International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

SEMANTIC PLAGIARISM DETECTION SYSTEM FOR ENGLISH TEXTS

Anupama Nair[1], Asmita Nair[2], Gayatri Nair[3], Pratiksha Prabhu[4], and Prof. Sagar Kulkarni[5]

[1-5]*Department of Computer Engineering, Pillai College of Engineering, Navi Mumbai, India – 410206*

Abstract— *Plagiarism is one of the major aspects that is considered when it comes to academics, literature as well as other fields where it is necessary to check if an idea is original. Plagiarism, when simply put, means the act of copying someone's work and portraying it as your own. It is ethically incorrect and is considered as a crime. For the purpose of finding plagiarism, many tools are available which can be downloaded or can be directly used online. These tools check the similarity at lexical and sentence level only. Hence, they only do statistical comparison whether the sentence is plagiarised or not, and not whether the idea is plagiarised. This project deals with detecting plagiarism at semantic level as well as identifying paraphrases, and ignoring the Named Entities which add to unnecessary plagiarism percentages. For the purpose of achieving this, we use Latent Semantic Analysis and a Bidirectional LSTM model for paraphrase detection. The final plagiarism uses a neural network to check plagiarism for an input paragraph which is done against a corpus.*

Key Words: **Plagiarism, Natural Language Processing, Semantic level, Machine Learning**

1. INTRODUCTION

Plagiarism is the act of showing someone else's work as your own and not giving credit to the original creator. It is considered as an act of dishonesty. Thus, making it a legal offense as well. Plagiarism is not always the copy of someone else's work in all entirety, but taking ideas from another source without properly citing it, or changing some form of the original work as well. Plagiarism is ethically incorrect and a serious crime. A Plagiarism Detection System checks the plagiarism of a document by comparing it with other documents and computing the amount of content that is similar or copied. As the volume of information on the Internet continues to increase, there is also an increase in the rate of plagiarism and thus the need for a plagiarism detection system. The objective of this project is to measure the semantic similarity of the document uploaded by the user with existing documents and derive a score that determines the degree to which the document is plagiarized.

2. LITERATURE SURVEY

A. An NLP Based Plagiarism Detection Approach for Short Sentences [1]

This technique is used by Shikha Pandey, Arpana Rawal. In semantic calculation, synonyms of arguments are compared between sentences. Typed dependency relationship (TDR), based on Natural Language Processing is presented for detecting plagiarism on short sentences. The disadvantage of this method is that it considers only synonyms and antonyms for semantic calculations.

B. SEMILAR: The Semantic Similarity Toolkit [2]

This technique is used by Vasile Rus, Mihai Lintean, Rajendra Banjade, Nobal Niraula, and Dan Stefanescu. It computes text-to-text similarity. The problems faced are for word-to-word similarity, phrase-to-phrase similarity, sentence-to-sentence similarity, paragraph-to-paragraph similarity, or document-to-document similarity. Implementation is done by part-of-speech tagging, phrase or dependency parsing, etc., semantic similarity methods (word-level and sentence-level), classification components for qualitative decision making with respect to textual semantic relations (Naïve Bayes, Decision Trees, Support Vector Machines, and Neural Network),

kernel-based methods (Sequence Kernels, Word Sequence Kernels, and Tree Kernels).

C. Semantic Plagiarism Detection in Text Document Using POS Tags [3]

This technique is used by Dnyaneshwar Ratan Bhalerao. Plagiarism detection methods used are Wordnet Similarity, Cosine Measurement, Integrated Sentence Similarity. The semantic similarity detection algorithm applied to the Pan-PC-11 data-set (considering 10 documents) to detect plagiarized sentences. It can detect copy paste words and plagiarized sentences when they are replaced by their synonyms. This algorithm considers nouns and verbs as the main feature for similarity but while dealing with semantics of the sentence there is a need to look at other syntactic features too.

D. Intrinsic Plagiarism Detection in Digital Data [4]

This methodology aims at detecting Paraphrasing, Idea, Mosaic, and 404 Error textual types of plagiarism that may possibly be observed in the submitted research paper. The proposed system detects almost all the plagiarized sections in the paper by analyzing the grammar used by the author of the paper. Implementation is done by Tree Construction. Each sentence of the digital text document format of the paper is parsed by its syntax, which results in a set of grammar trees. These trees are then compared against each other and then with the rest of the sentences in the document.

E. Plagiarism Detection using Semantic Analysis [5]

This technique is used by Eman Salih Al-Shamery and HadeelQasem Gheni. In this technique, the synonyms of each are found using WordNet. These synonyms will be considered as the appearance of the word itself when used to detect plagiarism. The only drawback is that synonyms are the only semantic aspect that is taken into consideration.

