GÜFBED/*GUSTIJ* (2018) CMES 2018 SEMPOZYUM EK SAYISI: 67-78
DOI: 10.17714/gumusfenbil.432582 Araştırma Makalesi / *Research Article*

Semantic Relation's Weight Determination on a Graph Based WordNet

Çizge Tabanlı WordNet Ağı Üzerinde Anlamsal İlişki Ağırlıklarının Tespiti

Çağatay TÜLÜ*[1, a] Umut ORHAN[2,b], Erhan TURAN[3,c]
[1]*Adana Science and Technology University, Information Technologies Department, Adana*
[2]*Çukurova University, Faculty of Engineering and Architecture, Computer Engineering Department, Adana*
[3]*Osmaniye Korkutata University, Faculty of Engineering, Computer Engineering Department, Osmaniye*

• Geliş tarihi / *Received*: 10.06.2018 • Düzeltilerek geliş tarihi / *Received in revised form*: 16.10.2018 • Kabul tarihi / *Accepted*: 24.10.2018

Abstract
Determination of semantic relatedness between two textual items is one of the crucial phases in many Natural Language Processing applications. In this study, a new approach to lexicon based semantic relation determination methods was experienced using WordNet 3.0 and Men's real-life similarity dataset. Men's test collection was used for the determination of the relation weights and determined weights were used in semantic relatedness computation. RG65 similarity dataset was used for a benchmark of the proposed method and Spearman correlation 0.81 was gained, taking into account that retrieving the relations weight using a large scale dataset and testing them with another real-life dataset promises new perspectives to the determination of the relations weight and to the relatedness computation.

Keywords: Natural language processing, Semantic relation, WordNet

Öz
Birçok doğal dil işleme uygulamasında metinsel iki öğenin anlamsal ilişkisinin tespit edilmesi çok önemli bir aşamadır. Bu çalışmada WordNet 3.0 ve Men's veri seti kullanarak sözlük tabanlı anlamsal ilişki belirleme metodları için yeni bir yaklaşım sunulmaktadır. Anlamsal ağırlıkların hesaplanmasında Men's veriseti kullanılmış ve bulunan değerler anlamsal ağırlık hesaplanmasında kullanılmıştır. Önerilen metodun doğruluğunu ölçmek için RG65 benzerlik veriseti kullanılmış, kayıslama sonucunda 0.81 Spearman korelasyon değeri elde edilmiştir. Büyük boyutlu bir verisetinin geliştirme ve test için kullanılıp, diğer önemli bir verisetinin de kıyaslama amaçlı olarak anlamsal ilişki tiplerinin ağırlıklarının belirlenmesi ve anlamsal ilişkinin hesaplanmasında kullanılması anlamsal benzerlik ve anlamsal ilişki hesaplanmasına farklı bir bakış açısı getirmektedir.

Anahtar kelimeler: *Doğal dil işleme, Anlamsal ilişki, WordNet*

*[a] Çağatay TÜLÜ; cagataytulu@gmail.com; Tel: (0322) 4550010; https://orcid.org/0000-0002-4462-3707
[b] orcid.org/ 0000-0003-1882-6567 [c] orcid.org/0000-0002-2953-2698

1. Introduction

Scaling the relatedness of two different concepts always takes attention from the researchers, assigning a value to the relatedness of two concepts brings new research opportunities especially for expert systems, smart applications, automatic question answering systems, text summarization, short answer grading, and many other Natural Language Processing (NLP) applications. Developing a mathematical model for real-life problems enables to generate algorithm and helps to solve them easily and consistently by the machine. So it is very important to use tools and resources that are compatible with mathematical operations.

