Similarity Evaluation with Wikipedia Features

Shahbaz Wasti, Jawad Hussain, Guangjiang Huang, Yuncheng Jiang

Similarity Evaluation with Wikipedia Features

Shahbaz Wasti[1,2], Jawad Hussain[1], Guangjiang Huang[1], and Yuncheng Jiang[1]

[1] South China Normal University, Guangzhou 510631, Guangdong, China
[2] University of Education, Lahore 54000, Pakistan.
shahbazwasti@gmail.com

Abstract. Wikipedia provides rich semantic features e.g., text, link, and category structure. These features can be used to compute semantic similarity (SS) between words or concepts. However, some existing Wikipedia-based SS methods either rely on a single feature or do not incorporate the underlying statistics of different features. We propose novel vector representations of Wikipedia concepts by integrating their multiple semantic features. We utilize the available statistics of these features in Wikipedia to compute their weights. These weights signify the contribution of each feature in similarity evaluation according to its level of importance. The experimental evaluation shows that our new methods obtain better results on SS datasets in comparison with state-of-the-art SS methods.

Keywords: Semantic Similarity· IC· *tfidf* · Vector Representation.

1 Introduction

Semantic similarity (SS) assessment between words or concepts is a critical issue in natural language processing. The semantic features of Wikipedia concepts (e.g., article title, text, hyperlinks, and categories) have encouraged several researchers to develop word similarity methods. These different features complement each other in expressing a particular Wikipedia concept, e.g., the title represents a single concept, the article text narrates the subject matter, the hyperlinks give the related articles, and categories categorize the article. Moreover, the underlying statistics of these features can be also be exploited for weight computation. But some of the Wikipedia-based methods [4][5][6] either rely on a single feature or ignore the important statistical details of these features. In this paper, we propose a novel vector representation of a Wikipedia concept by combining multiple features. The entries of a vector are the assigned weights of the features computed using information content (IC) [11] and *tfidf* weighting schemes. The weights of the features will reflect their level of importance in similarity evaluation while multiple features will enhance the semantics of a concept. The rest of the paper is organized as: In Section 2, we present our methods. In Section 3, the detail about the experiment and evaluation criteria is provided. In Section 4, we present the result of our methods. Section 5 provides the related work on Wikipedia-based similarity measures. Section 6 concludes the paper.

2 Proposed methods

Wikipedia is the largest freely available encyclopedic knowledge resource. It covers more than five million articles in various domains of life. Each of these articles describes its corresponding concept. In this paper we will refer a Wikipedia article as "concept", which comprises on multiple features such as title, text (or words), hyperlinks (or links) and categories etc. In order to represent a Wikipedia concept as a concept vector, we propose following weighting methods to compute the weights of the features.

2.1 Measurement of the weights of the features

The IC-based method to compute the weights of the links and categories in Wikipedia is defined as:

Definition 1 (IC of features). *Let f_i be a feature (link or category) of a Wikipedia concept, $Fr(f_i)$ be its frequency and N is the total number of Wikipedia concepts. Then, th IC of the feature f_i is computed as:*

$$IC(f_i) = \log\left(\frac{1}{P(f_i)}\right) = -log(P(f_i)) = -log(\frac{Fr(f_i)}{N}), \tag{1}$$

where $P(f_i) = \frac{Fr(f_i)}{N}$ is the probability of the feature f_i.

The $tfidf$ (term frequency (tf) and inverse document frequency (idf)) is widely used to compute the weights of words in a corpus. These weights quantify the strength of association between words and concepts [2]. We use $tfidf$ weighting metric to compute the weights of the words appearing in the gloss of a Wikipedia concept. We first convert a gloss into a set of individual words (we remove all the stop words, special characters and numbers). The weights of the words can be computed as:

Definition 2 ($tfidf$ weight of gloss words). *Let w_i be a word in a gloss G of a Wikipedia concept. The $tfidf$ weight of w_i is computed as:*

$$tfidf(w_i, G) = tf(w_i, G) \times log(\frac{N}{G_{w_i} + 1}), \tag{2}$$

where $tf(w_i, G)$ is the term frequency of ith word in gloss G, G_{w_i} is the number of glosses (document frequency) in Wikipedia that contain the word w_i and N is the total number of Wikipedia concepts.

