Plagiarism Detection Based on Citation Contexts

Sidik Soleman[1,a)]   Atsushi Fujii[1]

Abstract: Plagiarism, which violates another person's intellectual property, is also problematic in academia. We model the plagiarism for academic literature, by means of the similarity between textual contents and citation relations. We also propose our method for plagiarism detection and evaluate its effectiveness.

1. Introduction

Digital archives for academic publications have enabled us to efficiently access a large volume of scientific information. However, its misuse and misconduct have of late become a crucial problem. Plagiarism is "the act of using another person's words or ideas without giving credit to that person"[*1], which results in discouraging innovation and losing trust in the scientific research community. To alleviate this problem, a number of methods for detecting plagiarisms specifically for academic publications have been proposed.

In a broad sense, plagiarism detection is a task to identify whether a document in question was produced by means of plagiarism or not, and is often requested to present one or more source documents as evidences for the plagiarism. However, in this paper we consider only cases where an input document is a plagiarized one and focus only on identifying one or more source documents for the input document.

As with an adversarial information processing like filtering spam e-mails, a person who conducts plagiarism, or a plagiarist for short, usually intends to hide the plagiarism, for example, by means of editing and summarizing source documents. As a result, plagiarism detection is a cat-and-mouse game between plagiarists and people who develop plagiarism detection systems.

Whereas the above scenario is associated with intentional plagiarisms, detecting unintentional plagiarisms are also important to avoid innocent mistakes. Fang et al. [3] investigated approximately 2 000 papers that were once indexed by PubMed but retracted later and found that 9.8% of them were retracted due to being judged as a plagiarized paper. Irrespective whether those papers are associated with intentional or unintentional plagiarism, effective methods for plagiarism detection will have a significant impact on our society.

2. Related Work

Many methods have been proposed by researchers for detecting plagiarism. Related to scenario of PD that we focus on, Stamatatos et al. [7] used n-gram based on stopword to search for source documents of an input document. Although stopwords, such as *the, of, a, on, in,* etc, do not have important meaning, they argued that the pattern of stopwords do not change in plagiarized document, because plagiarist tend to change words that have synonym.

HaCohen-Kerner et al. [5] compared various fingerprinting methods based on word n-gram. They also compare abstract and reference section of documents, arguing that these sections are important in scientific literature. They reported that comparing only reference section produced many false positives in PD. It means that two documents cite the same papers and one of them is judged as plagiarism, but actually it is not plagiarism [5]. As text modification methods such as paraphrase are often used by plagiarist, in order to handle this, Chong et al. [2] compared documents by generalizing word based on sysnsets using a lexical database.

Another method, which uses structural information/component (e.g. *introduction, method, evaluation section,* etc.) of scientific literature, is proposed by Alzahrani et al. [1]. They argued that the term distribution indicates the importance of structural component. Hence, the term distribution in structural components is used to estimate a weight that describes the importance of a structural component [1]. In addition, they used this weight to re-weight terms in input document and documents in collection during their comparisons.

Gipp et al. [4] introduced PD based on citation, inspired from bibliographic coupling. They compared citation anchor patterns that spread out in both input document and document in collection as bag of anchors. The latest work that we are aware of, is proposed by Pertile et al. [6]. They combined textual similarity and citation based PD.

With the respect to citation relation, there are two types of usage. First, the existence of citation relation is to cancel the plagiarism decision [1]. Hence, if input document cites a document in collection, this document is not considered as source of plagiarism of the input document.

Second, citation relations are used to decide whether a docu-

---

[1] Tokyo Institute of Technology, Meguro–ku, Tokyo 152–8550, Japan
[a)] soleman.s.aa(at)m.titech.ac.jp

[*1] https://www.merriam-webster.com/dictionary/plagiarism

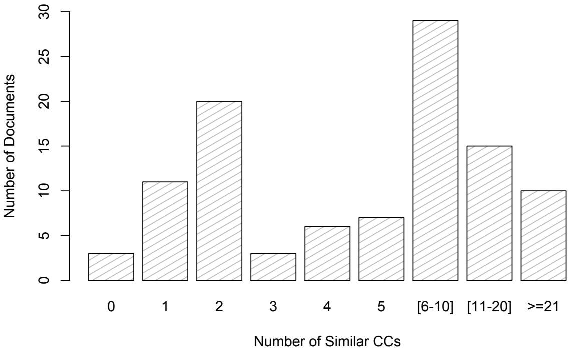

Fig. 1 Number of documents based on similar CCs

ment in collection is the source of plagiarism of an input document. Therefore, if the document and the input document cite similar documents, the document is considered as the source of plagiarism. This second type of usage is reflected in [5] [4] [6].

