Studies in Applied Linguistics & TESOL at Teachers College, Columbia University, Vol. 21, No. 2, pp. 54–60
The Forum

Automatic Detection of Plagiarism in Writing

Mahshad Davoodifard
Teachers College, Columbia University

INTRODUCTION

Plagiarism is defined as "using words, ideas, or work products attributable to another identifiable person or source without attributing the work to the source from which it was obtained' (Fishman, 2009, p. 5). While investigating plagiarism is relevant in different fields, from forensic linguistics to literary works, verification of original authorship has also attracted attention in academia and L2 learning and assessment contexts. Generally associated with academic misconduct and dishonesty, plagiarism in writing can take many shapes and be hard to detect. In addition to being a very time-consuming task, detecting plagiarism is not straightforward because committing plagiarism usually involves using techniques other than word-for-word copying and pasting the exact words, sentences, or other pieces of writing. That is why computer scientists are looking for more efficient and fast automatic detection tools. There are two major approaches to detecting plagiarism reported in the literature: the intrinsic approach, which investigates the internal features of the text for any different textual or stylistic element within a text; and the external approach, which compares the text to possible sources of plagiarism (source documents) and reports on degrees of similarity or overlap. Some researchers (e.g., Krause, 2016) consider a third approach to detecting plagiarism, namely, the cross-lingual method, which investigates instances of plagiarism translated from another language.

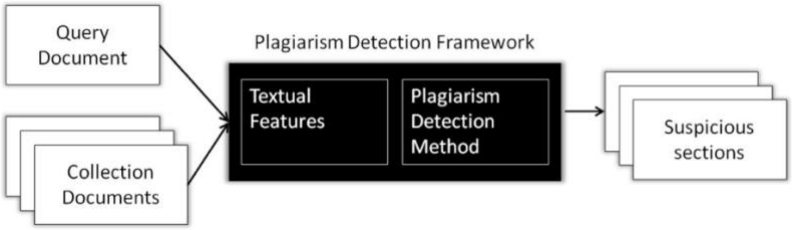
External plagiarism detection approaches face significant challenges, including a large number of potential source documents (the whole World Wide Web) and obfuscation techniques such as summarizing, paraphrasing, and translating the plagiarized contents (Potthast, et al., 2010). The biggest challenge yet is finding a valid corpus of plagiarized data that can be used in research, replication of techniques, and reporting the results. Due to ethical issues, having access to such corpora is almost impossible. To overcome this challenge, a corpus of artificially plagiarized documents, known as PAN-PC has been produced by the PAN competition organizers, which will be discussed later in this paper.

The current plagiarism detection methods use a wide variety of stylistic and linguistic features such as *n*-grams, semantic meanings, lexical and syntactic information, *tf-idf* and similarity measures and distance matrices (such as cosine similarity, Jaccard distance or Euclidean distance measures) to find similarity between documents and possible sources. These features are then embedded into plagiarism detection algorithms to produce tools that can find the closest possible match in as shortest time as possible, or in case of intrinsic methods, to find abnormalities within a text. There are also many commercial plagiarism checkers available on

the internet (e.g. *copyleaks* and *plagscan*), the algorithms of most of which remain a secret to the public due to proprietary rights.

The most common method used for developing external plagiarism detection tools includes preparation or pre-processing of the data, finding similarity levels, clustering and candidate document selection, and detailed analysis and reporting of plagiarized texts. This method is depicted in Figure 1.

FIGURE 1
General plagiarism detection framework (Kraus, 2016)



As mentioned above, there is no general agreement among researchers regarding which features would work best and yield the most accurate results in plagiarism detection algorithms. However, it seems that lexical features have been included in most works. Developing an external plagiarism detection tool, Abdi, et al. (2017) combined several linguistic features to capture the meaning of sentences, detect passive and active sentences, and bridge the lexical gaps between source and suspicious documents from the PAN-PC-11 corpus. In another external approach, Mahdavi et al. (2014) used *n*-grams along with cosine similarity and other distance coefficients within a Vector Space Model (VSM) to retrieve the best matches to a corpus of plagiarized texts in Persian. Also, basing their method on a VSM and a similarity score, Rao et al. (2011) proposed an external detection by indexing all the source documents and giving each suspicious document a query to that index. More recently, Stefanovic et al. (2019) employed an *n*-gram approach to visualize and cluster texts.

