model, for learning paraphrase identification using SVM. The sentences were first tokenized and then POS tagged, while stemming was performed only on nouns and verbs where performance improvement of 0.8% was observed. They also employed modified edit distance which incorporated (Jiang and Conrath, 1997) semantic similarity measure which resulted in 0.6% of improvement. They achieved the highest accuracy of 0.7496 on MSRPC using the combination of all measures experimented with. Individually, best accuracy of 74.20 was obtained by POS measure, which in fact modified PER to learn weighting each word's grammatical role, where consideration of both similarities and dissimilarities between sentences were found to be more useful than taking into account only similarities or dissimilarities.

Brockett and Dolan (2005) employed SVM for recognition of paraphrases and corpus construction. They created feature vectors around string similarity features, co-occurrence of morphological variants, synonyms and hypernyms extracted from WordNet and word association pairs injection by semi-automatically creating a lexicon of possibly-synonymous word pairs. Authors reported precision and recall values of 86.76 and 86.39% respectively, on MSRPC.

Zhang and Patrick (2005) used an approach based on hypothesis that if two sentences are true paraphrases of each other they have more chances of transformation into same surface texts than a pair of false paraphrase sentences. Their approach was based on two steps, text canonicalization and supervised learning. Supervised machine learning step utilized decision tree learning and scores obtained from longest common substring of words, longest common subsequence, word edit distance and modified N-gram precision based on words to train the classifier. Through evaluation of nominalization and lexical similarity measures used in baseline system B2 it was found that true paraphrase pairs had higher lexical overlap than false paraphrase pairs. Best performance was achieved using passive to active transformation with accuracy of 0.719 and F-measure of 0.807 on MSRPC test set.

Kozareva and Montoyo (2006) created feature vectors around lexical and semantic similarity attributes to train SVM, k-Nearest Neighbour technique and Maximum Entropy classifiers. The features they used were common consecutive n-grams between two texts, skipgrams, longest common subsequence, cardinal number attribute, proper name attribute and semantic similarity information based on WordNet Similarity package, where all features used were bidirectional. They experimented in three different settings with all three classifiers stated above. First, word overlap information and word similarity information was employed in different feature sets. Higher performance accuracy was achieved with word overlap information than word similarity information and as well as baseline system (Corley and Mihalcea, 2005). Second, word overlap and word semantic similarity information in same feature set were employed, which resulted in significant increase of 1%. In last experimental setting majority voting scheme was implemented to obtain highest accuracy of 0.7664, on same features set as in second experimental setting. SVM was found to out-perform all classifiers in all experimental settings.

Qiu *et al*. (2006) employed a two phase approach to PR. In first phase information content shared by sentences was identified by a similarity detection and pairing module. In this study, information content was said to be information nuggets presented in form of predicate argument tuples. Sentences were first parsed using a syntactic parser and then fed into a semantic role labeller. A thesaurus based on corpus statistics was used for detection of predicate argument tuples similarity. Weighting was applied to targets as they carried more importance than arguments of predicate argument tuples. They showed 0.720 accuracy and 0.816 F-measure on MSRPC and manual annotation on a sample of 200 sentence pairs found agreement in 0.935 (187/200) cases.

Fernando (2007) applied and experimented with various NLP techniques, which had not been adapted for machine learning based solution to PI. Methods used for the said task mainly included, cosine similarity metric, semantic similarity matrix and comparison of dependency parse trees using tree kernel methods. Cosine similarity metric, a purely lexical metric used in Information Retrieval (IR) to determine similarity between a query and a document, was used for analyzing similarity between sentences in a sentence pair. Three different weighting schemes were used, a tf-idf weighting (*cosSimTFIDF*), second tf weighting (*cosSimTF*) and last with no weights (*cosSim*). Out of these variants best results were achieved with no weighting scheme. Best performance was achieved using semantic similarity matrix based on Lin WordNet Similarity measure, with accuracy of 0.738 and F-Measure 0.822. Simple Cosine similarity metric (*cosSim*) performed very well, resulting in accuracy of 0.727 and F-Measure 0.822.

