Strategies for Retrieving Plagiarized Documents

Benno Stein
Faculty of Media, Media Systems
Bauhaus University Weimar
99421 Weimar, Germany
benno.stein@
medien.uni-weimar.de

Sven Meyer zu Eissen
Faculty of Media, Media Systems
Bauhaus University Weimar
99421 Weimar, Germany
sven.meyer-zu-eissen@
medien.uni-weimar.de

Martin Potthast
Faculty of Media, Media Systems
Bauhaus University Weimar
99421 Weimar, Germany
martin.potthast@
medien.uni-weimar.de

ABSTRACT

For the identification of plagiarized passages in large document collections we present retrieval strategies which rely on stochastic sampling and chunk indexes. Using the entire Wikipedia corpus we compile n-gram indexes and compare them to a new kind of fingerprint index in a plagiarism analysis use case. Our index provides an analysis speed-up by factor 1.5 and is an order of magnitude smaller, while being equivalent in terms of precision and recall.

Categories and Subject Descriptors: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*Retrieval models, search process*; H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing—*Abstracting methods, Indexing methods*

General Terms: Theory, Performance

Keywords: plagiarism analysis, hash-based indexing, fuzzy-fingerprinting

1. INTRODUCTION

In the generic plagiarism analysis situation we ask the following question:

"Did the author of a document d_q commit a plagiarism offense?"

Approaches for computer-based plagiarism analysis break down this question into manageable parts:

"Given a collection D of documents,
does d_q contain a passage p_q
for which one can find a document $d_x \in D$
that contains a passage p_x
such that under some retrieval model $\mathcal{R}$
the similarity $\varphi_{\mathcal{R}}$ between p_q and p_x is close to 1?"

The respective algorithms require the collection D stored in a preprocessed form, typically as a tailored chunk index $\mathbf{D}$. A chunk index is an inverted file whose vocabulary (the left-hand side of the mapping) contains larger portions of the document, such as n-grams, shingles, or other multi-term features. Based on such an index, we propose to organize the retrieval of plagiarized documents as a three-stage process (cf. Figure 1):

1. *Heuristic Retrieval.* Samples of the suspicious document d_q are extracted and looked-up in a chunk index. The set of matching samples indicates postlists, which in turn refer to candidate documents with possibly plagiarized passages.

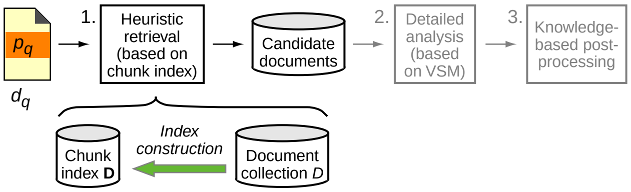

p_q
d_q
1.
Heuristic retrieval (based on chunk index)
Candidate documents
2.
Detailed analysis (based on VSM)
3.
Knowledge-based post-processing
Chunk index **D**
Index construction
Document collection *D*

Figure 1: A three-stage process for plagiarism analysis.

2. *Detailed Analysis.* The possibly plagiarized passages in each candidate document are compared to the matching passages in d_q. This analysis happens under a "stronger" retrieval model $\mathcal{R}$, which is the vector space model along with cosine similarity here.

3. *Knowledge-based Post-processing.* It is analyzed whether identical passages have been properly quoted, and hence establish no plagiarism offense.

In this paper we concentrate on the first step: We present a new kind of chunk index, based on so-called fuzzy-fingerprints, and compare it to a standard n-gram index with respect to storage requirements and retrieval efficiency. In our analysis we have indexed the entire Wikipedia corpus[1], for which we achieve a storage reduction by a factor >10 and a retrieval speed-up by factor 1.5.

Related Work. Research on automated plagiarism analysis focuses mainly on Step 2 in Figure 1, using technologies from the field of near-duplicate detection [1, 5]. New similarity hashing techniques from [4, 6] have not yet been considered for plagiarism analysis, despite their promising retrieval properties. Contributions related to tailored indexes for near-duplicate detection come from [1, 3]. The former introduce an index based on the so-called SPEX algorithm which allows for the identification of duplicate chunks within a closed document collection but cannot be employed to open retrieval situations. The latter develop a variant of an inverted file index where duplicate text-passages are indexed only once, achieving size reductions of 30%.

