Article

Experimental data for computing semantic similarity between concepts using multiple inheritances in Wikipedia category graph

Muhammad Jawad Hussain\textsuperscript{a,\,*}, Shahbaz Hassan Wasti\textsuperscript{a,b}, Guangjian Huang\textsuperscript{a}, Yuncheng Jiang\textsuperscript{a,\,*}

\textsuperscript{a}School of Computer Science, South China Normal University, Guangzhou 510631, China
\textsuperscript{b}Division of Science and Technology, University of Education, Lahore, Pakistan


doi:10.1016/j.dib.2020.105377
2352-3409/© 2020 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license.

https://doi.org/10.1016/j.dib.2020.105377
2352-3409/© 2020 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license.

This data article compiles the detailed and descriptive experimental data of Wikipedia-based semantic similarity approach called as Neighbourhood Aggregated Semantic Contribution (NASC), presented in Husain, et al. [1]. The JWPL (Java Wikipedia Library)-DataMachine and JWPL WikipediaAPI are used to extract the required Wikipedia features from Wikipedia dump. The dataset presents the disambiguated Wikipedia concepts of the gold standard word similarity benchmarks MC30 (English), RG65\textsubscript{es} (Spanish) and RG65\textsubscript{fr} (French) and their associated set of categories in the corresponding Wikipedia category graph (WCG). The dataset also contains the number of ancestors, common ancestors, pages, and common pages in the k-neighbourhood of the associated categories for different levels of parameter k in the English, Spanish, and French WCGs. The presented dataset can be used to assess the semantic similarity between Wikipedia concepts in English (MC30), Spanish (RG65\textsubscript{es}), and French (RG65\textsubscript{fr}) languages benchmarks. Moreover, the dataset will be useful for the further analysis and comparison of the taxonomic structures of the English, Spanish, and French WCGs.

© 2020 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license.

(http://creativecommons.org/licenses/by/4.0/)

Keywords:
Semantic similarity
Wikipedia category graph
Multiple inheritances
Information content

Corresponding authors.

E-mail addresses: jawad@m.scnu.edu.cn (M.J. Hussain), ycjiang@scnu.edu.cn (Y. Jiang).

(*Corresponding authors.

E-mail addresses: jawad@m.scnu.edu.cn (M.J. Hussain), ycjiang@scnu.edu.cn (Y. Jiang).

https://doi.org/10.1016/j.dib.2020.105377
2352-3409/© 2020 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license.

(http://creativecommons.org/licenses/by/4.0/)
| Specification Table                                      |
|---------------------------------------------------------|
| Subject                                                | Information systems                                      |
| Specific subject area                                   | Artificial Intelligence, Natural Language Processing, Information Retrieval |
| Type of data                                            | Tables, Graphs, Figures                                  |
| How data were acquired                                  | JWPL (Java Wikipedia Library)-DataMachine and JWPL WikipediaAPI were used to extract the required information from Wikipedia. |
| Data format                                             | Raw and processed                                        |
| Parameters for data collection                          | We removed cycles and all the hidden categories (administrative categories) from the corresponding Wikipedia category graph. |
| Description of data collection                          | We downloaded Wikipedia dump and used JWPL (Java Wikipedia Library)-DataMachine to extract the required features from Wikipedia such as page ids, titles, and page categories. Then we built the corresponding WCG by using JWPL WikipediaAPI. We traversed WCG to get the required data such as: k-neighbourhood, k-ancestors, category pages, and descendant of a particular category. |
| Data source location                                    | School of Computer Science, South China Normal University, Guangzhou 510631, China. |
| Data accessibility                                      | Repository name: Mendeley data repository                 |
| Data identification number                              | 10.17632/hnmb43sj5s.1                                    |
| Direct URL to data                                      | http://dx.doi.org/10.17632/hnmb43sj5s.1                  |
| Related research article                                | Author’s name: Muhammad Jawad Hussain, Shahbaz Hassan Wasti, Guangjian Huang, Lina Wei, Yuncheng Jiang, Yong Tang |
| Title                                                   | “An approach for measuring semantic similarity between Wikipedia concepts using multiple inheritances” |
| Journal                                                 | Information Processing & Management                      |
| DOI                                                     | https://doi.org/10.1016/j.ipm.2019.102188                |

**Value of the Data**

- The presented experimental data is useful to measure the semantic similarity between Wikipedia concepts.
- The data is beneficial for all the scientists who are exploiting Wikipedia as a Knowledge Resource.
- The provided data can be manipulated for the further analysis of the taxonomic structures and comparison among the English, Spanish, and French versions of the Wikipedia category graphs.