WordNet (Fellbaum, 1998) is a lexical semantic knowledge base designed for the English language. It is in a graph structure, so it can be adapted to perform mathematical operations. WordNet's core elements are concepts and relations. While concepts represent the unique semantic entities, relations are the semantic signs to address the relations between two concepts. There are twenty-six different type of relations defined in WordNet version 3.0. The most common relation is the Hypernym/Hyponym relations. Each concept is connected to neighbor concepts through relations. By these connections and concepts, WordNet can be represented as a graph network. Concepts are the nodes (vertex) of the graph and relations between concepts are the edges of the graph. This structure enables that relatedness between concepts in the graph network can be calculated using graph metrics. One of the important metrics is the path between two concepts. The path is the combination of nodes and edges. Relatedness can be measured by the length of the path. Type of the relation is important; WordNet-based semantic similarity measurement studies generally use hypernym/hyponym relation types, because they have well defined hierarchical structure. Hirst and St Onge (Hirst and St-Onge, 1998) have used direction change of the path in addition to path length, Wu and Palmer (Wu and Palmer, 1994) have used Lowest Common Subsumer (LCS) of the concepts and depth of the concepts from the top entity in the hierarchy.
A real-life dataset for the semantic relatedness consist of the word pairs (word1,word2) and relatedness value for each pair, they are evaluated by native English speakers, by the way, they give judgment of relatedness for a word pair (word1,word2), judgment can be made with numerical value or as true-false approach. There are several real-life datasets like RG65 (Rubenstein and Goodenough, 1965) consists of 65-word pairs, MC30 (Miller and Charles, 1991) with 30-word pairs, WordSim353 (Finkelstein et al, 2001) with 353-word pairs, Mens 3000 (Bruni et al, 2014) with 3000-word pairs.

In this study, we have focused on determining the weight of semantic relations which connects one concept to another in the WordNet as illustrated in Figure 1. Men's 3000 (Baroni et al, 2014) real-life dataset is used in the development phase of the proposed system. Using Men's 3000 real-life datasets in the proposed method, the weight of the relations in the WordNet is determined. Determined relations weights are applied to the real-life dataset RG65 simply multiplying the weights of the relations in the path and relatedness score is obtained. To evaluate the success of the found relatedness values in the RG65 real-life data set is used.

Rest of the paper is organized as follows; section 2 gives brief literature history of the semantic relatedness and semantic weight determination methods, section 3 presents detailed information about the method of the study, section 4 explains benchmark and evolution tests, section 5 is results and discussion part. And finally, section 6 is the conclusion of the study and gives a future aspect of the proposed method.

2. Related Works

Semantic relatedness, semantic similarity or semantic distance is the terms used for the same purposes, it aims to determine whether two textual items are related or not in a semantic manner. These textual items can be a single sense (concept), single words or short texts like sentences or paragraphs or documents. Relatedness of two textual items can be computed using several approaches; the first one is statistical approaches (Li et al, 2003; Sultan et al, 2014); it uses corpus statistics of the textual items and compares them according to retrieved statistical values from the corpus and generates relatedness score. The other one is knowledge-based approaches (Agirre and Soroa, 2009; Hughes and Ramage, 2007); it uses the features of hierarchical lexical knowledge bases. WordNet and conceptNet (Speer and Havasi, 2012) are the important ontology-based structured lexical knowledge bases. Wiktionary (en.wiktionary.org) are lexicon based free multilingual knowledge base developed collaboratively by people around the World and Babelnet (Navigli and Ponzetto,

2012) is another type of the knowledge base using multiple sources with cross-lingual structure. The last approach for relatedness is the hybrid approach(Meng et al, 2012) that combines knowledge-based and corpus-based approaches together.

Knowledge-based semantic relatedness computation methods use a resource that is consist of unique entities. These entities are called concepts. Each concept contains lexical or/and encyclopedic information about the concept and relation information to the other concepts. WordNet contains semantic relations among the concepts (known as synset) and their POS (Part of Speech) information. WordNet based methods are used for the semantic similarity measurement of short texts like senses, words, sentences and short paragraphs. Semantic relations in the WordNet are one of the key elements to the semantic relatedness computation. In the hierarchical structure of the WordNet, these relations provide the ability to reach from one concept to another. From originating synset to the destination synset in the WordNet, retrieving a path that consists of nodes and relations gives us the ability to make mathematical operations. Assigning a numerical value to the nodes and relations enables similarity calculation to be more accurate.

