CopyCaptor : Plagiarized Source Retrieval System using Global word frequency and Local feedback

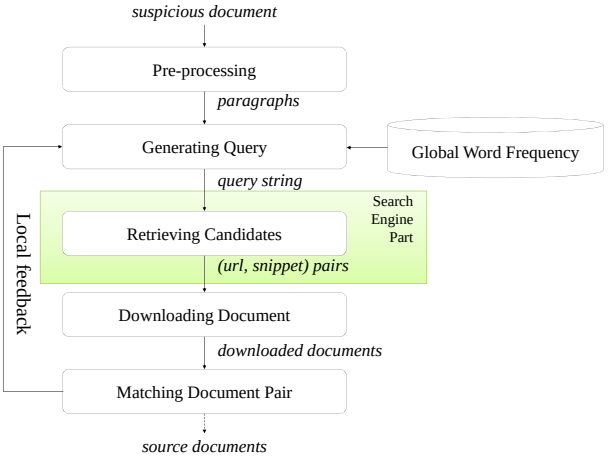
Taemin Lee, Jeongmin Chae, Kinam Park, and Soonyoung Jung

Introduction

- In the information age, research articles become easy to access at any time, and anywhere via open internet network. This environmental change made plagiarize others works easier than ever.
- Plagiarized documents are made from web resources (at least, 4.63 billion web pages). Therefore, a plagiarism detection method should consider characteristic of web resources.
- Since 2012, PAN (Plagiarism analysis, Authorship identification and Near-duplicate detection) research community made two core tasks 'source retrieval' and 'text alignment' for plagiarism detection.
- 'Source retrieval' is the key task to solve detecting plagiarized documents in consideration of web resources. The task focuses finding source documents from the given suspicious document using a web search engine.
- In this paper, we propose a plagiarized source retrieval system called 'CopyCaptor' which uses global word frequency and local feedback.

Method

Framework of CopyCaptor



- In the 'Pre-processing', suspicious document divided into paragraphs, and each word in paragraphs are tokenized and stemmed. Also, stop-words are removed.
- For the 'Retrieving candidate', our system uses Indri search engine and also retrieve snippet of candidate documents using ChatNoir API. If our system seen the same snippet beforehand or snippet has no words, its URL is discarded.
- In the 'Downloading document', we download top-k URLs from candidates and also download URLs which frequently appeared in candidates URLs from different query strings.
- In the 'Matching document pair', we align (suspicious document, download document) pairs using simple n-gram match method. If the matching ratio (how many share the same n-gram between suspicious and a download document pair) is over some threshold (e.g. 5% or 100 words), we accept it as a source document.
- When the most part of the suspicious document appeared in source documents or a number of querying on a suspicious document is over the number of paragraphs, the system stop retry querying and returns gathered source documents.

Generating query using global word frequency and local feedback

- Our generating query method acquired on some heuristics from an analysis of PAN 13' training data set. We set up three heuristics as follows:
  1. The most unique query is the best query.
  2. A query should be differ from the previously executed queries.
  3. A query formed with contiguous words in a phrase
- To find out 'most unique query', we define the uniqueness of a query as follow:

Definition 1. The uniqueness of a Query: Given a query formed with words Q_W (w_1, w_2 … w_n), and global frequency of words Q_{GWF} (GWF[w_1], GWF[w_2], …, GWF[w_n]), we will say uniqueness of a query is inverse proportional to the product of Q_{GWF}.

- GWF(Global Word Frequency) is a dictionary that contains a word as a key and an occurrence of a word in different documents on a very large corpus (e.g. Google N-gram) as a value. Lower occurrence means more uniqueness on a corpus.
- We get a local feedback from previous executed queries to generate a more effective query.
- Because a suspicious documents comply with multiple source documents and the source documents have different words, we prefer choosing a word not exists in previous query string and not exists in match words from 'Matching Document Pair' process.
- Query strings are made by contiguous k words in our system.

Experiment

- Using the system, we evaluated of the performance of the proposed system with 'Source Retrieval' task on PAN'13 corpora.
- PAN'13 corpora contain 40 suspicious documents for training and 58 suspicious documents for the test.
- We used training corpus to find out appropriate parameters only. We used parameters for the system as follows: *number of download documents for a query* = 2, *number of query words* = 8, *number of n of n-gram* = 4.
- Our system uses Indri search engines built on ClueWeb09 corpus which contains approx. 1 billion web pages.
- Overall, CopyCaptor system achieved 0.34 F1 score in source retrieval task.

Retrieval Performance			Workload		Time to 1st Detection	
F1	Precision	Recall	Queries	Downloads	Queries	Downloads
0.35	0.50	0.33	44.04	11.16	7.74	1.72

Conclusion

- In this paper, we design and implemented a system called CopyCaptor for source retrieval task on PAN'13.
- Retrieval performance of CopyCaptor shows that the system is based on simple heuristics but it well suited for solving the problem.
- However, also results shows that the research in this field is not yet conquered.
- Furthermore, The performance of our proposed system will be improved by applying of the better text alignment algorithm because matching results from text alignment affects query generation by local feedback.

Dept. of Computer Science Education, Korea University, Republic of Korea
Withcat, Knowledge Engineering Lab. (http://withcat.net)

Contact: Taemin Lee (taeminlee@korea.ac.kr)