Plagiarism detection has other applications than merely finding text overlap for general education purposes. In forensic authorship analysis, for example, plagiarism detection algorithms have been used for author verification using text features, a technique known as stylometry. In their study of authorship verification of short online messages, Brocardo et al. (2013) proposed a supervised learning technique combined with an *n*-gram analysis approach to check author identity. Their results showed higher accuracy for texts of longer characters and more encouraging results compared to similar techniques used before. Brocardo et al. (2013) based their argument on only one stylometric feature, which, as they mentioned, could describe their error rates. In another research, Stewart (2012) used stylometry and keystroke biometrics to verify the test takers' identity in online examinations. Defining and extracting different features, Stewart (2012) examined the effect of each feature on the performance of his model and similar to Brocardo's (2013), concluded that the length and number of texts has a significant effect on the performance of the model. Stewart (2012, p. 111) maintained that "ideally, an algorithm allowing mix and match or heuristic evaluation of different combinations of features to measure the effect on accuracy in authorship attribution would be preferred." In another study of stylometry-based detection of plagiarism, Krause (2015) analyzed a large corpus of blogs to

verify students' authorship. He selected a set of lexical and syntactic features including character frequency, word and sentence-length frequency, POS tag frequency, and word specificity. Krause (2015) analyzed the data at the sentence level and used a support vector machine and showed a high prediction quality with binary SVMs.

The performance of a plagiarism detection tool, as Potthast et al. (2010) explain, cannot be directly evaluated by applying precision and recall. In the case of external algorithms, for example, we need to evaluate whether the source documents corresponding to the suspicious documents have been accurately retrieved. Plagiarism detection algorithms do not return a single and unique result set for each plagiarized document; they rather consider the whole available sections, and therefore, the precision of the algorithm is calculated differently based on retrieval. The PAN organizers provide a comprehensive performance analysis tool for evaluating the plagiarism detection algorithms submitted (see Potthast et al. 2016).

Following the current trend of research, the present paper aimed at developing an external plagiarism detection model, analyzing 500 suspicious documents from the PAN-PC-11 corpus while finding similarity levels between them and a corresponding 500 source documents. According to PAN, the suspicious documents may or may not include plagiarism and it is up to the code to reveal the extent to which the texts are similar. Based on the positive findings reported in the literature regarding the effectiveness of lexical-related features, the model is based on *tf-idf* and cosine-similarity measures using a Vector Space Model. The findings of the algorithm are then presented followed by a discussion of the results.

METHOD

Data

As discussed earlier in this paper, most plagiarism detection studies base their model and analysis on artificially plagiarized corpora, since access to a large corpus of naturally occurring plagiarized data is still a challenge given practicality as well as ethical issues. The most frequently used and cited data for plagiarism detection is the PAN Plagiarism Corpus (PAN-PC). The corpus is made up of both manual and automatic insertion of texts into a set of documents, called suspicious documents. The possible sources of the corresponding plagiarism cases are also included in the corpus. Because PAN-PC includes both the suspicious and the external source documents, it allows for both intrinsic and external approaches to detecting plagiarism. The parameters varied in the construction of PAN-PC are depicted in Figure 2. Since the focus of the present paper is on developing an external detection tool, only the first part of the external data set was used.

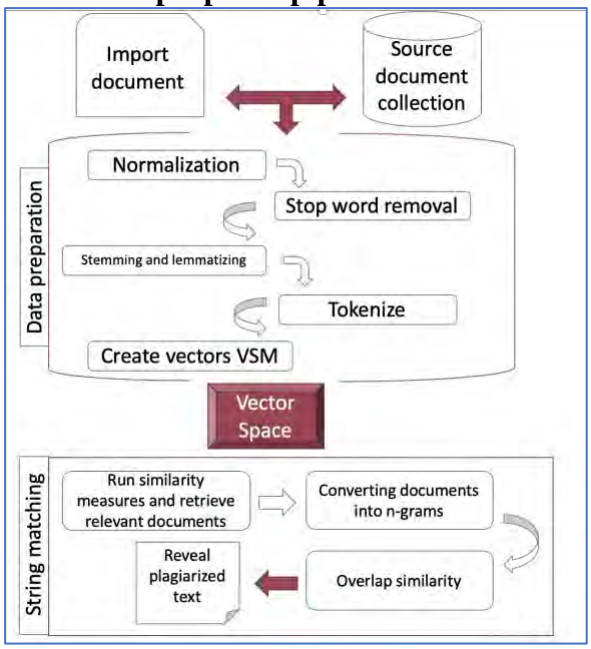
FIGURE 2
PAN-PC-11 corpus statistics (26,939 documents and 61,064 plagiarized texts)