There are several studies in the literature about weighting the relations in the relatedness computation of WordNet concepts, one of older is the Hirst&St-Onge (Hirst G. and St-Onge D, 1998) method, in this method relatedness between two concepts in the WordNet is calculated using the path length and direction change of the path from originating to the destination concept. In this approach, relation type hypernym and autonomy are taken into account and only noun concepts are used and the weight of these relations are given as numerical value 1. Another approach is the Wu-Palmer (Wu Z., Palmer M., 1994) method that uses both path length and taxonomic depth of the concepts. Yang and Powers (Yang and Powers, 2005) have proposed edge counting techniques for the weight of the relation types. In this method, two kinds of searching methods in the graph are used, bidirectional depth-limit search (BDLS) and uni-directional bread-first search (UBFS), combining the metrics taken from this searches, similarity/relatedness value is generated. This method uses the relation type in three different level, the first one is an identical level where two concepts are identical relation weight is 1.0. The second one is synonym/antonym level where relation weight is taken as 0.9, this is also called intermediate level, lower level relation weight is taken as 0.85, relation types hypernym/hyponym and holonym/meronym are the examples of lower level relation types. Searching depth is another parameter in this method, and the value of searching depth depends on the type of relation. This method also covers the verb similarity. It has been tested in 28 noun pair dataset (Miller and Charles, 1991), and the correlation of this method to human judgment is found as 0.921.

The other important study about relation weighting has been performed by Ahsaee et al (Ahsaee et al, 2014) in which they suggested using the weight of an edge as 1.0 in the leaf nodes, and decreasing the weight of node according to proximity to the root. For example, a leaf node takes the weight as 1.0, one step up of the node to root is taken as 0.9, two steps upward are taken as 0.81. It is assumed that when a concept goes up from the leaf node to the root node, its subjectivity decreases and objectivity increases. This method also achieved high correlation in the RG65 dataset, this method takes hypernym/hyponym relation and cover just nouns.

Siblini and Kosseim (Siblini and Kosseim, 2013) have categorized the relations and assign a numerical weight into each category. Semantic categories are Similar, Hypernym, Sense, Gloss, Part, Other Instance. They have created a semantic network using the WordNet 3.1 as a source. In addition to the synsets in the created graph, they have defined node type word to point relation between synset and concept for the sense and gloss relation types. Sense relation from synset to the word is created for the different word forms of a synset and gloss relation from synset to the word is created for each keyword in the synset's definition. For example, synset (automobile) is connected to the car node with sense relation and Wheel node with gloss relation. Generated graph network in the mentioned study consists of around 265k node and around 2 million relations. In order to measure semantic relatedness of given two concepts, that have used path cost from originating to the destination node, the path cost is calculated using relation types and weights in the path and also some other constant parameters are used. This method had given a performance in a MC30 dataset with Pearson Correlation value=0.93, but it gives poor performance in WordSim353 dataset with correlation 0.5. In the other WordNet-based approach systems like PageRank (Brin and Page, 1998), weights of the relations are generally taken as numerical value 1.0. These methods generally focus on other graph parameters than relation weights.

Machine-learned vector space model (Speer and Chin, 2016) has used which combines word embedding that is produced by GloVe (Pennington et al., 2014) and word2vec(Mikolov et al., 2013) using tightly structured semantic networks like conceptNet(Speer and Havasi, 2012). Kartsaklis et al (Kartsaklis et al, 2018) have proposed a method which maps the natural language texts to the knowledge-based entities, they have enhanced LTSM model with a dynamic disambiguation mechanism on the input word embeddings that address polysemy issue. This method has gained state of the art performance in many word-similarity evaluations.

3. Methodology

Initially, we have examined "The MEN Test Collection" (Bruni et al, 2014) relatedness dataset, it was prepared for the benchmark purposes of the similarity/relatedness studies. During the experiment, we have observed that relatedness between two concepts is mostly affected by the relation type and path length. Generally, relatedness and path length have a negative correlation. The shortest path between two concepts is the key parameter to prove the strength of relation in WordNet. But the strength of the relation of two concepts depends not only on the shortest path. Daily usage and cultural diversity imply the strength of the relations between two concepts in the human mind. Instead of a shortest path, all possible paths between two concepts are taken into account in this study, this approach is more close to human mind judgment.

