AN EFFICIENT APPROACH FOR SEMANTICALLY-ENHANCED DOCUMENT CLUSTERING BY USING WIKIPEDIA LINK STRUCTURE

Iyad AlAgha[1] and Rami Nafee[2]

[1]Faculty of Information Technology, The Islamic University of Gaza, Palestine
[2]Faculty of Information Technology, The Islamic University of Gaza, Palestine

ABSTRACT

Traditional techniques of document clustering do not consider the semantic relationships between words when assigning documents to clusters. For instance, if two documents talking about the same topic do that using different words (which may be synonyms or semantically associated), these techniques may assign documents to different clusters. Previous research has approached this problem by enriching the document representation with the background knowledge in an ontology. This paper presents a new approach to enhance document clustering by exploiting the semantic knowledge contained in Wikipedia. We first map terms within documents to their corresponding Wikipedia concepts. Then, similarity between each pair of terms is calculated by using the Wikipedia's link structure. The document's vector representation is then adjusted so that terms that are semantically related gain more weight. Our approach differs from related efforts in two aspects: first, unlink others who built their own methods of measuring similarity through the Wikipedia categories; our approach uses a similarity measure that is modelled after the Normalized Google Distance which is a well-known and low-cost method of measuring term similarity. Second, it is more time efficient as it applies an algorithm for phrase extraction from documents prior to matching terms with Wikipedia. Our approach was evaluated by being compared with different methods from the state of the art on two different datasets. Empirical results showed that our approach improved the clustering results as compared to other approaches.

KEYWORDS

Document Clustering, Semantic Similarity, Ontology, Wikipedia.

1. INTRODUCTION

Traditional techniques of document clustering are often based on word co-occurrence while it ignores the semantic relationships between words[1,2]. Thus, if two documents use different collections of words to represent the same topic, the clustering approach will consider these documents different and will assign them to different clusters despite the semantic similarity between the core words.

Exiting research on document clustering has approached the lack of semantics by integrating ontologies as background knowledge [3,4]. An Ontology is a hierarchy of domain terms related via specific relations. Ontologies can improve document clustering by identifying words that are probably synonyms or semantically related even though they are syntactically different[5]. Comparing terms in documents using ontology relies on the fact that their corresponding concepts within the ontology may have properties in the form of attributes, level of generality or specificity, and their relationships.

WordNet is an example of ontologies that is widely used as background knowledge for document clustering. Many efforts have explored the use of WordNet to incorporate semantics into the bag of words (BOW)[6]. This can be done by matching and replacing words in the document with the most appropriate terms from WordNet. Then, the attributes of and relationships between these terms can be exploited to enhance clustering. However, WordNet-based similarity has not proven to enhance clustering as first expected. While some works showed that WordNet has the potential to improve clustering results[5,7,8], other works have reported that the ontological concepts add no value and sometimes impair performance of document clustering[9]. Besides, some researchers indicate that WordNet does not provide good word similarity data and that its structure does not fit well for this task[10].

Some clustering techniques have used domain ontologies such as Mesh Ontology to cluster documents related to specific domains of knowledge[3]. However, these ontologies have limited coverage and it is unlikely to cover all concepts mentioned in the document collections, especially when documents are from general domain.

Rather than relying on domain specific and limited coverage ontologies, some approaches have used Wikipedia concepts and category information to enrich document representation and handle the semantic relationships between document terms[11-13]. Wikipedia is much more comprehensive than other ontologies since it capture a wide range of domains, is frequently updated and well structured. Wikipedia can be seen as an ontology where each article represents a single ontology concept, and all concepts are linked together by hyperlinks. In addition, Wikipedia has a hierarchal categorization that resembles the structure of an ontology whereas each article belongs to one or more information categories.

In this paper, we propose an approach to improve document clustering by explicitly incorporating the semantic similarity between Wikipedia concepts into the document's vector space model. Our approach is distinguished over similar approaches in terms of the way we used to efficiently map the document content to Wikipedia concepts and the low-cost measure we adapted to determine semantic similarity between terms. In the following section, we discuss similar efforts that also exploited knowledge from Wikipedia to enhance document clustering, and compare their approaches with ours.

