Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 151 (2019) 1170–1175

Procedia Computer Science

www.elsevier.com/locate/procedia

International Workshop on Web Search and Data Mining (WSDM)
April 29 – May 2, 2019, Leuven, Belgium

# A new semantic similarity approach for improving the results of an Arabic search engine

Amine EL HADI[a]*, Youness MADANI[b], Rachid EL AYACHI[a], Mohamed ERRITALI[a]

[a]*TIAD laboratory Departement of Computer Sciences Faculty of Sciences and Techniques Sultan Moulay Slimane University Beni Mellal, Morocco*

[b]*GI laboratory Departement of Computer Sciences Faculty of Sciences and Techniques Sultan Moulay Slimane University Beni Mellal, Morocco*

## Abstract

Determining semantic similarity between documents is crucial to many tasks such as plagiarism detection, automatic technical survey and semantic search. In this paper, we have mainly focused on detecting the semantic similarity between documents in large documents collection and queries based on an Arabic search engine, we investigated MapReduce as a specific framework for managing distributed processing in dataset pattern and semantic similarity measures of documents. Then we study the state of the art of different approaches for computing the similarity of documents. We propose an approach based on parallel algorithm of semantic similarity measures using MapReduce and WordNet after translation phase to detect the relevant documents in the face of the Arabic query. The numerical results obtained and presented showed the efficiency and the performance of the technique adopted.

"Keywords: Arabic search engine; Document similarity; Semantic measure; Big data; MapReaduce programming model"

* Corresponding author.
*E-mail address:* elhadi.amine@gmail.com

10.1016/j.procs.2019.04.167

## 1. Introduction

Text similarity measures play an increasingly important role in text related research and applications in tasks such as similarity recommendation [1], copy detection [2], social network mining [3] and others. Finding similarity between words is a fundamental part of text similarity, which is used as a primary stage for sentence, paragraph and document similarities.

Google approach is primary based on the matching process between the user's keywords and the texts under which it is indexed in terms of words [4]. Words are treated as a set of symbols, not words with meanings to human users. While plain linguistic measures, such as stemming and structured data search are used to enhance results, in nature Google and such search engines are still "Symbolic Computing" machines.

In third generation of search engines, Natural Language Processing technologies are applied in searching extensively, because in the first place, the search is seen as a language understanding process. This approach for search is paradigmatically different and a level higher in terms of the degrees of system difficulty and complexity. And it offers more accurate and consistent search results than the second generation search engines through its intelligence in language understanding. In this context many researchers have considered this aspect [5][6][7].

In this paper, we are interested in the phase of the documents indexation, each document represented by an intermediate representation. The Information Retrieval System (IRS) directly operates this representation. It describes the contents of the document by descriptors. These descriptors are significant units in the document. In our context, to find the relevant documents by comparison with an Arabic document query, the IRS compares the representation of this query to the representation of each document to get the most significant documents in our Arabic search engine.

The rest of the paper is organized as follows: Section 2 presents a state of the art of measuring document similarity; section 3 we describe our proposed approach for calculating the similarity semantic between documents in an Arabic search engine based on MapReduce model using Hadoop Framework. In section 4, we present and discuss our experiment results. Finally, in the section 5 we present the conclusion and the perspectives of this work.

## 2. Related Works

Many studies have been presented on measuring document similarity in recent years for facilitating the search for information in complex information systems such as a search engine.

Slimani [8] describes for us the existing semantic similarity methods based on structure, information content and feature approaches based on the quantifying similarity approaches, such as Path length based measure. This similarity measurement between concepts is based on the path distance separating the concepts, Thus similarity is computed by shortest path and the degree of similarity is determined based on path length [9].

Depth relative measure is a shortest path approaches, but it considers the depth of the edges connecting the two concepts in the overall structure of the ontology, The various depth relative measures are, Wu and Palmer measure [10] Leacock and Chodorow Similarity measure [11].

