Afrika Statistika

Afrika Statistika
Vol. 12(1), 2017, pages 1199–1218.
DOI: http://dx.doi.org/10.16929/as/2017.1199.100

ISSN 2316-090X

The exact probability law for the approximated similarity from the Minhashing method

Soumaila Dembele [1,2] **and Gane Samb Lo** [2,3,4]

[1] DER d'Economie, Université des Sciences Sociales et de Gestion de Bamako (USSGB), Mali
[2] LERSTAD - Université Gaston Berger,Saint-Louis, Sénégal
[3] Affiliated to LASTA - Université Pierre et Marie Curie, Paris France
[4] Associate professor at African University of Sciences and Technology (AUST), Abuja, Nigeria

Received March 1, 2017; April 3, 2017

Abstract. We propose a probabilistic setting in which we study the probability law of the Rajaraman and Ullman *RU* algorithm and a modified version of it denoted by *RUM*. These algorithms aim at estimating the similarity index between huge texts in the context of the web. We give a foundation of this method by showing, in the ideal case of carefully chosen probability laws, the exact similarity is the mathematical expectation of the random similarity provided by the algorithm. Some extensions are given.

Résumé. Nous proposons un cadre probabilistique dans lequel nous étudions la loi de probabilité de l'algorithme de Rajaraman et Ullman *RU* ainsi qu'une version modifiée de cet algorithme notée *RUM*. Ces alogrithmes visent à estimer l'indice de la similarité entre des textes de grandes tailles dans le contexte du Web. Nous donnons une base de validité de cette méthode en montrant que pour des lois de probabilités minutieusement choisies, la similarité exacte est l'espérance mathématique de la similarité aléatoire donnée par l'algorithme *RUM*. Des généralisations sont abordées.

Key words: Minshashing, algorithms, similarity, estimation, probability laws, convergence of algorithm.
AMS 2010 Mathematics Subject Classification : 62E15; 62F12; 68R05; 68R15; 68Q97.

1. Introduction

In this paper, we are concerned with the evaluation of an important algorithm destined to provide the approximation of the exact similarity of two texts, in the frame of Web mining.

* Corresponding author Soumaila Dembele: dembele.soumaila@ugb.edu.sn
Gane Samb Lo : gane-samb.lo@ugb.edu.sn, gslo@aust.edu.ng

Rajaraman and Ullman (2011) proposed a detailed algorithm we denote here as the RU one. This algorithm is based on *minhashing* methods. To fix the ideas, let us consider two sets S_1 and S_2, whose total cardinality is n. The Jaccard similarity between S_1 and S_2 is defined by:

$$p=\frac{\#(S_1 \cap S_2)}{\#(S_1 \cup S_2)} \tag{1}$$

Although this expression is simple, its computation is extremely time consuming in the context in Web mining, where the data may be huge. For this reason, approximations based on probability theory and statistical methods are used.

Before we come back the our precise subject, it may be useful to say some words on the general matter. The concept of similarity has been studied and is still studied by researchers from a variety of disciplines: (see e.g. Stein and Essen, 2006, Gionis *et al.*, 1999, for visual similarity, Cha, 2007 for the use of density functions in similarity detection, Gower and Legendre, 1986 and Zezula *et al.*, 2006 for the metric space approach, Strehl *et al.*, 2000, Formica, 2005 in the context of information sciences, Bilenko and Mooney, 2003 and Theobald *et al.*, 2008 for focus similarity on large-web collections).

The current work uses a *minhashing* method (see e.g. de França, 2014) on the Iterative Universal Hash Function Generator for Minhashing, and the resemblance and containment of documents (see e.g. Broder, 1997).

One way to deal with such problems is to transform the data into a low dimension representation, supposed to preserve enough information, and to derive the similarity index on the transformed data. In order to reduce the dimensionality of a data set, some methods consist of introducing variables and feature selections or, of using a probabilistic dimension reduction technique (see e.g. Guyon and Elisseeff, 2003, Guyon, 2006, Lawrence, 2008, etc). The method we work on it in this papers uses the technique of *signatures*. Let us explain this.

Consider m subsets of S: $S_1, ..., S_m$. These sets can be represented as in Table 1, that we will call the *representation matrix* or simply the *signature* of $S_1, ..., S_m$. This representation is set up as follows.

Elements	S_1	S_2	...	S_h	...	S_l	...	S_m
1	1	0	...	0	...	1	...	1
2	0	0	...	1	...	0	...	0
...	0	...	...	...	...	...	...	...
i	1	0	...	1	...	1	...	1
...	...	...	...	...	...	...	...	...
n	0	0	...	0	...	0	...	1

Table 1. Representation matrix of $S_1, \ldots, S_m$

– We form a rectangular array of $m + 1$ columns.

– We put $S, S_1, ..., S_m$ in the first row.
– We put in the column of S all the elements of S, that we might write from 1 to n in an arbitrary order.
– In each column S_h, $1 \leq \ell \leq m$, we will put 1 or 0 on the row i depending on whether the i^{th} element of S is in S_h or not.

It is immediate from Table 1 that the following properties hold, for each couple (i, j) such that $1 \leq h \neq \ell \leq m$:

(a) the cardinality of $(S_h \cup S_\ell)$ is the number of rows in Table 1 crossing columns S_h and S_ℓ at least with a unity value.
(b) the cardinality of $(S_h \cap S_\ell)$ is the number of rows in Table 1 crossing both columns S_h and S_ℓ with a unity value.

Hence, the representation matrix allow to get, visually, the similarity between S_h and S_ℓ. In particular, by denoting $S_h = (S_{ih}, 1 \leq i \leq n)^T$ for $1 \leq h \leq m$, where ^{T} stands for the transpose of a matrix, we have

$$sim(S_h, S_\ell) = \frac{\#\{i, 1 \leq i \leq n, S_{ih} = S_{i\ell} = 1\}}{\#\{i, 1 \leq i \leq n, (S_{ih} + S_{i\ell} = 1) + (S_{ih} = S_{i\ell} = 1)\}}, \tag{2}$$

which can be written as

$$sim(S_h, S_\ell) = \frac{\#\{i, 1 \leq i \leq n, S_{ih} + S_{i\ell} = 2\}}{\#\{i, 1 \leq i \leq n, (S_{ih} + S_{i\ell} = 1) + (S_{ih} + S_{i\ell} = 2)\}}. \tag{3}$$

The RU algorithm consists of reducing this matrix to a much less one with the help of *minhashing* functions and, of computing the similarity between two columns which is meant to approximate the similarity between the sets represented by these columns.

