Experiments on Document Chunking and Query Formation for Plagiarism Source Retrieval

Notebook for PAN at CLEF 2014

Amit Prakash, Sujan kumar Saha

Department of Computer Science and Engineering, Birla Institute of Technology, India
aprakash@bitmesra.ac.in, sujan.kr.saha@gmail.com

Abstract. This paper presents the details of the system we prepare as a participant of the PAN 2014 task on 'Source Retrieval: Uncovering Plagiarism, Authorship, and Social Software Misuse'. Our work is focused on intelligent chunking of suspicious documents and a hybrid approach of query formation. A method based on term frequency and word co-occurrence is proposed to extract query terms from a non-overlapping chunk of topically related sentences. The queries are then submitted to the ChatNoir search API to retrieve documents that are likely to be the sources of plagiarism. Finally a snippet matching and duplicate download restriction based filtering technique reduces the number of downloads. The evaluation results of the PAN14 Source Retrieval task show that the performance of our system is highly promising. The f-measure accuracy of the system is .3871 with a recall of .5083 which is the highest among all the participants.

1 Introduction

The World Wide Web has become the most popular source of information. Exponential increase in the amount of information available on the web and improved access to this via the Internet has tremendous potential and a lot to offer in terms of services. Internet now is a virtual treasure trove of information about every subject known to man. However one of the major disadvantages of this ease of access of vast amount of information lead to a serious problem called plagiarism [1].

Plagiarism is defined as the act of using the ideas or work of another person or persons as if they were one’s own, without giving credit to the source [2]. Here the word “work” can be defined as variety of things which include ideas, words, opinion, etc. Anything that is seen as an unethical and unattributed use of another’s original creation can be defined as plagiarism [3]. However this definition is not always consistent, different industries follow their own standard to define plagiarism. Our work is concerned with the cases of natural language text plagiarism whose potential source is World Wide Web.

Reports suggest that the Internet has led to a dramatic increase in plagiarism over the past decade due to the easy availability of resources on the internet that allow plagiarists to find materials from which to copy and turn in as their own. Plagiarism is a serious problem in all levels of academia. Numerous studies show that the trend to copy existing information from Internet is increasing day-by-day among students [2] [4]. The survey of Pew Internet & American Life Project (2011) [5] reported that, 55 percent of college presidents accepted that there was a noticeable increase in the numbers of plagiarized works in their colleges. Of that 55 percent, 89 percent believe that computers and the Internet have played a major role in this trend.

Due to absence of controlled evaluation environment to compare results of the algorithms, plagiarism detection is still a challenging task. So far various conferences and shared tasks have been organized to deal with plagiarism problem. PAN [7] is one of them, which has been organizing an international competition on plagiarism detection since 2009. It provides a real world scenario and standardized evaluation framework for researchers to develop and evaluate their systems. We participated in source retrieval sub-task of PAN 2014 competition where the goal is to retrieve documents (candidate documents) which serve as possible source of plagiarism for a given plagiarized document (suspicious document) from a web like scenario. For evaluation of such systems five evaluation measures have been considered by the PAN organizers: 1) number of queries submitted, 2) number of web pages downloaded, 3) precision and recall of web pages downloaded regarding the actual sources, 4) number of queries until the first actual source is found, v) number of downloads until the first actual source is downloaded.

The system we develop is consists of four core modules namely, chunking, query generation, downloading and filtering. During the design of the individual modules we mainly focused on maintaining a high recall of the system. Additionally, we targeted to keep the number of queries and number of downloads as low as possible so that the system achieves moderate performance with respect to all the evaluation metrics. The detail of the system is discussed in this paper.

Our approach is mainly focused on intelligent chunking of documents and a hybrid approach of keyword extraction from them, using two well known term extraction strategies: term frequency and word co-occurrence. First, we split the suspicious documents in variable length chunks. From these chunks a subgroup of topically related sentences formed based on co-occurrences of top frequent words. We have extracted nouns from these subgroups to form queries of maximum 10 words. We optionally submit four queries per chunk to ChatNoir [8] search engine and download maximum 10 documents per query. To further reduce the retrieved documents set we have applied a download filter based on 5-gram similarity check with 500 character snippet. Evaluation using TIRA [7] experimentation platform shows that using an average work load our system retrieves more than 50% of plagiarism sources with an accuracy of 38.24%. The following sections give the detail of methods used in the development of our system.