As mentioned by HaCohen-Kerner et al., [5], the second type of citation relation usage may produce false positives. One of the explanations why the document in collection and the input document are considered as false positive, is that both documents cite similar documents, but for different arguments. It means that their citation contexts are different. Therefore, in this paper, we propose to use citation contexts (CCs) for PD. Hence, a document in collection is considered as the source of plagiarism of an input document, if both of them use similar citation contexts in their documents. To the best of our knowledge, there has not been any attempt to use CCs for PD.

3. Citation Contexts Similarity Investigation

Because we propose to use CCs for PD, first, we have to investigate to what extent, plagiarized document and its source documents have similar CCs. We analysed 100 input documents that have been manually judged as positive to be aware of plagiarism, from PD dataset [6]. We found that 97 of these input documents are containing similar CCs with the candidate source documents. CC is fragment of document, which cites another document. The following text fragment is an example of CC from [6], which cites Alzahrani et al. [1]:

"... Taking the structure of a scientific paper into consideration was central to the approach by Alzahrani et al. (2012). ..."

Because there is possibility that CC is edited by plagiarist, we use cosine similarity to measure the similarity between two CCs. Two CCs are considered as similar, if their similarity scores are more than equal to 0.8.

Furthermore, we counted the number of CCs that are similar to the candidate source documents for each input document, as we presented in Figure 1. In average, the similar CCs are 11 between the input document and the candidate source document, and in some input documents, the similar CCs can be more than 21.

4. Proposed Approach

In order to compare input document and documents in document collection, we performed several processes for those documents. The following processes are used in our proposed approach:

(1) **Sentence classification**: because our proposed approach is based on CC, we have to split document into two fragments, that are CC and non-CC, by performing classification. We classify a sentence in document whether it is CC by the existence of citation anchor. Therefore, if a sentence contains citation anchor, it is CC, otherwise, it is non-CC. We use regular expression to recognise the citation anchor.

It looks simple, but there are many formats of citation anchors in document. Moreover, there is format of citation anchor that may also appear in non-CC. Hence, this classification may also produce false positive, meaning that the sentence is not CC but idenfified as CC. The following citation anchor formats and their examples are captured in our regular expression:

- Combination of author name and year of publication: *(author, 2010), (author, 2010a), (author1, 2010; author2, 2010b), author (2010), author (2010a), [author, 2010], [author, 2010a],* and *[author, 2010; author, 2010b].*
- Combination of author name, year of publication, and page number: *(author, 2010, p.1), (author, 2010, para.1),* and *(author, 2010, p.i).*
- Citation anchor is a key or a number of document identification in reference list: *[1], [LIZ2], [LIZ2a],* and *(1).*
- Combination of author name, year of publication, and a number of document identification in reference list: *[author, 2010 (1)]*

(2) **Lowercasing**

(3) **Stopword removal**

(4) **Stemming**

(5) **Document comparison**: input document and document in document collection is compared for each fragment, independently. Hence, for each document comparison, we have two similarity scores from CC and non-CC.

In order to compare two fragments, they are converted to vector based on term frequency in the corresponding fragment and inverted document frequency in the document collection (TFxIDF). Therefore, the weight for term t in fragment with type c is defined as follow:

$$w_{t,c} = f_{t,c} \; log \frac{N}{n_t} \tag{1}$$

with

- $f_{t,c}$: total number of term t that appears in fragment with type c, $c \in$ {CC, non-CC}.
- N: total number of documents in document collection.
- n_t: total number of documents in document collection that contain term t.

Then, two fragments with type c from input document i and document j from document collection, are compared by using the following formula:

$$sim(d_{c,i}, d_{c,j}) = \frac{\sum_{t=0}^{N} w_{t,c,i} \; w_{t,c,j}}{\sqrt{\sum_{t=0}^{N} w_{t,c,i}^2}\sqrt{\sum_{t=0}^{N} w_{t,c,j}^2}} \tag{2}$$

Finally, we model document similarity for PD as combination of two similarities that are CC and non-CC. Therefore, the final document similarity score between input document i and document j from document collection is formulated as:

$$score(d_i, d_j) = \alpha sim(d_{CC,i}, d_{CC,j}) + (1 - \alpha)\ sim(d_{non-CC,i}, d_{non-CC,j}) \quad (3)$$

with α is constant between 0 to 1.
Therefore by using α, we are able to prioritize CC over non-CC, or vice versa in this model.