Then we have decided to get all the paths between two concepts including shortest path, but retrieving all the paths between two concepts are time-consuming process for such a graph consist of the 117k node and more than 700k relations. So the length of the paths should be limited, we have performed tests on the graph DB and finally, we have determined path length interval as, interval more than 3 is getting the system into extreme CPU load and memory consumption that is causing in abnormal behavior and critical exceptions in the application. Query execution time is increased exponentially when path length is increased.

pathLenghtInterval=(shortestPath,shortestpath+3) (1)

We have collected all the paths between two words of each word pair in Men's dataset and retrieved paths are encoded with their relationship type, WordNet relation types and corresponding relation code can be found in Table 6. We have collected all paths of each word pair in Mens data collection. Path collection is performed as explained in the following example;

relatedness("*mushroom*", "*tomato")=0.74* given in Men's dataset.

As a first step, we find mushroom and tomato as nodes in WordNet graph db. Then we determined the shortest path length between this two node, the shortest path length is 3 and we retrieve all the path between mushroom and tomato nodes with *pathLenght=3+3=6*. Later we collected all the encoded paths and assign the value 0.74 to each of them as in the following;

findAllPath("mushroom", "tomato")= [bcjbck, bcjbck, bcjbck, bcjbck, bjcckb, bjckbc, brsbbc, bccbc, bjck, bcc, bcccjb, bccjbc, bccjbc, bcjbbc, bccjbc, bcjbbc, bcjbbc, bcjbbc, bccbcj, bcjbc, bcjbc, bcjbc, bcjbc, bccj, bjc]=0.74

All the paths in the above list are assigned with value 0.74, the same procedure is done all pairs in the Men's data, all dataset is consist of 3000-word pairs. Same paths with different relatedness score are grouped together and assigned a single score to show it as a relatedness score of that path. Consequently, we have assigned median and mean of the grouped values for a path. By the way, each path is represented with a numerical value between {0,1}, details of this process are explained in subsections of this section

We have filtered values of single length paths (pathLength=1) from the path list, then we have assigned these values to the relations, some relations are not found uniquely (as path length=1) their values are extracted from the paths. At this point, we have determined weights of each relation type. This phase is the first step of the relation weight determination process. In order to get strength the relation weights, we have calculated the path values using found relation weights in the first phase, in the second phase we have found many path values for a single word pair, then we have taken the nearest values to the real values of the relatedness. As in the above process, the unique path is taken directly as a relation weight and weight of the other relations are extracted using the weights found before.

3.1. Men's Data

Mens data (Bruni et al, 2014) is taken as a trusted dataset for the development phase of the semantic relations weight determination phase. This data consists of 3000 pairs of words, there are words from all part of speech; nouns, verbs, adjectives, and adverbs. Word pairs are evaluated by fifty native English speakers, every pair of words was asked to the humans to give judgment whether they are related or not related. And relatedness score was given according to this result between 0-50, while 0 (zero) shows all evaluators have given "not related"judgment, and 50 shows all evaluators have given "related" judgment. In this study, these values are normalized between 0-1 by dividing them into 50.

3.2. WordNet 3.0

WordNet 3.0 is another important component of the study. In this study, all the synsets and relations in WordNet are mapped into the graph network. In order to achieve this step a simple java application is developed, this application parses the data files from WordNet 3.0, and takes synset id, senses (literals of the sunset), pos type of a synset and creates a node in Graph database and assigns these values as a feature of the node, using the relation information; edges from source node to destination node are defined. A graph database is created in the Neo4j application (https://neo4j.com), graph network consists of around 117K nodes and 771K relations. All the relation types in WordNet are distributed in a graph network. Some relations were only unidirectional (for example, pertainym) in WordNet since proposed network to be bidirectional, these type of relations are converted into bidirectional type just by adding reverse relation, for example, reverse of the pertainym relation is converted to pertained pertainym<->pertainedBy. By the way, the bidirectional structure is completed.

3.3. Path Weight Determination Process for Each Path

We have used Men's 3000 data as trusted data to develop proposed system. For each word pair (w1, w2) in Men's data, we have taken all the possible paths using some limitations. We have set maximum length parameter in the possible path lengths which are shortestPath+3, by the way, we have taken all the paths between w1 and w2 according to this limitation. We have encoded each relation type (edge) in these paths, for example, a path between w1 and w2 of the following in Figure 1, is encoded as "BBC" where b corresponds relation type Hypernym and c corresponds relation type Hyponym. All the relation type codes can be found in Table 6.