Wikipedia Category Graph (WCG) is considered as a very large semantic network, where categories are organized via semantic relationships (hypernymy (hypers) and hyponymy) (hypos). These semantic relationships can be used in similarity computation [4][5]. In Wikipedia some similar concepts don't have common categories but some of their categories do have common hypers in WCG. The intuitive idea is that two concepts will be more similar if their categories have common hypers as well. However, the huge size of WCG poses two challenges, i.e., large search space and strongly connected upper regions [5]. Therefore, instead

of traversing whole WCG, researchers preferred to restrict the search space to a limited depth [4][5]. In this paper, we also extract the hypers of a category c in its limited search space ($k-neighborhood$ of c). Intuitively, the $k-neighborhood$ is a subgraph, i.e., it is the set of all the categories that can be traversed from hypers and hypos of the category c via at most k edges [5]. The weight of hypers is computed with Equation 1.

2.2 Vector Construction

Definition 3 (Features vector). *Let con be a Wikipedia concept and w_p, l_q, c_r and h_s be its gloss words, links, categories and hypers respectively. Let $tfidf_{weight}$ (Equation 2) be the weight of words and IC_{weight} (Equation 1) be the weights of links, categories and hypers. The features vectors are defined as:*

$$\begin{aligned}
&\boldsymbol{v_w} = (tfidf_{weight}(w_1), tfidf_{weight}(w_2), ..., tfidf_{weight}(w_p)),\\
&\boldsymbol{v_l} = (IC_{weight}(l_1), IC_{weight}(l_2), ..., IC_{weight}(l_q)),\\
&\boldsymbol{v_c} = (IC_{weight}(c_1), IC_{weight}(c_2), ..., IC_{weight}(c_r)),\\
&\boldsymbol{v_h} = (IC_{weight}(h_1), IC_{weight}(h_2), ..., IC_{weight}(h_s)).
\end{aligned} \tag{3}$$

we propose three novel representations of the Wikipedia concept as a concept vector, e.g., (1) $\boldsymbol{GLC_{con}}$, (2) $\boldsymbol{GLH_{con}}$, and (3) $\boldsymbol{hypers_{con}}$.

Definition 4 (Concept vector). *Let con be a Wikipedia concept and $\boldsymbol{v_w}$, $\boldsymbol{v_l}$, $\boldsymbol{v_c}$, and $\boldsymbol{v_h}$ be its features vectors. The concept vectors of con are:*

$$\begin{aligned}
&\boldsymbol{GLC}_{con} = \boldsymbol{v}_{w_i} \oplus \boldsymbol{v}_{l_i} \oplus \boldsymbol{v}_{c_i},\\
&\boldsymbol{GLH}_{con} = \boldsymbol{v}_{w_i} \oplus \boldsymbol{v}_{l_i} \oplus \boldsymbol{v}_{h_i},\\
&\boldsymbol{hyper}_{con} = \boldsymbol{v}_{h_i},
\end{aligned} \tag{4}$$

where $\oplus$ represents the concatenation of different features vectors.

2.3 Semantic Similarity Computation

Definition 5 (Semantic similarity). *Let con_i be a pair of Wikipedia concepts, and $\boldsymbol{GLC_{con_i}}$, $\boldsymbol{GLH_{con_i}}$, and $\boldsymbol{hyper_{con_i}}$ be its concept vectors respectively. The similarity between two vectors is defined as:*

$$\begin{aligned}
&Sim_1(con_1, con_2) = cosine(\boldsymbol{GLC}_{con_1}, \boldsymbol{GLC}_{con_2}),\\
&Sim_2(con_1, con_2) = cosine(\boldsymbol{GLH}_{con_1}, \boldsymbol{GLH}_{con_2}),\\
&Sim_3(con_1, con_2) = cosine(\boldsymbol{hyper}_{con_1}, \boldsymbol{hyper}_{con_2}).
\end{aligned} \tag{5}$$

Fig. 1 illustrates the similarity computation between con_1 and con_2 using $\boldsymbol{GLC}_{con}$ concept vector. The elements of each vector are the weights of the features of corresponding concepts. To give more importance to the common words between two concept vectors, we will use the maximum $tfidf_{weight}$ for that particular word in both the concept vectors [12].