2 Related Work

The research on plagiarism detection started with the detection of plagiarism in large piece of software codes [1]. As with the improvement of plagiarism cases in academics the researcher's interest shifted towards the plagiarism involving natural language texts. The research on natural language text plagiarism detection began in mid-1990 and has made a significant progress till date. The early researches were carried out on relatively small corpora consist of hundreds to few thousands of documents. However, now researchers consider the whole web as a possible source of plagiarism and generally use a search engine to retrieve the sources of plagiarized text. This leads to the development of online plagiarism detection services like plagiarism.org [9], turnitin.com [10] etc.

Compare to program code, detection of plagiarism in natural language text is a more challenging task due to the absence of formal syntax and ambiguity at various levels [1]. Natural language text can be plagiarized in number of ways. Beside simple copy and paste one can rearrange words, obfuscate or paraphrase the reused sentences. A lot of work has been done on simple copy paste detection, but still other problems have not received much attention.

The task of plagiarism detection has been divided into two main categories external plagiarism detection and intrinsic plagiarism detection [6]. In external plagiarism detection the contents of suspicious document is checked against a collection of external documents that have been used as source of plagiarism. On the other hand, in intrinsic plagiarism detection the plagiarized text is identified by investigating the changes in writing style within the same document. Since 2012, PAN separated the external and intrinsic plagiarism tasks. Intrinsic plagiarism detection migrated under author attribution task and external plagiarism detection task further divided into two subtasks source retrieval and detailed comparison.

A brief discussion of approaches used for source retrieval task can be found in the overview papers of previous PAN tracks [6]. Most approaches starts with the separation of large document text into smaller chunks. After that a keyword extraction method is applied on chunks to extract terms in order to formulate queries. Queries are formed in various ways so that it can retrieve the similar documents with maximum probability. This followed by a search controller which dynamically adjust the search based on the results of previously submitted queries. The final step of this process is download filtering. A download filter further reduces the document set returned by search engine by removing all the documents that are not worthwhile being compared in detail with suspicious document.

3 Methodology

Our approach involves four main steps: 1) Document Chunking, 2) Term Extraction, 3) Query Formation and Search Control, 4) Document Downloading and Filtering.

3.1 Document Chunking

A close analysis of suspicious documents show the text is categorized among various titles. Our document chunking strategy is based on the idea that the paragraphs under same title are topically related. We have considered text separated by two newline characters as a paragraph and a paragraph of length less than nine words as title. In order to form a chunk, first we partition the document text into paragraphs. In next step we merge these paragraphs to form non-overlapping chunks of variable lengths. We have used following two strategies for conditional merging of paragraphs.

1) Starting from the first paragraph, whenever we encounter a title or a paragraph of more than 100 words we form a new chunk. However, we avoid this when such paragraphs just proceeds after a title.

2) We merge the proceeding paragraphs in existing chunk till we don't encounter a paragraph necessary for creating a new chunk discussed above. While merging paragraphs we continuously check for the size of chunk. In case the size exceeds 200 words we stop merging paragraphs in existing chunk and create a new chunk form next paragraph.

3.2 Keyword Extraction

Our keyword extraction approach is based on two well known term extraction strategies term frequency and word co-occurrence. This section describes these approaches in detail.

The term frequency reflects the importance of each word of the document by counting their number of occurrences. The top frequent words (after removing the stop-words) can be used to define the center of attraction in a particular piece of text. These are the words around which the whole text is written. Based on this hypothesis we have extracted top 5 frequent words of a document and named it *document level tf* and the most frequent word of each chunk and referred it as *chunk level tf*. Before extracting frequent words we preprocess the documents by removing the stop-words and the words of length less than three characters.