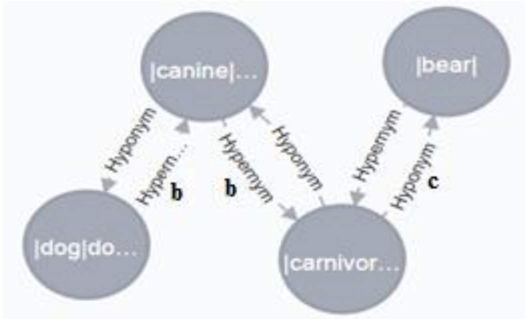

Figure 1. Sample path encoding between two concepts w1= "dog" and w2= "bear" in WordNet graph DB, encoded path= "BBC", all relation codes in the paths are given in Table 6.

By taking possible paths between each word pairs, we have gathered and encoded them into a text file, in that file we have collected more than 400k paths for the 3000-word pairs. We have assigned w1, w2 relatedness value to the all paths of two concept w1, w2.

For example, suppose *w1= "bedroom" and w2= "kitchen",* possible paths between them are taken in the following list;

findAllPath ("bedroom", "kitchen")=[bcjk, bcjk, bcjk, bcjk, bcjk, bcjk, jkbc, jkbc, jkbc, jkbc, jkbc, jkbc, bc, jk]

According to Men's data *rel("bedroom", "kitchen")=0.6*, so we assign this value into each retrieved path list for this pair; taking into account using their number of occurrences as;

Table 1. Pathlist generated for *paths("bedroom","kitchen"), rel ("bedroom", "kitchen") = 0.6* is taken from Men's dataset.

Path	Occurance	Importance factor	Value
bcjk	6	1	0.6
jkbc	6	1	0.6
bc	1	16	0.6
jk	1	16	0.6

We have also taken into account that the short path is an important path, that is the reason that we have given importance factor in the path list table, this factor is calculated as;

$$IF(path(\text{"bedroom"},\text{"kitchen"}))=n^{n-k} \quad (2)$$
(IF:importance factor)

Where n is the length of the maximum length path in the path list of the word pair, k is the length of the current path in the list. n and k values are limited to 5, the even maximum path length is greater than 5 it is fixed to 5 because increasing maximum path length will increase the importance factor exponentially and looping it in the application, hangs the program and causing exceeded memory consumption and generate out of memory exception error.

For example, the importance factor for path "*bc*" in *paths("bedroom", "kitchen");*

The first step is to find the maximum length path in the path list, so it is 4. Then path length of "*bc*" is 2, so Importance factor is faound as $4^{4-2} = 16$. Importance factor is used during the determination of the optimal path value.

3.4. Path Consolidation and Relation Weight Determination Process

In this process, optimal path value is found for all paths in the generated path pool of the Men's data. For each word pair, we have generated all possible paths including short path and assigned human judgment value to the all of the determined paths. Since there are 3k word pairs, we have collected 3k word pair list. These lists are consisting of paths in any size and any lengths. For the optimal path value determination for each path, we have grouped same paths with their corresponding human judgment values, occurrences and importance factors.

findAllPath ("ceiling", "wall")= [jcjk, jkjk, jkjk, jjk, jk, jk]

Table 2. Pathlist generated for paths("ceiling","wall"), value of rel("ceiling", "wall")=0.7 taken from Men's dataset.

Path	Occurance	Importance factor	Value
jcjk	1	1	0.7
jkjk	2	1	0.7
jjk	1	4	0.7
jk	2	16	0.7

Path consolidation is done to get optimal path value by combining each path. Consolidated paths and values for *final path("bedroom", "kitchen") and final path("ceiling", "wall")* are found as in Table 3.