Information content-based measure: Both the path length and depth relative measure use the knowledge solely captured by ontology to computationally determine the similarity between concepts. In this section, the knowledge revealed by corpus is used to augment the information already present in the ontologies or taxonomy. The various information content based measures are Resnik Measure [12] Lin Measure [13] Jiang and Conrath measure [14].

Matveeva [15] presented a Vector Space Model (VSM) algorithm to compute the similarity using Cosine measurement of the vector.

Zhang et al. [16] presented a sequence based method to detect partial similarity of web page using MapReduce, which consisted of two sub-tasks as sentence level near duplicate detection and sequence matching [17], this approach is considerate as a parallel-based method which it is focused on the MapReduce model.

## 3. Our Proposed MapReduce Algorithm

In the following, we propose our new MapReduce algorithm for computing the similarity measure of documents and queries. The first step is to get the query from our web search engine and apply the Arabic stemmer [18] on each word of it, then translate each steam from Arabic to English and stock the data in a new file to work on it. The same process will applied on each document of our corpus; the reason of this treatment is to compare the steaming of each word of a query Q with all the words in each document D from the corpus to get the similarity semantic measure between Q and D with our approach. The figure 1 present all the process of our application to get similarity semantic score between query and document.

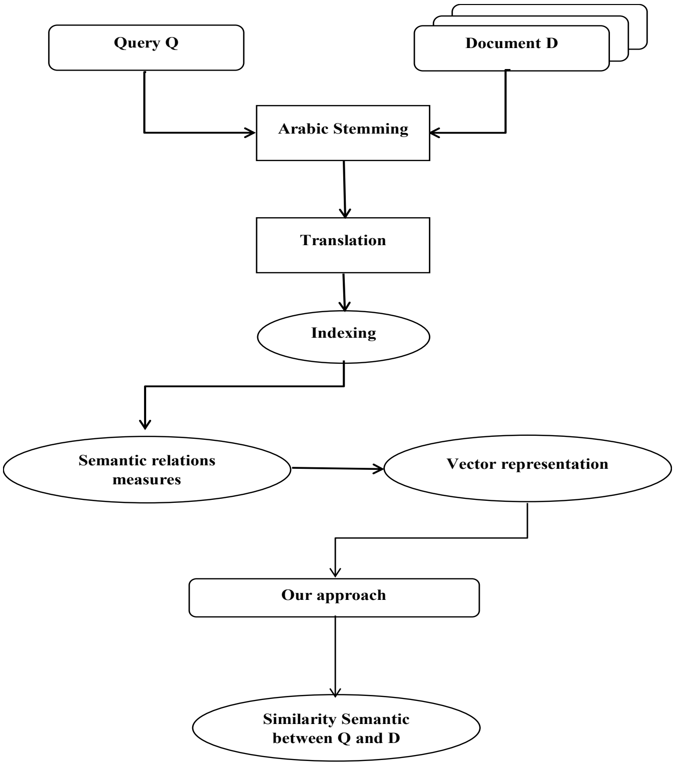

Figure 1 : The process of our methodology to compute the semantic similarity measure

### 3.1. Arabic Stemmer

By its morphological and syntactic richness, the Arabic language is considered among the most difficult languages to deal with it in the field of information search, in Arabic language we know that a word doesn't lose its sense when we convert it to its stemming, for that, we think about stemmer to get the best result in our work.

### 3.2. Translation

In this work we have a translation phase from Arabic to English to use Wordnet as an external network semantic resource, we have used the Yandex translation API, to translate any document word by word for not losing any word's sense, and get the correct translation.

### *3.3. Proposed approach*

After getting the steam of each word in the corpus and translate it from Arabic to English, we will apply our proposed approach based on a MapReduce model. The Map phase is for each term of document, the mapper emits the document ID as the key, and the steam of his words as the value. The second phase, named Reduce, will take the output of the Map function, and will computes the similarity relation between each collection of values of each document and the query. This similarity measure computed by our algorithm with the use of the weight of the words and Learock and Chodorow approach [11].

