# A Novel Approach for Developing Paraphrase Detection System using Machine Learning

Rudradityo Saha
Vellore Institute of Technology, Chennai,
Tamil Nadu, India - 600127

G. Bharadwaja Kumar
Professor
Vellore Institute of Technology, Chennai,
Tamil Nadu, India - 600127

## ABSTRACT
Plagiarism detection is difficult since there can be changes made to a sentence at several levels, namely, lexical, semantic, and syntactic level, to construct a paraphrased or plagiarized sentence posing as original. To identify cases of plagiarism and hence discourage the same, this paper presents a novel Supervised Machine Learning based Paraphrase Detection System developed by conducting experiments using Microsoft Research Paraphrase (MSRP) Corpus and assessed on the same. The proposed paraphrase detection system has achieved comparable performance with existing paraphrase detection systems. The major contributions of this paper are the utilization of a unique combination of lexical, semantic, and syntactic features, utilization of Shapley Additive Explanations (SHAP) Feature Importance Plots in XGBoost, and application of a soft voting classifier comprising of the top 3 performing standalone machine learning classifiers on the training dataset of MSRP Corpus. Another major contribution of the paper is the finding that applying data augmentation techniques degrades performance of machine learning classifiers.

## General Terms
Natural Language Processing

## Keywords
Natural Language Processing, Paraphrase Detection, Machine Learning, Classification, Supervised Learning

## 1. INTRODUCTION
The advent of Internet has made easy and instant access to vast amounts of information possible. The downside to this is that this has led to a significant increase in incidents of plagiarism. There are two main types of plagiarism namely, literal plagiarism where the source sentence is copied with only a few words modified or active voice is converted to passive voice or vice versa and, intelligent plagiarism where the source sentence is modified thoroughly with significant changes in sentence structure by means such as paraphrasing, translating or adopting idea [1].

Detecting plagiarism is a difficult task since there can be numerous major or minor changes made to a sentence at several levels, namely, lexical, semantic, and syntactic level, to construct a paraphrased or plagiarized sentence posing as original. To identify cases of plagiarism and hence discourage the same, a novel Supervised Machine Learning based Paraphrase Detection System has been developed.

## 2. LITERATURE REVIEW
An in-depth Literature Review is carried out to study prior work in the domains of paraphrase detection systems, and performance enhancement techniques for paraphrase detection which are detailed in the following sections.

### 2.1. Literature Review: Paraphrase Detection Systems
Research in paraphrase detection gained traction from 2005, attracting a large number of researchers using approaches namely, unsupervised learning, supervised machine learning, and supervised deep learning categorized in[2] which has resulted in a rich body of research. As this study is restricted to supervised machine learning paraphrase detection systems, this literature review is focused primarily on the same.

The authors in [3] have applied machine translation evaluation techniques namely, Bilingual Evaluation Understudy (BLEU), National Institute of Standards and Technology (NIST), Word Error Rate (WER), and Position Independent Word Error Rate (PER) as features which are used as input to support vector machine classifier to detect paraphrases in MSRP Corpus.

The authors in [4] have utilized lexical and semantic features to develop a machine learning based paraphrase identification method which is assessed using MSRP Corpus. Some lexical features are the ratio of common consecutive n-grams between two sentences and the total number of words in the two sentences, and Longest Common Subsequence. A noun-verb semantic similarity measure based on WordNet is used as a semantic feature. A voting classifier composed of three classifiers namely, Support Vector Machine, K-Nearest Neighbors, and Maximum Entropy is made use of for paraphrase detection. The paper concludes that the application of a voting classifier instead of individual classifiers leads to better performance in paraphrase detection. The paper suggests including syntactic features in the feature set to improve performance of suggested method.

A supervised two-phase framework proposed by authors in [5] detects paraphrases in MSRP Corpus by finding dissimilarities with the help of predicate argument tuples between the sentences in a sentence pair and then deciding whether the dissimilarities are significant or not. Predicate argument tuples are found out and labeled in a sentence pair using a syntactic parser and a semantic role labeler. The similarity of the sentence pair is measured by the extent of insignificance of unpaired predicate argument tuples. The method has achieved performance comparable to existing paraphrase detection methods.

