Identification of Original Document by Using Textual Similarities

Prasha Shrestha and Thamar Solorio

University of Houston
Department of Computer Science
4800 Calhoun Rd. Houston, TX, 77004
pshrestha3@uh.edu, solorio@cs.uh.edu

Abstract. When there are two documents that share similar content, either accidentally or intentionally, the knowledge about which one of the two is the original source of the content is unknown in most cases. This knowledge can be crucial in order to charge or acquit someone of plagiarism, to establish the provenance of a document or in the case of sensitive information, to make sure that you can rely on the source of the information. Our system identifies the original document by using the idea that the pieces of text written by the same author have higher resemblance to each other than to those written by different authors. Given two pairs of documents with shared content, our system compares the shared part with the remaining text in both of the documents by treating them as bag of words. For cases when there is no reference text by one of the authors to compare against, our system makes predictions based on similarity of the shared content to just one of the documents.

Keywords: original document, bag-of-words, document provenance, plagiarism.

1 Introduction

When two documents have shared content, the first question that arises is whether it was the author of one document or the other that produced the original content. The answer to this question has important implications in terms of establishing provenance and authorship of the information in the shared content. When the presence of this shared content has been found by plagiarism detection systems, identifying the original document can help somebody to be exonerated of plagiarism. This will especially be useful in the academic scenario when two students are found to have similar content in their assignment. Usually both are held under blame. But in same cases, the student whose work was plagiarized might not even be aware of it. Another obvious use is when a person makes a claim of plagiarism of their work when is no information about which version of the document came first. In this case a system that finds the original document can help to settle the dispute.

Identifying the original document can also be a first step towards establishing the provenance of a document. Provenance is important because it has critical applications in security. There has been a lot of work in recording provenance for different types of data in e-science [1]. Several methods have also been proposed for developing automatic provenance recording systems in the cloud. There have been standards set on the

A. Gelbukh (Ed.): CICLing 2015, Part II, LNCS 9042, pp. 643–654, 2015.
DOI: 10.1007/978-3-319-18117-2_48

properties that these provenance recording systems must satisfy [2] and all the details that provenance information for cloud processes should contain [3]. There have even been work done for recording provenance of experimental workflows and even to establish provenance for art [4]. But document provenance has hardly had any research effort devoted to it. It would be easy to record document provenance but this is rarely done and in the instances of plagiarism, people are likely to try to hide this information rather than to document it. If the provenance has not been recorded, establishing provenance from a pool of documents is the only option left and it is a very hard problem. The problem is more tractable if the modifications on the document have been made by different authors. For cases when a document written by an author gets subsequently modified by other authors, our method can be useful to extract provenance. If the whole document or parts of the document has been modified by another author, our system can compare the modified section with other works from both authors to decide which version of the document is the original one. Our method can be applied for all the documents in question pairwise until the entire lineage is traced.

The problem we are dealing with and authorship attribution are also closely related. But one major difference is that in authorship attribution, the document or piece of text that we are trying to attribute to an author is untouched by any other author. It has been written solely by that author. But in our case, we have a piece of text that has been written by one author and in most cases, modified by another author to use in his own work. We are trying to attribute the text used by both authors to one of them. This adds a layer of complexity to our problem. Nonetheless, the ideas used in this work can also be applied in the scenario when the authorship of a piece of text is disputed between two authors. Given that text with disputed authorship and other documents written by these authors, one can use our system without modification in order to attribute the work to one of the authors.

We have used a simple yet effective method in order to solve the problem of finding the original document out of two documents. We first separate the content shared by them from both documents. We then divide the rest of the text in the documents into segments and create a bag of word representation of these segments and also of the segment with the shared content. We then extract the top most frequent words from each of these segments. The next step is to find the overlap between the top words from the shared content and the top words of all of the segments of both documents. The document whose segments have the higher average overlap with the shared content will be classified as the original document. Similarly, from the perspective of document provenance, the shared segment will have originated from this document and thus will be the predecessor of the other document.

