Knowledge-Based Systems 000 (2017) 1–18

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Minmax Circular Sector Arc for External Plagiarism's Heuristic Retrieval stage

Fellipe Duarte[a,c,*], Danielle Caled[a], Geraldo Xexéo[a,b]

[a] Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Caixa Postal 68.511, Rio de Janeiro 21941-972, RJ, Brazil
[b] Departamento de Ciência da Computação, IM/UFRJ, Caixa Postal 68.530, Rio de Janeiro 21941-590, RJ, Brazil
[c] Departamento de Ciência da Computação, IM/UFRRJ, Nova Iguaçu 26020-740, RJ, Brazil

ARTICLE INFO

Article history:
Received 4 January 2017
Revised 28 June 2017
Accepted 12 August 2017
Available online xxx

Keywords:
External Plagiarism
Heuristic Retrieval
Locality-sensitive hashing
High-dimensional spaces
Pattern clustering
Approximate nearest neighbor search
Hashing method
Hashing time reduction
Min–max hash method
Pairwise Jaccard similarity estimation
Scalable similarity search
Approximation algorithms
Computational efficiency
Nearest neighbor searches
Jaccard similarity

ABSTRACT

Heuristic Retrieval (HR) task aims to retrieve a set of documents from which the External Plagiarism detection identifies plagiarized pieces of text. In this context, we present Minmax Circular Sector Arcs (*MinmaxCSA*) algorithms that treats HR task as an approximate *k*-nearest neighbor search problem. Moreover, *MinmaxCSA* algorithms aim to retrieve the set of documents with greater amounts of plagiarized fragments, while reducing the amount of time to accomplish the HR task. Our theoretical framework is based on two aspects: (i) a triangular property to encode a range of sketches on a unique value; and (ii) a Circular Sector Arc property which enables (i) to be more accurate. Both properties were proposed for handling high-dimensional spaces, hashing them to a lower number of hash values. Our two *MinmaxCSA* methods, Minmax Circular Sector Arcs Lower Bound (CSA_l) and Minmax Circular Sector Arcs Full Bound (*CSA*), achieved Recall levels slightly more imprecise than *Minmaxwise hashing* in exchange for a better Speedup in document indexing and query extraction and retrieval time in high-dimensional plagiarism-related datasets.

1. Introduction

Plagiarists' behavior involves handling information, ideas, or another person's expression, in order to acquire some kind of advantage, such as better grades. Plagiarism is intellectual theft and, therefore, it is fraudulent behavior [1]. In 2012, more than 23,000 students across the U.S participated in a survey, which revealed that 74% of students had copied other students homework, while 32% had plagiarized a classroom assignment from the Internet [2]. The Internet is an endless source of plagiarism, given that it has a huge amount of available data and an aggressive growth rate. For instance, in September 2016, Wikipedia contained more than 41.8 million articles, with a rate of 18,551 new articles per day [3], which indicates that manual detection of Internet plagiarism is infeasible.

Plagiarism detection research in natural language evolved by taking advantage of the development of related fields such as Natural Language Processing (NLP), Information Retrieval (IR), and Cross-Language Information Retrieval (CLIR) [4]. Most current approaches are unable to satisfy the runtime requirements of high-dimensional datasets, because the approaches are not time-efficient for high-dimensional data [5]. For plagiarism detection, such as for *k*-nearest neighbor search, a widely used method to handle high-dimensional data is locality-sensitive hashing (LSH), which groups nearby items using a family of hashing functions [5,6].

The goal of LSH [7,8] methods is to hash all the items several times until similar items are more likely to be in the same set of hashes than the dissimilar ones [9]. In LSH techniques, the computational time of the nearest neighbor search is drastically reduced, at the cost of a small probability of failing to find the absolute closest match [10]. Furthermore, LSH ensures that the probability of

* Corresponding author at: Departamento de Ciência da Computação, IM/UFRRJ, 26020-740 Nova Iguaçu, RJ, Brazil.
E-mail addresses: duartefellipe@cos.ufrj.br (F. Duarte), dcaled@cos.ufrj.br (D. Caled), xexeo@cos.ufrj.br (G. Xexéo).

http://dx.doi.org/10.1016/j.knosys.2017.08.013
0950-7051/

Table 1
Throughput (documents per hour) to index PAN-PC-11 documents.

k	Throughput *Min* (doc./hour)	Throughput *Minmax* (doc./hour)	Throughput CSA_L (doc./hour)	Throughput *CSA* (doc./hour)
48	99.60	119.30	130.13	135.60
96	49.99	60.19	65.24	67.75
192	25.00	30.06	32.76	33.91
384	12.47	14.83	16.33	16.95
768	6.18	7.36	8.06	8.30

collision is much higher for near objects than for those that are distant [8]. In other words, this technique addresses a simple concept: if two objects are close, their vector projection on a subspace of lower dimensionality leads to two neighboring points [10]. Thus, to identify an object ob_j's closest neighbors, LSH method generates ob_j hash values to be compared with the hash values of all items in a document collection [8].