An attempt to effectively filter out false paraphrases or irrelevant sentences generated by paraphrase generation methods to improve the quality of generated text is made by

authors in [6] by developing a paraphrase detection method which is assessed using MSRP Corpus. Lexical features such as n-gram based features, BLEU based features, and syntactic features such as dependency relationbased features in addition to simple features like absolute length difference between sentences in a sentence pair are utilized which are given as input to a support vector machine classifier for detecting paraphrases. The best performance is observed using all features except lemmatized unigrams. The study concludes that the suggested method has achieved comparable performance with the existing paraphrase detection methods. The authors suggest using more advanced weighted dependency-based features for further improvement of the suggested method.

The authors in [7] have utilized a generative model for paraphrase detection which constructs a paraphrase of a given sentence and uses probabilistic inference to decide whether a sentence pair is a paraphrastic pair or a non-paraphrastic pair. The method uses syntactic features and lexical features with the help of quasi-synchronous dependency grammars which is given as input to logistic regression classifier to perform paraphrase detection on MSRP Corpus.

The paraphrase detection method suggested in [8] utilizes two modules namely, feature set module and classifier module. The feature set module consists of similarity features based on monotonic and non-monotonic techniques and semantic features such as contradiction and polarity. The classifier module consists of support vector machine and logistic regression to perform supervised classification. The paper concludes that the suggested method has achieved comparable performance with existing paraphrase detection methods.

The authors in [9] have used a meta-classifier trained on features based on 8 machine translation metrics namely, BLEU, NIST, Translation Edit Rate (TER), Translation Edit Rate Plus (TERp), METEOR, SEPIA, BADGER and Maximum Similarity (MAXSIM) to develop a paraphrase detection method which is assessed using MSRP Corpus. The meta-classifier is composed of three standalone classifiers namely, logistic regression, sequential minimal optimization of a support vector machine, and a lazy, instance-based classifier that extends the nearest neighbor algorithm. The meta-classifier utilizes the average of the unweighted probability estimates from the constituent standalone classifiers as criteria for classifying a sentence pair into either a paraphrastic pair or a non-paraphrastic pair. The paper concludes that the suggested method has achieved comparable performance with existing paraphrase detection methods.

The paraphrase detection system proposed in [10] uses semantic heuristic features to enable the proposed system to perform better than the existing paraphrase identification systems. This is done by implementing better pre-processing techniques and a feature set containing additional features. A variant of the baseline system suggested in [8] is implemented which has more features than the baseline system. In the pre-processing phase, sentence pairs are Parts of Speech (POS) tagged, stop-words are removed using a custom stop-words list and then filtered based on POS tags. Features based on monotonic and non-monotonic alignments, and semantic features, namely Boolean features are utilized in the suggested method. The suggested system has achieved comparable performance with existing paraphrase detection systems on MSRP Corpus.

The authors in [11] have developed a support vector machinebased paraphrase recognition system with the help of various lexical, syntactic, and semantic features to detect paraphrases in MSRP Corpus. A feature selection technique namely, genetic algorithm is utilized to not only improve the accuracy of suggested paraphrase recognition system but to also attain similar performance with fewer features. Lexical features namely, Longest Common Subsequence, BLEU based features, and Skip-gram based features are utilized. Syntactic features such as Dependency Tree Edit Distance, Dependency Relation Overlap, and Parts of Speech Enhanced Position Independent Word Error Rate (POSPER) are employed. Semantic features like features based on WordNet are made use of to compute similarity between pair of nouns, verbs, adjectives, and adverbs between sentences in a sentence pair. The authors have been able to get a significant increase in performance of suggested system with fewer features through the application of genetic algorithm.

The authors in [12] have explored the usage of syntactic representations for learning relations between a set of two texts which can be sentences or paragraphs. They have constructed syntactic and semantic structures representing the text pairs and then applied graph and tree kernels to the structures for automatic creation of features which are then given as input to support vector machine for detecting paraphrases in MSRP Corpus.

The Supervised Machine Learning based Paraphrase Identifier developed in [13] has utilized a decision tree learning classifier to perform paraphrase identification on MSRP Corpus with the help of several lexical features namely, Longest Common Substring, Longest Common Subsequence, Edit Distance and Modified N-gram Precision. Various techniques of text normalization such as lemmatizing, conversion from passive to active voice, and replacement of named entities with generic tags have been initially employed on the sentence pairs of MSRP Corpus after which the normalized sentence pairs are given as input to decision tree classifier for paraphrase identification.

The authors in [14] have proposed a paraphrase identification system which makes use of a logistic regression classifier to identify paraphrases in MSRP Corpus with the help of numerous syntactic features based on tree edit sequences extracted by utilizing a tree kernel as a heuristic in a greedy search algorithm.