2. RELATED WORK

Techniques that employ Ontological features for clustering try to integrate the ontological background knowledge into the clustering algorithm. Ontology based similarity measures are often used in these techniques to calculate the semantic similarity between document terms. There is a plenty of Ontology-based similarity measures that exploit different ontological features, such as distance, information content and shared features, in order to quantify the mutual information between terms (reader is referred to [3] for a review and comparison of ontology based similarity measures). Distance between two document vectors is then computed based on the semantic similarity of their terms.

A number of research efforts explored the use of Wikipedia to enhance text mining tasks, including document clustering[11,14,15], text classification [16] and information retrieval[17]. Few approaches have explored utilizing Wikipedia as a knowledge base for document clustering. Gabrilovich et al. [18] proposed and evaluated a method that is based on matching documents with the most relevant articles of Wikipeda, and then augmenting the document's BOW with the semantic features. Spanakis et al. [19] proposed a method for conceptual hieratical clustering that exploits Wikipedia textual content and link structure to create compact document representation.

However, these efforts do not make use of the structural relations in Wikipeida. As a result, the semantic relatedness between words that are not synonyms is not considered when computing the similarity between documents.

Huang et al. [13] proposed an approach that maps terms within documents to Wikipedia's anchors vocabulary. Then they incorporated the semantic relatedness between concepts by using Milne and Witten measure [20] which takes into account all of the Wikipedi's hyperlinks. Our work is similar in that it also uses a similarity measure that is based on the Wikipedia's hyperlinks. However, their approach did not tackle the issue of frequent itemsets, and they instead used a less efficient approach by examining all possible n-grams. Another difference is the way the document similarity is measured: while they augmented the measure of document similarity, our approach augments the document's vector by reweighting the tf-idf score of each word according to its relatedness to other document's words. This makes our approach independent of, and can be used with, any measure of document similarity since the reweighting process is carried out before computing similarity between document pairs.

Another work that can be compared to ours is presented by Hu et al.[11]. They developed two approaches: exact match and relatedness-match, to map documents to Wikipedia concepts, and further to Wikipedia categories. Then documents are clustered based on a similarity metric which combines document content information, concept information as well as category information. However, their approach requires pre-processing of the whole Wikipedia's textual content, a thing that leads to substantial increase in both runtime and memory requirements. Instead, our approach does not require any access to the Wikipedia's textual content, and relies only on the Wikipedia's link structure to compute similarity between terms.

Hu et al. [12] proposed a method that mines synonym, hypernym and associative relations from Wikipedia, and append that to traditional text similarity measure to facilitate document clustering. However, their method was developed specifically for the task and has not been investigated independently. They also built their own method of measuring similarity through Wikipedia's category links and redirects. We instead used a similarity measure that is modeled after the Normalized Google Distance [21] which is a well-known and low-cost method of measuring similarity between terms based on the link structure of the Web.

Wikipedia has been employed in some efforts for short text classification. For example, Hu et al. [22] proposed an approach that generates queries from short text and use them to retrieve accurate Wikipedia pages with the help of a search engine. Titles and links from the retrieved pages are then extracted to serve as additional features for clustering. Phan et al. [23] presented a framework for building classifiers that deal with short text. They sought to expand the coverage of classifiers by topics coming for external knowledge base (e.g. Wikipedia) that do not exist in small training datasets. These approaches, however, use Wikipedia concepts without considering the hierarchical relationships and categories embedded in Wikipedia.

3. AN APPROACH FOR WIKIPEDIA-BASED DOCUMENT CLUSTERING

The pseudo code of our approach for Wikipedia-based document clustering is shown in Figure 1, and consists of three phases: The first phase includes of a set of text processing steps for the purpose of determining terms that best represent the document content. In the second phase, each document is represented by using the tf-idf weighted vector. Document terms are then mapped to Wikipedia concepts. In the third phase, the similarity between each pair of Wikipedia concepts is measured by using the Wikipedia link structure. The tf-idf weights of original terms are then reweighted to incorporate the similarity scores obtained from Wikipedia. By the end of the algorithm, the tf-idf representation of each document is enriched so that terms that are

semantically related gain more weight. Documents can then be clustered using any traditional clustering method such as k-means. These phases are explained in detail in the subsequent sections.

Prior to applying our approach, Wikipedia's vocabulary of anchor text is retrieved from the Wikipedia dump, which is a copy of all Wikipedia content, and stemmed in order to be comparable with the stemmed document content. For measuring similarity between Wikipedia concepts, all outgoing hyperlinks, incoming hyperlinks and categories of articles are also retrieved. Note that this task incurs a one-time cost, thus allowing the clustering algorithm to be invoked multiple times without the additional overhead of reprocessing the Wikipedia content.