Uribe (2008) proposed an approach based on tree alignment algorithm to maximize structural isomorphism in typed dependency parse trees of sentences in a sentence pair. Tree alignment algorithm was based on clause overlaps instead of word coupling. WordNet synonymy information was also utilized in node coupling process. Logistic Regression was used as learning model where four features were used, each accounting for one of the first four levels of each dependency tree. Uribe (2008) used a randomly selected 500 sentence pair subset of MSRPC for evaluation purposes, where stratified k-fold cross-validation was used. Researcher reported best precision and recall of 0.79 and 0.81, respectively.

Fernando and Stevenson (2008) used Semantic Similarity Matrix method used by Fernando (2007) for paraphrase identification. The only difference between their and Fernando (2007) was comparison of all word senses, opposed to only first word sense in Fernando (2007) while obtaining similarity scores using WordNet Similarity package. To constrain word-to-word similarity measures from resulting in spurious similarities a threshold of 0.8 was used for each knowledge-based similarity measure. Best performance accuracy of 0.741 with F-measure 0.824 was achieved using Jiang and Conrath WordNet similarity measure.

Malakasiotis (2009) used three methods for learning to identify paraphrases. First method, called INIT, applied nine standard string similarity measures to shallow abstractions of the sentences. Sentence abstractions included stemmed representations of sentences, tokens replaced by POS tags and tokens replaced by soundex codes and other such variations. Length ratio and Negation features were also used. INIT included total of 133 features. Second method named as INIT+WN used INIT and WordNet to treat synonyms as identical words resulting in higher lexical similarity. INIT+WN included total of 133 features. Third method, termed INIT+WN+DEP, used INIT+WN and dependency parser to obtain typed dependency parse trees, to calculate similarity of dependency relations at higher level. INIT+WN+DEP used 136 features in total. Maximum Entropy classifier was used to learn paraphrase identification. Best performance accuracy of 0.7617 was achieved using INIT+WN+DEP, with F-measure of 0.8288.

Dias and Smith (2004) used generative model that creates paraphrases of sentences and probabilistic inferencing to reason about whether or not two sentences have paraphrase relationship. Model applied used quasi-synchronous dependency grammars effectively incorporating syntax and lexical semantics. They also experimented with combination of their model with a complementary logistic regression model using product of experts. Highest performance accuracy of 0.8342, with 1.0000 precision and 0.9529 recall was achieved using oracle ensemble. It was the first system to meet human inter-annotator agreement level of MSRPC.

Uribe (2009a, b) followed where alignment and monotonicity analysis module and semantic heuristics were combined under feature-set definition module. Same preprocessing as (Uribe, 2009a) was performed. The only divergence from Uribe (2009a) was machine learning module for learning to identify paraphrases. Like Uribe (2009a) experimentation was performed on a 500 pair subset of MSRPC containing 68% of true paraphrase pairs. Best reported performance accuracy was 0.7360 and 0.7460 under monotonic and non-monotonic alignment respectively, using Logistic Regression Model as machine learning classifier. LRM was found to out-perform SVM against same feature set used.

Rajkumar and Chitra (2010) used combination of purely lexical, syntactic, lexical semantic and lexical-syntactic features to construct feature vector to train a Back Propagation Network for Paraphrase Recognition. Lexical-semantics based on modified string edit distance (lexical measure) computed using the Jiang and Conrath measure (semantic) were used. In lexical features skip-grams with skip distance k as 4 and adapted BLEU metric were used. Moreover dependency tree edit distance was employed to capture syntactic features. Parts of Speech enhanced Position Error Rate was used to detect semantic similarity where one feature for matches and another for non-matches

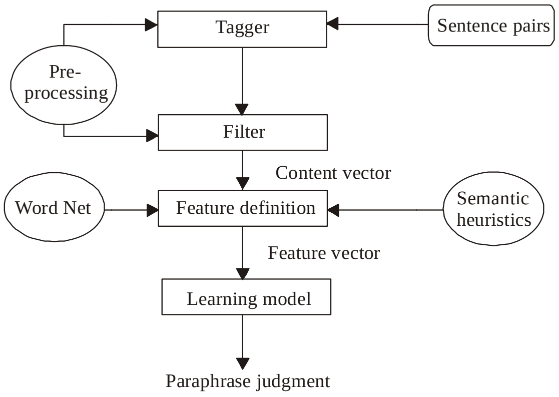

Fig. 1: Baseline system architecture

was included. Explicit negations were handled using binary negation attribute and implicit negations were handled by finding antonym relationships. The complex scenario of implicit and explicit negations altogether was reported as yet to be handled. The system being under implementation, so no performance measures were reported.