We have used co-occurrence to extract sentences from chunk in order to form subgroups. For each chunk we form two subgroups based on the co-occurrence of frequent words extracted earlier. The first subgroup has been formed using the word co-occurrences of document level tf only. Whenever two or more words of document level tf co-occur in a sentence we include that sentence in subgroup. In case the subgroup contains less than 5 sentences we include the sentences that contain any word of document level tf.

We form second subgroup based on the co-occurrences of chunk level tf word with document level tf words. Whenever the most frequent word of chunk co-occurs with any document level tf words in a sentence we include that sentence in subgroup. In case the subgroup contains less than 5 sentences we include the sentences containing the word of chunk level tf only.

We POS tag these subgroups using Maximum Entropy Part-of-Speech Tagger [11] and extract all the nouns as keywords. The reason behind taking only the nouns is to minimize the number of keywords and the hypothesis that nouns are sufficient enough define a piece of text uniquely in most of the cases.

3.3 Query Formation and search control

Forming query for ChatNoir search engine is a challenging task due to the fact that ChatNoir allows maximum 10 words per query to retrieve the sources. We form maximum four queries from each chunk and conditionally submit them to ChatNoir in order to minimize the workload. Before submitting each query we ensure that the 60% of current query terms differs from any previously submitted query otherwise we drop the current query.

If the subgroups return at least one noun we form a query from them. We have formed first two queries using this strategy taking the first 10 nouns extracted from subgroups. In case these queries contain less than 6 words, we append the nouns returned by tagging the chunk itself to make the query of 10 unique words. We form the third query from the nouns returned by tagging the chunk itself and submit it only if first query couldn't be formed or returns no result. The fourth query constitute the top 10 frequent words of a chunk and we submit it only if second query couldn't be constructed or dropped.

3.4 Document Downloading and Filtering

`Number of downloads' is considered as one of the metrics for evaluation of the system. Therefore we aim to keep the 'number of downloads' as low as possible. To achieve this we have adopted a two-stage approach. In the first stage we use a snippet based pre-checking of the retrieved documents. Initially we have retrieved 10 candidate documents for each query. For each of these documents we generate a 500 character snippet using ChatNoir's snippet generator facility. Then we check whether the snippet is containing any 5-gram from the suspicious document. If not, then we reject the document. Otherwise we log and download the corresponding document using ChatNoir API for detailed comparison. As a second stage, we restrict the system from duplicate download. We observe that many of the queries share common terms; that may lead to same download from two different queries. Once a document is downloaded by one query, it is not anticipated to be downloaded again by another query. To restrict this we maintain a list of downloaded documents which is checked before downloading the documents. A document is downloaded only if the corresponding entry is not there in the list.

4 Evaluations and Performance

We implemented our approach in Java programming language with the help of OpenNLP [12] natural language processing library. During system development we performed all the experiments on training corpus [13] only. The developed system then deployed on virtual machine for evaluation on test corpus [13] using TIRA experimentation platform. The test data was not revealed to participants in order to avoid the result optimization based on data set.

Table 1. PAN 2014 Source retrieval results

Users	Downloads	Downloads Until First Detection	F-Measure	No Detection	Precision	Queries	Queries Until First Detection	Recall	Runtime
elizalde14	33.2	3.9	0.3432	7	0.4002	54.5	16.4	0.3860	04:02:00
kong14	207.1	24.9	0.1197	6	0.0756	83.5	85.7	0.4820	24:03:31
prakash14	38.76	3.76	0.3871	7	0.3824	59.95	8.08	0.5083	19:47:45
suchomel14	237.3	38.6	0.1062	2	0.0775	19.5	3.1	0.3984	45:42:06
williams14	14.41	2.33	0.4726	4	0.5716	117.13	18.82	0.4762	39:44:11
zubarev14	18.61	2.25	0.4483	3	0.5378	37.03	5.39	0.4475	40:42:18

Table 1, shows the performance of systems participated in source retrieval sub-task of PAN 2014. Our approach achieved precision and recall of 0.5083 and 0.3824 respectively. The recall is highest among all the participants and we got fourth position in terms of precision. However, we achieved third position in terms of f-measure which is considered as the tradeoff between precision and recall.