The author in [15] has developed a paraphrase recognizer by applying string similarity measures to abstractions of a sentence pair along with synonym detection via WordNet and dependency similarity measures. A subset of features through the use of a feature selection technique, are used as input to a Maximum Entropy classifier to recognize paraphrases in MSRP Corpus. The author has suggested incorporating additional features such as BLEU based features and word alignment-based features for better performance of suggested system.

The study carried out in [2] provides a comprehensive discussion on the recent research carried out on paraphrase detection methods and presents the comparative performance results of paraphrase detection methods grouped under categories namely, unsupervised learning, supervised machine learning, and supervised deep learning.

## 2.2. Literature Review: Performance Enhancement Techniques

Literature Review carried out to explore performance enhancement techniques for paraphrase detection methods are as follows:

The study carried out in [16] have explored the impact of commonly used preprocessing tasks such as removing stop-words, lowercasing and stemming in all possible combinations in two languages, namely Turkish and English, on two different domains, specifically news and e-mails for text classification by keeping in consideration aspects such as accuracy and dimension size. The study concludes that text preprocessing in text classification is as important as feature selection for improving performance of classifier, some text preprocessing tasks such as lowercasing improve classification performance regardless of domain and language, all possible combinations of preprocessing tasks must be tried for finding out the best combination, and removing stop words leads to decrease in performance of classifier.

The authors in [17] have presented Easy Data Augmentation (EDA) techniques namely, random insertion, random swap, random deletion, and synonym replacement for improving text classification performance by augmenting or adding textual data to the training dataset of a corpus. The authors have shown that EDA techniques enhance performance for deep learning classifiers such as convolutional neural networks for performing text classification.

# 3. RESEARCH METHODOLOGY

The proposed Supervised Machine Learning based Paraphrase Detection System consists of three modules namely, Text Preprocessing Module, Feature Engineering Module, and Classification Module. The proposed system is developed and assessed using MSRP Corpus.

## 3.1. Text Preprocessing Module

Various text preprocessing techniques are explored specifically, removing trailing and ending whitespaces, lowercasing, removing punctuation marks, converting accented characters to ASCII characters, removing special characters or substituting special characters with their literal counterpart such as '$' to "dollar", removing numbers, removing stop words from a custom stop words list, stemming or lemmatizing, and expanding contractions.
Techniques namely, removing trailing and ending whitespaces, lowercasing, removing punctuation marks, and converting accented characters to ASCII characters give the best performance and hence, utilized in the proposed system.

## 3.2. Feature Engineering Module

An overview of the features utilized by the proposed system is given below:

### *3.2.1 Lexical Features*

Features based on Fuzzy String Similarity Measures, N-gram based features, Skip-gram based features, BLEU based features, and other features namely Absolute Length Difference between sentences in a sentence pair, and Normalized Longest Common Subsequence are used.

Fuzzy String Similarity Measures consist of various string matching methods which make use of Levenshtein Distance similarity ratio. N-gram based features consist of several features which utilize ratios of length of intersection of sets of n-grams and total number of words or length of union of sets of n-grams with n being 1, 2, 3, and 4 for a sentence pair. Skip-gram based features comprise of features which consist of ratios of length of intersection of sets of k-skip-n-grams with degree as k and skip distance as n and total number of words or length of union of sets of skip-grams with n and k being 1 or 2 in all combinations for a sentence pair. BLEU based features consist of features which compute the cumulative n-gram score of degree n with n being 1, 2, 3, and 4 for a sentence pair. The 7th smoothing technique given in [18] is applied on each BLEU based feature so as to obtain better paraphrase detection performance.

### *3.2.2 Semantic Features*

A distance metric based feature derived from pre-trained Word2Vec Embeddings of Google News Corpus namely, Normalized Word Mover's Distance is utilized. The Word Mover's Distance is conceptualized in [19]. Features based on Distance Metrics applied on Sentence Embeddings of Universal Sentence Encoder [20] are also utilized. The various distance metrics applied on sentence embeddings are Minkowski Distance, Cosine Distance, Canberra Distance, City Block Distance, and Bray Curtis Distance, Euclidean Distance, and Jaccard Distance.