METHODOLOGY

Baseline system: The system proposed by Uribe (2009 b) which is based on Uribe (2009 a) is used as a baseline system for validation of our proposed approach (Fig. 1). The baseline system has analyzed mainly the syntactic patterns of MSRPC and argued that MSRPC might not be a syntactically rich dataset. In this section, implementation details of this system are presented.

In the pre-processing phase, sentence pairs were first Part of Speech (POS) tagged followed by removal of stop words using list of common words and as last pre-processing step filtration was performed on the basis of POS tags. POS tagging was performed using Stanford Log-linear Part Of Speech Tagger (Toutanova *et al.*, 2003). In this activity, the tokenization was implicitly performed by the POS tagger, therefore an explicit tokenization was not required. A small stop word list was used to remove commonly occurring closed-class words that carried insignificant or no meaning in sentences.

The stop word list applied is as follows:

$$stopWordList = \begin{Bmatrix} a, an, and, as, at, by, for, from, in, \\ is, it, its, of, on, that, the, to \end{Bmatrix}$$

where each lexical term *ti* is represented by a word and its POS category, given by:

$$s = \{t_1, t_2, \ldots, t_n\}$$

and POS*i* is constrained to the following condition:

$$POS_i \in \{Verb, Noun, Adjective, Adverb\}$$

After pre-processing step, the feature set definition module defines feature-set based on monotonic and non-monotonic alignments and uses semantic heuristics to define features for identification of false paraphrase pairs. Order preserving lexical coupling or alignment under monotonicity constraint was implemented as Longest Common Subsequence (LCS). LCS, which can be implemented on the basis of increasing order character or word commonality, was used in word setting and termed as LCS of words. Opposed to monotonic alignment, non-monotonic alignment relaxes the order preserving constraint so alignments can contain cross links instead of just increasing order lexical couplings. A simple bag-of-words approach was used to meet this requirement, where a pair of content vectors produces as result a pair of strings of aligned lexical terms with POS tags.

The second main feature of feature-set definition module was application of semantic heuristics to identify false paraphrase pairs. Since both LCS of words and bag of common words contain lexical terms as well as POS tags, this Boolean feature translates to a simple search problem. Formally, a set of common terms "c" is given by:

$$c = \{t_1, t_2, \ldots, t_i\}$$

$$\text{and } \nexists t_i Verb(t_i) \;:\; t_i = {}^{w_i}/_{POS_i}$$

Use of negation modifiers can account for a contradictory relationship between two or more sentences, which can play important role in identification of false paraphrase pairs. In baseline system, polarity or orientation of sentences has been analyzed by consideration of negation modifiers in POS tagged sentences, not the content vectors which are further pre-processed forms of original sentences.

Separate feature vectors were created for monotonic and non-monotonic alignment based experimentation. Features extracted for each sentence pair have been given in Table 3. Later, the 'Machine Learning phase' was carried out by using Weka (Witten and Frank, 2005; Hall *et al.*, 2009) which is a frequently used collection of machine learning algorithms for data mining tasks. Algorithms included

Table 3: Features included in feature vectors for baseline system

Monotonic alignment based feature-set	Non-monotonic alignment based feature-set
Cosine similarity for monotonically aligned sentence pair	Cosine similarity for non-monotonically aligned sentence pair
Overlap coefficient for monotonically aligned sentence pair	Overlap coefficient for non-monotonically aligned sentence pair
Event mismatch	Event mismatch
Polarity	Polarity
Verb or adjective antonym match	Verb or adjective antonym match

in Weka can be applied directly to a dataset or may be called from Java code. Weka also contains tools for data pre-processing, classification, clustering and visualization. As Logistic Regression Model (LRM) resulted in higher performance accuracy than Support Vector Machines (SVM) in Uribe (2009b) so logistic regression model was used for classification purposes. LRM has been implemented under stochastic gradient descent (weka.classifiers.functions.SGD) where loss function used was log loss (logistic regression). For evaluation purposes, MSRPC corpora was used in the experimentations.