5. Experiment

5.1 Dataset

There are several existing datasets for PD. However, not all these datasets are suitable for our research purpose, because only some of these datasets have citation relations and use scientific literature as document collection. We use two types of dataset for evaluation, namely auto-simulated and manually judge dataset.

The auto-simulated dataset is produced and used by Alzahrani et al. [1] for PD. Because it is difficult to obtain set of documents that is verified to be plagiarism for the purpose of research, they constructed documents by means of plagiarism automatically. They simulated four aspects of plagiarism for every constructed document. First, they may use more than one source document for one constructed document. Second, texts from different parts of source document are used in one constructed document and third, these texts are obfuscated using some text modification methods [1]. The last, they controlled the length of text fragment that is plagiarism in constructed document.

To obfuscate text fragments from source document, Alzahrani et al. [1] used: verbatim copy-paste, sentence and word shuffling, part-of-speech based word shuffling, word insertion and deletion, synonym replacement using a lexical database, back-translation (e.g. *English-Japanese-English*), double back-translation, and auto-summarization. They may also combine auto-summarization with other text modification methods. Before contructing the documents, they divided document collection into two groups: input and target source documents. They started constructing these document by selecting a document from input randomly. Therefore, the constructed document is initially not plagiarism. Next, they selected random document from target source documents, and also randomly select text fragments from it. Then, they obsfuscated the text fragments before inserting them at random section of the constructed document. Lastly, they recoded about these insertions, in order to have list of source documents for each constructed document. To construct this dataset, they used document collection from Directory of Open Access [*2] with science and technology as their main topic.

In short, this dataset consists of three type of documents: input, target document, and document that records the relationship between input and source document. **Table 1** provides detail information about this dataset.

The manually judge dataset is created and has been used by Pertile et al. [6] for PD. They created this dataset by investigating documents in document collections exhaustively. They compared document one by one using several document similarity measurements, in order to select top n document pairs in document collections. Then, they asked 10 annotators to judge whether a pair of documents is worth to be aware of plagiarism based on the plagiarism level defined by IEEE [*3]. Hence, the pair of document with positive annotation is considered as the pair of input and source document [6]. They used two document collections from ACL [*4] and PubMed [*5] for dataset construction.

The manually judge dataset contains similar types of document with the auto-simulated dataset. Table 1 shows the detail information for this dataset. Because the documents in this dataset are still in PDF format, we converted them to *.txt* format by using PDFbox [*6].

5.2 Evaluation Method

To measure the performance of PD methods, we prefer to use Mean Average Precision (MAP), because it is better if the source documents have good rank in the document list. Thus, the user of PD system is able to identify source document as soon as possible. We calculate MAP by the following formula:

$$MAP = \frac{1}{|D|} \sum_{d=0}^{|D|} \frac{1}{|src_d|} \sum_{i=0}^{n} p(L_{d,i}) \quad (4)$$

$$p(L_{d,i}) = \frac{|\{s \in src_d \cap L_{d,i}\}|}{i} \quad (5)$$

with

- p: precision
- $L_{d,i}$: top i documents of document list, produced by input document d
- src_d: set of source documents for input document d
- D: set of input documents.

5.3 Result

Baseline: In the experiment, we compared documents as whole, as the baseline method. It means that we do not perform sentence classification, unlike in our proposed approach. After documents are lowercased, stopwords are removed, and all terms are stemmed, the documents are transformed into document vector using TFxIDF weighting, similar to the Equation 1. Finally, documents are compared using cosine similarity, similar to the Equation 2.

In this experiment, we tried to answer the following research questions:

(1) Does comparing CC improve PD?
(2) When we combine CC and non-CC in our model, does prioritizing CC also improve PD?
(3) How much should we prioritize CC?