WordNet similarity based features are also employed. Some preparation steps are undertaken on the sentences in a sentence pair in order to compute similarity using a WordNet based Measure between the sentences in the sentence pair. The steps for the same are that each sentence in the sentence pair is tokenized and POS tagged, then filtered and converted into a set of terms containing only noun, verb, adjective, and adverb POS tags, and thereafter an appropriate synset is derived from each word using Lesk algorithm [21]. A synset is a synonym from a set of synonyms that share a common meaning. Each WordNet based similarity measure utilized as feature is normalized by dividing the obtained similarity score for a sentence pair by the highest similarity score possible for that similarity measure. Various WordNet based similarity measures categorized under four categories namely, path length based measures, information content based measures, feature based measures, and hybrid measures have been described and compared in [22] along with their advantages and disadvantages, and they opine that there is no best way to evaluate the performance of any WordNet based similarity measure but that the criteria for selecting an evaluation metric depends on the application.

### *3.2.3 Syntactic Features*

Features based on overlap of dependency relations between the sentences in a sentence pair are used. A dependency tree is created for each sentence in a sentence pair from which dependency relations are extracted. Each dependency relation constitutes a triple comprising of head, dependent, and relationship between them. The features based on precision and recall are formed by computing the ratios of unique shared triples between the sentences and all unique triples in the first and the second sentence respectively in a sentence pair.The feature based on F-measure is given in Eq. (1).

$$f = \frac{2 * p * r}{p + r} \quad (1)$$

Here f is Dependency Relation Overlap F-measure, p is

Dependency Relation Overlap Precision, and r is Dependency Relation Overlap Recall.

## 3.3. Classification Module

Various machine learning classifiers are applied to compare their performance in paraphrase detection, in accordance with directions for future work in [23]. A total of 9 machine learning classifiers are used which are XGBoost, Logistic Regression, Gradient Boosting, AdaBoost, Random Forest, K Nearest Neighbors, LightGBM, Support Vector Machine, and Baseline majority class classifier. A soft voting classifier is then constructed out of the top 3 performing standalone machine learning classifiers on the training dataset of MSRP Corpus based on their Cross-Validation Accuracy Scores, in order to improve classification performance. Each constituent classifier in the soft voting classifier provides a probability value for both class labels namely, non-paraphrastic pair (0) and paraphrastic pair (1) for a sentence pair. For each class label, the predictions given by each constituent classifier are averaged. The class label which is nearer to the max of two averages is then taken as the predicted class label for the sentence pair.

## 3.4. Assessment using MSRP Corpus

The performance of the proposed systemis assessed on MSRP Corpus[24][25] by conducting experiments on the same. Since, paraphrase detection is a binary classification problem, each sentence pair is assigned a value "1" or "0" as quality, with "1" indicating that the sentences in a sentence pair are paraphrases, and "0" indicating that the sentences in a sentence pair are not paraphrases.

## 3.5. Experiments

A total of 5 experiments are carried out using MSRP Corpus. In Experiments 1, 2, and 3, performance of various standalone classifiers and a soft voting classifier are assessed with the help of various lexical, semantic, and syntactic features respectively.In Experiment 4, a feature-set composed of a combination of selected lexical, semantic, and syntactic features are utilized to evaluate the performance of various standalone classifiers and a soft voting classifier. In Experiment 5, EDATechniques [17] are applied on the training dataset of MSRP Corpus in order to boost the performance of various standalone classifiers and a soft voting classifier using the feature-set utilized in Experiment 4.

From Experiments 1, 2, and 3, the features with zero contribution found with the help of SHAP [26] Feature Importance Plot in XGBoost are excluded from the feature-set utilized in Experiment 4. A SHAPFeature Importance Plot gives the importance values of various features in descending order present in the feature-set using the mean (|Tree SHAP|) values given by the Tree SHAP algorithm. The Tree SHAP algorithm utilizes tree-based classifiers such as RandomForest, XGBoost, and LightGBM,however, XGBoosthas been proven to exhibit better classification performance than other tree-based classifiers [27]. Hence, Tree SHAP algorithmutilizing XGBoost, is used to generate SHAPFeature Importance Plot. The SHAPFeature Importance Plot in XGBoost generated in Experiments 1, 2, and 3 are given in Figures 1, 2, and 3 respectively.