Input: A set of documents $D = \{d_1, .., d_n\}$
Begin
{**Phase 1**: Pre-processing and extraction of frequent phrases}
for each document $d \in D$ do
  Apply stemming and stopword removal
end for
Concatenate all documents
Apply Apriori algorithm to extract frequent itemsets

{**Phase 2**: Construct tf-idf weighted vectors and map terms to Wikipedia concepts}
for each document $d \in D$ do
  Tokenize d
  Discard tokens that overlap with frequent phrases
  Discard rare terms
  Build the BOW of d where BOW = Retained tokens ∪ frequent phrases
  for each term $t \in BOW$ do
    Calculate *tf-idf* for t
    if t matches Wikipedia concept(s) then
      Replace t with matching Wikipedia concept(s)
    end if
  end for
end for

{Phase 3: {Reweighting *tf-idf* weights}
for each document $d \in D$ do
  for each term $t_i \in$ BOW of d do
    for each term $t_j \in$ BOW of d AND $t_i \neq t_j$ do
      Compute similarity between $t_i and t_j$ using equation 1
      if $t_i or t_j$ are ambiguous then
        Perform word-sense disambiguation (see section 6)
      end if
    end for
    Reweight $tdidf(d, t_i)$ using equation 2
  end for
end for

{Document clustering}
Apply any conventional clustering algorithm (e.g. k-means)
End

Figure 1. Pseudo code of our algorithm of document clustering

4. CONSTRUCTION OF DOCUMENT'S VECTOR SPACE MODEL

The first step of our clustering approach is to represent each document as a bag of words (BOW). Note that traditional clustering algorithms treat a document as a set of single words, thus losing valuable information about the meaning of terms. When incorporating semantics in document clustering, it is necessary to preserve phrases, the consecutive words that stand together as a conceptual unit. Without preserving phrases, actual meanings of document terms may be lost, making it difficult to measure semantic similarity in between. For example, the phrase "big bang theory" refers to a concept that is entirely different from what its individual tokens refer to. Thus, we aim to create a document's BOW representation whose attributes include not only single words but also phrases that have standalone meanings. This phase starts with some standard text processing operations including stopword removal and word stemming. Stopwords are words that occur frequently in documents and have little informational meanings. Stemming finds the root form of a word by removing its suffix. In the context of mapping with Wikipedia concepts, stemming allows to recognize and deal with variations of the same word as if they were the same, hence detecting mappings between words with the same stem.

To construct the document's bag of words and phrases, we used a simple method based on Apriori algorithm [24] to find frequent occurring phrases from a document collection. A phrase is defined as frequent if it appears in n number of documents (For our task we set n = 3). The Apriori algorithm consists of two steps: In the first step, it extracts frequent itemsets, or phrases, from a set of transactions that satisfy a user-specified minimum support. In the second step, it generates rules from the discovered frequent itemsets. For this task, we only need the first step, i.e., finding frequent itemsets. In addition, the algorithm was restricted to find itemsets with four words or fewer as we believe that most Wikipedia concepts contain no more than four words (this restriction can be easily relaxed).

After extracting frequent itemsets, we perform word tokenization to break the document text into single words. Many of the resulting tokens can be already part of the extracted itemsets. Therefore, we remove tokens that overlap with any of the retrieved frequent itemsets. Stemmed tokens as well as frequent itemsets that occur in the document will be combined together to form the BOW representing the document. Rare terms that infrequently occur in the document collection can introduce noise and degrade performance. Thus, terms that occur in the document collection less than or equal to a predefined threshold are discarded from the document's BOW.
It is worth noting here that similar works that exploited Wikipedia for document clustering often did not consider mining frequent itemsets occurring in the document [11 , 13]. Instead, they extract all possible n-grams from the document by using a sliding widow approach and match them with the Wikipedia content. In contrast, our approach of extracting frequent itemset prior to the concept-mapping process is more time-efficient as it avoids the bottleneck of matching all possible n-grams to Wikipedia concepts.