## 1. Data

Figs. 1–3 show the graphs of the Pearson correlation values of our proposed Neighbourhood Ancestor Semantic Contribution (NASC)-based semantic similarity methods in gold standard word similarity benchmarks of English, Spanish, and French languages. The Pearson correlation values are shown on different settings of parameter k for MC30 (English) [2], RG65es (Spanish) [3], and RG65fr (French) [4] benchmarks.

Tables 1–3 present the number of categories and common categories for the selected Wikipedia concept pair (Coast, Forest) from MC30 (English) and its equivalent pairs (Costa, Bosque), and (Cote geographic, Foret) from RG65es (Spanish) and RG65fr (French) on different values of parameter k. Moreover, these Tables also highlight the structural differences among English, Spanish, and French WCGs in terms of size and branching factor on different values of parameter k.

Fig. 4 shows the directory structure of all the supplementary data provided with this article on Mendeley data repository [5]. These data files can be used to reproduce the experiments of our methods and for the further analysis on English, Spanish, and French WCGs structures [1]. The folder “Benchmarks_results_graphs” contains all the data related to the graphs that are
**ENGLISH BENCHMARK (MC30)**

![Graph](image1)

**Figure 1.** The Pearson correlation of our methods with different settings of parameter $k$ on English MC30 benchmark.

**SPANISH BENCHMARK (RG65)**

![Graph](image2)

**Figure 2.** The Pearson correlation of our methods with different settings of parameter $k$ on Spanish RG65es benchmark.
Table 1
The number of categories, common categories of English Wikipedia concepts (Coast, Forest) on different settings of parameter k using English WCG.

| K  | Categories of Coast | Categories of Forest | Common categories |
|----|---------------------|----------------------|-------------------|
| 1  | 12                  | 17                   | 0                 |
| 2  | 30                  | 54                   | 2                 |
| 3  | 62                  | 120                  | 8                 |
| 4  | 111                 | 221                  | 22                |
| 5  | 183                 | 356                  | 44                |
| 6  | 267                 | 540                  | 74                |
| 7  | 363                 | 765                  | 124               |
| 8  | 482                 | 999                  | 175               |

Table 2
The number of categories, common categories of Spanish Wikipedia concepts (Costa, Bosque) on different settings of parameter k using Spanish WCG.

| K  | Categories of Costa | Categories of Bosque | Common categories |
|----|---------------------|----------------------|-------------------|
| 1  | 8                   | 9                    | 1                 |
| 2  | 18                  | 24                   | 4                 |
| 3  | 34                  | 55                   | 12                |
| 4  | 54                  | 101                  | 24                |
| 5  | 78                  | 158                  | 38                |
| 6  | 104                 | 222                  | 51                |
| 7  | 135                 | 286                  | 70                |
| 8  | 173                 | 343                  | 91                |

Fig. 3. The Pearson correlation of our methods with different settings of parameter k on French RG65fr benchmark.
Table 3
The number of categories, common categories of French Wikipedia concepts (Cote geographic, Foret) on different settings of parameter k using French WCG.

| K | Categories of Cote geographic | Categories of Foret | Common categories |
|---|-------------------------------|---------------------|-------------------|
| 1 | 3                             | 18                  | 0                 |
| 2 | 5                             | 39                  | 0                 |
| 3 | 11                            | 67                  | 2                 |
| 4 | 19                            | 101                 | 7                 |
| 5 | 31                            | 137                 | 12                |
| 6 | 46                            | 174                 | 21                |
| 7 | 59                            | 206                 | 28                |
| 8 | 69                            | 239                 | 33                |

Fig. 4. The Mendeley data directory structure of supplementary data files.

included in this article. The folders “French_RG65”, “MC30”, and “Spanish_RG65” have all the necessary pre-processed data files to execute the python based program to compute the semantic similarity between English, Spanish, and French Wikipedia concepts according to our methods. For example, as shown in Fig. 4, the folder “French_RG65” contains: (1) the experiments on RG65fr benchmark in the sub-folder named as “French_RG65_results”, (2) the data required
for the computation of $IC^k_{\text{neigh}}$ and $IC^k_{\text{neigh}}$ [1] in the sub-folder named as “predata_fr”, (3) the disambiguated French Wikipedia concepts in the file named as “disambiguated_benchmark.csv”, (4) the French Wikipedia concepts page ids in the file named as “fr_RG65_pageid.csv”, (5) the French Wikipedia page associated categories in the file named as “fr_RG65_page_categories.txt”, (6) the source code to compute the semantic similarity between the concepts of French Wikipedia using $IC^k_{\text{neigh}}$ in the file named as “RG_French_Sim_IC_hypos.txt”, (7) the source code to compute the semantic similarity between the concepts of French Wikipedia using $IC^k_{\text{neigh}}$ in the file named as “RG_French_Sim_IC_pages.txt”, and (8) the source code to reproduce the data associated to Table 3 in the file named as “Table3_French.txt”.

