External & Intrinsic Plagiarism Detection : VSM & Discourse Markers based Approach
Notebook for PAN at CLEF 2011

Sameer Rao, Parth Gupta, Khushboo Singhal, and Prasenjit Majumder

DA-IICT,
Gandhinagar, India
{rao_sameer,gupta_parth,
khushboo_singhal,p_majumder}@daiict.ac.in

Abstract. This paper aims to explain the performance of plagiarism detection system which can detect External as well as Intrinsic Plagiarism in text. It reports the results on PAN-PC-2011 test corpus. We investigated Vector Space Model based techniques for detecting external plagiarism cases and discourse markers based features to detect intrinsic plagiarism cases.

1 Introduction

Automatic plagiarism detection has gained immense attention of the researchers because of an absence of the one state-of-the-art algorithm and hence every year many systems are being tested in PAN. In the external setting of plagiarism detection, system has to find evidence of plagiarism from the pool of source documents. Sometimes there are no source documents available for suspicious documents to compare with. In such cases intrinsic plagiarism detectors play a major role. We present a Vector Space Model(VSM)[2] based approach for external plagiarism detection and discourse markers based approach for internal plagiarism detection.

2 External Plagiarism Detection

For external plagiarism detection setting the Dataset PAN-PC-2011[1] contains 11093 suspicious documents and 11093 source documents. The literal size of the corpus is 4.5 GB.

2.1 Algorithm

We convert all the non-english documents to english by a two stage strategy. First, we identify language of the document using Google Language Identifier[2]

[1] Dataset:PAN-PC-2011, http://pan.webis.de/

[2] Google Language Identifier: http://code.google.com/p/language-detection/

and then translate all non-english documents into english using Google Translator API[3]. We notice that some of the words had character level differences in our system and hence were not properly translated in turn translation of the sentence was not proper.

Candidate Selection : We use VSM based approach to select the candidate documents. All the source documents are indexed and each suspicious document is given as query to this index. We consider top 250 source documents in the ranked list as candidate or those with Similarity Score greater than 0.01, whichever is less. This strategy of involving two parameters for upper bound works good because there were many suspicious documents which were not at all plagiarised. For such documents the similarity score rapidly goes below 0.01 and hence we save computational power by not analysing all 250 candidate documents. Anyway we analyze at least top 20 documents for because we found some suspicious documents which have very small amount of plagiarism have similarity score below 0.01 even for top documents. Here, similarity score is typically the Dot Product of source document (d) and suspicious document(q).

$$cos\theta = \frac{d_2.q}{\|d_2\|.\|q\|} \tag{1}$$

Detailed Analysis : Last year we tried overlapping 7 word-grams to compare the sections of suspicious and source documents[1]. This time we used a window based similarity score to detect plagiarism. First, we take a 7-word gram of the suspicious document and look for it in source document. If it matches, we believe there can be a case of plagiarism because 'seven consecutive words match' is a potential evidence. Now, from that matching point we take 25 words window in both suspicious and source documents and calculate the similarity score. We remove a small set of stop-words from that window. We chose 25 words window because smallest case of plagiarism can be of 200 characters and which is explained by 25 words. We choose the similarity threshold to consider plagiarism as 0.50 which reveals at least 50% of words match in the window. We stop matching the windows if 8 consecutive windows have similarity score below 0.50. Keeping 8 tolerence windows helps to improve the granularity if obfuscation is very high for some sentences in between. Another reason to keep a tolerence window is, it becomes possible to keep a high similarity score to avoid false positives and still maintaining the granularity.

We merge the consecutive plagiarism cases if they are 500 characters apart. This helps in improving the granularity if algorithm has detected one case as split in many small cases due to obfuscation. If a suspicious document annotated by our algorithm has no plagiarism cases of length greater than 160 characters, we consider that document as plagiarism-free.

[3] Google Language Translator: http://translate.google.com/

3 Results and Analysis

Type	PlagDet	Recall	Precision	Granularity
External	0.1990889	0.1618067	0.4541152	1.2949292
Intrinsic	0.0693820	0.1080543	0.0783903	1.4787234

Table 1. Results on PAN-PC-2011 Test Data for External Plagiarism Detection

Table 1 shows the performance evaluation of our algorithm. Our overall recall score is low due to the fact that many candidate documents were not fetched in the first phase. In case of our automatic translation strategy, the offsets of sentences in the original and translated documents were different. Our system did not handle this fact in detailed analysis phase. One of the reason for low precision was tolerence windows being high which allowed many non-plagiarised sections to be marked as plagiarised which in turn affected the overall precision. These evaluation measures are explained in [6].