Although it may be empirical observed that the approximation may be relevant, it does not exist, up to our knowledge, an theoretical evaluation of the discrepancy between the exact similarity and the estimated similarity provided by the RU method. The papers will fill this gap. Beyond that, it lays out a probabilistic frame to handle the problem and opens new research trends.

The rest of the paper is organized as follows. In Section 2, the full description of the RU algorithm is given. Interesting remarks and properties will be addressed. Computation aspects will also be highlighted in this section. New forms, more appropriate to address the probability problem, will be given and a slightly modified algorithm, named RUM, is proposed. In section 3, the RU algorithm will be approached in a probability theory frame and the probability law of the random similarity is given and some consequences, among them the deviation from the true similarity, is characterized. Next, we study some conditions under which convergence of the RU is explained. Finally, in a pure and random scheme, we completely justify the RU algorithm in the probabilistic approach. Concluding remarks are stated in Section 4.

Element	S_1	S_2	...	S_m	$\mathcal{Z}_1$	...	$\mathcal{Z}_k$	
1	1	0	...	0	$z_1(1)$	...	$z_k(1)$	$\mathcal{Z}^{(1)}$
2	0	0	...	1	$z_1(2)$	...	$z_k(2)$	$\mathcal{Z}^{(2)}$
.	0	.	...	.	.	...	.	.
i	1	0	...	1	$z_1(i)$	...	$z_k(i)$	$\mathcal{Z}^{(i)}$
.	.	.	...	.	.	.	.	.
n	0	0	...	0	$z_1(n)$	...	z_k(n)	$\mathcal{Z}^{(n)}$

Table 2. Extension of the representation matrix by minhashing columns

minhashes	$t(S_1)$	$t(S_2)$	...	$t(S_m)$
1	c_{11}	c_{12}	...	c_{1m}
2	c_{21}	c_{22}	...	c_{2m}
...	...	.	...	.
k	c_{k1}	c_{k3}	...	c_{km}

Table 3. Signature matrix

2. RU and RUM algorithms

It is based on the notion of minhashing to reduce sets of huge sizes into sets of small sizes called signatures. The computation of the similarity is done on their compressed versions, i.e, on their signatures. To better explain this notion, let us consider m subsets of a reference set S of size $n \geq 1$ and let us use their *representation matrix* as in Table 1.

Let us consider $k \geq 1$ functions z_α $(\alpha = 1, ..., k)$ from $\{1, ..., n\}$ to itself in the following form:

$$z_\alpha(i) = x_\alpha i + y_\alpha \bmod n, \tag{4}$$

where x_α and y_α, $1 \leq \alpha \leq k$, are given integers. We modify this function in the following way: $z_\alpha(i) = n$ when the remainder of the Euclidean division is zero.

Next, we extend Table 1 by adding k columns $\mathcal{Z}_1, ..., \mathcal{Z}_k$, such that each $\mathcal{Z}_i$ is the transpose of $(z_\alpha(1), ..., z_\alpha(n))$. The resulting table is Table 2. Let us denote by $\mathcal{Z}$ the $(n \times k)$-matrix whose columns are $\mathcal{Z}_1, \mathcal{Z}_2, ..., \mathcal{Z}_k$ and let us denote its lines rows by $\mathcal{Z}^{(1)}, \mathcal{Z}^{(2)}, ..., \mathcal{Z}^{(n)}$.

The *RU* algorithm replaces Table 1 by a shorter one called *minhashing* signature represented in Table 3

Table 3 is obtained as follows, according to the method described in Rajaraman and Ullman (2011), page 65.

Algorithm of filling the columns S_j.

1. Set all the $c_{\alpha j}$ equal to ∞.
2. For each column S_j, proceed like this
   2-a. for each element i, from 1 to n, compute $z_1(i), z_2(i), ..., z_k(i)$.
   2-b. if i is not in S_j, then do nothing and go to $i+1$
   2-c. if i is in S_j, replace all the rows $(c_{\alpha j})_{1\leq r\leq k}$ by the minimum: $\min(c_{\alpha j}, h_\alpha(i))$.
   2-d. go to $i+1$
3. Go to $j+1$
4. End.

At the end of the procedure, each column will contain only integers between 1 and n. The estimated similarity between two subsets S_h and S_ℓ, $1 \leq h \leq\neq \ell \leq m$, based on this compressed table, is taken as the similarity of the columns $t(S)_h$ and $t(S)_\ell$ in the signature matrix which is

$$simRU(S_h, S_\ell) = \frac{\#(t(S_h) \cap t(S_\ell))}{k}. \tag{5}$$

The sets $t(S_h)$ and $t(S_\ell)$ are subsets of $\{1, 2, ..., n\}$ and stand for transformed sets of S_h and S_ℓ through the *minhashing* procedure.

From there, it is very important to give this remark. The algorithm is meant to gain time and to get an approximation of the similarity. It is based on the representation matrix. But, if we spent the required time to get it, there is nothing else to do, since the exact similarity is automatically read in virtue of Formulas 2 and 3. We have to modify the RU algorithm form a practical point of view.

The resulting modification, called RUM, consists of the following. Suppose that we want to find and estimated similarity between S_h and S_ℓ. We proceed as follows.

1. Form one set $S_{h,\ell}$ by putting the elements of S_h and then the elements of S_ℓ by putting twice elements of the intersection.

1. Form the representation matrix with $N = n_1 + n_2$ lines.
2. Apply the RU algorithm to this collection by using Criterion (C).

We do not seek to find the intersection. Elements of the intersection are counted twice here. The result is that we do not loose time in forming the representation matrix.

But, we will now have two approximations. First, we replace the representation of the RU approach by that of the RUM one. Next, we replace the latter by the signature matrix.

How is affected the original similarity? The RUM algorithm actually seeks at estimating the modified similarity between two subsets S_h and S_ℓ, $1 \leq h, \ell \leq m$. Let us denoted by $simM(S_h, S_\ell)$. It is immediately seen that we still have a zero similarity index, that is $simM(S_h, S_\ell) = 0$, if the two sets S_h and S_ℓ are disjoint, and a 100% index if the sets are identical. In the general case, the number of rows is now $\#(S_h \cup S_\ell) + \#(S_h \cap S_\ell)$ and the

common elements of the columns S_h and S_ℓ is $\#(S_h \cap S_\ell)$. The modified similarity between S_h and S_ℓ, is

$$simM(S_h \cap S_\ell) = \frac{2\#(S_h \cap S_\ell)}{\#(S_h \cup S_\ell) + \#(S_h \cap S_\ell)}, \tag{6}$$

which gives

$$simM(S_h \cap S_\ell) = \frac{2sim(S_h \cap S_\ell)}{1 + sim(S_h \cap S_\ell)} \tag{7}$$

and, reversely,

$$sim(S_h \cap S_\ell) = \frac{simM(S_h \cap S_\ell)}{2 - simM(S_h \cap S_\ell)} \tag{8}$$

It is also clear that from the previous formulas that sim and $simM$ take any of the value zero and one simultaneously.