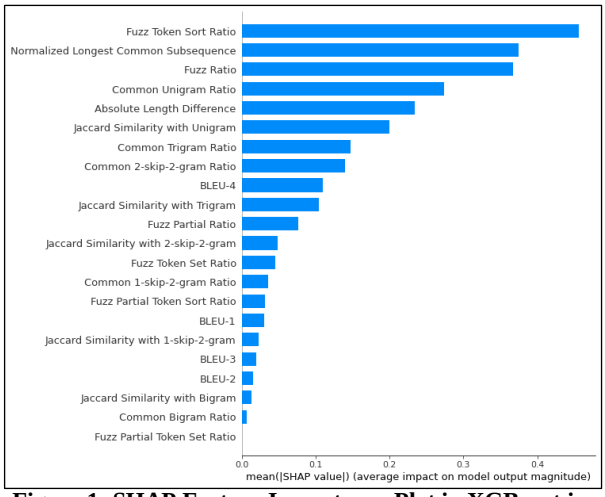


**Figure 1: SHAP Feature Importance Plot in XGBoost in Experiment 1**

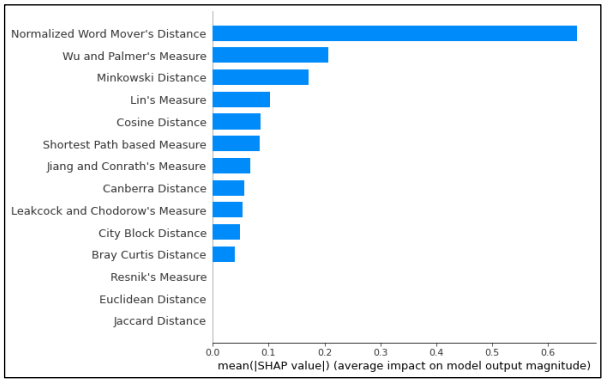


**Figure 2: SHAP Feature Importance Plot in XGBoost in Experiment 2**

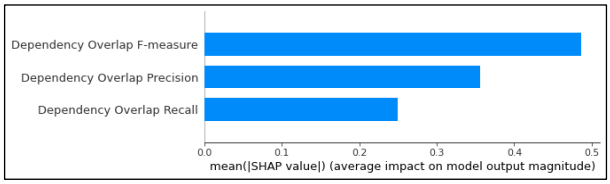


**Figure 3: SHAP Feature Importance Plot in XGBoost in Experiment 3**

In Experiment 1, 22 lexical features are used out of which 1 feature having zero contribution (from the bottom as seen in Figure 1) namely, Fuzz Partial Token Set Ratio is excluded from the feature-set taken for Experiment 4. In Experiment 2, 14 semantic features are used out of which 3 features with zero contribution (from the bottom as seen in Figure 2) namely, Resnik's Measure, Euclidean Distance, and Jaccard Distance are excluded from the feature-set taken for Experiment 4. In Experiment 3, 3 syntactic features are used. Since no feature has zero contribution (as seen in Figure 3), no feature is excluded from the feature-set taken for Experiment 4.Hence, total 35 features consisting of a combination of various lexical, semantic and syntactic features are included in feature-set taken for Experiment 4. The same feature-set is also used for Experiment 5.

# 4. DISCUSSION

A summary of results of the above 5 experiments is given in Table 1.

**Table 1: Summary of Results of Soft Voting Classifiers on Test Dataset of MSRP Corpus**

| Experiment | Soft Voting Classifier | Accuracy (%) | Precision (%) | Recall (%) | F1 Score (%) |
|---|---|---|---|---|---|
| 1 | Support Vector Machine, Logistic Regression, Random Forest | 76.23 | 78.19 | 89.10 | 83.29 |
| 2 | Support Vector Machine, Random Forest, K Nearest Neighbors | 71.42 | 74.12 | 87.62 | 80.30 |
| 3 | Gradient Boosting, Logistic Regression, XGBoost | 69.62 | 70.84 | 92.33 | 80.17 |
| 4 | Logistic Regression, Random Forest, Support Vector Machine | **77.10** | **79.42** | **88.49** | **83.71** |
| 5 | Random Forest, Support Vector Machine, K Nearest Neighbors | 74.72 | 75.05 | 92.85 | 83.01 |

Here, the soft voting classifier comprises of the top 3 performing standalone classifiers based on their Cross-Validation Accuracy Scores on training dataset of MSRP Corpus in each experiment, and the classification evaluation metrics namely, Precision, Recall, and F1 Score are computed for paraphrastic pairs i.e., sentence pairs labeled as "1".