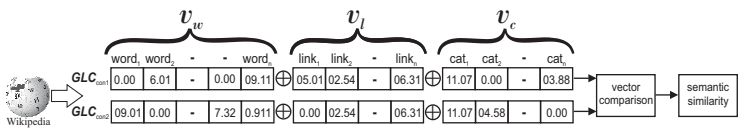

Fig. 1. $\boldsymbol{GLC}_{con}$ concept vector of Wikipedia concept pair.

3 Evaluation

We implement our approach using a Wikipedia snapshot as of December, 2018. We use Java Wikipedia Library to extract Wikipedia features, e.g., articles, categories and link structure. We remove the stop words, rare words and hyperlinks. To improve the efficiency of our approach, we compute the weights of the features offline. For $k - neighborhood$ we get the optimal results with $k = 4$.

3.1 Datasets and Evaluation Metric

We use the Pearson correlation coefficient metric to analyze the accuracy of our SS approaches on RG65[10], MC30[8], WS203[1], and SimLex[3] benchmarks. We normalize the similarity judgment scales of different datasets in the range of [0,1]. Some of the terms in above benchmarks are ambiguous in Wikipedia, i.e., Wikipedia may have more than one articles for a term, .e.g., minister, agony, journey, or crane etc. We adopted a simple strategy to disambiguate such concepts, i.e., we compute SS between all the associated ambiguous concepts using SS method Sim_2 and select the concept pair with highest SS score [9].

3.2 Comparison Systems

We compare our approaches with following well-known SS methods: the first system is $wpath$, it has two methods $wpath_{graph}$ and $wpath_{corpus}$ [13]. The methods measure the semantic similarity between the concepts in Knowledge Graphs (KGs) such as WordNet and DBpedia. However, in this paper we implemented $wpath$ methods in WCG. The second system is Word2Vec [7], where a neural network is used to learn continuous representations of word embedding. The third system is ESA [2] that represents the meaning of texts as high-dimensional weighted vectors of Wikipedia-based concepts. Finally, we select three methods from our previous works, $SimFou_{con}$ [6] and $SimSec_{con}$ and $SimSec_{cat}$ [9]. We implement all the comparison systems on the same Wikipedia version.

4 Results and Discussion

Table 1 shows the Pearson correlation performance of our methods and comparison systems on gold standard SS benchmarks. As we can see, both Sim_1 and Sim_2 achieve good results on all the benchmarks. Sim_2 performs better than

Table 1. Pearson correlation coefficient of proposed and comparison methods

Methods	MC30	RG65	WS203	SimLex
Sim_1	0.874	0.852	0.719	0.514
Sim_2	**0.885**	**0.868**	0.749	**0.515**
Sim_3	0.793	0.829	0.651	0.416
$wpath(corpus)$	0.514	0.781	0.482	0.356
$wpath(graph)$	0.582	0.824	0.508	0.381
$SimFou_{con}$	0.824	0.811	0.647	0.449
$SimSec_{con}$	0.845	0.827	0.712	-
$SimSec_{cat}$	0.842	0.836	0.686	-
$Word2Vec$	0.833	0.853	**0.763**	0.458
ESA	0.577	0.563	0.423	0.159

Sim_1. The reason is that combining hypers of the categories with other features (gloss words and links) yields a better semantic representation of a Wikipedia concept. It is because the concept pair will be more similar if their categories have common hypers even though they don't have a set of common categories. Our best method Sim_2 outperforms all the comparison methods on all MC30, RG65, SimLex. On benchmark WS203, Word2vec obtains the best correlation while Sim_2 shows the second best performance. All of our methods and other comparison methods relatively under-perform on SimLex as compared to other benchmarks. It is because in SimLex dataset the antonym pairs are rated dissimilar. While in KRs antonyms have a high degree of similarity.