We adopt the following rule : We use the modified RU algorithm in place of the original one. We will avoid to find the intersection, which in fact would stop our procedure since the similarity is already found, and by then, we gain a huge amount of time. At the end of the RU algorithm implementation on the modified set, we apply Formula 8, to get the approximation

$$simRU(S_h \cap S_\ell) = \frac{simRUM(S_h \cap S_\ell)}{2 - simRUM(S_h \cap S_\ell)} \tag{9}$$

Now, the question is how accurate is the approximation? Empirical studies strongly support the method. For instance, the four canonical Gospels have been compared with the target of assessing the hypothesis of the existence of a hidden or lost sources, named **Q** source, from which the current gospels are derived. The Gospel have been transformed into sets of words of $p = 3$ letters (named *p-shingles*). The numbers of 3-shingles of the four gospels are at least 55.000 and at most 110.000. The shortest time to compute the exact similarity between two gospels is around eight (8) minutes while the computation of the similarity between John and Matthews Gospels requires 3080 seconds (around 51 minutes). By using the RUM algorithm with only a small number $k = 5$ of *minhashing* functions, estimations of the similarity indices are obtained with a much smaller time, around 20 seconds. The estimated values showed clear trends for the exact values.

Clearly here, the choice of the *minhashing* functions is arbitrary. Without saying it, their choice is subject to a probability law. Implicitly, the uniform law is assumed. Even in that implicit choice, we did not see a study on the exact final probability law.

In the forthcoming section, we deal with the probability law of $simRU$, considered as a random variable. We will study it with respect to the probability law of the random coefficients a_i and b_i, $i = 1, ..., k$.

3. Probabilistic approach

Let us give a probabilistic approach of the similarity.

I - The similarity as a conditional probability.

We adopt the notation introduced in the previous section, in particular the representation matrix in Section 1. Now, we suppose that we pick at random a row X from the number of lines in Column one in the Table 1 and for each h, $1 \leq h \leq m$, let $X_{k,\ell}$ be the Bernoulli random variable taking the value at the crossing between the column S_h and the row X. We are going to see that the random variable X guide the similarity index.

Theorem 1. *Let us randomly pick a row X among n rows. Let $S_{X,h}$ be the value of the row X at the crossing with a column S_h, $1 \leq h \leq m$ in the representation matrix. Then the similarity between two sets S_h and S_ℓ, $1 \leq h \leq \ell$, is the conditional probability of the event $(S_{X,h} = S_{X,\ell} = 1)$ with respect to the event $(S_{X,h} + S_{X,k} \geq 1)$. i.e*

$$sim(S_\ell, S_h) = \mathbb{P}[(S_{X,h} = S_{X,\ell} = 1)/(S_{X,h} + S_{X,\ell} \geq 1)].$$

Proof. We first observe that for the defined matrix below, the set of rows can be split into three classes, based on the columns S_ℓ and S_h:

1. The rows (A) that cross both columns S_ℓ and S_h with unity values.

2. The rows (B) that cross S_ℓ and S_h with a unity value and a null value.

3. The rows (C) that cross both S_ℓ and S_h with null values.

Let us show that $sim(S_\ell, S_h) = \mathbb{P}[(S_{X,h} = S_{X,\ell} = 1)/(S_{X,h} + S_{X,\ell} \geq 1)]$.

Clearly, the similarity is the ratio of the number of rows (A) to the sum of the numbers of rows X and the number of rows (B). The rows (C) are not involved in the similarity between S_h and S_k. Thus

$$sim(S_\ell, S_h) = \frac{\#\{i, 1 \leq i \leq n, S_{X,h} = 1, S_{X,\ell} = 1\}}{\#\{i, 1 \leq i \leq n, (S_{X,h} + S_{X,\ell} = 1) + (S_{X,h} = 1, S_{X,\ell} = 1)\}}.$$

Then, by dividing the numerator and the denominator by n, we will have

$$sim(S_\ell, S_h) = \frac{\frac{\#\{i, 1 \leq i \leq n, S_{X,h} = 1,\ S_{X,\ell} = 1\}}{n}}{\frac{\#\{i, 1 \leq i \leq n, (S_{X,h} + S_{X,\ell} = 1) + (S_{X,h} = 1, S_{X,\ell} = 1)\}}{n}}.$$

Hence we get the result

$$sim(S_\ell, S_h) = \mathbb{P}[(S_{X,h} = S_{X,\ell} = 1)/(S_{X,h} + S_{X,\ell} \geq 1)].$$

This theorem will be the foundation of the statistical estimation of the similarity as a probability.

Important remark. When we consider the similarity of two subsets, say S_h and S_k and we use the global space as $S_h \cup S_\ell$, we may see that the similarity is, indeed, a probability. But when we simultaneously study the joint similarities of several subsets, say at least S_h, S_k and S_ℓ with the global set $S_h \cup S_k \cup S_\ell$, the similarity between two subsets is a **conditional** probability. Then, using the fact that the similarity is a probability to prove the triangle inequality is not justified, as claimed in Rajaraman and Ullman (2011), page 76.

II - Expected or Normal Similarity.

Before we begin, we stress that the notation k and m are not related to those in the other sections. These notation should stay specific to the problem handled here.

We shall use the language of the urns. Suppose that we have a reference set of size n that we take as an urn **U**. We pick at random a subset X of size k and a subset Y of size m. If m and k have not the same value, the picking order of the sets does have an impact on our results. We then proceed at the beginning by picking at random the first subset, that will be picked all at once, next put it back in the urn **U** (reference set). Then we pick the other subset. Let us ask ourselves the question : what is the expected value of the similarity of Jaccard?