Table 3 Consolidated path and value list for *paths("bedroom","kitchen")&paths("ceiling","wall")*

Path	ValueList(as ArrayList)
jcjk	[0.7]
jkbc	[0.6 0.6 0.6 0.6 0.6 0.6]
bcjk	[0.6 0.6 0.6 0.6 0.6 0.6]
jkjk	[0.7 0.7]
jjk	[0.7 0.7 0.7 0.7]
jk	[0.6…0.6(16 times 0.6) 0.7..0.7(32 times 0.7)]
bc	[0.6…0.6(16 times 0.6)]

Values in *ValueList* are calculated for each path, for example in the Table 1 value for *jk=0.6* and *occuranceXimportance factor=1X16=16,* so we add 16 times 0.6 into value list of path *jk,* in the Table2 *jk* value 0.7 and *occuranceXimportance factor=2X16=32* so we add 32 times 0.7 into value list of path *jk*.

After determining the value list for each path, the next step is to get optimal value. For this process we have used some statistical methods, we have determined optimal value in two separated manner, one for mean and other for a median of the value list. After that, we have got paths and optimal values. By scanning all the paths we filter the paths with path length=1, this means that if there is a path with length=1, it means the weight of this relation type is determined because path with length=1 means a single relationship type is weighted.

3.5. Weight Determination of Missing Relations

In the previous process, we have determined the weight of the single length paths (path length=1) in the path list. But there are some relations that are not found as a single path with path length=1, they have combined inside multiple length paths. For example, suppose there is no unique path for relation type "*x*", but there are many paths with multiple lengths like "*a*", "*jlx*", "*MBA*". In such a situation weight of the *x* determined using the known relations in the path, for example, if weight of relation *a=0.7* and value for path *xa=0.5* then value for x is found as *0.5/0.7=0.71*, if we can't determine the value of the relation by using this method, relation weight is assigned manually and manual relation's weight is given as 0.85. This value is taken from the edge counting technique of Yang and Powers (Yang and Powers, 2005). In that method, lower level relation weight is taken as 0.85.

With the above-explained processes we have extracted weight of each relation given in WordNet 3.0, just we have given manual value to the relation type *e(Instance_Hyponym), s(Member_of_this_domain_USAGE), q(Member_of_this_domain_REGION)*. That is 0.85, all other weights are determined by the algorithm.

3.6. Second Step in Relation Weight Determination

As a second step of the relation weight determination process, we applied found weight values in the first step to the Men's collection dataset. We have calculated the possible path values list for each word pair. From this list, we have taken the nearest value to the real value (human judgment) of the word pair. So we determined the nearest path for each word pair. By the way, we had a 3k path for 3k word pair. Then we take the single length paths (path length=1), the same procedure is applied as explained in the previous subsection, and optimal weights are determined.
Consequently, we had two set of the relation weights determined from the first and second steps both given in Table7. In the next chapter, we will test both of these relation weights set and evaluate the determined scores with benchmark data.

3.7. Development Environment

For the development of the proposed system, we have used the Java environment and Neo4j graph db. Using developed java codes, WordNet 3.0 data is imported into Neo4j DB in a graph structure. All queries are performed on the Neo4j database using cipher query language. Shortest paths between two nodes and all possible paths between two nodes have been determined through cipher queries. The operating system of the development environment is windows 7 professional 64-bit virtual machine with 2 core CPU and 8 GB memory.

4. Evaluation & Benchmark Tests

Using the determined weights of each relation, we tested RG65 real-world dataset, we have collected all possible paths for each word pair in this dataset, by the way we have got a list of the possible paths of each word pairs, and paths in a list are grouped using importance factor and number of occurrence for each path. We have calculated the value of each path in the list using given relation weights and then get the mean and median of the list one by one, found value from mean or average for each word pair that represents the relatedness value of the corresponding word pairs. In order to make a benchmark of the determined related values with the real RG65 values, we have used Spearman rank correlation because this method has used in most of the relatedness studies for the evaluation of the benchmark tests.

Table 4. Correlation values in the benchmark tests using RG65 data with mean and median of path value lists determined in step 1 and step 2.

	Correlation using means of path value list	Correlation using the median of path value list
Relation weights determined in first step	0.81	0.78
Relation weights determined in second step	0.79	0.74

We get 0.81 correlation using RG65 real values with relation weights found in the first step of the proposed method. In this step, we have statistically take the mean of the path values. In the first step, we take the median of the path values and we have found 0.79 correlation. Weights found in the second step are not better than the values found in the first step. We might comment that weights found in the first step are better promising. The second step of the method has performed in order to see whether we can get a better correlation. We get 0.78 correlation value using the relation weights found in the first step and taking the median of the path value list for each word pair. We continued testing the relation weight data taken in the second step. We get 0.79 correlation using the relation weights found in the second step and taking the mean of the path value list for each word pair. We get 0.74 value using the relation weights found in the second step and taking the median of the path value list for each word pair.