After identifying the document's BOW, the next step is to map terms within the BOW to Wikipedia concepts: Each term is compared with Wikipedia anchors, and matching terms are replaced by the corresponding Wikipedia concepts. Terms that do not match any Wikipedia concept are not discarded from the BOW in order to avoid any noise or information loss.

Formally, let $D = \{d_1, .., d_n\}$ be a set of documents and $T = \{t_1, .., t_m\}$ be the set of different terms occurring in a document . Note that T includes both: 1) Wikipedia concepts which replace original terms after the mapping process. 2) Terms that do not match any Wikipedia concepts. The weight of each document term is then calculated using tf-idf (term frequency-inverted document frequency). Tf-idf weighs the frequency of a term in a document with a factor that

discounts its importance when it appears in almost all documents. The tf-idf of term t in document d is calculated using the following equation:

$$\mathrm{tfidf}(d,t) = \log(\mathrm{tf}(d,t)+1) * \log\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right)$$

The document's vector representation $\overrightarrow{t_d}$ is then constructed from the tf-idf weights of its terms:

$$\overrightarrow{t_d} = \left(\mathrm{tfidf}(d,t_1), \ldots \mathrm{tfidf}(d,t_m)\right)$$

5. MEASURING SEMANTIC SIMILARITY BETWEEN WIKIPEDIA TERMS

After representing the document as a vector of term tf-idf weights, the next step is to augment these weights so that terms gain more importance according to their semantic similarity to the other document terms.

To measure similarity between document terms, we used a measure that is based on the Normalized Google Distance Measure (NGD) [21]. The NGD measure first uses the Google search engine to obtain all Web pages mentioning these terms. Pages that mention both terms indicate relatedness, while pages that mention only one term indicate the opposite. The NGD denotes the distance or dissimilarity between two terms: the smaller the value of NGD, the more related the terms are semantically. For this work, the measure is adapted to exploit Wikipedia articles instead of the Google's search results. Formally, the Wikipedia-based similarity measure is:

$$sim(s,t) = 1 - \frac{\max\{\log(S),\log(T)\} - \log|S \cap T|}{\log(R) - \min\{\log(S),\log(T)\}} \quad (1)$$

where s and t are a pair of Wikipedia concepts. S and T are the sets of all Wikipedia articles that link to s and t respectively, and R is set of all Wikipedia concepts. The output of this measure ranges between 0 and 1, where values close to 1 denote related terms while values close to 0 denote the opposite. Note that the advantage of this measure is its low computational cost since it only considers the links between Wikipedia articles to define similarity.

After computing the similarity between each pair of terms, the tf-idf weight of each term is adjusted to consider its relatedness to other terms within the document's vector representation. The adjusted weight w of a term t is calculated using the following equation:

$$w(d,t_i) = tdidf(d,t_i) + \sum_{\substack{j=0, j \neq i \\ sim(t_i,t_j) \geq threshold}}^{N} tdidf(d,t_j) * sim(t_i,t_j) \quad (2)$$

where $sim(t_i,t_j)$ is the semantic similarity between the terms t_i and t_j, and is calculated using Equation 1. N is the number of co-occurred terms in document d. The threshold denotes the minimum similarity score between two terms. Since we are interested in emphasizing more weight on terms that are more semantically related, it is necessary to set up a threshold value.
Note that this measure assigns an additional weight to the original term's weight based on its similarity to other terms in the document. The term weight remains unchanged if it is not related to any other term in the document or if it is not mapped to any Wikipedia concept. The final document's vector $\overrightarrow{t_d}$ is:

$$\overrightarrow{t_d} = \left(w(d,t_1), \ldots w(d,t_m)\right)$$

After constructing the semantically-augmented vectors for all documents, any conventional measure of document similarity, such as the cosine measure, can be used to measure similarity between document pairs. Note that in our approach we incorporate the similarity scores in the document representation before applying the document similarity measure. Thus, our approach is independent of, and hence can be used with, any similarity measure or clustering algorithm.

6. WORD SENSE DISAMBIGUATION

Concepts mentioned in Wikipedia are explicitly linked to their corresponding articles through anchors. These anchors can be considered as sense annotations for Wikipedia concepts. Ambiguous words such as “eclipse” are linked to different Wikipedia articles based on their meanings in the context where they occur (e.g. eclipse "astronomical event", eclipse "software suite", eclipse "foundation"). When mapping document terms to Wikipedia concepts, it is necessary to perform word sense disambiguation to identify the correct word sense. Failing to do so may result in false results when measuring similarity between terms.