The answer at this question allows us later to appreciate the degree of similarity between the texts. We have the following result :

Proposition 1. *Let U be a set of size n. Let us randomly pick two subsets X and Y of U, of respective sizes m and k according to the scheme described above. We have*

$$\mathbb{P}(Card(X \cap Y) = j) = \frac{1}{2}\left(\frac{C_k^j\, C_{n-k}^{m-j}}{C_m^n} + \frac{C_m^j\, C_{n-m}^{k-j}}{C_k^n}\right)\mathbb{I}_{(0\leq j\leq \min(k,m))}. \tag{10}$$

For all $p \leq 1$, the p-th moment of the random similarity $sim(X,Y)$ is given by

$$\mathbb{E}(sim(X,Y)) = \sum_{j=0}^{\min(k,m)} \frac{j}{2(m+k-j)}\left\{\frac{C_k^j\, C_{n-k}^{m-j}}{C_m^n} + \frac{C_m^j\, C_{n-m}^{k-j}}{C_k^n}\right\}. \tag{11}$$

Proof. Let us use the scheme described above. Let us first pick the set X. We have $L = C_n^k$ possibilities. Let us denote the subsets that would take X by $X_1, ..., X_L$. The searched probability becomes

$$\begin{aligned}\mathbb{P}(Card(X \cap Y) = j) &= \sum_{s=1}^{L} \mathbb{P}((Card(X \cap Y) = j) \cap X_s) \\ &= \sum_{s=1}^{L} \mathbb{P}((Card(X \cap Y) = j)/X_s)\mathbb{P}(X_s).\end{aligned}$$

Once X_s is chosen and fixed, we get

$$\mathbb{P}((Card(X \cap Y) = j)/X_s) = \frac{C_m^j \quad C_{n-m}^{k-j}}{C_m^k} .$$

To explain this, we start with the fact that X_s is fixed and contains C_k^j combinations of j elements. Now, we have to choose a combination $\mathcal{C}$ of m elements from n elements which contains one of the C_k^j combinations of X_s in such a way that none of the other elements of $\mathcal{C}$ is in X_s. This means that on should choose first a combination of j among the k elements of X_s with C_k^j ways, and next one completes with a combination of $m-j$ elements among the $n-k$ elements of the complement of X_s.

Now, since $\mathbb{P}(X_s) = 1/C_n^k = 1/L$, we conclude that

$$\mathbb{P}(Card(X \cap Y) = j) = \sum_{s=1}^{L} \frac{C_k^j \; C_{n-k}^{m-j}}{C_m^n}(1/L) = \frac{C_k^j \; C_{n-k}^{m-j}}{C_m^n}.$$

The result corresponding to picking up Y first, is obtained by symmetry of roles of k and n. We then get (10). The formula (11) comes out immediately since

$$sim(X,\ Y) = \frac{\#(X \cap Y)}{\#(X \cup Y)} = \frac{\#(X \cap Y)}{m + k - \#(X \cap Y)}. \tag{12}$$

■

III - Approximated Similarity Based on the Strong Law of Large Number.

Since the similarity is a conditional probability in according to Theorem 1, we can deduce a strong law of Large numbers, which is by the way a Glivenko-Cantelli property in the discrete case, in the following way.

Theorem 2. *Let $sim(S_1, S_2)$ be the similarity between two subsets S_1 and S_2 of a set whose size is considered very large. Let us pick at random a subset $S_{1,n}$ from S_1 with size $n(1)$ and a subset $S_{2,n}$ from S_2 of size $n(2)$ and let us consider the random similarity $sim_n(S_1, S_2) = sim(S_{1,n}, S_{2,n})$ between $S_{1,n}$ and $S_{2,n}$. Then $sim_n(S_1, S_2)$ converges almost-surely to $sim(S_1, S_2)$ with at rate of convergence in the order of $(n(1) + n(2))^{-1/4}$ when $n(1)$ and $n(2)$ become simultaneously large.*

That is a direct consequence of the classical theorem of Glivenko-Cantelli.

Finally, we come to the probability law induced by the RU algorithm.

IV - Probability law induced by the *minhashing* method and application.

A - Two other alternative versions A simple criterion.

Before to give two alternate versions of the *RU/RUM* algorithm. The first will be particularly useful while addressing the probability law. Also it leads to a new procedure that will be the base of the implementation of the algorithm in computer packages.

(a) A simple criterion.

If we look carefully at the algorithm, we may see that we have the following criteria.

Criterion (C). The transpose of each column

$$t(S_h) = [(c_{\alpha h})_{1\leq\alpha\leq k}],\ 1 \leq h \leq m,$$

in Table 3 is the minimum of the rows $\mathcal{Z}^{(i)} = (z_1(i), ..., z_k(i))$ of Table 2, when i covers the elements i of S_h, where the minimum is operated coordinate-wisely.

The proof comes easily by looking at simple cases with small cardinalities. The induction to arbitrary cardinalities is immediate.

This simple remark allows to set up programs in a much easier way through a kind of Markov process.

(b) A version if form a Markov process.

We want to form the final transformed matrix signature as defined in Table 3 by denoting U_h as the column associated with $t(S_h)$ and U_ℓ as the column associated with $t(S_\ell)$. We remind that U_h and U_ℓ are vectors of dimension k. This procedure will be implemented is easy to implement into computer packages.

We remind that in the original matrix signature, the elements of $S_h \cup S_\ell$ are given in an arbitrary order $(\sigma_0(i),\ 1 \leq i \leq N)$. We denote by $C(i, h)$ the value at which the row i and the column h cross each other in Table 2. In what follows, for any Boolean variable C, $\mathbb{I}(C)$ stands for the indicator function of C, which takes the value one if C holds and zero otherwise.

Let us express *RU* algorithm as a final step of Markov process.

We fix h, $1 \leq h \leq m$. The following procedure iteratively forms the final value of U_h.

Step 1. Do for each h, $1 \leq h \leq m$: $(1a)$. Take $U_h^0 = (n+1, n+1, ..., n+1)^t \in \mathbb{R}^k$. Put U_h^0 in the column $t(S_h)$ of Table 3.

Sub-step $(1b)$. For each i from 1 to n, we take

$$U_h^i = U_h^{i-1}\mathbb{I}(C(i,h) = 0) + \min(U_h^{i-1}, (\mathcal{Z}^{(i)})^T)\mathbb{I}(C(i,h) = 1).$$

.
Step 2. For each $1 \leq h \neq \ell \leq m$, compute the estimated similarity :

$$simRU(S_h, S_\ell, \sigma_0) = \frac{1}{k}\sum_{\alpha=1}^{k} I(U_h(\alpha) = U_\ell(\alpha)). \tag{13}$$

This second algorithm is more simple to implement.

B - Probability laws.

Now we are going to compute the estimated similarity *simrum* in a complete randomly experience. We consider two subsets S_ℓ and $S_{\ell'}$ of S. We allow the elements of S be ordered according to a permutation σ of $\{1, 2, ..., n\}$. We denote set of permutations of $\{1, 2, ..., n\}$ as $\mathcal{S}_n$ and consider the probability space $(\mathcal{S}_n, \mathcal{P}(\mathcal{S}_n), \mathbb{P}_0)$, where $\mathcal{P}(\mathcal{S}_n)$ is the power set of $\mathcal{S}_n$ and $\mathbb{P}_0$ is the uniform probability measure on $\mathcal{S}_n$ defined by $\mathbb{P}_0(\{\sigma\}) = \frac{1}{N!}$ for $\sigma \in \mathcal{S}_n$.