The formula for Accuracy is given in Eq.(2).

$$A = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

The formula for Precisionis given in Eq. (3).

$$P = \frac{TP}{TP + FP} \quad (3)$$

The formula for Recall is given in Eq. (4).

$$R = \frac{TP}{TP + FN} \quad (4)$$

The formula for F1 Score is given in Eq. (5).

$$\mathrm{F} = \frac{2 * \mathrm{P} * \mathrm{R}}{P + R} \quad (5)$$

Here A (Accuracy) is the number of correctly predicted sentence pairs divided by the total number of sentence pairs, TP (True Positive) refers to a sentence pair being correctlypredicted as a paraphrastic pair, TN (True Negative) refers to a sentence pair being correctly predicted as a non-paraphrasticpair, FP (False Positive) refers to a sentence pair being incorrectly predicted as a paraphrastic pair, and FN (False Negative) refers to a sentence pair being incorrectly predicted as a non-paraphrastic pair, P (Precision) is the number of correctly predicted paraphrastic pairs divided by the number of sentence pairs predicted as paraphrastic pairs, R (Recall) is the number of correctly predicted paraphrastic pairs divided by the number of actual paraphrastic pairs, and F1 Score is the harmonic mean of Precision and Recall.

The performance of soft voting classifiers of developed methods in Experiments 1, 2, and 3 are improved upon with the soft voting classifier of developed method in Experiment 4 based on Accuracy Score, Precision Score, and F1 Score. The soft voting classifier of developed method in Experiment 5 has obtained lower Accuracy Score, lower Precision Score and lower F1 Score than the soft voting classifier of developed method in Experiment 4. However, the Recall Score obtained by soft voting classifier of developed method in Experiment 5 is the highest. From this result, it can be inferred that the EDATechniques [17] applied on the developed method in Experiment 5 improves the Recall Score but degrades the Accuracy Score, Precision Score and F1Score of the machine learning classifiers. This is a newfinding since the authors in [17] have demonstrated improvement of Accuracy Scores by applying their proposed EDA Techniques on only deep learning classifiers namely, Convolutional Neural Networks and Recurrent Neural Networks, and not on machine learning classifiers for text classification.

The soft voting classifier of developed method in Experiment 4 has obtained the best Accuracy Score of 77.10%, the best Precision Score of 79.42% and the best F1 Score of 83.71%, while the Recall Score of 88.49% has been obtained.

Hence, the developed method in Experiment 4 is selected as the proposed Supervised Machine Learning based Paraphrase Detection System.

Further, the soft voting classifier consisting of Logistic Regression, Random Forest, and Support Vector Machine of proposed system has obtained higher Accuracy Score,

comparable Precision Score, higher Recall Score, and higher F1 Score than those of its constituent standalone classifiers on test dataset of MSRP Corpus as given in Table 2. This finding is in agreement with the results of authors in [4] who have shown improvement in Accuracy Scores of standalone classifiers through the use of voting classifier consisting of the same standalone classifiers.

**Table 2. Performance Comparison of Soft Voting Classifier with its Constituent Classifiers of Proposed System**

| Sl. No. | Classifier | Accuracy (%) | Precision (%) | Recall (%) | F1 Score (%) |
|---|---|---|---|---|---|
| 1 | Logistic Regression | 76.87 | 79.78 | 87.36 | 83.40 |
| 2 | Random Forest | 75.65 | 78.42 | 87.45 | 82.69 |
| 3 | Support Vector Machine | 76.46 | 78.83 | 88.32 | 83.31 |
| 4 | Logistic Regression, Random Forest, Support Vector Machine | **77.10** | **79.42** | **88.49** | **83.71** |

Here the classification evaluation metrics namely, Precision, Recall, and F1 Score are computed for paraphrastic pairs i.e., sentence pairs labeled as "1".

The code created for the proposed system which has been developed and executed on Google Colaboratory using Python 3 Programming Language, is uploaded at https://github.com/Rudradityo/Paraphrase-Detection.

## 5. EVALUATION

The performance of the proposed Supervised Machine Learning based Paraphrase Detection System on MSRP Corpus is evaluated using classification evaluation metrics namely, Accuracy, Precision, Recall, and F1 Score where Precision, Recall, and F1 Score are computed for paraphrastic pairs i.e., sentence pairs labeled as "1".The comparison of performance of existing paraphrase detection systems with the proposed Supervised MachineLearning based Paraphrase Detection System on the test dataset of MSRP Corpus is given in Table 3.