This paper also deals with the case when between the two documents, in one of them all of the text is similar to parts of the other document. In this case, there is no additional reference text to compare against for one of the authors. Here, the prediction needs to be done only based upon the similarity or dissimilarity of the shared text to the text of only one of the authors. This scenario can happen in real life as well where all of the text written by an author has been fully lifted from one or more sources without adding any original content. This is a much harder problem and will generally have

lower accuracy than when text from both authors is available. Our system can be used, although cautiously for this scenario as well.

2 Related Work

At the time of this writing, we were able to find only one previous work that deals with a similar problem as ours. Grozea and Popescu (2010), in their work, have proposed a solution for finding the direction of plagiarism [5]. The idea behind their approach is that the n-grams present in the plagiarized passage will repeat more throughout the original document than in the plagiarized document. This makes it very likely for these n-grams to occur much earlier in the source document than in the plagiarized document. They have used character 8-grams and only considered the first one of the n-gram matches between the plagiarized and the non-plagiarized sections. Then they plotted these matches and then found the asymmetry in the plots. Their work is a continuation of the system they submitted to the PAN 2009 External Plagiarism Detection Competition and they used the same data for this experiment as well. They were able to obtain an overall accuracy of 75.42% on this dataset.

The above work is the only one we could find that deals with the exact same problem as the one we are trying to solve. But the work on plagiarism detection: both intrinsic and extrinsic, problems dealing with authorship and the problem of anomaly detection are relevant to our task.

Our problem is very similar to the intrinsic plagiarism detection problem. In intrinsic plagiarism detection, the task is to figure out if a document has been plagiarized or not by using the text in just that document as the reference. So, in this problem as well as our problem requires the checking of how similar parts of a document are as compared to other parts of the same document. For the intrinsic plagiarism detection problem, Stamatatos (2009) proposed that the inconsistencies within the document, mainly stylistic, can point towards the plagiarized passage [6]. They use bag of character trigrams of automatically segmented passages in the document and use a sliding text window to compare the current text in the window to the whole document. They only deal with documents that have less than half plagiarized content because otherwise the style function will represent the style of the plagiarist and not of the true author. As in this method, most of the approaches to plagiarism detection, both intrinsic and extrinsic make use of n-grams. Barrón-Cedeño and Rosso (2009) have tried to investigate the best value for n when performing an n-gram comparison [7]. They used word n-grams in their method and found out that low values of n generally work better for n-gram based methods.

Intrinsic plagiarism detection can also be modeled as a one class classification problem, with the non-plagiarized text falling under the target class and all other plagiarized texts being the outliers [8]. Stein et al. (2011) used this approach along with a large number of lexical, syntactic and semantic features. In order to perform outlier identification, they assumed that the feature values of the outliers have uniform distribution and then use using maximum likelihood estimation. They also employ as a post-processing step, a technique called unmasking by Koppel and Schler (2004) [9]. This method works by removing the most discriminating features gradually such that, after a few iterations, the remaining features cannot properly discriminate between texts written by the same author but can still discriminate between texts from different authors.

Our problem as well as the problem of intrinsic plagiarism is similar to the problem of anomaly detection as well, since a plagiarized passage behaves like an anomaly. Guthrie et al. (2007) too have used a large variety of stylistic, sentiment and readability features in order to find an anomalous segment in a text [10]. The rank features used by them are particularly unique and they use rank correlation coefficient rather than similarity measures for the rank features. They rank a list of articles, prepositions, conjunctions, pronouns, POS bigrams and POS trigrams and then calculate the Spearman rank correlation coefficient. They found out that their accuracy improves as the segment size increases.

The problem of authorship attribution is related to our problem because both involve examining a text with undoubted authorship to check if another piece of text having unknown or dubious authorship is also written by the same author. In our problem, for the case when there is no reference text from one of the authors the problem becomes even more similar to authorship attribution, albeit on text written or changed by both authors. Stamatatos (2009) noted that although many kinds of lexical, character, syntactic and semantic features are used in authorship attribution, lexical features are the most prominently employed features in authorship attribution systems [11]. They also noticed that most systems considered the text as a bag of words with the stopwords being the most discriminating and most widely used features.