Table 2 presents the experiment results for the auto-simulated dataset. From this table, we know that method that compares CC is superior compare to other methods at any cut off level. Additionally, we conducted 2 tailed paired t test for all the methods at cut off 100. We found that the differences among all the methods

---

[*2] http://doaj.org

[*3] https://www.ieee.org/documents/Level_description.pdf

[*4] http://aclanthology.info/

[*5] https://www.ncbi.nlm.nih.gov/pubmed/

[*6] https://pdfbox.apache.org/

Table 1 Statistics of the datasets

Type	ACL (manually judge)	PubMed (manually judge)	Auto-simulated
Topic	computation linguistics	biomedical and life science	science and technology
Target document	4 685	1 440	8 657
Input document	40	60	3,950
Avg. word (target)	2 557.7	2 868.8	4 417
Avg. word (input)	2 797	3 732	5 263
Source/input document	1.025	1.05	2.5
Kappa	0.675 *(substantial)*	0.524 *(moderate)*	—
Agreement rate	84%	80%	—

Table 2 MAP scores on the auto-simulated dataset

Cut off	Baseline	CC	non-CC
10	0.3077	**0.3787**	0.3131
30	0.3143	**0.3837**	0.3195
100	0.3184	**0.3858**	0.3235
200	0.3196	**0.3862**	0.3247
500	0.3205	**0.3865**	0.3255
1000	0.3207	**0.3866**	0.3258

Table 3 MAP scores on manually judge dataset

Cut off	Baseline	CC	non-CC
		PubMed	
10 or more	**0.9694**	0.9436	0.9625
		ACL	
10	0.8958	**0.9375**	0.8958
30	0.8958	**0.9386**	0.8987
100 or more	0.8979	**0.9386**	0.8987

are significant at level 1%.

Table 3 shows the MAP scores for the manually judge dataset. The baseline performance is good enough. Therefore, it may be difficult to improve it. One reason that may explain why the baseline performance is high, is because during the creation of this dataset, Pertile et al. [6] only focused on document pairs that have large amount of verbatim copy or paraphrased text. Therefore, they ignored relatively small paraphrased texts.

Except for PubMed sub-dataset, we found that the method that compares CC improves the performance of PD and the result is consistent at any cut off level. Related to the first question that we address, according to the results on Table 2 and Table 3, it suggests that comparing CC improves the PD performance. Therefore, combining methods that compare CC and non-CC as in our model may also improve PD.

In our model, we combined two methods that compare CC and non-CC, and we set a weight from them using α with range between 0 to 1. The PD method with α equal to 0 is equivalent to the method that compares non-CC, and if α is equal to 1, the PD method is equivalent to the method that compares CC. **Table 4** shows the MAP scores from the auto-simulated dataset for various values of α. We see that all the methods that combine CC and non-CC are better than the baseline for any α at any cut off level.

Prioritising CC means that we set α more than 0.5. Based on the results on Table 4, we see that the MAP scores for α above 0.5 are better than the baseline and the methods with α less than equal to 0.5. Additionally, the PD performances with α equal to 0.8 and 0.9 are better than the methods with α less than 0.8, the baseline, and the methods that only compare CC and non-CC. Hence, these results indicate that by proritising CC, it improves PD.

We found the best α in this dataset is 0.9. We also conducted 2 tailed paired t test among this method, the baseline, and the methods that compare CC and non-CC for their results at cut off 100, we found that their differences are significant at level 1%.

Table 5 presents the results for the manually judge dataset for various values of α. We see that it is difficult to improve the PD performance in PubMed sub-dataset, although the MAP differences are little, especially from α equal to 0.2 to 0.8, which is only about 0.0047 in average. However, we observe some improvements in ACL sub-dataset. When α is more than 0.5, the MAP scores of these methods are better than the baseline, the methods with α less than 0.5, and the methods that compare CC and non-CC. In average, these methods improve the baseline for about 0.062. We found that the best α for this dataset based on the results on ACL sub-dataset is 0.7.

Related to the second and the third question, we observe that if we prioritize the method that compares CC, the PD performance is improved, it is suggested from the results on the auto-simulated dataset and ACL sub-dataset from the manually judge dataset. We found the best values of α for the auto-simulated and the manually judge dataset are 0.9 and 0.7, respectively.

6. Conclusion

In this paper, proposed PD based on citation contexts and evaluate its effectiveness, because it is likely that plagiarists reuse CCs from other document to their documents.

We conducted the experiment on two kinds of datasets, namely auto-simulated and manually judge dataset. Both datasets have different characteristics, according to their creation processes. The auto-simulated dataset is constructed by simulating plagiarism activity when a document that is created by means of plagiarism contains many text fragments from more than one document. These text fragments have various lengths and are obfuscated by using some text modification methods, such as automatic summarisation, and synonym replacement.

