Engineering Applications of Artificial Intelligence 45 (2015) 246–258

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

# On retrieving intelligently plagiarized documents using semantic similarity

Syed Fawad Hussain*, Asif Suryani

*Faculty of Computer Science and Engineering, GIK Institute of Engineering Sciences and Technology, Topi 23640, Pakistan*

ARTICLE INFO

*Article history:*
Received 20 January 2015
Received in revised form
1 June 2015
Accepted 9 July 2015
Available online 29 July 2015

*Keywords:*
Plagiarism detection
Semantic similarity
Support Vector Machine
Information retrieval

ABSTRACT

Plagiarism in text documents can be done in many ways. The most common form of plagiarizing a text document is to copy a chunk of text and alter it intelligently, thereby making it look original. Such cases are hard to detect since they require semantic analysis of the document. External sources of knowledge such as WordNet have been employed to help detect such cases. However, such an approach might often miss the contextual significance of the employed words, as well as suffer from the issue of synonymy and polysemy. We propose an architecture that uses a semantic similarity measure that exploits the semantic similarity of words, as mined from within the data corpus, thereby using localized contextual information. In this work, an approach for detecting plagiarism in text document has been proposed using a semantic similarity measure with a Nearest Neighbor (NN) search, and using a kernel in multiclass support vector machine. We test our approach on a plagiarism dataset specially developed to test the efficacy of the solution with varying level of plagiarism. The results have been compared with that of well-known commercial software, Turnitin®, having access to a large database. Our experiments suggest that using semantic kernels can help detect plagiarism, which can outsmart available techniques.


## 1. Introduction

Innovations and ingenious ideas are always accepted in the knowledge industry. The fundamental efforts of the knowledge industry have been to endorse these innovations and ideas which arose across the world. With this also came a common problem of the originality and legitimacy of those ideas and work. Some individuals, groups and institutions affianced themselves in academic dishonesty by taking credit away from inventors. More precisely, this practice includes crafting the alias of ideas and work without crediting the original contributors. The specific term given to such acts of academic dishonesty is "Plagiarism" (Park, 2003). The term plagiarism can be defined more broadly; *it is typically denoted to be the theft of words or thoughts that cannot be considered as universal knowledge. Plagiarism includes the limited borrowing without acknowledgment of another's unique and substantial research findings* (Vinod, 2011).

Text is the most common source in which everything is expressed. Each year hundreds of thousands of research works are published in several conferences, journals and books. This necessitates a need to develop an automated computerized system to detect potential plagiarism. Early work in this regard is found in the 1990s, such as the work of Brin et al. (1995) and (Shivakumar and Garcia-Molina (1996), which was mainly focused on detecting plain copying, using statistical and computer based methods. During the last decade, many plagiarism strategies have evolved including copy-paste dangling references, word replacement and changing syntactic structure of document, etc. More advanced algorithms are required to detect such smarter plagiarism. Lately, automated plagiarism detection in natural languages has been an active area of research, and several sophisticated techniques have been proposed owing to recent advances in different fields like information retrieval (IR), cross language information retrieval (CLIR), text mining, etc. An excellent survey regarding such plagiarism detection algorithms can be found in Maurer et al. (2006) and some more recent algorithms in Alzahrani et al. (2012), for cross languages in Barrón-Cedeño et al. (2013).

Most current plagiarism detection methods use statistical and natural language processing techniques that results in a sophisticated plagiarism detection tool. However, there still exist some tricks that can be easily applied to beat such systems. Two such possible approaches could be (1) to use synonyms and replace keywords in the copied text by its synonym. For instance, the word "beautiful" can be replaced with "charming", "adorable" or "handsome"; and, (2) to paraphrase sentences, i.e. take key ideas from a text or a paragraph which are then re-written by the author as their own. The latter case can be termed as *intelligent plagiarism* is much harder to detect, since neither the word statistical count nor the natural language grammar will match. Fortunately, recent development in the field of text mining have produced several algorithms that match text not only on their statistics but also on their semantic level, such as Altınel et al. (2015), Bär et al. (2012),

* Corresponding author. Tel.: +92 938 281026x2396.
*E-mail address:* fawadhussain@giki.edu.pk (S.F. Hussain).

http://dx.doi.org/10.1016/j.engappai.2015.07.011
0952-1976/

Hussain et al. (2010) and Uddin et al. (2012). These algorithms include a very important factor: the search of semantic similarity among texts determined from within the corpus, as opposed to external knowledge bases. Many algorithms, e.g. Bisson and Hussain (2008) and Hussain et al. (2010) measure semantic similarity of textual data by finding cluster of documents and words, thereby analyzing words with respect to the class of documents in which they occur.

Plagiarism detection usually involves two parts – first, to retrieve the set of documents from which a given document might be plagiarized, and second, find the extent of plagiarism (if any) from these documents. In this work, we aim to incorporate semantic measures and develop them further in order to detect intelligent text plagiarism. Therefore, our work will focus on the first part where we retrieve the set of documents from which the potential plagiarism has taken place. The second part usually involves complicated syntactic analysis using natural language processing and syntactic structure, e.g. Galitsky (2014), Hadj Taieb et al. (2014) and Schuhmacher and Ponzetto (2014) among many others, which is beyond the scope of this paper. Indeed, as mentioned earlier, we are interested in finding plagiarism where the syntactic structure might be changed by the idea or main context is quite similar. This set of potentially plagiarized documents can further be analyzed by an expert (either human or system) for verification.