F. Detecting Plagiarism based on the Creation Process [6]

Authors put up some strategies of heuristic retrieval and evaluate the performance of the models for the detailed analysis. A graphical user interface is developed to conveniently access the system.

The GUI is written in java. The interface allows the user to input words, and to submit for semantic similarity calculation. But plug-ins need to be added to get the logs required. A disadvantage is that there are higher chances of false positives.

G. Similarity Measures Based on Latent Dirichlet Allocation [7]

The MSRP corpus is the largest publicly available annotated paraphrase corpus and has been used in most of the recent studies that addressed the problem of paraphrase identification. Allocation is a probabilistic approach and Latent Semantic Analysis. A combination of LSA and LDA is possible**.**

H. Deep Paraphrase Detection in Indian Languages [8]

This technique is proposed by Rupal Bhargava, Gargi Sharma and Yashvardhan Sharma. They have used six approaches in total - CNN, CNN-WordNet, 1-Layer-LSTM, 2-Layer LSTM, 1-Layer-BiLSTM, 2-Layer-BiLSTM. They have taken English, Hindi, Malayalam, Tamil and Punjabi into consideration. CNN-WordNet gives the best accuracy for English Language. The corpus used is MSRPC for English language and DPIL for Indian Regional Languages. A highest accuracy of 83% is achieved for English language.

2.1 Summary of Related Work

The summary of the methods used in the literature survey is given in Table 1.

Table 1: Summary of the literature survey

Literature	Semantic Similarity	Sentence Similarity	Statistical Method
An NLP Based Plagiarism Detection Approach for Short Sentences [1]	Yes	No	No

SEMILAR: The Semantic Similarity Toolkit [2]	No	Yes	No
Semantic Plagiarism Detection in Text Document Using POS Tag [3]	Yes	Yes	No
Intrinsic Plagiarism Detection in Digital Data [4]	Yes	Yes	Yes
Cross Lingual Plagiarism Detection [5]	Yes	No	Yes
Plagiarism Detection using Semantic Analysis [6]	Yes	No	No
Detecting Plagiarism based on the Creation Process [7]	No	No	Yes
Similarity Measures Based on Latent Dirichlet Allocation	Yes	Yes	No
[8]			

3. PROPOSED WORK

The proposed approach takes input as a set of English text documents. The documents undergo pre-processing steps which include tokenization, stop word removal, lemmatization, and POS tagging. Then the pre-processed data is given as input to compute semantic similarity for plagiarism detection. The final output is a report produced based on a set threshold value which decides whether both the documents have similar content or not. The proposed architecture system is shown in Fig. 1.

Each block of the system architecture is described in this section.

3.1 Input Documents

A suspicious English text document will be given as input to the system. The system will compare the similarity of this document with all the reference documents and produce a plagiarism report.

3.2 Pre-processing

The first step in the pre-processing is to present the English documents into clean word format and the output data will only consist of useful phrases. In the next steps of plagiarism detection, the documents are represented by a large number of features.

Commonly the steps taken are:

3.2.1 Tokenization

Tokenization is a process of converting sentences into a chain of words so that processing word byword can be easily performed. Here we have a tendency to use white space characters for tokenization. These separate tokens are also called lexicons. The document can be tokenized using the lexical analyzer. Tokenization helps to work with each word separately.

3.2.2 Stop Word Removal

The most frequently occurring words which slow down the processing of documents are called stop words. These words are irrelevant. Such words include articles, conjunctions, 22 prepositions, and