5. Results and Discussion

This study might give a new perspective to the relatedness computation of two concepts in WordNet; even success of this approach seems not satisfied in comparison to the other WordNet-based approaches shown in Table 5. The method uses simple mathematical and statistical operations. Optimal path determination process and weight determination of missing relations are difficult in terms of algorithm generation and software development.

Table 5. Spearman Correlation values of several successful WordNet Based relatedness methods using RG65 dataset

Method	**Correlation**
ADW (Pilehvar, Navigli 2015)	0.868
Hughes and Ramage (2007)	0.838
Agirre at al. (2009)	0.830
Wu and Palmer (1994)	0.78

Given approach in this study is simple, only complex and time-consuming part is a determination of the possible paths between two concepts in the WordNet. In the Neo4j graph database, shortest path determination is easy using a cipher query language, but determining all the possible lengths in a limited path length takes several minutes for any two synsets because graph database is consist of the 117k node and more than 700k relations, querying in such a graph database takes time. This process takes more than ten hours for the 3k word pairs in Mens dataset which is used for the development of this study. After path determination, rest of the method is not time-consuming part, just we find the values of each path by multiplying the relation types in the path, then found results are put into array list and their mean or median is calculated easily.

Another aspect we have taken from this study for the future studies is that; we have generated path list and optimal values in the first phase of the study; by this approach, we might have a path database of the human judgments. We can extend this database using other real-life similarity datasets. In the new coming studies, we can detect the optimum path of two concepts and extract the optimum path value from this database, doing this makes the similarity/relatedness computation faster and more accurate.

Future studies about this approach might be done using other metrics, not only path length but also the depth of the concepts, depth of the lowest common subsumer (LCS) of the concepts. Contributions of such parameters to the relatedness computation might be in positive and this might increase the success of proposed method and bring it into the top of relatedness methods list is given in Table5. We set weight value 0.85 for the relation types that cannot be determined our algorithm, by changing this value in upward or downward we can observe the contributions of this coefficient to the method in a positive or negative way. Moreover, information content value that might be taken from an external corpus about the concepts and can be added to the nodes in the path. By the way, computation would be more complex; the success of this hybrid approach also might give promising results.

6. Conclusion

This study uses WordNet as source and takes the Men’s data to determine the weights of the WordNet relations. Basic mathematical and statistical operations are used to determine and generalize the relation weights. Using these relations weights, the relatedness of the two concepts defined in WordNet is measured. Relatedness score is produced between [0,1]. Evaluated using RG65 that is a widely used benchmark dataset for semantic relatedness, found Spearman Correlation value is 0.81, in the literature, there are other methods outperforms this values. Most successful WordNet-based method’s correlation value is 0.86 (Pilehvar and Navigli, 2015). By improving this method using some other features of WordNet’s graph structure, this method might be improved and might generate more successful correlation values. It is a new approach for weighting all relation types instead of grouping or categorizing the relations. This study is measuring relatedness for two lexical items in any relation type in WordNet since most of the relatedness study uses limited relation types; this is the advantage of the study. Since most of the studies measure the relatedness of noun pairs, this study measures the lexical items without part of speech information; this is also another positive part of the study.

Acknowledgment

This study is a part of the research programme with project number 212E526, which is financed by the Scientific and Technological Research Council of Turkey (TUBITAK).