The goal of this work is to improve the accuracy of detecting the most common form of plagiarism in text document, i.e. paraphrasing and word replacement, and to retrieve a set of potentially plagiarized documents. Our main contribution is as follows:

- Firstly, we perform a comparative analysis of traditional and recently developed semantic measures.
- Secondly, we enhance an existing semantic similarity measure to include prior knowledge from an existing corpus that could facilitate plagiarism detection by retrieving potentially plagiarized documents.
- Finally, we analyze the behavior of our proposed plagiarism detection framework on a purpose built dataset. We test the effectiveness of our proposed approach and compare it to commercially available software, Turnitin, on varying degree of plagiarism and analyze our results.

The rest of the paper is organized as follows: in Section 2, we compare and analyze several traditional and semantic similarity measures. In Section 3, we develop on the existing similarity measures and propose an enhanced supervised similarity measure. Section 4 describes the detailed architecture of our proposed approach to plagiarism detection in text documents, and improve it for faster computational time. Section 5 analyzes the results on several benchmark and specific dataset. We finally conclude the paper with some future research directions in Section 6.

## 2. Background and related work

### 2.1. Basic notation and definitions

In document plagiarism, we are interested in comparing chunks of the document for plagiarism rather than the whole document itself. Hence, a slightly modified approach to the usual information retrieval task is used where both the source documents $D_{src}$ and plagiarized document $D_{plg}$ are divided into small parts ($S_{src}$ and $S_{plg}$ respectively), usually at sentence or paragraph levels.

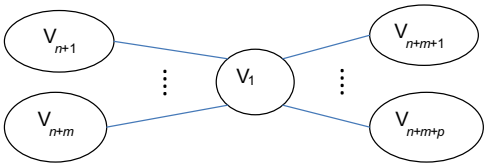


**Fig. 1.** Tri-partite graph of sentence $V_1$.

We first present the document-sentence-word tri-partite graph model for a set of given documents, based on which the ranking-based sentence retrieval framework is developed. Let $G=\{V, E\}$, where $V$ is the set of vertices that consists of the sentence set $S=\{s_1, s_2,\ldots, s_n\}$, the document set $D=\{d_1, d_2,\ldots, d_m\}$, and the term set $W=\{w_1, w_2,\ldots, w_p\}$, i.e., $V=D\cup S\cup W$, $m$ is the number of documents, $n$ is the number of sentences and $p$ is the number of terms, and $E$ is the set of edges. A sample graph G is presented in Fig. 1. For ease of illustration, we only demonstrate the edges between a sentence vertex $V_1$ and other vertices which represent documents or terms. $V_{n+1}\ldots V_{n+m}$ represents one of the $m$ documents, while $V_{n+m+1}\ldots V_{n+m+p}$ represents the word set. Therefore, $V_1$ may contain any of the $p$ words while being part of one (or more) of the $n$ documents.

Let $\mathbf{w}_i$ denote a term vector in the set of terms used to index the set of documents. Then, the set of all term vectors $\{\mathbf{w}_i\}$ $(1\le i\le p)$ is the generating set of the vector space, thus the space basis. A document vector $\mathbf{d}_i$ is given by

$$\mathbf{d}_i=\left[w_{i1}, w_{i2}, \ldots, w_{ip}\right] \tag{1}$$

### 2.2. Semantic similarity measures

Traditional similarity measures (such as Cosine, Euclidean, etc. ) can be used to compute similarity between documents (or paragraphs from documents). However, they will merely compare common words that occur in both documents and cannot detect semantic similarity. To overcome this issue, various approaches have been suggested that take into account the semantic relationship occurring within the dataset. These algorithms are usually referred to as *semantic based similarity measures*. Some of the popular methods are described below.

#### 2.2.1. Latent semantic analysis

LSA was proposed by Deerwester et al. (1990) as a least square projection method based on the mathematical technique termed as Singular Value Decomposition (SVD). LSA works by decomposing the document collection matrix $\mathbf{D}_{src}$ into component matrices:

$$\mathbf{D}_{src}=\mathbf{U}\sum\mathbf{V}^T \tag{2}$$

a left orthonormal matrix, $\mathbf{U}$, containing the document strength with concepts; a singular diagonal matrix, $\mathbf{\Sigma}$, representing the strength of the concepts; and, a right orthonormal matrix, $\mathbf{V}^T$, corresponding to word strengths against each concept. The dimensionality of the data is reduced by removing the least singular values (concepts) and reconstructing an approximate data matrix, $\mathbf{D}_{src}'$. Similarity between documents is then computed using the Cosine similarity measure on the approximated data matrix, resulting in new similarities that would not have been found using the simple VSM model.

Several other semantic similarity measures have been proposed in the literature. Most of these algorithms represent the document corpus matrix, $\mathbf{D}_{src}$, as a bi-partite graph between two set of objects - the documents and the words. These similarity measures then estimate the similarity based on some property of nodes in the graph, such as the random walk (Erkan, 2006), a higher order