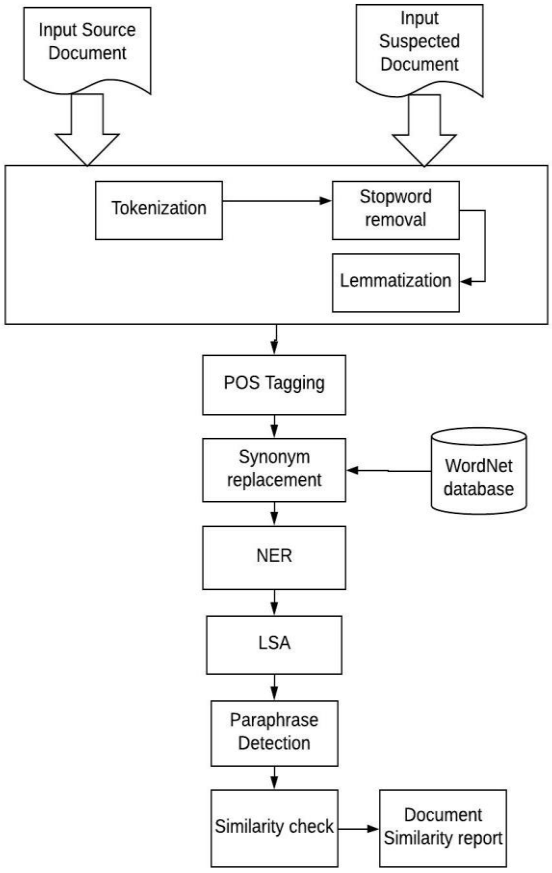

Fig. 1 Proposed system architecture

Other function words. Hence, stop word removal is done to enhance the speed of processing. A corpus of stop words is used to filter these out from the documents.

3.2.3 Morphological Analysis

In this step, the internal structure of the word is identified. A morphological analyzer gives the output as the root word of the given token. Devanagari script is very morphologically rich as it contains case markers and proposition markers as suffixes. Thus the stem and root word may vary in their forms.

The stem words are checked for inflections by creating appropriate rules. If it is inflected, then the root is formed by adding a replacement character with the stem word. The perfect match is searched from the created set of rules.

3.2.4 POS Tagging

In corpus-based studies, part-of-speech tagging is the process of converting sentences into the form (word, tag). The tag represents the part of speech associated with the word. This can be done for identification of words as nouns, verbs, adjectives, adverbs, etc. POS tagging is a supervised learning approach that uses attributes like the previous word, next word, is the first letter capitalized, etc. The most popular tag set is the Penn Treebank tagset.

3.3 Named Entity Recognition

In any text document, there are some terms representing specific entities that are more informative and are most essential in a sentence. These entities are known as named entities, which refer to terms that represent real-world objects like people, places, organizations, and so on, mostly denoted by proper nouns. The basic approach could be to find these by identifying the noun phrases in text documents. Named entity recognition (NER), also known as entity chunking/extraction, is a popular technique used to identify the named entities and classify them under various predefined classes.

3.4 Latent Semantic Analysis

Latent semantic analysis (LSA) is a method of understanding the context or underlying meaning of a document and the words present in it. The term 'latent' refers to the hidden features in the data. These hidden features represent some vital information present in the textual data. Latent semantic analysis is an unsupervised learning method. The output of LSA are topics or contexts are representations of the input data. Basically, it returns the contextual meaning of what that data is about. According to LSA, the words that are similar in meaning tend to appear in similar kinds of text. LSA works in two steps: Document Term Matrix and Single Value Decomposition. It produces latent features as topics. The comparison of the topics of both the suspected document corpus and the input data gives a semantic similarity percentage between both the documents.

3.5 Paraphrase Detection using Bidirectional Long Short Term Memory (BiLSTM) Model

A Recurrent Neural Network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. LSTM is an RNN that has a feedback network. An LSTM model can learn from its previous experiences, and retain necessary information so as to be used in future data. This makes

the LSTM model ideal for plagiarism detection using semantic analysis. This is because a sentence would be similar in the context to that of the previous sentence. A Bidirectional LSTM simply traverses in both directions i.e., from front to end and then end to front. This makes the neural network to learn better. Thus, we apply the BiLSTM model for paraphrase identification between an input and the corpus.

3.6 Doc2Vec

Numeric representation in Natural Language Processing is employed for several purposes, for example, document retrieval, web search, spam filtering, topic modeling, etc. However, this illustration of text documents could be a difficult task. Doc2vec is a simple and easy to use technique which gives accurate results. The goal of Doc2vec is to make a numeric illustration of a document, despite its length. Using Gensim Doc2vec is very straightforward. Using Gensim, the model will be initialized and trained for a few epochs. Then, the similarity of every unique document to every tag will be assessed.

4. IMPLEMENTATION

The objective of our project is to create a plagiarism detection system that considers semantic similarity, paraphrases as well as does not consider named entities while checking for plagiarism. Named entities unnecessarily add to plagiarism percentages. They can be ignored accordingly. We have used StanfordNER for the purpose of finding out the named entities in the document corpus and the input. These are ignored and only the remaining parts are considered for following steps.