4 Intrinsic Plagiarism Detection

The main idea behind the Intrinsic Algorithms is to find out the sections which are not in the harmony of the whole document in terms of writing style and/or author style. This year we also tried to address this issue.

4.1 Algorithm

The Algorithm tries to calculate the distance between two normalized feature vectors: One is composed of the whole document while the other representing the partially overlapping sections of the documents of 2000 characters window with 200 step size. All the sections for which the style change value comes out to be greater than 2.0 are marked to be plagiarized. Consecutive plagiarized sections which are 500 characters apart are merged to form a single plagiarized case to maintain proper granularity value.

4.2 Features

Frequent character n-grams based feature to detect style change was used in [3], while frequency of different pronouns, closed class words, stem suffixes, punctuation marks, average length of a statement were used to classify author style in [4, 5]. We have combined these features and also added frequency of discourse markers. We believe some authors use some words more often and these words are generally discourse markers.

Discourse markers Discourse markers are words that do not change the meaning of the text. They are either used as filler element in the text or out of author's habit. People use them frequently in the text and most likely twice every 2 or 3 sentence. So frequency of such words can help us in detecting author's style change. Few discourse markers in English language are "well", "actually", "basically" , "then", "means", etc. Such commonly used discourse markers are added as a different dimension in our stylometric feature vector.

Style change function Distance between normalized stylometric feature vectors is calculated using style change function as

$$d_1(A, B) = \sum_{g \epsilon P(A)} \left(\frac{2(f_A(g) - f_B(g))}{(f_A(g) + f_B(g))} \right)^2 \qquad (2)$$

where A and B are normalized vectors for complete document and extracted section of document respectively and g is different dimension of stylometric vector. Further details of the function can be found in [3]

4.3 Results and Analysis

The corpus has 4753 number of documents for intrinsic setting. Performance results are reported in Table 1. The major problem with our weak performance is low recall and large number of false positive detection. We fixed same feature dimensions for all documents but some of those features don't apply to a particular author and play a negative role in style change function calculation.

5 Conclusion

We tested the performance of VSM based approach for the external plagiarism detection and learnt that VSM can better handle obfuscation but one has to carefully tackle the precision of the system. we plan to further investigate the issue to improve on precision. VSM based technique to pull candidate documents is very fast at the same time one has to go deep in the ranked list. Our external plagiarism detection system seriously needs parameter tuning which we plan to execute in near future. We also tried novel discourse markers based features along with some well known features and successfully detect intrinsic plagiarism. We would consider other features and techniques that help in removing false positives, for which we need to analyze the fact of how much uniform an author style can be when writing a document.

References

1. Parth Gupta and Sameer Rao and Prasenjit Majumder. External Plagiarism Detection: N-Gram Approach using Named Entity Recognizer: Lab Report for PAN at CLEF 2010. In Braschler et al. ISBN 978-88-904810-0-0.

2. G. Salton , A. Wong , C. S. Yang, A vector space model for automatic indexing, Communications of the ACM, v.18 n.11, p.613-620, Nov. 1975
3. Stamatatos, E.: Intrinsic Plagiarism Detection Using Character n-gram Profiles. In: 3rd PAN Workshop. Uncovering Plagiarism, Authorship and Social Software Misuse. pp. 38-46 (2009)
4. Mario Zechner,Markus Muhr,Roman Kern and Michael Granitzer.External and Intrinsic Plagiarism Detection Using Vector Space Models.In 3rd PAN Workshop Uncovering Plagiarism,Authorship and Social Software Misuse.pp.47-55(2009).
5. Markus Muhr, Roman Kern, Mario Zechner, and Michael Granitzer. External and Intrinsic Plagiarism Detection using a Cross-Lingual Retrieval and Segmentation System. Lab Report for PAN (2010).
6. Potthast M., Barrn-Cedeo A., Stein B., Rosso P. *An Evaluation Framework for Plagiarism Detection.* In: Proc. of the 23rd International Conference on Computational Linguistics, COLING-2010, Beijing, China, August 23-27, pp. 997-1005