Fig. 5 shows the image of our python-based functions named as “get_Sweight ()” and “get_SV ()”. These functions are used to compute the semantic weight and semantic value of a category according to its k-neighbourhood in the corresponding WCG respectively.

Fig. 6 presents the image of our python-based functions named as “get_AggSweight ()” and “get_SS ()”. The first function returns the aggregated semantic weight of a category. The second function computes the similarity between two comparing categories in the corresponding WCG.
```python
def compute_SS(c1,c2,k):
    result=[]
    maxSimCat=[]
    avgSimCat=[]
    p1 = get_Page_ID(c1)
    p2 = get_Page_ID(c2)
    cats1=get_CategoryList(p1)
    cats2=get_CategoryList(p2)
    minSize=0
    if (p1==p2):
        result.append(1.0)
        result.append(1.0)
        result.append(1.0)
    else:
        if (len(cats1)<len(cats2)):
            minSize=len(cats1)
            set1=cats1
            set2=cats2
        else:
            minSize=len(cats2)
            set1=cats2
            set2=cats1
        for i in set1:
            simCat=[]
            for j in set2:
                if (i==j):
                    simCat.append(1.0)
                else:
                    simCat.append(get_SS(i,j,k))
            simCat = np.array(simCat)
            maxSimCat.append(simCat.max())
            avgSimCat.append(simCat.mean())
        maxSimCat=np.array(maxSimCat)
        total=0.0
        total=maxSimCat.sum()
        result.append(total/minSize)#1st Similarity Score
        result.append(maxSimCat.max())#2nd Similarity Score
        avgSimCat=np.array(avgSimCat)
        result.append(avgSimCat.max())#3rd Similarity Score
        result=np.array(result)
    return result
```

**Fig. 7.** The image of the Wikipedia concepts semantic similarity computation function.

**Fig. 7** shows the image of the semantic similarity computation function named as “compute_SS ()”. This function computes the semantic similarity between two Wikipedia concepts for a specific value of parameter \( k \) in the corresponding WCG.

## 2. Experimental design, materials, and methods

### 2.1. Data extraction

Firstly, we used JWPL (Java Wikipedia Library)-DataMachine to extract Wikipedia features from Wikipedia dump. JWPL is an open-source, Java-based application programming interface that allows access to all the information contained in Wikipedia. JWPL extracts the Wikipedia features such that: page ids, page categories, redirects (synonyms) and category structure etc.,
from Wikipedia dump file and stores these features in MYSQL tables. Secondly, we constructed Wikipedia category graph (WCG) using JPL WikipediaAPI. This WikipediaAPI constructs acyclic WCG and removes all the hidden (administrative) categories from it [6]. Finally, we explored the taxonomic structure of this constructed WCG to get the related data such as k-neighbourhood, hypernyms, hyponyms, and k-ancestors. We stored all the required data in the panda data frames to implement our python-based program to compute semantic similarity between Wikipedia concepts.

2.2. The parameter k and implementation of our methods

We used Wikipedia category graph (WCG) as a semantic network in our methods. However, traversing whole WCG is not only computationally expensive but also reduces the accuracy of multiple inheritance-based semantic similarity methods [1]. Therefore, we only traversed a sub-graph of WCG (referred to as k-neighbourhood) for a particular category (including itself) to define its semantic space. The parameter k is a positive integer such that 1 ≤ k ≤ max_depth(WCG), which defines the size of the sub-graph or k-neighbourhood of a category in the corresponding WCG. Intuitively, the k-neighbourhood of a category (node) a ∈ WCG (k-neighbourhood of (a)) represents the set of all nodes (ancestors or descendants) of the category ‘a’ which can be traversed via at most k edges [7].

We only aggregated the IC-based semantic contribution weights of the k-ancestors of a particular category to achieve the notion of multiple inheirtances. Where the k-ancestors represents the ancestors of a category in its k-neighbourhood.

\[
k - IC_p(u) = -\log\left(\frac{\sum_{v \in k-hyponyms(u) \cup |u|} |page(v)| + 1}{\sum_{v \in graphV(u)} |page(v)| + 1}\right)
\]

(1)

\[
k - IC_h(u) = 1 - \frac{\log(k - hyponyms(u) + 1)}{\log(|k_graphV(u)|)}
\]

(2)

\[
SC^{k_{neigh}}(v)(u) = \frac{IC^{k_{neigh}}(u)}{1 + IC^{k_{neigh}}(u)}
\]

(3)

We used Eq. (3) to compute the semantic contribution weight of a particular category in the corresponding WCG and assigned a numerical value to it. Note that we implemented Eq. (3) by using two types of ICs (see Eqs. (1) and (2)) [8]. Fig. 5 shows the image of the function which implements Eq. (3). The function “get_Sweight (catid, hid)” returns the IC-based (using Eq. (1) to compute the IC) semantic contribution weight of the ancestor of a category. The function “get_SV (catid, k)” aggregates the semantic contribution weights of all the k-ancestors of a category to compute its semantic value.