The manually judge dataset is created by investigating two document collections from ACL and PubMed. Some textual similarity methods are used to pool some pairs of documents from these document collections. After that, annotators manually judge these pairs whether the pair should be aware of plagiarism.

Our experiment results suggested that comparing CC improved the PD performance. Additionally, in our model, prioritising it also improved the PD performance. The suggestions are based on the results on the auto-simulated dataset and ACL sub-dataset from the manually judge dataset. We also found the best weights for CC comparison in our model are 0.9 and 0.7 for the auto-simulated and the manually judge dataset, respectively.

Table 4 Experiment results on the auto-simulated dataset by tuning α

		α										
Cut off	Baseline	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
10	0.3077	0.3131	0.3195	0.3289	0.339	0.3504	0.3599	0.3688	0.3758	0.3801	0.3828	0.3787
30	0.3143	0.3195	0.3263	0.3365	0.3473	0.3587	0.3677	0.3758	0.3822	0.3857	0.3879	0.3837
100	0.3184	0.3235	0.3308	0.3409	0.3516	0.3628	0.3717	0.3795	0.3854	0.3885	0.3902	0.3858
200	0.3196	0.3247	0.332	0.3422	0.3528	0.3639	0.3727	0.3805	0.3864	0.3893	0.3902	0.3862
500	0.3205	0.3255	0.3328	0.3429	0.3535	0.3647	0.3734	0.3812	0.387	0.3899	0.3913	0.3865
1000	0.3207	0.3258	0.333	0.3431	0.3537	0.3649	0.3737	0.3814	0.3873	0.3902	0.3915	0.3866

Table 5 Experiment results on manually judge dataset by tuning α

		α										
Cut off	Baseline	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
		PubMed										
10 or more	0.9694	0.9625	0.9556	0.9667	0.9653	0.9653	0.9653	0.9644	0.9644	0.9617	0.9561	0.9436
		ACL										
10	0.8958	0.8958	0.8958	0.925	0.9321	0.9396	0.9417	0.9458	0.95	0.95	0.9625	0.9375
30	0.8958	0.8987	0.8996	0.9298	0.9331	0.9406	0.9427	0.9635	0.9635	0.9635	0.9635	0.9386
100 or more	0.8979	0.8987	0.8996	0.9298	0.9331	0.9406	0.9427	0.9635	0.9635	0.9635	0.9635	0.9386

References

[1] Alzahrani, S., Palade, V., Salim, N., and Abraham, A.: Using Structural Information and Citation Evidence to Detect Significant Plagiarism Cases in Scientific Publications, *Journal of the American Society for Information Science and Technology. Wiley Subscription Services, Inc., A Wiley Company*, Vol.63, No.2, pp.286–312 (2012).

[2] Chong, M. and Specia, L.: Lexical Generalisation for Word-level Matching in Plagiarism Detection, *International Conference Recent Advances in Natural Language Processing*, Hissar, Bulgaria, pp.704–709 (2011).

[3] Fang, F.C., Steen, R.G., and Casadevall, A.: Misconduct Accounts for the Majority of Retracted Scientific Publications, *Proceedings of the National Academy of Science. National Academy Sciences*, Vol.109, No.42, pp.17028–17033 (2012).

[4] Gipp, B. and Meuschke, N.: Citation Pattern Matching Algorithms for Citation-based Plagiarism Detection: Greedy Citation Tiling, Citation Chunking and Longest Common Citation Sequence, *Proceedings of the 11th ACM Symposium on Document Engineering (DocEng'11). ACM*, Mountain View, California, USA, pp.249–258 (2011).

[5] HaCohen-Kerner, Y., Tayeb, A., and Ben-Dror, N.: Detection of Simple Plagiarism in Computer Science Papers, *Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics*, Beijing, China, pp.421–429 (2010).

[6] Pertile, S.D.L., Moreira, V.P., and Rosso, P: Comparing and Combining Content-and Citation-based Approaches for Plagiarism Detection, *Journal of the Association for Information Science and Technology. Association for Information Science and Technology*, Vol.67, No.10, pp.2511–2526 (2016).

[7] Stamatatos, E.: Plagiarism Detection Based on Structural Information, *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, Glasgow, Scotland, UK, pp.1221–1230 (2011).