To compute the similarity between documents and query we apply the following algorithm:

```
Class Mapper
    Method Map(Corpus)
        For each document ∈ (Corpus)
            TP ← Text_Preprocessing(document)
        End for
        For each term ∈ TP
            Write(Docid,term)
        End for
Class Reducer
    Method Reduce(Docid, List(term))
        Vector(q) ← indexing(Query)
        Vector(d) ← indexing(List(term))
        C ← Cosine(Vector(q), Vector(d))
        LC ← 0
        Sim← 0
        For each n ∈ List(term)
            For each e ∈ List(Query)
                LC ← LC + SimLC(n,e)
            End for
        End for
        Sim ← $\frac{1}{2} \times (Cosine + LC)$
        Write(Docid, S)
```

Where:

- Text_Preprocessing(document): Applies the different text preprocessing methods on each document of our corpus.
- Indexing (Query): is a function that consists in indexing the query of our system for constructing the vector of the query.
- Cosine(Vector(q), Vector(d)) : is a function to calculate the cosine similarity between the query and each document f the corpus.
- SimLC(n,e) : calcul the semantic similarity between a word of the query and a word of a document, using the leackock and chadorow approach.

Our similarity measure presented by the following formula:

$$Sim(q,d) = \frac{1}{2} \times \left(\frac{\sum_{i=1}^{n} q \times d}{\sqrt{(\sum_{i=1}^{n} q^2)} \times \sqrt{(\sum_{i=1}^{n} d^2)}} + SimLC(q,d)\right) \quad (1)$$

**Cosine Measure:**

It uses the complete vector representation, that is to say the objects frequency (words). Two documents are similar if their vectors are combined. If two objects are not similar, their vectors form an angle (X, Y) whose Cosine represents the similarity value. The formula is defined by the ratio of the scalar product of vectors x and y and the product of the norm of x and y.

$$Sim_{Cosine}(\text{X,Y}) = \frac{\sum_{i=1}^{n} x \times y}{\sqrt{(\sum_{i=1}^{n} x^2)} \times \sqrt{(\sum_{i=1}^{n} y^2)}} \quad (2)$$

The measurement of Cosine quantifies the similarity between the two vectors as the cosine of the angle between two vectors.

**Wu and Palmer Measure:**

The principle of this measurement is given an ontology formed by a set of nodes and a root node (R). X and Y represent two ontology elements for which we will compute the similarity. The principle of similarity measurement is based on the distances (N1 and N2) which separate the X and Y nodes from the node R and the distance (N) which separates the Subsuming Concept (SC).

The Wu and Palmer measurement [10] is defined by this formula:

$$Sim(X,Y)_{Wu\ and\ Palmer} = \frac{2 \times N}{N1 + N2} \quad (3)$$

**Learock and Chodorow Measure:**

Another method presented by [19], which combines between counting of the arcs method and the informational contents method. The proposed measure by Leacock and Chodorowis based over the shortest way length between two synsets of Wordnet. This technique [11] is defined by the formula:

$$Sim_{lc}(X,Y) = -\log\left(\frac{cd(X,Y)}{2 \times M}\right) \quad (4)$$

M is the longest way length, which separates the concept root, of ontology, of the concept more in bottom. We indicate that $cd(X,Y)$ is the shortest way length, which separate X of Y.

## 4. Experimentation and Discussion of Our MapReduce Algorithm

To evaluate our approach based on the MapReduce programming model and using the HDFS file system, we conduct extensive performance study, the experiment is to run our MapReduce algorithm with a variable number of documents in input (in the corpus stored in HDFS). We compute the semantic similarity measure between the same Arabic query presented in document and all the documents in the corpus. The results show that the semantic similarity changes with the change of each approach. Our approach will be based on Leacock and Chodorow approach and cosine approach.

In the following, we present the practical results of our approach compared with existing approaches such as Wu and Palmer and Leacock and Chodorow.

Table 1 shows the results of the similarity between query and documents in our corpus using our approach, the approach of Wu and Palmer and the approach of Leacock and Chodorow.