One way to disambiguate words is to simply use the most common sense. The commonness of a sense is identified by the number of anchors that link to it in Wikipedia. For example, over 95% of anchors labelled as “Paris” link to the capital of France while the rest link to other places, people or even music. However, choosing the most common sense is not enough and it is not always the best decision. Instead, we used the same approach used in [21] which uses the two terms involved in the similarity measure to disambiguate each other. This is done by selecting the two candidate senses that most closely related to each other. We start by choosing the top common senses for each term (For simplicity, we ignore senses that contribute with less than 1% of the anchor’s links). We then measure the similarity between every pair of senses, and the two senses with the highest similarity score are considered.

7. EVALUATION

Wikipedia releases its database dumps periodically, which can be downloaded from http://download.wikipedia.org. The Wikipedia dump used in this evaluation was released on the 13th August 2014, and contains 12100939 articles. The data was presented in XML format. We used the WikipediaMiner [25] toolkit to process the data and extract the categories and outlinks out of Wikipedia dump.

7.1. Methodology

Our objective was to compare our approach with other approaches from the state of the art. Therefore, we used the same evaluation settings used by Hu et al. [12] in order to make our results comparable with theirs. The following two test sets were created:

- Reuters-21578 contains short news articles. The subset created consists of categories in the original Reuters dataset that have at least 20 and at most 200 documents. This results in 1658 documents and 30 categories in total.
- OHSUMed contains 23 categories and 18302 documents. Each document is the concatenation of title and abstract of a medical science paper.

Besides our method, we implemented and tested three different text representation methods, as defined below:

- Bag of Words: The traditional BOW method with no semantics. This is the baseline case.

- Hotho et al.'s method: this is a reimplementation of Hotho et al.'s WordNet-based algorithm[8]. The intention of considering this method is to compare how the use of Wikipedia as background knowledge influences the clustering results as compared to WordNet.
- Hu et al.: this is a reimplementation of the Hu et al.'s algorithm [12] which leverages Wikipedia as a background knowledge for document clustering. While Hu et al. built their own measure of document similarity based on Wikipedia's links, our approach uses a similarity measure that is based on the Normalized Google Distance.

To focus our investigation on the representation rather than the clustering method, the standard k-means clustering algorithm was used. We used two evaluation metrics: Purity and F-score. Purity assumes that all samples of a cluster are predicted to be members of the actual dominant class for that cluster. F-score combines the information of precision and recall which is extensively applied in information retrieval.

7.2. Results

Table 1 shows how the different methods perform in clustering on the two datasets. In general, the performance of BOW on both datasets is improved by incorporating background knowledge either from WordNet (Hotho et al.'s method) or Wikipedia (Hu et al. and our method). For instance, according to the F-score, for the Reuters dataset, our method and Hotho et al's method achieve 27.23% and 14.71% respectively.

On comparing the use of Wikipedia to WordNet, our approach and Hu et al.'s, which both use Wikipedia to enrich document similarity, outperformed the Hotho et al.'s approach for both datasets. This demonstrates the potential of integrating Wikipedia as a knowledge source as compared to the WordNet based method.

Comparing our approach with the other three methods, our approach achieves the best F-score and purity on both datasets. We applied t-test to compare between the performance of our approach and the others. Results show that our approach significantly outperformed all other methods on the Reuters dataset with the p-value < 0.05. When using the OHSUMed dataset, the difference between our approach and the BOW and Hotho et al. was also significant. However, there was no significant difference between the Hu et al. approach and ours ($p < 0.32$). In general, our approach provides slight improvement over the Hu et al.'s method. As both approaches exploit the Wikipedia link structure to enrich the document representation, we believe that the slight improvement in our approach stems from the adopted similarity measure that is based on the well-known Normalized Google Distance.