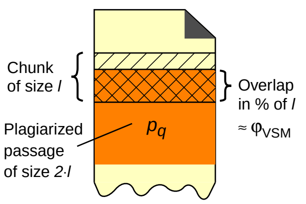
2. RETRIEVAL MODEL

In the plagiarism analysis use case, a comparison of a document d_q against a collection D is a problem of inherently quadratic runtime at the passage level: n-gram by n-gram of d_q must be looked-up in an inverted file index $\mathbf{D}$ of D. A small value for n leads to a large set of candidate documents implying a high post-processing effort, a large value for n minimizes the chance of detecting a plagiarized passage that has been slightly modified. Experience has shown that useful values for n are between 3 and 5 [5].

[1] A Wikipedia snapshot from November 4th, 2006 was indexed.

SIGIR'07, July 23–27, 2007, Amsterdam, The Netherlands.
ACM 978-1-59593-597-7/07/0007.

Document d_q:



Portion γ of reliable chunks:

$$\gamma = 2 \cdot \frac{1-\mathit{overlap}}{l_q} + \frac{l_q - 1}{l_q},$$

with l_q = size of plagiarized passage measured in multiple of l

overlap	l_q	γ
0.8	1	0.4
0.8	3	0.8

Figure 2: Only a portion γ of all chunks identifies a plagiarized passage p_q of size l_q. The figure shows the computation of γ.

Lower Bounds for Sample Extraction. Given an upper bound for the plagiarized portion p of d_q, a lower bound for the number of n-grams to be extracted can be stated such that plagiarized passages are covered with high probability.

Suppose that a chunk is drawn uniformly at random from d_q, and let X be the random variable that has value 1 if the chunk belongs to a plagiarized passage, and 0 otherwise. Then X has a Bernoulli distribution with parameter p. If the experiment is repeated r times with $X_1, \ldots, X_r$ denoting the respective outcomes, the variable $S_r = \sum_{i=1}^{r} X_i$ has a binomial distribution, and the probability that S_r takes value k is $P(S = k) = \binom{r}{k} p^k (1-p)^{r-k}$. We can determine a value for r such that the equation $P(S_r > k) \geq \alpha$ holds for a given k at a desired confidence threshold α:

$$\begin{aligned} P(S_r > k) \geq \alpha \quad &\Leftrightarrow \quad 1 - P(S_r \leq k) \geq \alpha \\ &\Leftrightarrow \quad P(S_r \leq k) \leq 1 - \alpha \\ &\Leftrightarrow \quad \sum_{i=1}^{k} \binom{r}{k} p^k (1-p)^{r-k} \leq 1 - \alpha \qquad (1) \\ &\Leftarrow \quad e^{-\frac{1}{2p}\frac{(rp-k)^2}{r}} \leq 1 - \alpha \qquad (2) \end{aligned}$$

Remarks. (1) k defines the number of plagiarized passages that are discovered in d_q. (2) In general holds $r \geq \frac{k}{p}$ since the expected value $E(S_r) = rp$. (3) The last implication results from the Chernoff Inequality applied to the binomial distribution; it is used to derive a closed form for a lower bound of r.

Chunking with Fuzzy-Fingerprints. The chunk size can be significantly increased, if a fuzzy match instead of an exact match is applied in the heuristic retrieval step. In this connection we pick up the concept of fuzzy-fingerprinting [6]: the vector space representation of a text passage is "condensed" toward a small set of prefix classes, where a prefix class comprises all terms with the same prefix. The observed prefix class frequencies are normalized and compared with their expected values from the British National Corpus. The resulting exact deviations are fuzzified with different fuzzification schemes and encoded as numbers. In this way, chunks of size l, with $l \in [40, 200]$, are encoded as two or three 8-byte numbers.

In our use case we assume for a suspicious paragraph p_q and its plagiarized counterpart p_x a similarity of $\varphi(p_q, p_x) \geq 0.8$ under the vector space model. Thus, to identify this similarity with two chunks $c_q \subseteq p_q$ and $c_x \subseteq p_x$, the overlap of c_q and c_x must also be at least 0.8, which in turn means, that only a portion γ of all chunks that intersect with p_q can be considered. Figure 2 illustrates the computation of γ; when working with fuzzy-fingerprints the value p in Inequality (1) and (2) must be multiplied by γ.