The proposed framework: In this section, implementation details of proposed approach have been presented along with corpora used for its evaluation.

We have proposed, implemented and experimented with a variant system of baseline described above which included more features than baseline system, as shown in Table 8. The system architecture is presented in Fig. 2

As mentioned earlier, the baseline system took into account POS tags of lexical terms while considering the similarity of sentences. On the other hand, we also experimented with similarity of content vectors without considering POS tags. Monotonic alignment was not further experimented, as it resulted in performance lower than non-monotonic alignment. Whereas nonmonotonic alignment used was supported by synonymy information contained in WordNet, as opposed to baseline system Uribe (2009b) which used WordNet only for finding antonym relationships between lexemes.

Antonym detection was performed on lexical terms which had failed to align in bag-of-words based alignment. Which resulted in performance optimization, since finding antonym relationships between lexical terms of complete content vectors would have resulted in performance overhead of extra search of antonym relationships for aligned lexical terms. As opposed to baseline system, our approach did not use overlap coefficient; instead along with cosine similarity measure dice coefficient and length ratio measures were used. A total of 9 features were included in feature vectors, as presented in Table 4.

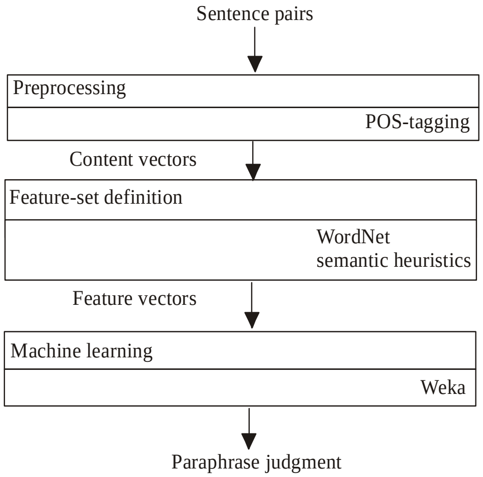

Fig. 2: System architecture for ParaDetect

Before discussing further the operational details of the proposed system, some terminologies are explained in the following paragraph.

Cosine similarity measure has previously been used by Fernando (2007) for PI, however we have used this measure to analyze pre-processing applied by baseline system Uribe (2009b) in a step-wise fashion. We have experimented with five slightly different systems developed for baseline pre-processing analysis. First, cosine similarity measure was applied on raw sentence pairs, called “cosSimRaw”. Secondly, cosine similarity measure was applied on tokenized string pairs, named “cosSimTok”. As tokenization was performed by POS tagger used, so “cosSimTok” used POS tags along with lexical terms of sentence pairs. Following that, cosine similarity measure was applied on tokenized string pairs and system was termed as “cosSimTokUnTagged”. This system differs from “cosSimTok” in absence of POS tags from sentence representations. In the “cosSimTokSwr” system, cosine similarity measure was applied on tokenized and stop word removed sentence pairs. Finally, cosine similarity measure was applied on completely pre-processed sentence pairs i.e., content vectors, called “cosSimPrep”. We produced these settings to compare results with “cosSim” Fernando (2007), the unweighted cosine similarity metric used for PI task. The

Table 4: Features included in feature vectors in proposed approach

ParaDetect features	
1	Cosine similarity extended to include synonymy information of non-monotonically aligned sentences, pre-processed with *cosSimTokUnTagged* pre-processing
2	Cosine similarity extended to include synonymy information of non-monotonically aligned sentences, pre-processed with *cosSimTok* pre-processing
3	Dice coefficient extended to include synonymy information of non-monotonically aligned sentences, pre-processed with *cosSimTokUnTagged* pre-processing
4	Dice coefficient extended to include synonymy information of non-monotonically aligned sentences, pre-processed with *cosSimTok* pre-processing
5	Event detection based on non-monotonic alignment of sentence pairs, pre-processed with *cosSimTok* pre-processing, with consideration of POS tags
6	Characters based length ratio of sentences, pre-processed with *cosSimTokUnTagged* pre-processing
7	Tokens based length ratio of sentences, pre-processed with *cosSimTokUnTagged* preprocessing
8	Polarity
9	Verb, adjective, noun or adverb antonym match applied to sentence dis-similarity