In order to determine if two documents have been written by the same author, rather than treating it as a one-class classification problem, Koppel and Winter (2014) have converted the problem to a many-candidates problem [12]. While in the one-class classification problem, we only have text written by the target author and we need to find out if a given document is written by this author vs any existing author. It is not possible to obtain text for every author in the world. So, they have created impostor documents and then tried to find out if the current document is more similar to a document written by the target author over any other impostor documents. So, the complexity of the problem is reduced from being a target vs outlier problem to a classification problem with a known set of classes.

3 Methodology

The input to our system is a pair of documents with known plagiarized content between them. In most real plagiarism cases, a single document might have passages taken from multiple source documents, which is also the case in the dataset we use. For this reason, we perform our classification on a per passage basis. Our system tries to attribute each one of the plagiarized passages to one of the documents separately. For example, if there are two documents containing similar passages, one of them will be the original document for this particular passage. But there might be several such passages inside a single document, originally appearing in several other documents. Thus, for each such passage, the original document might be different. For this reason, we perform our classification on a per passage basis. This shared or plagiarized content needs to be compared with only the text that has purely been written by the authors in question. For this reason, we also remove all other passages known to be shared with some other documents. After this, we are only left with the texts written by the two authors in

question. In most of the cases, we have enough text from both authors in order to make a comparison. But for a few cases, we only have text from one of the authors to compare against. The method we used for the case when we have some amount of text from both authors is described in Section 3.1. For the few cases where we only have the text from one of the authors, we describe the method we used is described in Section 3.2.

3.1 Overlap between Words

Two pieces of text written by the same author are more likely to have similar word usage patterns. We use this idea in order to compare the shared content with the rest of each authors' text. We first find the most frequent word tokens in the shared, plagiarized part and in the non-plagiarized parts of both of the documents, and then use the overlap between these tokens in order to decide which document the plagiarized passage was originally taken from. This is a two class classification problem, but with a very limited amount of data that is representative of the two classes.

In any document, and especially in the long ones, the writing style and word usages of an author can change subtly throughout the document. The particular passage that has been copied by one author from another author's document may be similar to some parts of the text, but not so much to the others. For this reason, we first divide the unplagiarized passage of both documents into segments. In most of the cases in our dataset, the text purely written by a single author i.e. the non-plagiarized part of the text is longer than the plagiarized part. We chunk the non-plagiarized text into equal length segments in such a way that there are enough segments to compare the plagiarized text against, while also keeping the segments similar in length to the plagiarized text. But for cases when the non-plagiarized text is very short in comparison to the plagiarized text, the whole text comprises a single segment. We then tokenize the segments into words and retain everything, including stopwords.

After obtaining the segments, we proceed on to extract f frequent words from each of the segments, including the plagiarized ones. We set the value of f according to the segment size so that we will have enough words to compare for documents or segments of any size. We set f to one fourth of the segment size except for the case when one of the segments is smaller than this value. In this case, f will be equal to the size of the smaller segment. Thus, for large segments, we end up taking only the most frequent words. But for small segments, there will not be many words to compare against if we just take the most frequent ones. For cases where the segments are very small, either due to the plagiarized passage being small or the unplagiarized content in one of the documents being short, we use all of the tokens.

With these most frequent words in hand, the next step is to check how similar the plagiarized passage is to the two sets of non-plagiarized text. This similarity score is calculated as shown in Equation 1 below.

$$avg\_overlap(p, u) = \frac{\sum_{i=1}^{len(u)} |fw(p) \cap fw(u_i)|}{len(u)} \tag{1}$$

The value for $avg\_overlap$ is calculated between a plagiarized segment p and the set of non-plagiarized segments u of a document. $fw(x)$ represents the most frequent

words in a segment x. This score will provide us with the extent of overlap between the plagiarized segment p and the set of non-plagiarized segments u in a document. The score calculates the overlap of the plagiarized passage with each of the non plagiarized segments of a document. It then computes the average overlap. This $avg\_overlap$ score is calculated for a plagiarized passage and the set of segments of both candidate documents. For a passage, the original document is taken as the document that produces a higher score with this passage.