3. ANALYSIS WITH WIKIPEDIA

For analysis purposes we have compiled three chunk indexes containing the 1.5 million Wikipedia articles. Two of the indexes, $\mathbf{D}_{\text{3-gram}}$ and $\mathbf{D}_{\text{4-gram}}$ store all 3- and 4-grams respectively; the third index, $\mathbf{D}_{FF}$, stores all fuzzy-fingerprints that were computed from chunks of size $l = 40$ having a 50% offset.

Chunk index type	Avg. postlist length	Size	Size ratio
$\mathbf{D}_{FF}, l = 40, \gamma = 0.8$	1.17 entries	**0.35 GB**	**0.07**
$\mathbf{D}_{\text{3-gram}}$	2.43 entries	3.42 GB	0.65
$\mathbf{D}_{\text{4-gram}}$	1.44 entries	5.25 GB	1

Table 1: Characteristics of the Wikipedia chunk indexes.

Index Data Structure. Since the Wikipedia collection is known in advance we designed a space efficient inverted file data structure based on minimal perfect hashing. The data structure implements a hash table $\mathcal{T}$ with $|\mathcal{T}|$ storage positions. A particular hash function $h : U \rightarrow |\mathcal{T}|$ is used to map the universe of chunks, U, perfectly onto a minimum number of storage positions, say, $|\mathcal{T}| = |U|$. h can be constructed in $O(|U|)$ time and space and requires additional storage of size $4.6 \cdot |U|$ bytes [2]. The overall sizes of the compiled indexes are shown in Table 1. Note that $\mathbf{D}_{FF}$ needs only a small fraction of the storage of the n-gram indexes. Also note that postlist compression based on a gap statistic evaluation cannot significantly change these relations.

Runtime Analysis. To compare the runtime performance we analyzed custom-built documents that contained plagiarized passages from Wikipedia articles. The documents' lengths ranged from 50 to 300 pages, the plagiarized portion p ranged from 0.1% to 6.5%. As determined by Equation 1 for $\alpha = 0.9$ and $k = 1$, between $r = 150 \ldots 3500$ chunks (size $l = 40$) and 4-grams were randomly extracted from the test documents and looked-up in the indexes $\mathbf{D}_{FF}$ and $\mathbf{D}_{\text{4-gram}}$ respectively. For each document the average number of result documents per query was recorded, whereas common 4-grams (occurring in more than 10 documents) were discarded. The left plot in figure 3 shows the averaged number of returned documents depending on the plagiarized portion p: the retrieval based on $\mathbf{D}_{FF}$ outperforms the 4-gram index by factor 1.5. Note that the result set size determines the true plagiarism analysis effort, since each result document must be loaded and compared in detail. The right plot shows precision and recall at certain similarity thresholds for fuzzy-fingerprinting. Note that at similarities above 0.8 we achieve a reasonable recall performance.

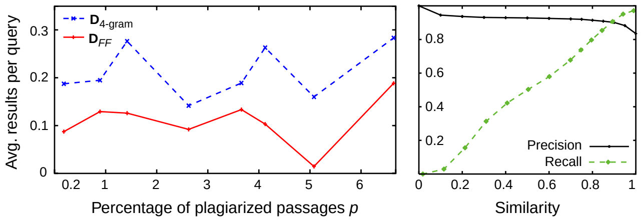


Figure 3: Runtime performance of the chunk indexes measured in the number of retrieval results to be post-processed.

4. REFERENCES

[1] Y. Bernstein and J. Zobel. A scalable system for identifying co-derivative documents. *SPIRE '04*, vol. 3246 of *LNCS*, pages 55–67.

[2] F. Botelho, Y. Kohayakawa, and N. Ziviani. A Practical Minimal Perfect Hashing Method. *WEA '05*, vol. 3505 of *LNCS*, pages 488–500.

[3] A. Broder, N. Eiron, M. Fontoura, M. Herscovici, R. Lempel, J. McPherson, R. Qi, and E. Shekita. Indexing Shared Content in Information Retrieval Systems. *EDBT '06*, pages 313–330.

[4] A. Gionis, P. Indyk, and R. Motwani. Similarity Search in High Dimensions via Hashing. *The VLDB Journal*, pages 518–529, 1999.

[5] T. Hoad and J. Zobel. Methods for Identifying Versioned and Plagiarised Documents. *J. of the ASIST*, 54(3):203–215, 2003.

[6] B. Stein. Fuzzy-Fingerprints for Text-Based Information Retrieval. *I-KNOW '05*, JUCS, pages 572–579. Know-Center.