Using Microsoft SQL Server platform for plagiarism detection

Vladislav Shcherbinin
American University of Nigeria
Lamido Zubairu way, Yola township by-pass,
PMB 2250, Yola, Nigeria
vladislav.scherbinin@gmail.com

Sergey Butakov
SolBridge International School of Business,
151-13 Samsung 1-Dong, Dong-gu, Daejeon,
300-814, South Korea
butakov@solbridge.ac.kr

Abstract: The paper presents an approach for plagiarism detection using Microsoft SQL Server platform in a large corpus of documents. The approach was used for participation in the first international plagiarism detection competition that was held as a part of PAN'09 workshop. The main advantages of the proposed approach are its high precision, good performance and readiness for deployment into a production environment with relatively low cost of the required third party software. The approach uses fingerprinting-based algorithm to compare documents and Levenstein's metric to markup plagiarized fragments in the texts.
Keywords: external plagiarism detection, Winnowing, document fingerprinting

1 Introduction

Digital plagiarism remains a burning issue both in academia and industry over the last two decades. Of course methods and tools of plagiarism uncovering have evolved a lot from the pioneering works on plagiarism uncovering in source codes in 1980s to web-enabled anti-plagiarism services of today.

Plagiarism detection methods at large can be split into two large groups: external document analysis methods and intrinsic plagiarism detection methods, or stylometry (Maurer, Kappe, & Zaka 2006). The method and software proposed in this paper aimed on the external plagiarism detection, e.g. revealing the text copied from other documents. The software was tested on the corpus of document provided for competition. The rest of the paper is organized as follows: the detailed description of the software platform and the detection process can be found in the second and third sections of the paper. Conclusion section summarizes the results and proposes directions for the future research.

2 Detection process

The document processing for the competition was performed by three nodes. Node 1 served as DBMS platform and Node 2 and Node 3 were used on the detection phase. The following subsections explain detection steps in details.

2.1 Loading and preprocessing of the documents

To perform the comparison on a large corpus of documents we decided to use the Winnowing, one of the well-known fingerprinting-based algorithms (Schleimer et al., 2003). According to this algorithm each document was substituted with a set of its hashes for the detection purposes.

The database designed to store documents and fingerprints consists of three tables: Folder, Document, and Fingerprint.

After loading documents and compiling their fingerprints the Fingerprint table was indexed with two indexes: one nonclustered index on hash value and document ID (index 1) and another clustered index on document ID, hash value and sequential number of a hash in the document (index 2). After the loading phase

Stein, Rosso, Stamatatos, Koppel, Agirre (Eds.): PAN'09, pp. 36-37, 2009.

the Fingerprint table was populated with 137,981,386 records. The most time consuming operation here was loading documents and compiling fingerprints.

2.2 Locating sources

The main objective of this step was to reduce the number of documents for comparison phase. This step selects all pairs of documents that share at least one fingerprint and stores these pairs in a table for more detailed analysis. After this step the table that links the pairs of possible matches in the documents was populated with only 44,532 records instead of 52,000,000 – possible number of pairs the search would have had to process if it compares all suspicious documents versus all source documents: 7214 * 7215 = 52,049,010. This step literally substituted the "one-vs-all" comparison with "one-vs-suspicions". As this step consists of only one query the better system performance could be achieved only by improving MS SQL Server hardware. This step uses index 1.

2.3 Detecting plagiarized passages

At this point all the required information is ready for the main step: detection of the common fragments in documents. The result from this step was used to identify exact plagiarized excerpts and to establish anchors for the further analysis. The main point here is the proper indexing of the Fingerprint table: on this step the clustered index created earlier (index 2) was used which provided the best possible execution plan.

After all common fingerprints have been identified and thus provided established anchors, the next task was to find common intervals for marking up the plagiarized passages. For better performance this process was distributed among two workstations (nodes 2 and 3), each running a console application performing the following steps:

1. Retrieve an unprocessed document from the Document table and corresponding records from the table that links it with possible sources.
2. For each record run the following steps:
    a. Execute the stored procedure to retrieve starting positions of the common excerpts.
    b. For each result skip forward character by character in both source and suspicious documents, while characters are equal. This will identify exact excerpt.
    c. Skip forward *n* characters, and compare excerpts using Levenstein's distance to identify near similar and obfuscated excerpts.
3. Save identified intervals into the DB.

Both nodes used several separate threads for this processing and each thread was processing a separate document, retrieved on the step 1 shown above. The detection time could be improved by increasing the computational power of the processing nodes (nodes 2 and 3) or by further increasing the number of nodes.

2.4 Compiling results

On the last step Microsoft SQL Server Integration Services was used to export information about detected plagiarism to XML files with the required format.

3 Conclusion

As the competition results indicate the proposed approach provides competitive results in terms of preciseness. Moreover it comes in the ready-to-deploy form that can be easily implemented on relatively inexpensive third party software (MS SQL Server). This will allow easy system integration with virtually any university-wide course management system. The required improvements to reduce the granularity of results are planned for implementation in the next version of the software. At this stage of the development the solution is publicly available for downloading as a desktop version at www.siberiasoft.info.

References

Maurer, H., Kappe F., Zaka B. (2006) Plagiarism – A Survey. Journal of Universal Computer Sciences, vol. 12, no. 8, pp. 1050 – 1084.

Schleimer S., Wilkerson D., and Aiken A. (2003). Winnowing: Local Algorithms for Document Fingerprinting. Proceedings of the ACM SIGMOD International Conference on Management of Data, pages 76-85, June 2003.