Document statistics		
Document purpose	Plagiarism per document	Document length
Source documents 50% Suspicious documents: • With plagiarism 25% • without plagiarism 25%	Hardly (5W–20%) 57% Medium (20%–50%) 15% Much (50%–80%) 18% Entirely (>80%) 10%	Short (1–10 pp.) 50% Medium (10–100 pp.) 35% Long (100–1000 pp.) 15%
Plagiarism case statistics		
Obfuscation	Case length	
None Paraphrasing: • Automatic (low) • Automatic (high) • Manual Translation: • Automatic • Automatic + manual correction	18% 32% 31% 8% 10% 1%	Short (< 150 words) 35% Medium (150–1150 words) 38% Long (> 1150 words) 27%

Proposed method

The overall structure of the steps followed in the present paper to detect plagiarism is shown in Figure 3. The first step, data preparation, is performed to convert a text input to a feature vector. In order to appropriately perform that function, the texts are first normalized with UTF-8 encoding to make sure that they represent the same writing standard. Next, stop words are removed, followed by stemming and lemmatization to minimize inflections and word adding. Finally, the texts are tokenized at the word level to be converted into vectors to be used in the vector space model.

FIGURE 3
The proposed pipeline



After the data is normalized, cleaned up and tokenized, the features are selected for building feature vectors. As mentioned above, in this paper each document is represented by its *tf-idf* score. The next step involves defining and measuring the cosine similarity score, which represents the correlation between the vectors. Cosine similarity measure ranges from 0 to 1, with 0 meaning that the two texts have nothing in common and 1 meaning that they are identical copies. In the literature, researchers have considered different thresholds for cosine similarity to reflect a case of plagiarism and avoid the retrieval of too many irrelevant documents. The combination of *tf-idf* and cosine similarity is a common method in the automatic detection of plagiarism. It allows for quick retrieval of source documents that best represent as possible matches with the suspicious (plagiarized) text. Based on the same concept, this method allows for checking how similar the text (query text) is to the available sources in the database. In this paper, I am reporting the results of the algorithm that performed the latter function.

The next step in developing the external detection tool should be building an algorithm or classifier that is able to retrieve documents based on the similarity measures calculated, and indicate, for example, what percentage of the document is plagiarized, and which source document(s) are the best matches for that calculation. While the algorithm developed for the present study was able to detect overlapping similarities, the results are limited to a manual examination of the similarity scores obtained for comparing each of the suspicious documents to the source documents (a total of 250,000 scores). The accuracy of the model can be evaluated by PAN's evaluation measures of precision, recall, and the combination F score, in order to see how accurately the model has been able to retrieve related documents.

RESULTS AND DISCUSSION

For the purpose of analysis, the data was limited to the first 24 texts of each data set (i.e., source and suspicious data), and cosine similarity was calculated. Appendix 1 shows the full matrix of the results, but here I explain the first three significant findings and manually retrieve and examine related documents to evaluate the accuracy of the model. From the matrix in Appendix A, it can be seen that the algorithm compared each suspicious document against all other suspicious and source documents, which is due to the fact that I migrated the sample 24 texts of each data set in one folder for faster analysis of the data (a score of 1 means the document was compared against itself, which is an identical copy). Based on the previous works on cosine similarity, I consider a score of 0.2 and above for cosine similarity as an indicator of plagiarism. After retrieving the related documents, I used online tools for comparison of documents to find similarities between the text documents and confirm the results.

For example, the results showed a cosine similarity score of 0.2 between source document 10 and suspicious document 1. The documents were compared online through "text-compare.com" and "copyleaks". While the text-compare website found a number of common words and phrases between the documents, *copyleaks* reported the suspicious document as original, which is not true, because a brief review of the actual texts (which are stories revolving around a letter received) can reveal many examples of common words. The detection tool also found a score of 0.4 between source 18 and suspicious 9 documents, and a score of 0.3 between suspicious 24 and source 18 documents. Again, *copyleaks* did not find the suspicious documents as plagiarized with the exception of finding 8 words copied from source 21 in suspicious 9.

Based on these results, it can be inferred that the detection tool developed here is functioning as expected in finding text similarities.