Table 5: Performance comparison of *ParaDetect* and various state-of-the-art systems using complete MSRPC training and test datasets

	Comparison of proposed approach with state of the art systems using MSRPC			
	Accuracy	Precision	Recall	F-Measure
Proposed approach	0.7467	0.7822	0.8578	0.8183
Zhang and Patrick (2005)	0.7190	0.7430	0.8820	0.8070
Finch (2005)	0.7496	0.7658	0.8980	0.8266
Kozareva and Montoyo (2006)	0.7664	0.9442	0.6876	0.7957
Fernando and Stevenson (2008)	0.7410	0.7520	0.9130	0.8240
Malakasiotis (2009)	0.7617	0.7935	0.8675	0.8288
Lintean and Rus (2010)	0.7206	0.7404	0.8928	0.8095

Table 6: Performance comparison of proposed approach and various systems using dataset A

	Dataset A			
	Accuracy	Precision	Recall	F-measure
Proposed approach	0.782	0.747	0.853	0.796
Cordeiro *et al*. (2007b)	0.782	--	--	0.809
Fernando (2007)	0.795	--	--	0.809

cosSimTokUnTagged system introduced here is very much similar to cosSim.

Preprocessing: The experimentation with cosine similarity based variant systems for baseline analysis ascertained that pre-processing in Urib (2009a, b) resulted in loss of performance as compared to "*cosSimTokUnTagged*", as presented in Table 8. So *ParaDetect* included preprocessing as that of "*cosSimTokUnTagged*", which opposed to baseline system pre-processing did not include stop word removal and POS constraint.

Corpora for paradetect evaluation: *ParaDetect* evaluation was performed on MSRPC training and test sets and {MSRPC and X1999 (For evaluation purposes, we used MSRPC and a dataset of 1999 false paraphrase sentence pairs provided by João Paulo C. Cordeiro called {X1999$^-$} in this study)}referred to as 'A' in this study and (Fernando, 2007). As original MSRPC contains 3900 true paraphrase pairs and 1901 false paraphrase pairs, inclusion of 1999 negative paraphrase pairs selected randomly from web news stories gave a balanced dataset of 3900 positive and 3900 negative examples, for experimentation.

RESULTS AND DISCUSSION

This section is divided into three main sub-sections to present results followed by a critical analysis of experimental findings and finally provide misclassification analysis. In the results section the results of experiments performed and a comparative diagrammatic and tabular analysis of those results has been presented. The analysis of misclassifications has been detailed later in this section. The results on MSRPC have been shown in Table 5 and Fig. 3 For comparison of results, performances achieved by some other state-of-the-art PI systems have also been reproduced and illustrated. Since *cosSimTokUnTagged* proved to be more useful in identifying paraphrases than *cosSimPrep*, so instead of using pre-processing done in *cosSimPrep*, we have pre-processing which was applied in *cosSimTokUnTagged* system, discussed in

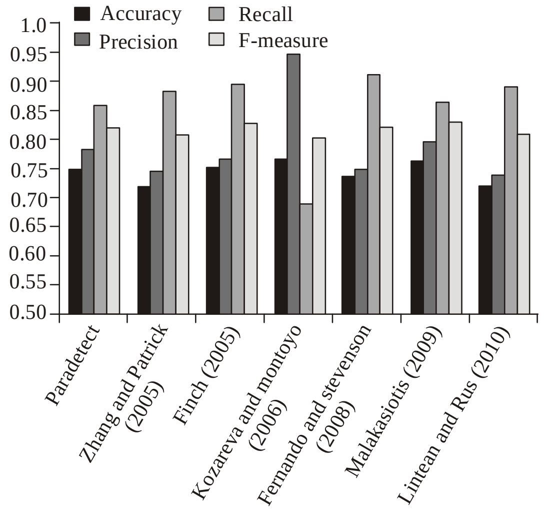

Fig. 3: Performance comparison of ParaDetect and various state-of-the-art system using Complete MSRPC training and test datasets