**Table 3: Comparison of Performance of Paraphrase Detection Systems**

| Reference | Accuracy (%) | Precision (%) | Recall (%) | F1 Score (%) |
|---|---|---|---|---|
| [5] | 72.00 | 72.50 | 93.40 | 81.63 |
| [4] | 76.64 | 94.42 | 68.76 | 79.57 |
| [10] | 74.67 | 78.22 | 85.78 | 81.83 |
| [3] | 74.96 | 76.58 | 89.80 | 82.66 |
| [6] | 75.63 | 77.00 | 90.00 | 82.99 |
| [9] | 77.40 | * | * | 84.10 |
| [7] | 76.06 | 79.57 | 86.05 | 82.68 |
| [23] | 80.41 | * | * | 85.96 |
| [12] | 79.13 | 80.70 | 90.10 | 85.14 |
| [11] | 76.97 | 80.47 | 88.09 | 84.11 |
| [13] | 71.90 | 74.30 | 88.20 | 80.66 |
| [14] | 73.20 | 75.70 | 87.80 | 81.30 |
| [15] | 76.17 | 79.35 | 86.75 | 82.89 |
| **Proposed System** | **77.10** | **79.42** | **88.49** | **83.71** |

Here, the values marked by * are not given by the respective authors, the classification evaluation metrics namely, Precision, Recall, and F1 Score are computed only for paraphrastic pairs i.e., sentence pairs labeled as "1".

In Table 3, only supervised machine learning based paraphrase detection systems have been compared with the proposed system since the proposed system has also been based on the same.

The proposed system has achieved better Accuracy Score than those of 10 out of 13 paraphrase detection systems, behind [9], [12], and [23]. The proposed system has achieved better Precision Score than those of 7 out of 11 paraphrase detection systems, behind [7], [11], [12], and [4], andbetter Recall Score than those of 7 out of 11 paraphrase detection systems, behind [3], [6], [12], and [5]. The proposed system has achieved better F1 Score than those of 9 out of 13 paraphrase detection systems behind [9], [11], [12], and [23].

## 6. CONCLUSION AND FUTURE WORK

The proposed Supervised Machine Learning based Paraphrase Detection System has achieved comparable performance with existing paraphrase detection systems. The major

contributions of this research are as follows:

1. A unique combination of lexical, semantic, and syntactic features has been utilized in the proposed system which, to the best of our knowledge, has not been explored previously in the same feature-set.

2. A feature selection technique has been employed by making use of SHAP [26] Feature Importance Plots in XGBoost to remove the features with zero contribution for enhancing the performance of proposed system. In all the papers referenced, there is, to the best of our knowledge, no usage of SHAP [26] Feature Importance Plot for eitherillustrating the contribution of various features or used as a feature selection technique.

3. EDA Techniques proposed in[17] have been applied, which have resulted in increased Recall Score but reduced Accuracy Score, Precision Score, and F1 Score of machine learning classifiers, contrary to the findings of the authors who have demonstrated improvement of Accuracy Scores by applying their proposed EDA Techniques on only deep learning classifiers namely, Convolutional Neural Networks and Recurrent Neural Networks, and not on machine learning classifiers for text classification.

4. The soft voting classifier consisting of standalone classifiers namely, Logistic Regression, Random Forest, and Support Vector Machine of proposed system has obtained higher Accuracy Score, Recall Score, and F1 Score than those of its constituent standalone classifiers. This finding is in agreement with the results of authors in [3] who have shown improvement in Accuracy Scores of standalone classifiers through the use of voting classifier consisting of the same standalone classifiers.

For future work, more advanced features of lexical, semantic, and syntactic categories in addition to the ones utilized can be added in the feature-set of the proposed Supervised Machine Learning based Paraphrase Detection System for better performance in paraphrase detection.

## 7. REFERENCES

[1] Alzahrani, Salha& Salim, Naomie & Abraham, Ajith. (2012). Understanding Plagiarism Linguistic Patterns, Textual Features, and Detection Methods. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on. 42. 133 - 149. 10.1109/TSMCC.2011.2134847.

[2] El Desouki, M. I., & Gomaa, W. H. (2019). Exploring the Recent Trends of Paraphrase Detection. International Journal of Computer Applications, 975, 8887.

[3] Finch, A., Hwang, Y. S., &Sumita, E. (2005). Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In Proceedings of the third international workshop on paraphrasing (IWP2005).