CONCLUSION

This paper reports on preliminary steps to create an external plagiarism detection tool. I used the PAN-PC-11 data sets and extracted *tf-idf* scores of text documents and cosine similarity measures between source and suspicious documents to find text overlap. The model was able to successfully create vectors and measure the similarity metrics. However, the algorithm was not extended further to automatically retrieve related documents to follow on the pipeline (converting texts to *n*-grams for detailed analysis and revealing the best match as a source of plagiarism and evaluating the accuracy of the model). The model produced a matrix of cosine similarity for all the documents, which I used to manually retrieve documents and check for overlap using online tools. While extending the algorithm based on the suggested pipeline would allow for a more accurate evaluation of the model, manual comparison of sample documents provided some validity of the model developed for the present study.

REFERENCES

Abdi, A., Shamsuddin, S. M., Idris, N., and Alguliyev, R. M. (2017). A linguistic treatment for automatic external plagiarism detection. Knowledge-Based Systems. 10.1016/j.knosys.2017.08.008.

Bird, S., Klein, E., & Loper, E. (2009). Natural language processing with Python: Analyzing text with the natural language toolkit. Sebastopol, CA: O'Reilly Media.

Brocardo, M., Traore, I., Saad, S. and Woungang, I. (2013). "Authorship verification for short messages using stylometry," 2013 International Conference on Computer, Information and Telecommunication Systems (CITS), Athens, 2013, pp. 1-6.

Chambers, B. (2014). Basic Statistical NLP Part 2 - TF-IDF And Cosine Similarity. Retrieved from http://billchambers.me/tutorials/2014/12/22/cosine-similarity-explained-in-python.html.

Clough, P. T. (2000). Comments on setting criteria for experimental writing. Qualitative Inquiry, 6(2), 278-291.

Fishman, T. (2009). We know it when we see it is not good enough: Toward a standard definition of plagiarism that transcends theft, fraud, and copyright. *Proceedings of 4th Asia Pacific Conference on Educational Integrity (4APCEI)* (pp. 1-5). Wollongong, Australia: University of Wollongong NSW Australia.

Huang, L. (2017). Measuring Similarity Between Texts in Python. Retrieved from https://sites.temple.edu/tudsc/2017/03/30/measuring-similarity-between-texts-in-python/

Kraus, C. (2016). Plagiarism Detection - State-of-the-art systems (2016) and evaluation methods. Retrieved from https://arxiv.org/pdf/1603.03014.pdf.

Krause, M. (2015). Stylometry-based Fraud and Plagiarism Detection for Learning at Scale.

Liang, H. (2014). Coevolution of political discussion and common ground in web discussion forum. Social Science Computer Review, 32, 155-169. doi:10.1177/0894439313506844

Mahdavi, P., Siadati, Z., Yaghmaee, F. (2014). Automatic external Persian plagiarism detection using vector space model. Proceedings of the 4th International Conference on Computer and Knowledge Engineering.

Pang, B., & Lee, L. (2004). Sentiment polarity dataset version 2.0 [Data file]. Retrieved from http://www.nltk.org/nltk_data/

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830. http://www.jmlr.org/papers/v12/pedregosa11a.html

Potthast, M., Stein, B., Barrón-Cedeño, A., and Rosso P (2010). An Evaluation Framework for Plagiarism Detection. In Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010), Beijing, China, August 2010. Association for Computational Linguistics.

Rao, S., Gupta, P., Singhal, K., & Majumder, P. (2011). External & intrinsic plagiarism detection: VSM & discourse markers based approach. *Notebook for PAN at CLEF*, *2011*.

Sanchez-Perez, M., Gelbukh, A., Sidorov, G. (2014). Adaptive Algorithm for Plagiarism Detection: The Best-Performing Approach at PAN 2014 Text Alignment Competition. CLEF 2015: 402-413

Stefanovič, P., Kurasova, O., & Štrimaitis, R. (2019). The n-grams based text similarity detection approach using self-organizing maps and similarity measures. *Applied sciences*, *9*(9), 1870.

Stewart, J. (2013) An Evaluation of the Application of Stylometry and the Keystroke Biometric to Identity Verification of Online Test-Takers. Ph.D. Dissertation. Pace Univ., New York, NY, USA.

Mahshad Davoodifard is a doctoral student in the applied linguistic program of Teachers College in the second language assessment track. Mahshad is interested in innovative approaches to test crafting (such as SBA) as well as the role of technology in enhancing our understanding and improvement of second language tests. Correspondence should be sent to E-mail: md3573@tc.columbia.edu