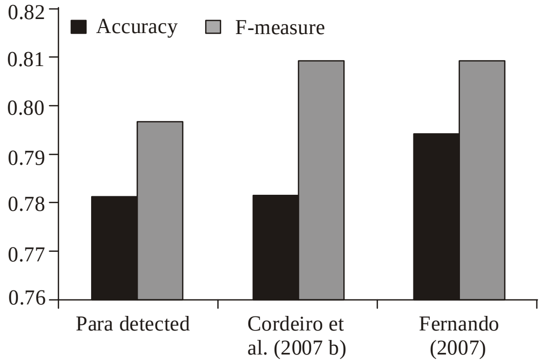

Fig. 4: Performance comparison of Para Detect and other system using dataset A

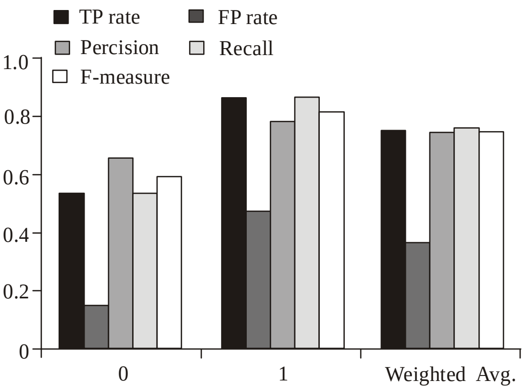

Fig. 5: Detailed accuracy results of ParaDetect

previous section. Our proposed approach performed significantly better than the baseline system. As compared to other state-of-the-art PI systems, the results of our proposed approach are fairly comparable when evaluated using complete MSRPC.

Moreover, in our approach, performance results obtained using dataset A with 10-fold cross validation have been produced in Table 6 and Fig. 4**.** Since Fernando (2007) and Cordeiro *et al*., (2007) have not produced precision and recall values, so these are missing in Table 8. Accuracy achieved by *ParaDetect* on dataset A is exactly equal to accuracy reported by Cordeiro *et al*., (2007) while f-measure is relatively lower. While, both accuracy and f-measure achieved by *ParaDetect* are relatively lower than accuracy and f-measure values reported by (Fernando, 2007).

Paradetect misclassification analysis: *Our proposed system* performs fairly well at identifying true paraphrase pairs, as given in Table 7 and Fig. 5 where 0.858 is the TP rate for true paraphrase pair class. On the other hand, TP rate for false paraphrase pair class is just 0.526 which is fairly low with respect to TP rate for true paraphrase pair class. We believe that this limitation of our approach is mainly due to higher lexical similarity in false paraphrase pairs, which makes them hard to be differentiated from true paraphrase pairs and existence of variety of relationships in false paraphrase pairs' class. On the other hand, misclassification in true paraphrase class is primarily due to lower lexical similarity in sentence pairs which have yet been given positive classifications in gold standard annotations. Table 8 shows "marginal" or "hard" instances, which might also be argued about their gold standard annotations. For example, sentence pair {1617861, 1617809} seems to be not quite a true paraphrase pair, yet the gold standard annotation classified this pair as true paraphrase pair. And sentence pair {229207, 229298} has been classified as false paraphrase pair even though it clearly seems to be a true paraphrase pair. Moreover, further examples also illustrate similar phenomenon.

As, aforesaid, the data set {X1999$^-$}includes 188 sentence pairs which are exact copy sentences and have been annotated as false paraphrase pairs, which does not coincide with definition of paraphrase used by MSRPC annotators. So, to consolidate this argument we trained *ParaDetect* using training part of MSRPC and applied {X1999$^-$} as test set, which resulted in 188 FP classifications, meaning that this 188 sentence pair subset contradicts the annotation guidelines followed by MSRPC annotators. Hence this 188 sentence pair subset should either be excluded from {X1999$^-$} or might not be used along with MSRPC.