As is the case in most plagiarized text, most of the plagiarized segments have been obfuscated. Due to this, we actually have two different versions of the same passage, in the two documents. We choose to make predictions for them individually and then combine the results of the predictions later. We calculate $avg\_overlap$ for the first version of the passage with both of the documents and obtain a prediction for that passage about the document it actually belongs to. We repeat the same for the other version of the passage. If both versions of the plagiarized segment predict the same document as the original, the final prediction is also the same document. But if they disagree, we go back to the $avg\_overlap$ values to make the final prediction. We have two $avg\_overlap$ scores for each version of the plagiarized passage, as calculated before. We take the higher of these scores for both passages and then again compare these two scores. Our system then uses the prediction for the version of the plagiarized segment that has higher $avg\_overlap$ score with its predicted original document.

3.2 Meta Learning for Predictions Using Single Documents

In cases where a document has been fully plagiarized, there is no reference text for one or both of the authors and the method described in the previous section becomes inapplicable. This scenario occurs in three cases. First, an entire document might be the product of content plagiarized from parts of another document. Second, the entire original document might be plagiarized into another document having some content of its own. The third and very rare case is when a document is fully plagiarized to form a new document and no extra content is added to it. For this last case, we do not have any reference to compare against in either of the two documents. This problem is nearly impossible to solve, and will require information outside of the two documents and is thus outside of the scope of our work.

For the other two cases, we have some reference text for one of the two authors. We make use of this author's text to perform a one class classification to decide whether the plagiarized text has also been written by the same author. The intuition behind this method is simple. A piece of text originally written by an author will resemble other content produced by the same author.

For the document having content additional to the plagiarized text, we first divide this non-plagiarized content into segments and then obtain the most frequent word unigrams in a similar way as the previous method. We also obtain these most frequent tokens for the plagiarized content in both documents in the same way. We then calculate the overlap score for both versions of the plagiarized passage with the set of segments obtained from the document having reference text. The scoring here also used the same formula as shown in Equation 1.

We will already have in hand the scores that we obtain for the case described in Section 3.1. We use these scores as our training data to train a logistic regression model. We then take the scores that we have just obtained for the one reference document and two versions of the plagiarized passage. We feed these to get the predictions from the model. Although there are only two classes that can be predicted for each of the passages, the documents that these two classes represent vary in every instance. This makes it a hard problem to get as good results as in the case discussed in the previous section.

4 Dataset

It is hard to find data for real cases of plagiarism or unintentional copying. For our experiments, we used the dataset from the text alignment subtask of PAN plagiarism detection task. This dataset consists of a set of documents with some content taken from one document and copied into another, either verbatim or with some changes. In this dataset, a document containing the plagiarized content is called a suspicious document and a document containing purely original content is called a source document. We removed the information about whether a document was source or suspicious and treated both documents equally in order to mimic the scenario of the problem we are trying to solve.

The plagiarism detection task at PAN has been taking place every year since 2009 and they have released a new or modified corpus in most years. We performed detailed experiments on the PAN 2009 corpus, in order to compare our results with Grozea and Popescu (2010), the only other known system dealing with the same problem [5]. But we also evaluated our system on all the other existing versions of the PAN dataset. In PAN 2009 corpus, the documents have been artificially plagiarized by using different methods of obfuscation as a human plagiarist would [13]. They have used replacement by synonyms, shuffling, text insertion and deletion. In the 2009 dataset, some documents have the plagiarized passages copied verbatim, while others have high or low levels of obfuscated plagiarized passages. In another form of obfuscation called translation obfuscation, they used a machine translator to translate English passages into a chain of other languages and then translated it back to English. In newer versions of the dataset, they have also used summary obfuscation. In summary obfuscation, the passages from one document are summarized before being inserted into the other document.

