International Journal of Latest Trends in Engineering and Technology
Vol.(10)Issue(2), pp.270-272
DOI: http://dx.doi.org/10.21172/1.102.43
e-ISSN:2278-621X

SEMANTIC PLAGIARISM DETECTION USING NATURAL LANGUAGE PROCESSING

Rashmi Bongirwar[1], Renuka Deshpande[2], Neha Battula[3]

Abstract- Plagiarism in the sense of "theft of intellectual property" has been around for as long as humans have produced work of art and research. However, these days a stupendous amount of data regarding every field is available over the Internet. With this easily available large amount of data, the problem of plagiarism has increased. Application of NLP can help resolve this kind of problem. But today, with the vast amount of data available, the problem of paraphrasing and obfuscation is also increasing. Hence, we have developed a semantic plagiarism detector where there will be semantic checking which includes checking of synonyms. The underlying syntactic structure and semantic meaning of two documents can be compared to reveal their similarity.
Keywords – plagiarism, paraphrasing, obfuscation, NLP, semantic meaning

1. INTRODUCTION

Plagiarism is unacceptable use of other people's work as accurate copy or with little modification. It is a fraud of profound extent that is increasing a lot these days and requires immediate attention as it leads to low education quality, and hampers creativity. For example, students just easily copy assignments from the internet, teachers copy work, PHD students also use existing work.

This project deals with academic plagiarism. Plagiarism could be of many types such as: duplication, paraphrasing, repetitive research, replication, misleading attribution, verbatim plagiarism or complete plagiarism. 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 which determines the degree to which the document is plagiarized. Previous works on this subject have focused on the syntactic similarity to determine plagiarism. This project intends to overcome the shortcomings of existing plagiarism detection techniques by using semantic analysis instead of syntactic analysis. We can detect semantic plagiarism with the help of Natural Language Processing (NLP) Tools. It is a branch of artificial intelligence that helps computers understand, interpret and manipulate human language. NLP draws from many disciplines, including computer science and computational linguistics, in its pursuit to fill the gap between human communication and computer understanding. It helps computers communicate with humans in their own language and scales other language-related tasks.

2. LITERATURE SURVEY

We studied various existing systems which computed semantic similarity for short texts but not the entire document, systems which used syntactic analysis in documents, semantic similarity detection at various levels of granularity like word, sentence, document, checking for word replacement whether it is exact, synonym, hypernym, meronym, holonym, etc, different measures of semantic distance in WordNet. Through surveying various techniques and systems, we realized the best way for us to proceed would be using WordNet and doing semantic similarity measurement for two documents in a sentence wise approach.

3. PROPOSED ALGORITHM

3.1 Overall algorithm -

First, we create a repository. The document in the repository can be a webpage. If it is a webpage, text from the url is extracted and stored in a file in repository. Then, store the document uploaded by user in our repository. Furthermore, pre-processing of documents is performed. The final step is computation of similarity percentage of documents.

3.2 Pre-processing-

Since text is the most unstructured form of all the available data, various types of noise are present in it and the data is not readily analyzable without any pre-processing. The entire process of cleaning and standardization of text, making it noise-free and ready for analysis is known as text preprocessing. Different data preprocessing activities performed are:

[1] Department of Computer Engineering, SIES GST, Nerul, Navi Mumbai, Maharashtra, India
[2] Department of Computer Engineering, SIES GST, Nerul, Navi Mumbai, Maharashtra, India
[3] Department of Computer Engineering, SIES GST, Nerul, Navi Mumbai, Maharashtra, India

- Conversion of text to lower case: This is to avoid distinguishing between words simply on the basis of case.
- Removal of punctuation except full stop: Punctuation can provide grammatical context which supports understanding but does not add any value. Full stops are not removed because they are needed later for sentence tokenization.
- Removal of special characters and alphanumeric characters: They are not relevant to our analysis.
- Removal of English stop words: Stop words are common words found in a language. Words like for, of, are etc are common stop words.
- Stemming: Transforms to root word. Stemming uses an algorithm that removes common word endings for English words, such as “es”, “ed” and “’s”.
- Lemmatization: Transformation to dictionary base form i.e., “produce” & “produced” become “produce”.

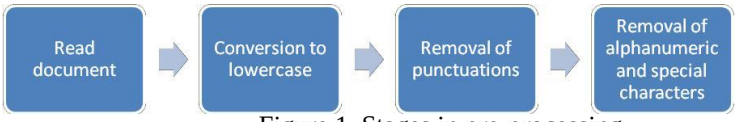

Figure 1. Stages in pre-processing

3.3 Computing semantic similarity-

- Break down the user and repository documents into sentences.
- Take each sentence of the user document and one at a time compare it with each sentence of a document from repository.
- Repeat this for all documents of repository; we get an array of similarity percentage scores for each document.
- Once we get the individual percentage similarity, we take average of all to get the final similarity percentage.