Table 7: Detailed accuracy results of proposed approach

Predicted class	TP rate	FP rate	Precision	Recall	F-measure
0	0.526	0.142	0.651	0.526	0.582
1	0.858	0.474	0.782	0.858	0.818
Weighted avg.	0.747	0.363	0.738	0.747	0.739

Table 8: Sentence pairs showing some “marginal” instances of MSRPC, for misclassification analysis

	Sentence1-Id	Sentence 1
Pair quality	Sentence2- Id	Sentence 2
1	1617861	Shares of Coke were down 49 cents, or 1.1%, at $43.52 in early trading Friday on the New York Stock Exchange.
	1617809	In late morning trading, Coke shares were down 2 cents at $43.99 on the New York Stock Exchange.
0	229207	NBC probably will end the season as the second most popular network behind CBS, although it's first among the key 18-to- 49-year-old demographic.
	229298	NBC will probably end the season as the second most-popular network behind CBS, which is first among the key 18-to-49-year-old demographic.
1	872807	Freddie Mac shares were down more than 16 % at $50.18 at midday yesterday.
	872885	Freddie Mac shares were off $7.87, or 13.2%, at $52 in midday trading on the New York Stock Exchange.
1	621407	The findings were reported online in the June 1 edition of scientific journal Nature Medicine.
	621315	The findings are published in today's edition of the journal Nature Medicine.
1	452845	The broader Standard and Poor's 500 Index .SPX gained 3 points, or 0.39%, at 924.
	452902	The technology-laced Nasdaq Composite Index <.IXIC> rose 6 points, or 0.41%, to 1,498.
1	2433757	Prime Minister Junichiro Koizumi must be counting his lucky stars.
	2433838	Prime Minister Junichiro Koizumi has all but popped the champagne bottle.
1	2949437	The report was found Oct. 23, tucked inside an old three-ring binder not related to the investigation.
	2949407	The report was found last week tucked inside a training manual that belonged to Hicks.
0	2229419	The department's position threatens to alienate social conservatives, who have provided strong political support for Mr. Ashcroft and President Bush.
	2229908	The department's stance disappointed some abortion opponents, and it threatens to alienate social conservatives who have provided strong political support for Ashcroft and President Bush.
0	197853	The dollar's slide against the yen was curbed by wariness that Japanese authorities could intervene to stem the yen's rise.
	197784	Despite hefty losses against the euro, the dollar's slide versus the yen was curbed by wariness that Japanese authorities could intervene to stem the yen's rise.

CONCLUSION AND FUTURE WORK

In this study, a paraphrase identification approach is presented based on improved pre-processing and semantic heuristics based enhanced features set. The system produces comparable or even better results than the state of the art systems in this category. Another important part of work is misclassification analysis which not only resulted in highlighting advantages and disadvantages of semantic heuristics based features used in this study, but helped bring to light some criticisable annotations of sentence pairs included in benchmark corpus like MSRPC, as well. It was also shown that the dataset {X1999$^-$} might not be used along with MSRPC due to bias of 188 exactly same false paraphrase pairs contained in {X1999$^-$}.

Paraphrase identification is a binary classification problem and in reality this is too restrictive in terms of classification of everyday text in which “marginal” cases do exist. We envisage, as a part of future work, to introduce more classes to study paraphrase relationships. Moreover, we intend to incorporate a more holistic antonym detection module in the proposed framework to account for implicit and explicit negations altogether.

REFERENCES

Brockett, C. and B. Dolan, 2005. Support vector machines for paraphrase identification and corpus construction. Proceedings of the 3rd International Workshop on Paraphrasing, pp: 1-8.

Bhagat, R., E. Hovy and S. Patwardhan, 2009. Acquiring paraphrases from text corpora. Proceedings of the 5th international Conference on Knowledge Capture, ACM New York, USA, pp: 161-168.

Corley, C. and R. Mihalcea, 2005. Measuring the semantic similarity of texts. Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, Association for Computational Linguistics Stroudsburg, PA, USA, pp: 13-18.

