R. Maluleka, I.V. Sochenkov

Query Formulation for Source Retrieval based on Named Entities and N-grams Extraction[1]

Abstract. This paper presents an approach for the source retrieval task using two distinct keyphrase extraction strategies, namely n-grams from chunked text and named entities. The proposed approach was evaluated on TIRA and performed well against other participants of PAN CLEF.

Keywords: source retrieval, named entity extraction, plagiarism detection.

Introduction

For the as long as people have been creating original works, there have been imitators. Whilst imitation is said to be the sincerest form of flattery, it is a serious problem in the areas of research and academia. With the spread of internet access, it is now easier than ever to plagiarise from a plethora of sources. In addition to this, the availability of online translators and synonymizers has made it possible to obfuscate plagiarism with little effort. For these reasons it is simply not practical to manually detect most instances of plagiarism. In fact, plagiarism involving paraphrasing and translation in particular still presents a formidable challenge in the active research area of automatic plagiarism detection [1]. There exist several different types for plagiarism, ranging from improper citation to cut-and-paste copying of another's work. Obfuscated plagiarism is the most difficult to detect as the plagiarist has attempted to transform the appropriated work enough to make it seem distinct from the original. The effectiveness of a plagiarism detection software can be measured by the kinds of plagiarism it can identify. This software drastically reduces the amount of effort, and time spent, in performing the detecting plagiarism. Automatic plagiarism detection has been an active research area since the 1970's and has attracted increased interest in recent years as advancements in computational speed, and access to sophisticated search engines, allow increasingly complex approaches to be implemented. These tools allow academics to quickly retrieve potential sources for a suspicious passages of text, which can then be further analysed. Making the task significantly more tractable.

This paper describes an algorithm for identifying key features of a suspicious document, building on the approaches of teams that competed in the PAN international competition on plagiarism detection [1-4]. The method closely follows the approach described by Williams, Chen, Choudhury and Giles in their 2013 paper [5], while also incorporating named entity n-grams similar to those used by Elizalde [6].

1. Related Work

The Uncovering Plagiarism, Authorship and Social Software Misuse Lab (PAN) has been held annually since 2007, and beginning in 2009 has been part of the Conference and Labs of the Evaluation Forum (CLEF). PAN aims to answers the questions: given a document, is is original? who wrote it? what are the author's traits? through experimentation on shared tasks. With the goal of

[1]The reported study was partially funded by RFBR, according to the research project No. 16-37-60048 mol_a_dk.

providing sustainable and reproducible evaluations of state-of-the-art-algorithms. The approach describes in this paper applies to the source retrieval sub-task of plagiarism detection. This task entails retrieving sources, given a potentially plagiarised document.

The framework of the PAN evaluation lab aims to emulate the real-life scenario of text reuse, where a plagiarist uses a web search engine to find source documents. To achieve this, organisers created a crowd-sourced corpus of manually written documents with instances of plagiarism. Instead of using the actual World Wide Web, authors were asked to use a static web-crawl of the web (the ClueWeb09[2]). They access ClueWeb through search engines (Indri[3] and ChatNoir [7]), and can browse it as if it were the real thing. This same setup is used for evaluation. Participants in the evaluation lab are then given API access to these search engines and a subset of documents from the plagiarism corpus on which to train their software. This allows participants to design software within a setup that is very similar to the real-world task of retrieving sources of plagiarism by programmatically accessing a web search engine, but with the reproducibility of working in a static environment. The task is then to retrieve source documents while minimising the retrieval cost.

Participants in the PAN/CLEF evaluation lab are required to submit their software on the TIRA experimentation platform [8], allowing organisers to compare the current year's submissions to those submitted in previous years (since 2012). Thus the outcome of the PAN evaluation labs is performance data about different approaches to the shared tasks and, additionally, a collection of state-of-the-art implementations of these assorted approaches [9].

One of the best performing software in the source retrieval task in 2013 and 2014 was that of Williams et al [5, 10]. Even though they did not submit a new version in 2015, their software still went unmatched in the 2015 lab. William's approach in 2013 made use of an unsupervised ranking method to rank the results returned by a search engine by their similarity with the suspicious document. In 2014 they switched to supervised method for ranking results. However, F_1 score increased insignificantly. Elizalde's 2013 approach makes use of a novel idea of extracting named entities across a document in an attempt to match highly obfuscated plagiarism. These named entity queries are of interest because they could compliment, and potentially improve, William's approach.

2. The Proposed Approach

The approach consists of several stages, namely: chunking, key-phrase extraction, query formulation, and download filtering.

Chunking: The suspicious document is first segmented into paragraphs of 5 sentences each. Each paragraph is pre-processed, removing all non-alphabetic characters.