Latent Semantic Analysis gives the similarity index between the document corpus elements and the input. Cosine similarity is used to find the similarity index. This step is followed by paraphrase detection. We use a single layer bidirectional LSTM model. This model is trained using the Microsoft Research Paraphrase Corpus (MSRPC). The model gave an accuracy of 69.33% while training 80% of the MSRPC training corpus. The paraphrase module returns 'Yes' or 'No' i.e., whether a sample input and the elements of the document corpus are paraphrases of each other or not. Finally, a text similarity is done between the input text and individual elements of the document corpus. Gensim's Doc2Vec is used for this purpose.

The generated document similarity report shows whether the input is plagiarized or not considering all of the above mentioned aspects. This similarity score generated can be used to determine and remove plagiarism.

5. CONCLUSION

Thus, we present the structure of a Plagiarism Detection System using semantic analysis and paraphrase identification. Plagiarism detection systems available in the market and online, at the present time, do not give very good accuracy due to a lot of factors that are ignored. Named Entity Recognition (NER) is one such factor that most of the systems are not able to avoid while considering the plagiarised content. A semantic level analysis is done using Latent Semantic Analysis(LSA). Paraphrase identification is done using a Bidirectional Long Short Term Memory (BiLSTM) Model. For the final plagiarism check, the input text undergoes a text similarity using Doc2Vec, to give a similarity score.

ACKNOWLEDGEMENT

We sincerely thank **Prof. Sagar Kulkarni**, the project guide, for his valuable guidance, encouragement, and support in carrying out this project. We take this opportunity to thank all our faculties who have directly or indirectly helped in our project. We are also thankful to **Dr. Sharvari Govilkar,** HOD of Computer Engineering department, for encouraging and allowing us to present the project on the topic Plagiarism Detection System. We deeply express our gratitude to **Dr. Sandeep M. Joshi,** Principal, PCE New Panvel, to provide us with this opportunity to learn and explore our technical knowledge.

REFERENCES

1. Shikha Pandey, Arpana Rawal, "An NLP Based Plagiarism Detection Approach for Short Sentences, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7 Issue-4, November 2018 https://www.ijrte.org/wp-content/uploads/papers/v7i4/E1831017519.pdf

2. Vasile Rus, Mihai Lintean, Rajendra Banjade, Nobal Niraula, and Dan Stefanescu, "SEMILAR: The Semantic Similarity Toolkit", Proceedings

of the 51st Annual Meeting of the Association for Computational Linguistics, pages 163–168, Sofia, Bulgaria, August 4-9 2013. Association for Computational Linguistics
http://www.cs.memphis.edu/~vrus/publications/2013/ACL-13.SEMILAR.DEMO.pdf

3. Dnyaneshwar Ratan Bhalerao, Semantic Plagiarism Detection in Text Document Using POS Tags,
https://www.academia.edu/16250097/Semantic_Plagiarism_Detection_in_Text_Document_Using_POS_Tags

4. Netra Charya, Kushagra Doshi, Smit Bawkar and Dr. Radha Shankarmani Information Technology Dept., Sardar Patel Institute of Technology, Andheri (W), Mumbai, India HOD, Information Technology Dept., Sardar Patel Institute of Technology, Andheri (W), Mumbai, India,"Intrinsic Plagiarism Detection in Digital Data "
http://www.ijiere.com/FinalPaper/FinalPaperIntrinsic%20Plagiarism%20Detection%20in%20Digital%20Data160148.pdf

5. Eman Salih Al-Shamery and Hadeel Qasem Gheni, "Plagiarism Detection using Semantic Analysis", Indian Journal of Science and Technology, Vol 9(1), DOI:10.17485/ijst/2016/v9i1/84235, January 2016, ISSN (Online): 0974-5645,
http://www.indjst.org/index.php/indjst/article/view/84235

6. Johannes Schneider, Abraham Bernstein, Jan vom Brocke, Kostadin Damevski, and David C. Shepherd, "Detecting Plagiarism based on the Creation Process",
https://www.semanticscholar.org/paper/Detecting-Plagiarism-Based-on-the-Creation-Process-Schneider-Bernstein/d856fa815fa7b17468f30ae000d58f5477db6277

7. Rus V., Niraula N., Banjade R. (2013) "Similarity Measures Based on Latent Dirichlet Allocation". In: Gelbukh A. (eds) Computational Linguistics and Intelligent Text Processing. CICLing 2013. Lecture Notes in Computer Science, vol 7816. Springer, Berlin, Heidelberg.

8. Rupal Bhargava, Gargi Sharma and Yashvardhan Sharma, WiSoc Lab, Department of Computer Science, Birla Institute of Technology, Pilani Rajasthan, India. 2017
IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, "Deep Paraphrase Detection in Indian Languages"
https://dl.acm.org/doi/pdf/10.1145/3110025.3122119