Cordeiro, J, G. Dias and P. Brazdil, 2007. New functions for unsupervised asymmetrical paraphrase detection. J. Softw., 2(4): 12-23.

Dale, R., 2010. Handbook of Natural language Processing. Nitin, I. and J.D. Fred (Eds.), 2nd Edn., CRC Press, Boca Raton, pp: 678.

Dias, D. and N.A. Smith, 2009. Paraphrase identification as probabilistic quasi- synchronous recognition. ACL '09 Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, Singapore, pp: 468-476.

Fernando, S. and M. Stevenson, 2008. A semantic similarity approach to paraphrase detection. Proceedings of the Computational Linguistics UK (CLUK 2008) 11th Annual Research Colloquium.

Fernando, S., 2007. Paraphrase identification. M.Sc. Thesis, University of Sheffield, UK.

Finch, A. Y. S. Hwang and E. Sumita, 2005. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. Proceedings of the 3rd International Workshop on Paraphrasing, (IWP2005), pp: 17-24.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I.H. Witten, 2009. The Weka Data Mining Software: An Update; ACM SIGKDD Explorations Newsletter, ACM New York, USA, 11(1): 10-18.

Jiang, J.J. and D.W. Conrath, 1997. Semantic similarity based on corpus statistics and lexical taxonomy. Proceedings on International Conference on Research in Computational Linguistics.

Kozareva, Z. and A. Montoyo, 2006. Paraphrase identification on the basis of supervised machine learning techniques. Proceedings of Advances in Natural Language Processing: 5th International Conference on NLP, pp: 524-533.

Lintean, M., V. Rus and A. Graesser, 2010. Paraphrase identification using weighted dependencies and word semantics informatica. An. Int. J. Comp Inf., 34(1): 19-28.

Malakasiotis, P., 2009. Paraphrase recognition using machine learning to combine similarity measures. Proceedings of the ACL-IJCNLP 2009 Student Research Workshop Annual Meeting of Association for Computational Linguistics, Stroudsburg, PA, USA, pp: 27-35.

Nirenburg, S., M. McShane and S. Beale, 2008. Resolving paraphrases to support modeling language perception in an intelligent agent. Proceedings of the 2008 Conference on Semantics in Text Processing, Stroudsburg, PA, USA, pp: 179-192.

Uribe, D., 2008. Recognition of paraphrasing Pairs. CERMA '08 Proceedings of the 2008 Electronics, Robotics and Automotive Mechanics Conference: IEEE Computer Society Washington, DC, USA, pp: 50-55.

Uribe, D., 2009a. Monotonicity analysis for paraphrase Detection. CERMA '09 Proceedings of the 2009 Electronics, Robotics and Automotive Mechanics Conference: IEEE Computer Society Washington, pp: 82-87.

Uribe, D., 2009b. Effectively using monotonicity analysis for paraphrase identification. 2009 Eighth Mexican International Conference on Artificial Intelligence, pp: 108-113.

Qiu, L., M.Y. Kan and T.S. Chua, 2006. Paraphrase recognition via dissimilarity significance classification. Proceedings of Empirical Methods in Natural Language Processing, Association for Computational Linguistics Stroudsburg, PA, USA, pp: 18-26.

Rajkumar, A. and A. Chitra, 2010. Paraphrase recognition using neural network classification. Int. J. Comput. Appl., 1(29): 42-47.

Toutanova, K., D. Klein, C. Manning and Y. Singer, 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. NAACL '03 Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Association for Computational Linguistics Stroudsburg, PA, USA, pp: 173-180.

Witten, I.H. and E. Frank, 2005. Data Mining: Practical Machine Learning Tools and Techniques. 2nd Edn., Morgan Kaufmann Publishers, Academic Press, New York.

Wubben, S., A. Van den Bosch and E. Krahmer, 2010. Paraphrase generation as monolingual translation: data and evaluation. Proceedings of the 10th International Workshop on Natural Language Generation (INLG 2010), Association for Computational Linguistics Stroudsburg, PA, USA, pp: 203-207.

Zhang, Y and J. Patrick, 2005. Paraphrase identification by text canonicalization. Proceedings of the Australasian Language Technology Workshop, pp: 160-166.