Keyphrase Extraction and Query Formulation: Two different methods are employed in forming keyphrases. The first attempt to find the most important features of the entire document, while the second forms queries based on individual chunks.

Named entity queries: Named Entities are identified over the whole text. They are then ranked in descending order of length. The longest are submitted as-is as queries to the search engine. As noted by Elizalde [6], the rationale behind is that the named entities are unlikely to change even if paraphrasing has been used to obfuscate plagiarism.

Chunk based queries: Each sentence in each paragraph is tokenized. All stopwords are removed, and only verbs, nouns, and adjectives are retained. Queries are formed by concatenating sequences of tokens to form disjunct sequential 10-grams. The first three 10-grams from each paragraph are submitted to the search engine.

Download Filtering: The ChatNoir search engine [7] allows one to request a snippet, of up to five hundred characters, of a specific document. Documents are only downloaded if they share at least five word 5-grams with the suspicious document.

Algorithm 1 shows the algorithm for the introduced approach. The algorithm was implemented using Python programming language, making use of the following non-standard libraries: BeautifulSoup4[4], NLTK[5], NumPy[6] and Shingling[7]. The implementation is publicly available through PAN's online code repository[8].

[2] http://lemurproject.org/clueweb09.php
[3] http://lemurproject.org/indri.php

[4] http://www.crummy.com/software/BeautifulSoup/bs4/
[5] http://www.nltk.org/
[6] http://www.numpy.org/
[7] https://pypi.python.org/pypi/shingling
[8] https://github.com/pan-webis-de/maluleka16

Algorithm 1 Source Retrieval Approach

```
1: procedure SOURCERET(text)
2:     NEs ← getNamedEntities(text)
3:     sources ← processNEResults(NEs)
4:     paragraphs ← splitText(text)
5:     for all p in paragraphs do
6:         p ← preprocess(p)
7:         queries ← extractTopQueries(p)
8:         for all q ∈ queries do
9:             results ← submitQueries(q)
10:        end for
11:        results ← rankResults(results)
12:        for all result in results do
13:            snippet ← getSnippet(result)
14:            if similarity(snippet) ≥ min_Sim then
15:                if previousSource(result) = False then
16:                    sources ← Download(result)
17:                end if
18:            end if
19:        end for
20:    end for
21: end procedure
```

3. Evaluation

There is no unified performance measure for a plagiarism detection task. Thus an approach is judged based on several scores taken as averages over a dataset [1]:

Number of queries submitted; number of web pages downloaded; precision and recall of web pages downloaded regarding actual sources of a suspicious document; number of queries until the first actual source is found; and the number of downloads until the first actual source is downloaded. The first three measures capture the overall behaviour of a system and measures. The last two assess the time to first result. The quality of identifying reused passages between documents is not taken into account here, however retrieving duplicates of a source document is considered a true positive, whereas retrieving more than one duplicate of a source document does not improve performance.

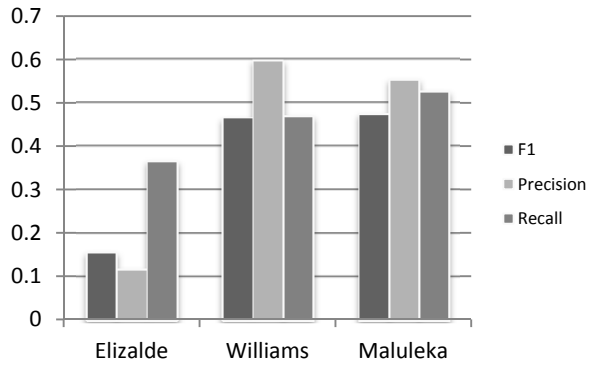

Fig. 1. Comparison of Related Approaches – Key Measures

Our algorithm was evaluated on TIRA against the PAN 2014 source retrieval test dataset 2. The same dataset used in the 2015 labs, allowing us to directly compare our results to the those of the competition participants. Table 1 shows a detailed comparison of our approach to those of Williams and Elizalde using data from the results of the PAN 2015 source retrieval task [9]. Figure 1 shows a graphical comparison using only the F_1, precision and recall measures.

From the results data we can conclude that considering named entity queries does indeed improve the approach suggested by Williams. As one can see from Table 1, the presented algorithm has comparable precision and recall, and the highest recall and overall F_1 score of the three approaches. It fact, it currently holds the top F_1 score of all evaluated approaches[9] (see Table 2).

Conclusion

This article suggests an approach to the source retrieval using a combination of two distinct keyphrase extraction strategies, namely 10-grams from chunks and named entities. The evaluation results show that this approach achieves a good compromise between precision and recall.

The introduced approach is based on the results of work done by participants in the PAN lab shared tasks. Certainly, any number of approaches could be derived from the many approaches that have been implemented as part of the task evaluations. This paper seeks to suggest a high performing approach, and make a state-of-the-art implementation publicly available to aid other researchers and practitioners.