Next, we choose the following *minhashing* function

$$Z_\alpha(i) = X_\alpha i + Y_\alpha \bmod n, \quad i = 1, ..., n.$$

with a random generation of the integers $(X_1, Y_1), \ldots, (X_k, Y_k)$.

From there, a number of possibilities may be conceived. Do we take the (X_α, Y_α)'s as independent? independent and identically independent? dependent according a what copula? etc. We may also discuss about the dependence between X_α and Y_α for each $\alpha = 1, \ldots, k$?

As a first step, let us suppose that :

(H) $(X_1, Y_1), \ldots, (X_k, Y_k)$ are independent and identically distributed with common probability law $\mathbb{P}_{(X,Y)}$.

We apply the *RUM* algorithm and observe the estimated random similarity between S_ℓ and $S_{\ell'}$

$$simrum(S_\ell, S_{\ell'}) = \frac{1}{k}\sum_{\alpha=1}^{k} I(U_\ell(\alpha) = U_{\ell'}(\alpha)).$$

If there is no risk of confusion, we simply write *simrum* at the place $simrum(S_\ell, S_{\ell'})$ as we also use *sim* and *simru* at the place of $sim(S_\ell, S_{\ell'})$ and $simru(S_\ell, S_{\ell'})$ respectively. We are now going to give the probability law of *simrum* after the following notations. The matrix $\mathcal{Z}$ in Table 2 is random now and we denote

$$\Gamma_\ell(\sigma) = \{i \in [1, n], \quad C(i, \ell, \sigma) = 1\}.$$

Introduce the following notation. Let $1 \leq t \leq k$ define $\mathcal{B}_t$ as the set of all t-tuples. For $(\beta 1, ..., \beta_t) \in \mathcal{B}_t$, define $(\beta_1, ..., \beta_t)^c$ as the complement of the set of $\{\beta_1, ..., \beta_t\}$ in $\{1, ..., n\}$.

Define also for $1 \leq p, q \leq n$, $1 \leq \ell, \ell' \leq m$, $\sigma \in \mathcal{S}$,

$$\overline{m}(p, q, \sigma, \ell) = \min\{p + iq \mod n,\ i \in \Gamma(\sigma)\}$$

and

$$B(\ell, \ell', \sigma) = \{(p, q), 1 \leq p, q \leq n, \overline{m}(p, q, \sigma, \ell) = \overline{m}(p, q, \sigma, \ell')\}$$

The probability law of the estimated similarity is the following.

Theorem 3. *By Criterion (C) U_h is given as follows.*

$$U_h = \min(\mathcal{Z}^{(i)},\quad i \in \Gamma_h(\sigma))^T \in \mathbb{R}^k,\ h = \ell, \ell', \tag{14}$$

where the minimum of rows is done by coordinate-wisely. Then $1 \leq \alpha \leq k$,the probability of the event $(U_\ell(\alpha) = U_{\ell'}(\alpha))$ is given by

$$\mathbb{P}(U_\ell(\alpha) = U_{\ell'}(\alpha)) = \frac{1}{n!} \sum_{\sigma \in S_n} \sum_{(p,q) \in B(\ell, \ell', \sigma)} \mathbb{P}(X_\alpha = p, Y_\alpha = q). \tag{15}$$

Moreover, the probability law of simrum is given by the discrete probability measure defined on $\mathcal{V} = \{s \in [0, 1], ks \in \mathbb{N}\}$ by

$$\mathbb{P}(simrum = s) = \sum_{C \in B_t} \prod_{\alpha \in C} \mathbb{P}((U_\ell(\alpha) = U_{\ell'}(\alpha)) \times \prod_{\alpha \notin C} \mathbb{P}((U_\ell(\alpha) \neq U_{\ell'}(\alpha)). \tag{16}$$

for $s \in \mathcal{V}$.

Proof. We are going to compute the probability of the event $(U_\ell(\alpha) = U_{\ell'}(\alpha))$. First, it follows from the algorithm that (14) is straightforward, that is

$$U_\ell = \min(\mathcal{Z}^{(i)},\ i \in \Gamma_\ell(\sigma))^T \in \mathbb{R}^k,\ \ell = 1, 2.$$

Now we are going to estimate the probability law of the event $(U_\ell(\alpha) = U_{\ell'}(\alpha))$. We point out that, conditionally on σ, the couples $(U_\ell(\alpha), U_{\ell'}(\alpha))$ are independent, and probabilities for events only depending on $(U_\ell(\alpha), U_{\ell'}(\alpha))$, are computed with $\mathbb{P}_{(X_\alpha, Y_\alpha)}$. We get

$$\begin{aligned}
&\mathbb{P}(U_\ell(\alpha) = U_{\ell'}(\alpha)) \\
=\ &\mathbb{P}(\min(Z_\alpha(i),\quad i \in \Gamma_\ell(\sigma)) = \min(Z_\alpha(i),\quad i \in \Gamma_{\ell'}(\sigma)) \\
=\ &\mathbb{P}(\min(X_\alpha + iY_\alpha \bmod N,\quad i \in \Gamma_\ell(\sigma)) = \min(X_\alpha + iY_\alpha \bmod N,\quad i \in \Gamma_{\ell'}(\sigma)).
\end{aligned}$$

By using the notation $\overline{m}(p, q, \sigma, \ell)$ and $B(\ell, \ell', \sigma)$ introduced above, we finally get

$$\begin{aligned}
\mathbb{P}(U_\ell(\alpha) &= U_{\ell'}(\alpha)) = \mathbb{P}_{(\sigma, X_\alpha, Y_\alpha)}(B(\ell, \ell', \sigma)) \\
&= \mathbb{P}_\sigma \otimes \mathbb{P}_{(X_\alpha, Y_\alpha)}(B(\ell, \ell', \sigma)) \\
&= \frac{1}{n!} \sum_{\sigma \in S_n} \mathbb{P}_{(X_\alpha, Y_\alpha)}(B(\ell, \ell', \sigma)) \\
&= \frac{1}{n!} \sum_{\sigma \in S_n} \sum_{(p,q) \in B(\ell, \ell', \sigma)} \mathbb{P}(X_\alpha = p, Y_\alpha = q)
\end{aligned}$$

and (15) is proved. Next, we recall that

$$simrum = \frac{\#\{1 \leq \alpha \leq k, \;\; U_\ell(\alpha) = U_{\ell'}(\alpha)\}}{k},$$

which entails

$$simrum \in \#\left\{\frac{t}{k}, \; 1 \leq \alpha \leq k\right\}, \; s = \frac{t}{k}, \; t = sk \in \mathbb{N}.$$