Table 1. Similarity of our Approach Compared with WP and LC.

| | Precision | Recall | F1 |
|---|---|---|---|
| Wu and Palmer (WP) | 40% | 33.34% | 36.75% |
| Learock and Chodorow (LC) | 40% | 50% | 44.45% |
| Our approach | 60% | 50% | 54.54% |

From this table we see that our approach gives us a great similarity than the other two approaches Wu and Palmer or Learock and Chodorow. We can conclude from these practical results that our approach is effective to an information retrieval system because it can easily give us semantically similar documents for an Arabic query.

## 5. Conclusion and Future Research

This paper discussed a new approach for semantic similarity measure based on MapReduce framework to compute the similarity between an Arabic query and the large corpus of documents existing in HDFS in an Arabic search engine and find the most pertinent documents. The results conclude that our MapReduce algorithm gives better results than the existing semantic similarity measure approaches. The future research involves integrating the proposed approach in a multilingual search engine using Hadoop multi-nodes to improve the running time of the search engine.

## References

[1] Khadija A. Almohsen, Huda Al-Jobori, Recommender Systems in Light of Big Data, International Journal of Electrical and Computer Engineering (IJECE), Vol. 5, No. 6, pp. 1553-1563, December (2015).

[2] Hoad T. C. and Zobel. J., Methods for Identifying Versioned and Plagiarized Documents, JASIST,vol. 54,pp. 203- 215, (2003).

[3] Spertus E.,Sahami M. and Buyukkokten O., Evaluating Similarity Measures: A Large-scale Study in the Orkut Social Network, proceedings of KDD, (2005).

[4] Brin S. and Page L. : The anatomy of a large hypertextual web search engine. Web publication, Stanford University (1998)

[5] Abuleil S., Alsamra K. : New Technique to Support Arabic Noun Morphology: Arabic Noun Classifier System (ANCS). Internatinal Journal of Computer Processing of Oriental Languages. Vol. 17 Issue 2. (2004)

[6] Aitao Chen and Fredric Gey : Building an Arabic stemmer for information retrieval. The Eleventh Text Retrieval Conference (TREC 2002), National Institute of Standards and Technology (NIST). (2002)

[7] Aljlayl M. : Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach. ACM Eleventh Conference on Information and Knowledge Management. (2002)

[8] Thabet Silmani. Description and evaluation of semantic similarity measures approaches. International Journal of Computer Applications(0975-8887). Volume 80- No.10, October (2013).

[9] Wei, T. T., Lu, Y. H., Chang, H. Y., Zhou, Q., & Bao, X. Y. A semantic approach for text clustering using WordNet and lexical chains. Expert Systems with Applications, 42(4), 2264–2275. (2015).

[10] Z. Wu and M. Palmer. Verb Semantics and Lexical Selection. In Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics (ACL'94), pages 133{138, Las Cruces, New Mexico. (1994).

[11] Leacock, C and Chodorow, M. Filling in a sparse training space for word sense identification. ms. March. (1994).

[12] Resnik O. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity and Natural Language. Journal of Artificial Intelligence Research, 11, 95-130. (1999).

[13] Lin, D. Principle-Based Parsing Without Overgeneration. In Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL'93), pages 112-120, Columbus, Ohio. (1993).

[14] Jiang, J.J. and Conrath. D.W. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In Proceedings of the International Conference on Research in Computational Linguistic, Taiwan. (1998).

[15] Matveeva I., Document Representation and Multilevel Measures of Document Similarity. Proceedings of ACLHLT. (2006).

[16] Qi Zhang,Yue Zhang,Hao. Yu and Xuan. Huang, Efficient PartialDuplicate Detection Based on Sequence Matching. Proc. of SIGIR. (2010).

[17] Qinsheng D., Wei Liu, Li G. and Yonglin T. Near duplicate detection using MapReduce. College of Computer Science and Technology, 2nd International Conference on Computer Science and Network Technology, pp 243-246, (2012).

[18] Madani, Y., Erritali, M., Bengourram, J.: Arabic stemmer based big data. J. Electron. Commer. Organ. JECO 16(1), 17–28. (2018).