[9] http://www.tira.io/task/source-retrieval/dataset/pan14-source-retrieval-test-dataset2-2014-05-14/

Table 1. Comparison of Related Approaches

Team	F1 Measure	Prec.	Rec.	Queries	Dwlds	Queries to 1st Detect.	Dwlds to 1st Detect.	No Detect.
Elizalde13	0.15622	0.11845	0.36621	**41.6**	83.9	**18.0**	18.2	**4**
Williams13	0.46597	**0.59656**	0.46919	117.1	**12.4**	23.3	**2.2**	7
Maluleka16	**0.47458**	0.55403	**0.52677**	138.4	18.7	20.9	**2.2**	6

Table 2. Comparison of Top Performing Approaches

Team	F1 Measure	Prec.	Rec.	Queries	Dwlds	Queries to 1st Detect.	Dwlds to 1st Detect.	No Detect.
Gillam13	0.05545	0.03831	0.14813	15.7	86.8	16.1	28.6	34
Haggag13	0.38303	**0.67290**	0.31370	41.7	**5.2**	13.9	**1.4**	12
Kong13	0.01119	0.00587	**0.58559**	47.9	5185.3	**2.5**	210.2	**0**
Maluleka16	**0.47458**	0.55403	0.52677	138.4	18.7	20.9	2.2	6

This software can be readily applied to real-world plagiarism detection; as modern search engines provide similar features to those used in experimentation. There is, of course, room for improvement. Rather than considering a small number of top results returned by the search engine, which is done for the sake of expediting experimentation, we could consider many more results. Indeed, [3] argues that this might improve performance with little added cost.

References

1. Potthast, M. et al. Overview of the 5th international competition on plagiarism detection. In: CLEF Conference on Multilingual and Multimodal Information Access Evaluation. CELCT. 2013, pp. 301-331.
2. Potthast, M. et al. Overview of the 4th International Competition on Plagiarism Detection. In: CLEF (Online Working Notes/Labs/Workshop). 2012.
3. Potthast, M. et al. Overview of the 6th International Competition on Plagiarism Detection. In: Working Notes Papers of the CLEF 2014 Evaluation Labs. Ed. by Cappellato, L. et al. CEUR Workshop Proceedings. CLEF and CEUR-WS.org, Sept. 2014.
4. Stamatatos, E. et al. Overview of the PAN/CLEF 2015 Evaluation Lab. In: Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 6th International Conference of the CLEF Initiative (CLEF 15). Springer, Berlin Heidelberg New York. 2015.
5. Williams, K. et al. Unsupervised Ranking for Plagiarism Source Retrieval. In: Notebook for PAN at CLEF 2013 (2013).
6. Elizalde, V. Using statistic and semantic analysis to detect plagiarism. In: CLEF (Online Working Notes/Labs/Workshop). 2013.
7. Potthast, M. et al. ChatNoir: a search engine for the ClueWeb09 corpus. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. ACM. 2012, pp. 1004-1004.
8. Gollub, T., Stein, B., and Burrows, S. Ousting ivory tower research: towards a web framework for providing experiments as a service. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. ACM. 2012, pp. 1125-1126.
9. Hagen, M., Potthast, M., and Stein, B. \Source Retrieval for Plagiarism Detection from Large Web Corpora: Recent Approaches". In: Working Notes Papers of the CLEF (2015), pp. 1613-0073.
10. Williams, K., Chen, H., and Giles, C. Supervised Ranking for Plagiarism Source Retrieval Notebook for PAN at CLEF 2014. In: Cappellato, L., Ferro, N., Halvey, M., Kraaij, W. (eds.) CLEF 2014 Evaluation Labs and Workshop - Working Notes Papers, 15-18 September, Sheffield, UK. CEUR Workshop Proceedings, CEUR- WS.org (Sept. 2014).

Малулека Рулани. Аспирант Российского университета дружбы народов (РУДН). Окончил РУДН в 2016 году. Область научных интересов: интеллектуальные методы поиска и анализа информации, методы поиска текстовых заимствований. E-mail: rhumaluleka@gmail.com

Соченков Илья Владимирович. Заместитель заведующего лабораторией ИСА ФИЦ ИУ РАН, Доцент кафедры информационных технологий, ст. научный сотрудник Российского университета дружбы народов (РУДН). Окончил РУДН в 2009 году. Кандидат физико-математических наук. Автор 50 печатных работ. Область научных интересов: интеллектуальные методы поиска и анализа информации, обработка больших массивов данных, контентная фильтрация, компьютерная лингвистика, распознавание образов. E-mail: isochenkov@sci.pfu.edu.ru