\[
Sim^{k_{neigh}}_{cat}(u, v) = \frac{\sum_{w \in ([k - A(u)] \cap [k - A(v)])} \left(SC^{k_{neigh}}(u) (w) + SC^{k_{neigh}}(v) (w)\right)}{Sem^{k_{neigh}}_{u}(u) + Sem^{k_{neigh}}_{v}(v)}
\]

(4)

To implement Eq. (4), the function “get_AggSWeight (a, b, k)” returns the aggregated semantic contribution weight of the common k-ancestors of two comparing categories ‘a’ and ‘b’ on a specific value of parameter k. The function “get_SS (a, b, k)” computes the similarity between two categories by aggregating the semantic contribution weights of the common ancestors of two categories ‘a’ and ‘b’ in the nominator and divides it by the individual semantic values of both the categories in the denominator for a specific value of parameter k as depicted in Fig. 6.

\[
S_{cat}(cs_1, cs_2) = \frac{1}{min(m, n)} \max_{cs \in CS(cs_1, cs_2)} \left(\sum_{\langle cat_1, cat_2 \rangle \in CS} Sim^{k_{neigh}}_{cat}(cat_1, cat_2)\right)
\]

(5)
\[ \text{MaxS}_{\text{cat}}(c_1, c_2) = \max_{c_{11} \in c_1, c_{2j} \in c_2} \left( \text{Sim}^{k_{\text{neigh}}}_{\text{cat}}(\text{Cat}_{11}, \text{Cat}_{2j}) \right) \]  

(6)

\[ \text{AvgS}_{\text{cat}}(c_1, c_2) = \max \left( \sum_{c_{11} \in c_1, c_{2j} \in c_2} \left( \text{Sim}^{k_{\text{neigh}}}_{\text{cat}}(\text{Cat}_{11}, \text{Cat}_{2j}) \right) \right) \]  

(7)

Finally, the function "compute_SS (c1, c2, k)" taking three inputs as parameters: the titles of two Wikipedia concepts and the value of parameter k. This function computes semantic similarity between two Wikipedia concepts by using different aggregation functions which are defined in Eqs. (5)–(7) [9,10]. These aggregation functions are implemented by using Numpy arrays as depicted below in Fig. 7. The actual source code and all other required data files are provided in the supplementary data.

**Acknowledgments**

The works described in this paper are 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.

**Conflict of Interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**References**

[1] M.J. Hussain, S.H. Wasti, G. Huang, L. Wei, Y. Jiang, Y. Tang, An approach for measuring semantic similarity between Wikipedia concepts using multiple inheritances, Inf. Process. Manag. 57 (2020), doi:10.1016/j.ipm.2019.102188.

[2] G.A. Miller, W.G. Charles, Contextual correlates of semantic similarity, Cogn. Process. 6 (1991) 1–28.

[3] J. Camacho-Collados, M.T. Pilehvar, R.Navigli, A framework for the construction of monolingual and cross-lingualword similarity datasets, ACL-IJCNLP 2015 - 53rd Annu. Meet. Assoc. Comput. Linguist. 7th Int. Jt. Conf. Nat. Lang. Process. Asian Fed. Nat. Lang. Process. Proc. Conf. 2 (2015) 1–7, doi:10.3115/v1/p15-2001.

[4] L. Gurevych, Using the structure of a conceptual network in computing semantic relatedness., Int. Conf. Nat. Lang. Process (2005) 767–778.

[5] M. Jawad Hussain, S.H. Wasti, G. Huang, Y. Jiang, Experimental data for computing semantic similarity between concepts using multiple inheritances in Wikipedia category graph, Mendeley (2020), doi:10.17632/hnmrb43j5s.1.

[6] S.H. Wasti, M.J. Hussain, G. Huang, A. Akram, Y. Jiang, Y. Tang, Assessing semantic similarity between concepts: A weighted-feature-based approach, Concurr. Comput. Pract. Exp. (2020) e5594, doi:10.1002/cpe.5594.

[7] Y. Jiang, X. Zhang, Y. Tang, R. Nie, Feature-based approaches to semantic similarity assessment of concepts using Wikipedia, Inf. Process. Manag. 51 (2015) 215–234.

[8] Y. Jiang, W. Bai, X. Zhang, J. Hu, Wikipedia-based information content and semantic similarity computation, Inf. Process. Manag. 53 (2017) 248–265.

[9] A. Formica, Ontology-based concept similarity in formal concept analysis, Inf. Sci. 176 (2006) 2624–2641.

[10] Z. Galil, Efficient algorithms for finding maximum matching in graphs, ACM Comput. Surv. 18 (1986) 23–38.