Hence,

$$\begin{aligned}\mathbb{P}(simrum &= s) = \mathbb{P}(simrum = t) \\ &= \mathbb{P}\left(\frac{\#\{1 \leq \alpha \leq k, \;\; U_\ell(\alpha) = U_{\ell'}(\alpha) = t\}}{k}\right).\end{aligned}$$

Let us put

$$B_t = \{\alpha_{i_1} < \alpha_{i_2} < ... < \alpha_{i_t} : U_\ell(\alpha_i) = U_{\ell'}(\alpha_i), \quad \forall 1 \leq i \leq t, U_\ell(r) \neq U_{\ell'}(r), \; \forall r \in \{1, ..., k\}, r \neq \alpha_i\}.$$

Let D_t be the ordered subsets of $\{1, ..., k\}$ of size t. If $(\alpha_{i_1}, \alpha_{i_2}, ..., \alpha_{i_t}) \in D_t$, we denote $(\alpha_{i_1}, \alpha_{i_2}, ..., \alpha_{i_t})^c$ as its complement in D_t.

Thus, we have

$$B_t = \{c = (\alpha_{i_1}, \alpha_{i_2}, ..., \alpha_{i_t}) \in D_t : \forall 1 \leq i \leq t, \; U_\ell(\alpha_i) = U_{\ell'}(\alpha_i), \; \forall r \in c^c \;\; , U_\ell(r) \neq U_{\ell'}(r)\}.$$

Let $B \in B_t$, we have

$$\begin{aligned}B(c) &= \bigcap_{\alpha \in c}(U_\ell(\alpha) = U_{\ell'}(\alpha)) \bigcap \bigcap_{\alpha \notin c}(U_\ell(\alpha) \neq U_{\ell'}(\alpha)) \\ &= B_1(c) \bigcap B_2(c).\end{aligned}$$

We conclude that

$$\begin{aligned}\mathbb{P}(simrum &= s) = \sum_{c \in B_t} \mathbb{P}(B_1(c) \bigcap B_2(c)) \\ &= \sum_{c \in B_t} \prod_{\alpha \in c} \mathbb{P}(U_\ell(\alpha) = U_{\ell'}(\alpha)) \times \prod_{\alpha \notin c} \mathbb{P}(U_\ell(\alpha) \neq U_{\ell'}(\alpha))\end{aligned}$$

This completes the proof of Theorem.■

We have the following consequence.

Corollary 1. *For any $p \in [0,1]$, we have*

$$\begin{aligned}\mathbb{P}(|simrum - p| \leq \varepsilon) &= \sum_{p-\varepsilon \leq s \leq p+\varepsilon} \mathbb{P}(simrum = s) \\ &= \sum_{[k(p-\varepsilon) \leq t \leq k(p+\varepsilon)]} \mathbb{P}(simrum = \frac{t}{k}).\end{aligned} \tag{17}$$

Proof. Let us put $p = s$ in (1).

We get

$$\begin{aligned}\mathbb{P}(|simrum - p| \leq \varepsilon) &= \mathbb{P}(p - \varepsilon \leq simrum \leq p + \varepsilon) \\ &= \sum_{p-\varepsilon \leq s \leq p+\varepsilon} \mathbb{P}(p_3 = s).\end{aligned}$$

Since $s = t/k$, we obtain

$$\mathbb{P}(|simrum - p| < \varepsilon) = \sum_{[k(p-\varepsilon) \leq t \leq k(p+\varepsilon)]} \mathbb{P}(simrum = \frac{t}{k}).$$

□

As a first application, Formula 16 allows to compute the p-th moment of $simrum$, for $p \geq 1$, which is

$$fi\mathbb{E}(simrum^p) = \sum_{s \in \mathcal{V}} s^p \mathbb{P}(simrum = s).$$

Let us denote by SIMRUM the mathematical expectation of $simrum$, that is $SIMRUM = \mathbb{E}(simrum)$. In the context of discrete random variables, we may use Formula 17 to find a 95%-confidence interval by using an iterative procedure.

In conclusion, this result allows to have relevant confidence intervals of random modified similarity, and by then, of the random similarity through Formula 9. The comparison between the true similarity and the estimated similarity will no longer be done with a sole observation, but with respect to the whole confidence interval. This makes the comparison more reliable.

Several interesting questions remain open. For example, what is the efficiency of the estimation? What is the impact of the probability law of $\mathbb{R}^{2k}$-random variable

$$((X_1, Y_1), \ldots, (X_k, Y_k)),$$

on the quality of the estimation? What happens as k gets bigger?

Answering all these questions are beyond the scope of this paper. But, at least, we are going to give definite results on the RU algorithm as a statistical method and lay out the general case.

C - Assessment of the RU algorithm as an estimation method.

Let us begin to give the main idea of the method. This time suppose that the set $S = \{1, ..., n\}$ is given in a fixed order in the representation matrix and we write it from yhe top to the bottom in its natural order. It is attempted to consider k random and independent permutations Z_α, $1 \leq \alpha \leq k$ of the set $S = \{1, ..., n\}$. Suppose we are able to get them. We may complete the algorithm. At the arrival, we have the probability law of *simrum* through the following notation. Let us introduce this new notation, for $L \subset \{1, 2, ..., n\}$, $\sigma \in \mathcal{S}_n$,

$$Min_L(\sigma) = \min_{i \in L} \sigma(i),$$

For $1 \leq \ell \leq n$, we make the following abuse of notation and write $Min_{\Gamma(\ell)} = Min_\ell$. Now, denote for $1 \leq \ell \leq n$, $1 \leq \ell \leq \ell' \leq m$,

$$B(\ell, \ell') = \{\sigma \in \mathcal{S}_n, \pi_\ell(\sigma) = \pi_{\ell'}(\sigma)\}.$$

The probability law *simrum* is still given by

$$\mathbb{P}(simrum = s) \sum_{c \in B_t} \prod_{\alpha \in c} \mathbb{P}(U_\ell(\alpha) = U_{\ell'}(\alpha)) \times \prod_{\alpha \notin c} \mathbb{P}(U_\ell(\alpha) \neq U_{\ell'}(\alpha))$$

with

$$\mathbb{P}(U_\ell(\alpha) = U_{\ell'}(\alpha)) = \mathbb{P}_{Z_\alpha}(B(\ell, \ell')).$$

for $ks \in \mathbb{N}$.