Table 1. Comparison with related work in terms of purity and F-score

	Reuters 21578		**OHSUMed**	
	Purity (Impr.)	**F-score (Impr.)**	**Purity (Impr.)**	**F-score (Impr.)**
Bag of Words	0.571	0.639	0.36	0.470
Hotho et al.	0.594 (4.02%)	0.685 (7.20%)	0.405 (12.5%)	0.521 (10.85%)
Hu et al.	0.655 (8.40%)	0.733 (14.71%)	0.475 (31.94%)	0.566 (20.43%)
Our Approach	0.703 (23.11%)	0.813 (27.23%)	0.483 (34.16%)	0.592 (25.95%)

8. CONCLUSION AND FUTURE WORK

In this work, we proposed an approach for leveraging Wikipedia link structure to improve text clustering performance. Our approach uses a phrase extraction technique to efficiently map document terms to Wikipedia concepts. Afterwards, the semantic similarity between Wikipedia terms is measured by using a measure that is based on the Normalized Google Distance and the Wikipedia's link structure. The document representation is then adjusted so that each term is assigned an additional weight based on its similarity to other terms in the document. Our approach differs from similar efforts from the state of the art in two aspects: first, unlink other works that built their own methods of measuring similarity through the Wikipedia's category links and redirects, we instead used a similarity measure that is modelled after the Normalized Google Distance which is a well-known and low-cost method of measuring similarity between terms. Second, while other approaches used to match all possible n-grams to Wikipedia concepts, our approach is more time efficient as it applies an algorithm for phrase extraction from documents prior to matching terms with Wikipedia. In addition, our approach does not require any access to the Wikipedia's textual content, and relies only on the Wikipedia's link structure to compute similarity between terms. The proposed approach was evaluated by being compared with different methods from related work (e.g. Bag of Words with no semantics, clustering with WordNet as well as clustering with Wikipedia) on two datasets: Reuters 21578 and OHSUMed.
We think that our approach can be extended to other applications that are based on the measurement of document similarity, such as information retrieval and short text classification. In our future work, we aim to further improve our concept-mapping technique: Currently, only the document terms that exactly match Wikipedia concepts are extracted and used for the similarity measure. Instead of exact matching, we aim to utilize the graph of Wikipedia links to build the connection between Wikipedia concepts and the document content even if they cannot exactly match. This approach can be more useful when Wikipedia concepts cannot fully cover the document content.

REFERENCES

[1] Hotho, Andreas, Maedche, Alexander & Staab, Steffen, (2002) "Ontology-Based Text Document Clustering", KI, Vol. 16, No. 4: pp. 48-54.

[2] Charola, Apeksha & Machchhar, Sahista, (2013) "Comparative Study on Ontology Based Text Documents Clustering Techniques", Data Mining and Knowledge Engineering, Vol. 5, No. 12: pp. 426.

[3] Zhang, Xiaodan, Jing, Liping, Hu, Xiaohua, Ng, Michael & Zhou, Xiaohua, (2007) "A Comparative Study of Ontology Based Term Similarity Measures on Pubmed Document Clustering", Advances in Databases: Concepts, Systems and ApplicationsSpringer. pp. 115-126.

[4] Logeswari, S & Premalatha, K, (2013) "Biomedical Document Clustering Using Ontology Based Concept Weight", Computer Communication and Informatics (ICCCI), 2013 International Conference on: IEEE. pp. 1-4.

[5] Hotho, Andreas, Staab, Steffen & Stumme, Gerd, (2003) "Ontologies Improve Text Document Clustering", Data Mining, 2003. ICDM 2003. Third IEEE International Conference on: IEEE. pp. 541-544.

[6] Wei, Tingting, Zhou, Qiang, Chang, Huiyou, Lu, Yonghe & Bao, Xianyu, (2014) "A Semantic Approach for Text Clustering Using Wordnet and Lexical Chains", Expert Systems with Applications, Vol. 42, No. 4,

[7] Wang, Pu, Hu, Jian, Zeng, Hua-Jun, Chen, Lijun & Chen, Zheng, (2007) "Improving Text Classification by Using Encyclopedia Knowledge", Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on: IEEE. pp. 332-341.

[8] Hotho, Andreas, Staab, Steffen & Stumme, Gerd, (2003) "Wordnet Improves Text Document Clustering", Proceedings of SIGIR Semantic Web Workshop: ACM. pp. 541-544.

[9] Moravec, Pavel, Kolovrat, Michal & Snasel, Vaclav, (2004) "Lsi Vs. Wordnet Ontology in Dimension Reduction for Information Retrieval", Dateso. pp. 18-26.