Apart from the PAN dataset, we also tested our system on a prominent case of plagiarism that had appeared in the media. Many works of a famous journalist works were alleged to have been plagiarized from other sources.[1] Those allegations were found to be true and the magazines where they were published issued statements expressing that those articles did not meet their standards and some even fired him. We only collected those news articles where he had plagiarized more than two sentences from another news article. We found three such cases among the plagiarism allegations against him.

[1] https://ourbadmedia.wordpress.com/2014/08/19/did-cnn-the-washington-post-and-time-actually-check-fareed-zakarias-work-for-plagiarism

4.1 Results and Analysis

The results we obtained for the PAN 2009 dataset are shown in Table 1. We obtained an accuracy of 85.56% on the overall test dataset. This is a lot higher than the 75.42% obtained by the only known previous work [5], who also used the same dataset. In the case of real plagiarism, the more obfuscated the text is, the more it deviates from the writing style of the original author and will reflect the writing style of the plagiarist. As expected, the results were better in the case of no obfuscation and the problem was harder for higher levels of obfuscation. When there is no obfuscation, our accuracy is 88.12% but it drops down to 79.54% for high obfuscation.

Table 1. Accuracy on the PAN 2009 dataset

Data Type	Number of Passages	Accuracy (%)
No Obfuscation	26855	88.12
Low Obfuscation	26628	86.04
High Obfuscation	13658	79.54
Translation Obfuscation	6381	85.72
Overall	73522	85.56
Grozea and Popescu (2010)	73522	75.42

The results for all of the PAN datasets are shown in Table 2. We obtained accuracy comparable to the PAN 2009 dataset for the PAN 2011 and 2012 datasets as well. But the accuracy on PAN 2013 data is notably lower than on all other datasets. To find the reason for this, we looked at the lengths of the documents in these datasets. We found that the length of documents in PAN 2013 dataset is significantly shorter than that of the other PAN datasets as shown in Table 3. When documents are short, it is harder to capture the writing style of an author given that small amount of information. Our segment size is also small and there are less segments to compare the plagiarized passage against. The top most frequent words obtained might not represent how the author truly writes for this case. This made our accuracy drop significantly. There might also be a bigger problem because in the older PAN datasets, the plagiarized passage and the document where it was inserted into to create simulated plagiarism were randomly chosen. As such, the plagiarized and non-plagiarized parts of the same document might have different topics. It is possible that in the experiments with the older datasets, our system might have been doing topic classification along with the detection of the original document. But even in this PAN 2013 dataset where the corpus creators have tried to stay within the same topic for both plagiarized and non-plagiarized text, our accuracy is fairly reasonable.

We performed an analysis of the effect of length on accuracy by using the PAN 2009 dataset. Since we are comparing plagiarized passages against non-plagiarized ones, both their lengths can affect our system. For example, if the non-plagiarized portion contains just five words while the plagiarized portion contains 5000 words, although the whole document will be long, our prediction might be hampered by the brevity of

Table 2. Accuracy on other PAN datsets

Dataset	Number of Passages	Accuracy (%)
PAN 2009	73522	85.56
PAN 2011	49621	82.14
PAN 2012	12495	85.98
PAN 2013	4007	74.87

Table 3. Average length of documents across PAN datasets

Dataset	Avg. # of words per document
PAN 2009	47653
PAN 2011	50582
PAN 2012	50315
PAN 2013	2462

the non-plagiarized passage. For this analysis, we considered the length of a document as the length of its shorter portion: either the plagiarized or the non-plagiarized part. We sorted the documents in ascending order by length and divided them into 10 buckets containing equal number of documents. The first bucket contains the shortest 10% of the documents, the second bucket contains the next shortest 10% and so on while the tenth bucket contains the longest 10% of the documents. We then looked at the accuracy for these buckets. Figure 1 shows the bucket index with the average number of words in the documents of that bucket and the accuracy obtained for that bucket of documents. The accuracy for the bucket with longer documents is considerably higher, 90 than that for the bucket with shorter documents, although the curve is not ascending uniformly. But the accuracy for the tenth bucket containing the longest documents is the highest at 90.99% while the accuracy for the first document is comparatively low at 82.65%. This also further shows that the length of the documents in the dataset plays a great role in the prediction accuracy.