Here, what is expected is that two different rows $\mathcal{Z}^i$ and $\mathcal{Z}^j$, $1 \leq i \leq j \leq$ will be probably disjoint, or at the least, that the probability they are not disjoints is very low. Suppose for a while that this is the case. Let us denote by $I(\ell - \ell') =: I_1$ the set of lines pertaining to elements of $S_\ell \setminus S_{\ell'}$, $I(\ell' - \ell) =: I_3$ the set of lines pertaining to elements of $S_{\ell'} \setminus S_\ell$ and $I(\ell + \ell') =: I_2$ the set of lines pertaining to elements of $S_\ell \cap S_{\ell'}$. It is clear that

$$U_\ell = \min \left(\min_{i \in I_2} \mathcal{Z}^i, \ \min_{i \in I_1} \mathcal{Z}^i \right)$$

and

$$U_{\ell'} = \min \left(\min_{i \in I_2} \mathcal{Z}^i, \ \min_{i \in I_3} \mathcal{Z}^i \right).$$

Thus, in the hypothesis of disjoint lines $\mathcal{Z}^i$'s, any event $BU(\alpha) = (U_\ell(\alpha) = U_{\ell'}(\alpha))$ is surely achieved through the part

$$\min_{i \in I_2} \mathcal{Z}^i,$$

meaning that $(U_\ell(\alpha) = U_{\ell'}(\alpha))$ is equivalent to the event

$$\left(\min_{i\in I_2} \mathcal{Z}^i_\alpha < \min_{i\in I_1\cup I_3} \mathcal{Z}^i\right)$$

and hence, by denoting

$$qrum = \#\{\alpha, BU(\alpha) holds\}.$$

Similarly to the previous steps, we may denote

$$t(B)(\ell,\ell') = \{\sigma \in \mathcal{S}_n, \min_{i\in I_2} \sigma(i) < \min_{i\in I_1\cup I_3} \sigma(i)\}. \tag{18}$$

The probability law *simrum* is still given by

$$\mathbb{P}(simrum = s) = \sum_{C\in B_t} \prod_{\alpha\in C} \mathbb{P}_{Z_\alpha}(t(B)(\ell,\ell')),$$

for $ks \in \mathbb{N}$.

Before we conclude, let us address two points.

Point (a) The previous developments are based on choosing random permutations. This is very time-consuming when n is large. Consider functions of the form $Z_\alpha(i) = iX_\alpha + Y_\alpha$ mod n, $1 \le i \le n$, $1 \le \alpha \le k$, is a way to quickly have almost permutations of a small number of repetitions among the set $\{Z_\alpha(i),\ 1 \le i \le n\}$.

Point (b) To achieve the target property in Point (a), we may simply consider a random variable

$$Z_\alpha = (Z_\alpha(i),\ 1 \le i \le n)$$

with values in some space D, with a size at least equal to n. For $1 \le i \ne j \le n$, $1 \le \alpha \le k$, denote the probability of the event that the two lines $\mathcal{Z}^i_\alpha$ and $\mathcal{Z}^j_\alpha$) have at least on common coordinate by

$$\begin{aligned} p_{i,j} &= \mathbb{P}(\mathcal{Z}^i_\alpha = \mathcal{Z}^j_\alpha)) \\ &= \mathbb{P}_{Z_\alpha}(\{\sigma \in \mathcal{S}_n, \sigma(i) = \sigma(j)\}) \end{aligned}$$

By independence and stationary, the probability of the complement of the event $D_{i,j}$ that the lines $\mathcal{Z}^i$ and $\mathcal{Z}^j$, $1 \le i \le j \le$, are disjoint is

$$\mathbb{P}(D_{i,j}) = (1 - p_{i,j})^k.$$

Next, denote by $\mathcal{C}_{n,r}$ the class of lines $1 \le i \ne j \le n$ such that the lines $\mathcal{Z}^i$ and $\mathcal{Z}^j$ have exactly $r \ge 1$ common coordinates. Denote the probability of the event D_n that all the lines are mutually disjoint each other. We have

$$\mathbb{P}(D_n) = 1 - \left(\sum_{r=1}^{k} \#(\mathcal{C}_{n,r}) p^r_{i,j} (1 - p_{i,j})^{k-r}\right),$$

or

$$\mathbb{P}(D_n) = 1 - \left(\sum_{r=1}^{k} \#(\mathcal{C}_{n,r}) \mathbb{P}_{Z_\alpha}(\mathcal{D}_{i,j})^r (1 - \mathbb{P}_{Z_\alpha}(\mathcal{D}_{i,j}))^{k-r} \right), \tag{19}$$

with

$$\mathcal{D}_{i,j} = \{\sigma \in \mathcal{S}_n, \sigma(i) = \sigma(j)\}.$$

Conclusion. We are now able to have a partial conclusion.

(a) If the probability in Formula (19), which is $\mathbb{P}(D_n)$ is zero, then set on which we compute the estimated similarity is given by Formula (18) and the probability law of the estimated similarity is given by Formula (3).

(b) If the probability in Formula (19), which is $\mathbb{P}(D_n)$, is small enough, we approximate the probability law of the estimated similarity is given by Formula (3).

(c) If we go back to the *minhashing* functions and consider the functions Z_α as random permutations picked on the uniform sampling, the estimated similarity, on the base of Point (a) and (b), the exact mathematical expectation of any event $U_\ell(\alpha) = U_{\ell'}(\alpha)$ for any $1 \leq \alpha \leq k$. Why?

In this pure and uniform random scheme, all the lines of $\mathcal{Z}$ are disjoint and the realization of the event

$$t(B)(\ell, \ell') = \{\sigma \in \mathbb{S}_n, \min_{i \in I_2} \sigma(i) < \min_{i \in I_1 \cup I_3} \sigma(i)\},$$

is a pure matter of combinatorics. To realize this event, we have to choose of the $\{\sigma(i), i \in I_2\}$ to be the unity and we have $\#(I_2) = \#(S_\ell \cap S_{\ell'})$ ways to do it. So the probability of having this is

$$\frac{\#(S_\ell \cap S_{\ell'})}{n}.$$

At the arrival, for all α, $1 \leq \alpha \leq k$, the binary random variable $S(\alpha)$ that is equal one if $U_\ell(\alpha) = U_{\ell'}(\alpha)$ and zero otherwise, is a Bernoulli random variable with parameter $sim = \#(I_2)/n$, the number we can uniformly choose a permutation σ such that $1 \in \{\sigma(i),\ i \in I_2\}$. We have k independent Bernoulli random variables. Besides, the random variable

$$S_k = \sum_{1 \leq \alpha \leq k} S(\alpha)$$

is k times the estimated similarity *simrum*. We may then apply the probability formulas :