[10] Passos, Alexandre & Wainer, Jacques, (2009) "Wordnet-Based Metrics Do Not Seem to Help Document Clustering", Proc. of the of the II Workshop on Web and Text Intelligente, S¦£o Carlos, Brazil. pp.
[11] Hu, Xiaohua, Zhang, Xiaodan, Lu, Caimei, Park, Eun K & Zhou, Xiaohua, (2009) "Exploiting Wikipedia as External Knowledge for Document Clustering", Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining: ACM. pp. 389-396.
[12] Hu, Jian, Fang, Lujun, Cao, Yang, Zeng, Hua-Jun, Li, Hua, Yang, Qiang & Chen, Zheng, (2008) "Enhancing Text Clustering by Leveraging Wikipedia Semantics", Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval: ACM. pp. 179-186.
[13] Huang, Anna, Milne, David, Frank, Eibe & Witten, Ian H, (2009) "Clustering Documents Using a Wikipedia-Based Concept Representation", Advances in Knowledge Discovery and Data MiningSpringer. pp. 628-636.
[14] Kumar, Kiran, Santosh, GSK & Varma, Vasudeva, (2011) "Multilingual Document Clustering Using Wikipedia as External Knowledge": Springer
[15] Roul, Rajendra Kumar, Devanand, Omanwar Rohit & Sahay, SK, (2014) "Web Document Clustering and Ranking Using Tf-Idf Based Apriori Approach", arXiv preprint arXiv:1406.5617, Vol., No.
[16] Zhang, Zhilin, Lin, Huaizhong, Li, Pengfei, Wang, Huazhong & Lu, Dongming, (2013) "Improving Semi-Supervised Text Classification by Using Wikipedia Knowledge", Web-Age Information ManagementSpringer. pp. 25-36.
[17] Han, May Sabai, (2013) "Semantic Information Retrieval Based on Wikipedia Taxonomy", International Journal of Computer Applications Technology and Research, Vol. 2, No. 1: pp. 77-80.
[18] Gabrilovich, Evgeniy & Markovitch, Shaul, (2006) "Overcoming the Brittleness Bottleneck Using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge", AAAI. pp. 1301-1306.
[19] Spanakis, Gerasimos, Siolas, Georgios & Stafylopatis, Andreas, (2012) "Exploiting Wikipedia Knowledge for Conceptual Hierarchical Clustering of Documents", The Computer Journal, Vol. 55, No. 3: pp. 299-312.
[20] Milne, David & Witten, Ian H, (2008) "Learning to Link with Wikipedia", Proceedings of the 17th ACM conference on Information and knowledge management: ACM. pp. 509-518.
[21] Cilibrasi, Rudi L & Vitanyi, Paul MB, (2007) "The Google Similarity Distance", Knowledge and Data Engineering, IEEE Transactions on, Vol. 19, No. 3: pp. 370-383.
[22] Hu, Xia, Sun, Nan, Zhang, Chao & Chua, Tat-Seng, (2009) "Exploiting Internal and External Semantics for the Clustering of Short Texts Using World Knowledge", Proceedings of the 18th ACM conference on Information and knowledge management: ACM. pp. 919-928.
[23] Phan, Xuan-Hieu, Nguyen, Le-Minh & Horiguchi, Susumu, (2008) "Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-Scale Data Collections", Proceedings of the 17th international conference on World Wide Web: ACM. pp. 91-100.
[24] Agrawal, Rakesh & Srikant, Ramakrishnan, (1994) "Fast Algorithms for Mining Association Rules", Proc. 20th int. conf. very large data bases, VLDB. pp. 487-499.
[25] "Wikipedia Miner". 20th September 2014]; Available from: http://wikipedia-miner.cms.waikato.ac.nz/.

Authors

Iyad M. AlAgha received his MSc and PhD in Computer Science from the University of Durham, the UK. He worked as a research associate in the center of technology enhanced learning at the University of Durham, investigating the use of Multi-touch devices for learning and teaching. He is currently working as an assistant professor at the Faculty of Information technology at the Islamic University of Gaza, Palestine. His research interests are Semantic Web technology, Adaptive Hypermedia, Human-Computer Interaction and Technology Enhanced Learning.

Rami H. Nafee received his BSc from Al-Azhar University-Gaza and his MSc degree in Information technology from the Islamic University of Gaza. He works as a Web programmer at Information Technology Unit at Al-Azhar University-Gaza. He is also a lecturer at the Faculty of Intermediate Studies at Al-Azhar University. His research interests include Data Mining and Semantic Web.