On our data collected from real plagiarism case as describe in 4, we were able to predict the original document correctly for two cases out of the three. We believe that the size of the documents might have again played a role in the result. The one case where we designate the wrong document as the original one is where we have the least amount of text. Although this dataset is too small to draw any conclusions, our method does seem to work well for real cases, given that there is enough text to compare.

For a small minority of documents not having reference text by one of the authors, the results obtained by using the method described in Section 3.2 is shown in Table 4. The accuracy for this method is not comparable to the case when we have reference text for both documents. This problem of identifying whether a piece of text is written by a particular author or not, given very small samples of text written by that author is an inherently hard problem and is thus inclined to suffer from lower accuracy. But as seen in the same table, this situation occurs in less than 0.5% of the data for PAN 2009-12 datasets. In real plagiarism cases as well, the plagiarist is likely to plagiarize some parts

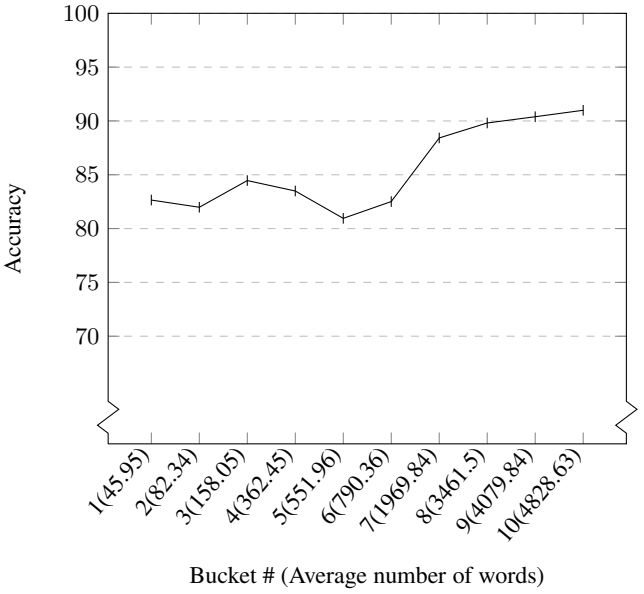

Fig. 1. Accuracy on buckets of documents sorted by length

of the document and also add his own content in order to not get caught. This was also the case for all of the real plagiarism data that we collected. The author had only copied certain parts of another article and inserted them into his own article. So, the situation where there is no other text in a document other than the plagiarized one is very less likely to occur.

Table 4. Accuracy for Meta Learning Method

Data Type	% of Total Documents	Accuracy (%)
PAN 2009	0.0054	52.65
PAN 2011	0.46	51.14
PAN 2012	0.46	43.10
PAN 2013	9.86	47.74

As is the case with systems dealing with authorship or author profiling that use word n-grams, stopwords were the most discriminating features. They were the most frequently occurring tokens in the overlap between the two segments. Apart from stopwords, the words that belong to the topic of the document also occurred in the overlap. For documents that were stories, there were also a lot of named entities present as the common words between the segments.

Apart from the documents for which we have results shown in the above tables, there are four document pairs in the dataset which belong to the third scenario as described in

Section 3.2. For these cases, neither of the methods gives a classification. This is a very hard problem and it occurred only in the 2009 dataset and that too in only four cases. A solution to this problem will surely require more information that what is available in the dataset as we will need to collect more data from the authors which out of the scope of our work.