(1) Moments :

$$\mathbb{E}(simrum) = sim \quad and \quad \mathbb{V}ar(simrum) = sim(1 - sim)/k.$$

(2) Tchebychev Inequality.

$$\mathbb{P}(|simrum\ sim| > \lambda) \leq \frac{sim(1-sim)}{k\lambda^2}, \lambda > 0.$$

(3) Gaussian Approximation. If the similarity is non-zero, we have

$$\sqrt{\frac{k}{sim(1-sim)}}\Big(simrtm - sim\Big) \to \mathcal{N}(0,1);$$

which gives the approximated confidence interval of 95% percent

$$sim \in [simrum - 1.96\frac{\sqrt{simrum(1-simrum)}}{k},\ simrum + 1.96\frac{\sqrt{simrum(1-simrum)}}{k}]$$

Of course, we may have given more properties of the Binomial law and apply them to the similarity.
We achieved the result we were targeting. Nevertheless, we want to give preliminary results for the general.

D - General Case.

We suppose that the Z_α's are independent observations a the random variable defined on $\{1, \ldots, 2\}$ represented by $(Z(1), \ldots, Z(n))$ taking with values set $\mathcal{Z}$ which is finite and let z_0 its minimum member. We may use again the reasoning above. Given the event D_n - the lines $\mathcal{Z}^i$'s disjoint - the random variable $S(\alpha)$ is one if and only if

$$z_0 \in \{Z_\alpha(i),\ i \in I_2\}$$

Let us denote by $\mathbb{P}_{D_n}$ the conditional probability on D_n, that is $\mathbb{P}_{D_n}(B) = \mathbb{P}_{D_n}(B/D_n)$ for all $B \subset \{1, 2, \ldots, n\}$, and by $\mathbb{E}_{D_n}$ the mathematical expectation with respect to $\mathbb{P}_{D_n}$.

We then have that S_k follows a binomial law with parameter k and

$$p = \mathbb{P}_{D_n}(z_0 \in \{Z_\alpha(i),\ i \in I_2\})$$

In this case, $simrum \to p$ in probability as $k \to +\infty$, since

$$\begin{aligned}\mathbb{P}(|simrum - p| > \lambda) &= \mathbb{P}_{D_n}(|simrum - p| > \lambda\ /\ D_n)\mathbb{P}(D_n)\\ &= \mathbb{P}(D_n)\frac{p(1-p)}{k\lambda^2},\ \lambda.\end{aligned}$$

With such a lay out, we will be able to find out possible other choices of the Z_α's we have very quickly while $\mathbb{P}(D_n)$ close to one as near as possible. It is thought that the choice $Z_\alpha(i) = iX_\alpha + Y_\alpha \bmod n$ is justified. We will come back to this.

E - Interesting remark.

At the light of what precedes, we see that the RU algorithm might have been done with maximum of the lines at the place of the minimum. The probability parameter of the Bernoulli random variables $S(\alpha)$ would the number of ways to uniformly chose a permutation σ such that $n \in \{\sigma(i),\ i \in I_2\}$. From then, we proceed as above.

4. Conclusions and perspectives

At the end, we unveil the validity of the RU algorithm, justified the convergence of estimated similarity index, and completely described its probability law of this index in the pure and uniform scheme. But, we applied the method by using it on the set that is easily formed. The building of the modified set allows to gain a great amount. A formula describing the similarity indices obtained from the direct algorithm and the modified one allows to work with the second and, at the end of procedure, to find the first. Beyond the uniform scheme, we extended the method to general probability law and provided the limit of the estimated similarity. From there, we provided a way to have a reasonable estimation based on random variables that may be formed in short times, to the contrary to random permutations. The impact of using minhashing function should be evaluated in the frame developed here.

References

Bilenko M. and Mooney R.J., 2003. Adaptive Duplicate Detection Using Learnable String Similarity Measures. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD)*.39-48, Washington DC.

Broder, A., 1997. On the resemblance and containment of documents. In : *Compression and Complexity of Sequences, Proceedings*. 21-29.

Cha S.H., 2007. Comprehensive Survey on Distance Similarity Measures between Probability Density Functions. *International journal of mathematical models and methods in applied sciences*. **4**(1), 300-307.

de França, F.O., 2014. Iterative Universal Hash Function Generator for Minhashing. Arxiv: arXiv:1401.6124.

Dembele, S. and Lo, G.S., 2015. Probabilistic, statistical and algorithmic aspects of the similarity of texts and application to Gospels comparison. *Journal of Data Analysis and Information Processing*. Vol. **3**, 112-127. Doi : 10.4236/jdaip.2015.34012.

Formica, A., 2005. Ontology-based concept similarity in Formal Concept Analysis, *Information sciences*. 2624-2641.

Gionis, A., Indyk, P. and Motwani, R., 1999. Similarity Search in High Dimensions via Hashing. In: *Proceedings of the 25th VLDB Conference*. 518-529 , Eds: Edinburgh, Scotland.

Gower, J.C. and Legendre, P., 1986. Metric and Euclidean properties of dissimilarity coefficients. *Journal of Classification*. **3**, 5-48.

Guyon, I. and Elisseeff, A., 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research*. Vol. **3**, 1157-1182.

Guyon, I., 2006. *Feature extraction: foundations and applications*, Vol. **207**, Springer-Verlag Berlin Heidelberg.

Lawrence, N.D., 2008. *Tutorial, Dimensionality reduction the probabilistic way*, ICML Tutorial, University of Manchester, U.K.

Stein, B. and su Eissen, S.M., 2006. Near Similarity Search and Plagiarism Analysis. In: *Spiliopoulou et al. (Eds.): From Data and Information Analysis to Knowledge Engineering Selected Papers from the 29th Annual Conference of the German Classification Society (GfKl) Magdeburg*. 430-437, Springer.

Strehl, A., Ghosh, J. and Mooney, R., 2000. Impact of Similarity Measures on Web-page Clustering. *American Association for Artificial Intelligence*. 58-64.

Rajaraman and Ullman J.U., 2011. *Mining of Massive Datasets*. California.

Theobold M., Siddharth J. and Paepcke A., 2008. SpotSigs: robust and efficient near duplicate detection in large web collections. In: *31st Annual ACM SIGIR 08 Conference*, Singapore.

ZEzula, P., Dohnal, V. and Amato, G., 2006. *Similarity Search The Metric Space Approach*, Springer.