We submitted an average 59.95 queries to download 38.76 sources per suspicious document. We formed four queries per chunk but their conditional submission to search engine further reduced the total number of queries submitted. As we retrieves 10 documents per query but an average 38.76 downloads per document show that our download filter performs quite well.

5 Conclusion

In this paper we have presented an approach to retrieve possible sources of reused text for a given plagiarized document. We have introduced an intelligent way of document chunking and a combination of two well known keyword extraction strategies to extract query terms. During the development of our system we experimented on the various parameters used, such as the title size, the size of paragraph need to create a new chunk, the size limit of chunk and the POS tags to be extracted after tagging the subgroups.

Our system performance is evaluated on PAN 14 test data set and compared with the systems of other participants. Results show that our system's performance is best in terms of finding the reused sources using an average workload. The plagiarism detection method we proposed does minimal computations and performs the task at a speed suitable enough for practical applications.

However, there are certain possibilities to improve the performance of our system. Our method succeeded in terms of recall, but we need to further reduce the total workload. For this purpose a deeper investigation into query formation and download filtering is required. A better performance can be achieved by using the advanced functionalities offered by ChatNoir search engine. These include the batch query service and addition parameters returned in search results. Furthermore our plan is to extend our approach to deal with cross-language plagiarism cases.

6 References

1. Clough, P.: Old and new challenges in automatic plagiarism detection, Plagiarism Advisory Service, University of Sheffield, (Feb 2003)
2. Maurer, H., Kappe, F., Zaka, B.: Plagiarism - a survey. Journal of Universal Computer Science 12 no. 8, 1050 – 1084 (2006).
3. Bailey, J., The Difference Between Copyright Infringement and Plagiarism, Plagiarism Today, (Oct 2013)
4. Mccabe, D.L.: Cheating among college and university students: A North American perspective. International Journal for Educational Integrity 1(1), 1–11 (2004)
5. Parker, K., Lenhart, A., Moore, K.: The Digital Revolution and Higher Education, Pew Research Center, (Aug 2011)
6. Potthast, M., Hagen, M., Gollub, T., Tippmann, M., Kiesel, J., Rosso, P., Stamatatos, E. and Stein, B., Overview of the 5th International Competition on Plagiarism Detection. In Forner, P., Navigli, R. and Tufis, D., editors, Working Notes Papers of the CLEF 2013 Evaluation Labs, ISBN 978-88-904810-3-1,(Sep 2013)
7. Gollub, T., Potthast, M., Beyer, A., Busse, M., Rangel, F., Rosso, P., Stamatatos, E. and Stein, B.: Recent Trends in Digital Text Forensics and its Evaluation. In Forner, P., Müller, H., Paredes, R., Rosso, P. and Stein, B., editors, Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 4th International Conference of the CLEF Initiative (CLEF 13), Springer. ISBN 978-3-642-40801-4,(Sep 2013)
8. Potthast, M., Hagen, M., Stein, B., Graßegger, J., Michel, M., Tippmann, M., Welsch, C.: ChatNoir: A Search Engine for the ClueWeb09 Corpus. In: Proceedings of the 35th International ACM Conference on Research and Development in Information Retrieval (SIGIR 12). p. 1004 (Aug 2012)
9. www.plagiarism.org
10. www.turnitin.com
11. Toutanova, K. and Manning, C.D.: Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000), 63-70 (2000)
12. Kottmann, J., Margulies, B., Ingersoll, G., Drost, I., Kosin, J., Baldridge, J., Goetz, T., Morton, T., Silva, W., Autayeu, A., Galitsky, B.: Apache opennlp. Online (May 2011), www.opennlp.apache.org
13. Potthast, M., Stein, B., Barrón-Cedeño, A. and Rosso, P.:An Evaluation Framework for Plagiarism Detection. In 23rd International Conference on Computational Linguistics (COLING 10), Association for Computational Linguistics, (Aug 2010).