4. EXPERIMENT AND RESULT

In the proposed algorithm, the documents are checked semantically sentence wise and the similarity percentage of user document with each document is shown and stored. Then, average of these documents is taken to generate the final similarity percentage with all documents. While conducting tests, two documents were compared and similarity score was generated. Say, user document has 45 sentences and document in repository has 60 sentences, each sentence in user document will be compared with each sentence in repository document. To generate the score, it will be divided by total number of sentences 45*60 = 2700. This is repeated for all documents in the repository. This will give us an overall score which gives us an idea of the degree of plagiarism whether it is low, moderate or high.

5. CONCLUSION

Plagiarism detection for text in natural languages is a challenge. We describe our preliminary research on semantic similarity measures and their possible usage for paraphrasing detection in the task of plagiarism identification. Semantic similarity plays an important role in natural language processing, information retrieval, text summarization, text categorization, text clustering and so on. There is a plethora of measures and approaches proposed for different purposes in NLP domain. This project describes an approach to detect semantic plagiarism which occurs in researches by using WordNet. In this approach, WordNet has proven as an effective way to identify the semantic plagiarism by giving the synonyms of words in the document to detect the plagiarism.

However, there is some future scope in our research. Some important words which are not included in WordNet lexicon will not be considered as concepts for similarity evaluation. In future work, we would like to perform our method on a larger knowledge base, such as Wikipedia. Also we would like to take into consideration the citations and references given by the user to check their document. Although this method has a few drawbacks, it performs fairly well.

6. REFERENCES

[1]. A. Islam and D. Inkpen, Semantic text similarity using corpus-based word similarity and string similarity, ACM Transactions on Knowledge Discovery from Data, vol. 2, no. 2, pp. 125, Jan. 2008.

[2]. U. Bandara and G. Wijayarathna, “A Machine Learning Based Tool for Source Code Plagiarism Detection,” International Journal of Machine Learning and Computing, pp. 337–343, 2011.

[3]. Eman Salih Al-Shamery and Hadeel Qasem Gheni. Plagiarism detection using semantic analysis. Indian Journal of Science and Technology.

[4]. [4]Alexander Budanitsky and Graeme Hirst. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. University of Toronto- Toronto, Ontario, Canada.

[5]. A. Anguita, A. Beghelli, and W. Creixell, Automatic cross-language plagiarism detection, 2011 7th International Conference on Natural Language Processing and Knowledge Engineering, 2011.

[6]. J. Agarwal, R.H. Goudar, P. Kumar, K. Sharma, V. Parshav, R.Sharma, A. Srivastava and R. Rao, Intelligent Plagiarism Detection Mechanism using Semantic technology: A Different Approach, IEEE International Conference on Advances in Computing , Communication and Informatic, Mysore,22-25 Aug. 2013, 779-783.

[7]. T. Vrbanec and A. Mestrovic, The struggle with academic plagiarism: Approaches based on semantic similarity, 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics(MIPRO), 2017.

[8]. Rada Mihalcea and Courtney Corley. Corpus-based and knowledge-based measures of text semantic similarity. American Association for Artificial Intelligence, 2006.

[9]. I. Atoum and A. Otoom, Efficient Hybrid Semantic Text Similarity using Wordnet and a Corpus, International Journal of Advanced Computer Science and Applications, vol. 7, no. 9, 2016.
[10]. A. H. Osman and N. Salim, An improved semantic plagiarism detection scheme based on Chi-squared automatic interaction detection, 2013 International Conference On Computing, Electrical And Electronic Engineering (Icceee),2013.
[11]. Chi-Hong Leung and Yuen-Yan-Chan. Nlp approach for automatic plagiarism detection. The Chinese university of Hong-Kong, 2011.
[12]. George A. Miller and Walter G. Charles. Contextual correlates of semantic similarity. Language and Cognitive Processes.
[13]. Salmon Run, Inter-Document Similarity with Scikit-Learn and NLTK. [Online]. Available: ttp://sujitpal.blogspot.in/2013/05/inter-document-similarity-with-scikit.html.
[14]. What is Plagiarism?, Plagiarismorg RSS. [Online]. Available:http://www.plagiarism.org/article/what-is- plagiarism.
[15]. WordNet Interface. [Online]. Available: http://www.nltk.org/howto/wordnet.html.
[16]. Sujitpal, sujitpal/nltk-examples, GitHub. Available: https://github.com/sujitpal/nltk-examples/blob/master/src/semantic/shortsentencesimilarity.py .
[17]. Compute sentence similarity using Wordnet -NLPFORHACKERS, NLP-FOR-HACKERS,08-Aug-2017. [Online]. Available: http://nlpforhackers.io/wordnet-sentence-similarity/.
[18]. The Power of WordNet and How to Use It in Python,XRDS, 20-Jul-2017. [Online].Available: http://xrds.acm.org/blog/2017/07/power-wordnet-use-python/.
https://googleweblight.com/i?u=https://en.m.wikipedia.org/wiki/Plagiarism_detection& hl=en-IN