[4] Kozareva, Z., &Montoyo, A. (2006, August). Paraphrase identification on the basis of supervised machine learning techniques. In International Conference on Natural Language Processing (in Finland) (pp. 524-533), Springer, Berlin, Heidelberg.

[5] Qiu, L., Kan, M. Y., & Chua, T. S. (2006, July). Paraphrase recognition via dissimilarity significance classification. In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (pp. 18-26). Association for Computational Linguistics.

[6] Wan, S., Dras, M., Dale, R., & Paris, C. (2006). Using dependency-based features to take the "para-farce" out of paraphrase. Proceedings of the Australasian Language Technology Workshop. 131-138.

[7] Das, D., & Smith, N. A. (2009, August). Paraphrase identification as probabilistic quasi-synchronous recognition. In 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: Volume 1-Volume 1 (pp. 468-476). Association for Computational Linguistics.

[8] Uribe, D. (2009, November). Effectively using monotonicity analysis for paraphrase identification. In 2009 Eighth Mexican International Conference on Artificial Intelligence (pp. 108-113). IEEE.

[9] Madnani, N., Tetreault, J., & Chodorow, M. (2012, June). Re-examining machine translation metrics for paraphrase identification. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 182-190). Association for Computational Linguistics.

[10] Ul-Qayyum, Z., & Altaf, W. (2012). Paraphrase identification using semantic heuristic features. Research Journal of Applied Sciences, Engineering and Technology, 4(22), 4894-4904.

[11] Chitra, A., & Rajkumar, A. (2013). Genetic algorithm based feature selection for paraphrase recognition. International Journal on Artificial Intelligence Tools, 22(02), 1350007.

[12] Filice, S., Da San Martino, G., &Moschitti, A. (2015, July). Structural representations for learning relations between pairs of texts. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 1003-1013).

[13] Zhang, Yitao& Patrick, Jon. (2012). Paraphrase Identification by Text Canonicalization. Proceedings of the Australasian Language Technology Workshop.

[14] Heilman, M., & Smith, N. (2010). Tree Edit Models for Recognizing Textual Entailments, Paraphrases, and Answers to Questions. In the 2010 Annual Conference of the North American Chapter of the ACL pages 1011-1019, Los Angeles, California.

[15] Malakasiotis (2009). Paraphrase Recognition Using Machine Learning to Combine Similarity Measures. In proceedings of the ACL-IJCNLP 2009 Student Research Workshop, pages 27-35, Suntec, Singapore.

[16] Uysal, A. K., &Gunal, S. (2014). The impact of preprocessing on text classification. Information Processing & Management, 50(1), 104-112.

[17] Wei, J. & Zou, K. (2019). EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on

Natural Language Processing, pages 6382–6388, Hong Kong, China, November 3–7, 2019.

[18] Chen, B., & Cherry, C. (2014, June). A systematic comparison of smoothing techniques for sentence-level bleu. In Proceedings of the Ninth Workshop on Statistical Machine Translation (pp. 362-367).

[19] Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015, June). From word embeddings to document distances. In International conference on machine learning (pp. 957-966).

[20] Cer, D., Yang, Y., Kong, S. Y., Hua, N., Limtiaco, N., John, R. S., Constant, N., Céspedes, M. G., Yuan, S., Tar C., Sung, Y. H., Strope B. & Kurzweil R. (2018). Universal sentence encoder. arXiv preprint arXiv:1803.11175.

[21] Lesk, M. (1986, June). Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In Proceedings of the 5th annual international conference on Systems documentation (pp. 24-26).

[22] Meng, L., Huang, R., & Gu, J. (2013). A review of semantic similarity measures in wordnet. International Journal of Hybrid Information Technology, 6(1), 1-12.

[23] Ji, Y., & Eisenstein, J. (2013, October). Discriminative improvements to distributional sentence similarity. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 891-896).

[24] Dolan, B., Quirk, C., & Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. Proceedings of the 20th International Conference on Computational Linguistics.

[25] Quirk, C, Brockett, C, & Dolan, W (2004). Monolingual Machine Translation for Paraphrase Generation. Conference: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, EMNLP 2004 142-149.

[26] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In Advances in neural information processing systems (pp. 4765-4774).

[27] Batunacun, Wieland, R., Lakes, T., and Nendel, C.: Using Shapley additive explanations to interpret extreme gradient boosting predictions of grassland degradation in Xilingol, China, Geosci. Model Dev., 14, 1493–1510, https://doi.org/10.5194/gmd-14-1493-2021, 2021.