5 Conclusions and Future Work

We have presented a method to identify the original document out of two when they have a piece of shared content between them. Our method is a good solution to the problem of finding the original document and it performs well across different datasets. As expected, the results were better for lower levels of obfuscation. Even for higher levels of obfuscation, our accuracy is close to 80%. Also, more correct predictions are made when there is sufficient text to capture the writing style of an author. As the text becomes shorter, the problem gets harder. But even for short documents, we obtain reasonable accuracy. Only in the case when one of the documents has been fully plagiarized, our accuracy is low. But since we are making predictions based on only one of the documents, lower accuracy is to be expected. There are rare cases where we cannot apply any of our methods due to neither of the two documents having any reference text. This is a problem that can have practical applications. This also relates to the problem of document provenance when there are only minor changes made to a document, as can happen when somebody is proofreading a document. This is an interesting problem and we would like to explore it in the future. We have also not dealt with the problem of self-plagiarism where an author reuses his/her own text. For this case, we cannot make use of the writing style of the author to determine which document is the original. This will require a completely different method. We also leave this for future work. But for now, we can surely say that when we have two different authors and text written by them to compare against, our method gives good performance.

Acknowledgments. We would like to thank the anonymous reviewers for providing us with the helpful feedback. This research is partially funded by The Office of Naval Research under grant N00014-12-1-0217.

References

1. Simmhan, Y.L., Plale, B., Gannon, D.: A survey of data provenance in e-Science. SIGMOD Rec. 34, 31–36 (2005)
2. Muniswamy-Reddy, K.K., Macko, P., Seltzer, M.: Provenance for the cloud. In: Proceedings of the 8th USENIX Conference on File and Storage Technologies, FAST 2010, pp. 15–14. USENIX Association, Berkeley (2010)
3. Muniswamy-Reddy, K.K., Holland, D.A., Braun, U., Seltzer, M.: Provenance-aware storage systems. In: Proceedings of the Annual Conference on USENIX 2006 Annual Technical Conference, ATEC 2006, p. 4. USENIX Association, Berkeley (2006)

4. Green, R.L., Watling, R.J.: Trace element fingerprinting of australian ocher using laser ablation inductively coupled plasma-mass spectrometry (LA-ICP-MS) for the provenance establishment and authentication of indigenous art*. Journal of Forensic Sciences 52, 851–859 (2007)
5. Grozea, C., Popescu, M.: Who's the thief? Automatic detection of the direction of plagiarism. In: Gelbukh, A. (ed.) CICLing 2010. LNCS, vol. 6008, pp. 700–710. Springer, Heidelberg (2010)
6. Stamatatos, E.: Intrinsic plagiarism detection using character n-gram profiles. In: 3rd PAN Workshop Uncovering Plagiarism, Authorship and Social Software Misuse, vol. 2, p. 38 (2009)
7. Barrón-Cedeño, A., Rosso, P.: On automatic plagiarism detection based on n-grams comparison. In: Boughanem, M., Berrut, C., Mothe, J., Soule-Dupuy, C. (eds.) ECIR 2009. LNCS, vol. 5478, pp. 696–700. Springer, Heidelberg (2009)
8. Stein, B., Lipka, N., Prettenhofer, P.: Intrinsic plagiarism analysis. Language Resources and Evaluation 45, 63–82 (2011)
9. Koppel, M., Schler, J.: Authorship verification as a one-class classification problem. In: Proceedings of the Twenty-First International Conference on Machine Learning, p. 62. ACM (2004)
10. Guthrie, D., Guthrie, L., Allison, B., Wilks, Y.: Unsupervised anomaly detection. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence, IJCAI 2007, pp. 1624–1628. Morgan Kaufmann Publishers Inc., San Francisco (2007)
11. Stamatatos, E.: A survey of modern authorship attribution methods. Journal of the American Society for Information Science and Technology 60, 538–556 (2009)
12. Koppel, M., Winter, Y.: Determining if two documents are written by the same author. Journal of the Association for Information Science and Technology 65, 178–187 (2014)
13. Potthast, M., Stein, B., Barrón-Cedeño, A., Rosso, P.: An evaluation framework for plagiarism detection. In: Coling 2010: Posters, pp. 997–1005. Coling 2010 Organizing Committee, Beijing (2010)