Minwise hashing [11] is the best known LSH-based permutation method. It aims to discover the similarity between items as a set intersection problem, which can be solved through a random sampling process conducted independently for each item [11]. For instance, consider a document represented as a set consisting of the position of words from a permuted word list, known in IR as "vocabulary". *Minwise hashing* selects the greatest lower bound from permutations of the document's numerical set. *Minwise hashing* and Jaccard similarity are highly correlated, because the probability that *Minwise hashing* will produce the same value for two sets is equal to their Jaccard similarity [9]. However, properties other than the greatest lower bound (infimum) can be used—*Minmaxwise hashing* combines the infimum and the least upper bound (supremum) values to obtain a faster and slightly more accurate solution to the problem of finding similar items [12].

This work proposes two Minmax Circular Sector Arcs (*MinmaxCSA*) methods: Minmax Circular Sector Arcs Lower Bound (CSA_L) and Minmax Circular Sector Arcs Full Bound (*CSA*). Both the CSA_L and *CSA* methods are based on our proposed triangular property, in which a unique numerical value encodes a set's interval of values for each random permutation. Our triangular property can represent with a unique value, two permuted sets, *A* and *B*, which have the same boundary values; that is, two sets with the same infimum and supremum values are represented by one hash value. Furthermore, our Circular Sector Arc geometric interpretation shows that both the CSA_L and *CSA* methods improve the triangular property accuracy by encoding a range of hash values into a unique integer.

It was proven in our novel theoretical background that the estimators of the *MinmaxCSA* methods only require pairwise equality checking and, therefore, that both methods fit in External Plagiarism's Heuristic Retrieval stage. Both CSA_L and *CSA* methods improve *Minwise hashing* time whereas, as Table 1 shows, CSA_L and *CSA* presented, respectively, 31% and 36% of improvement over *Minwise hashing*'s documents indexing Throughput. Moreover, in spite of *Minmaxwise hashing* method having a Throughput 20% higher than *Minwise hashing* method, the CSA_L and *CSA* methods spend a significantly shorter CPU time than *Minwise* and *Minmaxwise hashing* method to index a collection with many documents. Furthermore, CSA_L and *CSA* methods achieved Recall levels slightly more imprecise than *Minmaxwise hashing* in exchange for a better Speedup in document indexing and query extraction and retrieval time. Finally, it is worth noting that all evaluated methods (*Minwise hashing, Minmaxwise hashing,* CSA_L and *CSA*) fight against the curse of dimensionality by representing each document as a set of lower values of permutations when compared with the dataset vocabulary size of 520,600 words.

The rest of this paper is organized as follows: Section 2 describes the related work on the Heuristic Retrieval problem for the External Plagiarism detection task; Section 3 formalizes a workflow for locality-sensitive hashing methods based on Operator sets; in Section 4 we survey *Minwise* and *Minmaxwise hashing* as background for our proposed approaches; in Section 5 we present our triangular property and its improvement using Circular Sector Arcs and their geometric interpretation, and we also cover the formalization of the *MinmaxCSA* algorithm, computational costs, and Jaccard similarity analysis; and, finally, Sections 6 and 7 demonstrate the accuracy and time performance of *MinmaxCSA* methods in different plagiarism-related benchmarks.

2. Related work

Due to the vagueness of plagiarism's boundaries and its conceptual ambiguity, plagiarism is difficult to define [13]. For instance, we can explain plagiarism as an unacknowledged reproduction of the content generated by other individual intellectual or artistic effort. However, a plenty of plagiarism definitions can be found in the plagiarism detection literature. For instance, Maurer et al. [14] and Alzahrani et al. [4] organize the different types of plagiarism on a taxonomy in which plagiarism patterns are grouped into literal plagiarism or intelligent plagiarism.

Literal plagiarism practice, e.g. verbatim copy or phrase reordering, is common in the situations where little effort is made by the plagiarist in order to obfuscate that an unacknowledged copy was made [4]. Intelligent plagiarism practice, however, tries to hide the copy by obfuscating the original work and employing on it more complex techniques as, for example, paraphrasing, summarization, translation and active to passive voice transformations [15]. Moreover, it is difficult to identify intelligent plagiarism occurrences since the words and textual structure may be very different compared to the original composition [15].

Plagiarism Detection (PD) approaches aim to answer the following question: *Is there any kind of plagiarism in document* d_q? [16] Thus, PD approaches attempt to identify plagiarism (i) based exclusively on irregularities or inconsistencies of d_q [17] or (ii) identifying d_q plagiarism patterns, e.g. plagiarized chunks, sentences or paragraphs, on a collection of documents [18]. Besides, (i) is known as Intrinsic PD task and is out of the scope of this work, whereas (ii) is known as External, or extrinsic, PD task.

External PD task assumes that the source documents are available and reachable, i.e. if d_q has plagiarized text then all the source documents are accessible, in a collection of documents or on online sources, to the plagiarism detector system [19]. Furthermore, the aggressive increase on the size of the data makes the cost of storage and search, in massive collections of documents, a challenge to the External PD [20] task.

The External PD retrieval process is organized into three stages: (i) samples of d_q are extracted and a set of matching samples, from candidates documents, with possibly plagiarized passages are retrieved, then (ii) a set of plagiarism passages is identified through pairwise comparisons of retrieved passages and d_q samples. Finally, in (iii) all identified passages are analyzed to remove quoted passages and to join contiguous passages of text [16]. Moreover, each

Please cite this article as: F. Duarte et al., Minmax Circular Sector Arc for External Plagiarism's Heuristic Retrieval stage, Knowledge-Based Systems (2017), http://dx.doi.org/10.1016/j.knosys.2017.08.013