5 Related Work

Jiang et al. [5] proposed IC-based measures by treating WCG as a large semantic ontology. However, in these methods other Wikipedia features like text and hyperlinks are not considered. Moreover, due to multiple inheritance in WCG, it is difficult to identify a single least common subsumer of two comparing categories [4]. Hussain et al. [4] proposed SS methods using multiple inheritance feature in WCG. However, the limitation of their methods is that they also do not consider the semantic details of other Wikipedia features. Qu et al. [9] proposed a series of hybrid SS methods, they combine text and categories to compute SS of Wikipedia concepts. Their methods require fine tuning of five weighting parameters to balance the contribution of each feature w.r.t Wikipedia versions and benchmarks. This hampers their applicability as a general-purpose solution in Wikipedia. In contrast, our approaches combines different Wikipedia features to construct a concept vector. Our approaches do not require any parameter tuning. Finally the results of our methods show better performance in term of correlation with human judgment.

6 Conclusion

We represent Wikipedia concept as a concept vector by integrating multiple Wikipedia features and their statistics. We use IC and $tfidf$ weights for the features. Our methods obtain show better performance on gold standard benchmarks in comparison with other SS methods. Especially the method Sim_2 proved to be more robust on all the benchmarks. The empirical evaluation shows that the integration of multiple weighted features improves the similarity assessment between concepts.

Acknowledgments: This work is supported by The National Natural Science Foundation of China under Grant Nos. 61772210 and U1911201; Guangdong Province Universities Pearl River Scholar Funded Scheme (2018); The Project of Science and Technology in Guangzhou in China under Grant No. 201807010043.

References

1. Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., Soroa, A.: A study on similarity and relatedness using distributional and wordnet-based approaches. In: Proceedings of Human Language Technologies. pp. 19–27 (2009)
2. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: IJcAI. vol. 7, pp. 1606–1611 (2007)
3. Hill, F., Reichart, R., Korhonen, A.: Simlex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics **41**(4), 665–695 (2015)
4. Hussain, M.J., Wasti, S.H., Huang, G., a nd Yuncheng Jiang, L.W., Tang, Y.: An approach for measuring semantic similarity between wikipedia concepts using multiple inheritances. Information Processing & Management **57**(3), 102188 (2020)
5. Jiang, Y., Bai, W., Zhang, X., Hu, J.: Wikipedia-based information content and semantic similarity computation. Information Processing & Management **53**(1), 248–265 (2017)
6. Jiang, Y., Zhang, X., Tang, Y., Nie, R.: Feature-based approaches to semantic similarity assessment of concepts using wikipedia. Information Processing & Management **51**(3), 215–234 (2015)
7. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. Computer Science (2013)
8. Miller, G.A., Charles, W.G.: Contextual correlates of semantic similarity. Language and cognitive processes **6**(1), 1–28 (1991)
9. Qu, R., Fang, Y., Bai, W., Jiang, Y.: Computing semantic similarity based on novel models of semantic representation using wikipedia. Information Processing & Management **54**(6), 1002–1021 (2018)
10. Rubenstein, H., Goodenough, J.B.: Contextual correlates of synonymy. Communications of the ACM **8**(10), 627–633 (1965)
11. Shannon, C.E.: A mathematical theory of communication. Bell system technical journal **27**(3), 379–423 (1948)
12. Wasti, S.H., Hussain, M.J., Huang, G., Akram, A., Jiang, Y., Tang, Y.: Assessing semantic similarity between concepts: A weighted-feature-based approach. Concurrency and Computation: Practice and Experience **32**(7), e5594
13. Zhu, G., Iglesias, C.A.: Computing semantic similarity of concepts in knowledge graphs. IEEE Transactions on Knowledge and Data Engineering **29**(1), 72–85 (2017)