Table 6. Relation type, relation code, and determined relation weight in each step

Relation Type	Rel. Code	Determined Weight	
		Step1	Step2
Antonym	a	0.7	0.7
Hypernym	b	0.8	0.82
Hyponym	c	0.76	0.76
Instance_Hypernym	d	1.39	1.36
Instance_Hyponym	e	0.85	0.85
Member_Holonym	f	0.97	1.14
Member_Meronym	g	1.05	0.85
Substance_Holonym	h	1.07	1.1
Substance_Meronym	i	0.99	0.85
Part_Holonym	j	0.83	0.84
Part_meronym	k	0.78	0.8
Attribute	l	0.95	0.98
Derivationally_related_form	m	0.84	0.86
Domain_of_synset_TOPIC	n	0.90	1.02
Domain_of_synset_REGION	p	0.75	0.9
Member_of_this_domain_REGION	q	0.85	0.85
Domain_of_synset_USAGE	r	0.77	0.86
Member_of_this_domain_USAGE	s	0.85	0.85
Entailment	t	0.79	0.79
Cause	u	0.78	0.72
Also_see	v	0.82	0.95
Verb_Group	x	1.16	1.12
Similar_to	w	0.88	0.89
Participle_of_verb	y	1.44	0.85
Pertainym	z	1.48	1.30
No_Path_Determined*	%	0.21	0.21
Synonym*	&	0.84	0.79
PertainedBy*	!	0.85	0.85

*This relation types are not given in WordNet 3.0, it is created by the algorithm.

Table 7. RG65 benchmark dataset and determined score by a proposed algorithm using relatedness weight determined in the first step with a mean of the path length values (Spearman rank correlation is 0.81).

Word1	Word2	Rel Scr RG65	Rel. Score By Our Method
cord	smile	0.01	0.28
rooster	voyage	0.01	0.14
noon	string	0.01	0.25
fruit	furnace	0.01	0.26
autograph	shore	0.02	0.21
automobile	wizard	0.03	0.36
mound	stove	0.04	0.28
grin	implement	0.05	0.29
asylum	fruit	0.05	0.27
asylum	monk	0.10	0.24
graveyard	madhouse	0.11	0.16
glass	magician	0.11	0.30
boy	rooster	0.11	0.20
cushion	jewel	0.11	0.37
monk	slave	0.14	0.52
asylum	cemetery	0.20	0.24
coast	forest	0.21	0.46
grin	lad	0.22	0.35
shore	woodland	0.23	0.61
monk	oracle	0.23	0.34
boy	sage	0.24	0.25
automobile	cushion	0.24	0.62
mound	shore	0.24	0.41
lad	wizard	0.25	0.34
forest	graveyard	0.25	0.28
food	rooster	0.27	0.20
cemetery	woodland	0.30	0.30
shore	voyage	0.31	0.31
bird	woodland	0.31	0.33
coast	hill	0.32	0.48
furnace	implement	0.34	0.23
crane	rooster	0.35	0.30
hill	woodland	0.37	0.61
car	journey	0.39	0.43
cemetery	mound	0.42	0.34
glass	jewel	0.45	0.26
magician	oracle	0.46	0.59
crane	implement	0.59	0.35
brother	lad	0.60	0.33
sage	wizard	0.62	0.30
oracle	sage	0.65	0.39
bird	crane	0.66	0.57
bird	cock	0.66	0.76
food	fruit	0.67	0.41
brother	monk	0.69	0.62
asylum	madhouse	0.76	0.76
furnace	stove	0.78	0.65
magician	wizard	0.80	0.79
hill	mound	0.82	0.81
cord	string	0.85	0.75
glass	tumbler	0.86	0.76
grin	smile	0.87	0.79

References

Aguirre, E. and Soroa, A., 2009. Personalizing PageRank for Word Sense Disambiguation. Proceedings of the 12th conference of the European chapter of the Association for Computational Linguistics, March 2009, Athens, Greece, p. 34-41.

Ahsan, M.G., Naghibzadeh, M. and Naeini, S.E.Y., 2014. Semantic similarity assessment of words using weighted WordNet, Int. J. Mach. Learn. & Cyber, 5 (3), 479-490, https://doi.org/10.1007/s13042-012-0135-3.

Brin, S. and Page, L., 1998. The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 30, 107-117.

Bruni, E., Tran, N. K. and Baroni, M., 2014. Multimodal Distributional Semantics. Journal of Artificial Intelligence Research, 49, 1-47, https://doi.org/10.1613/jair.413.

Fellbaum, C., 1998. WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press, 422p.

Finkelstein, L., Gabrilovich, E., Matia,S., Y., Rivlin, E., Solan, Z., Wolfman, G. and Ruppin, E., 2001. Placing search in context: The concept revisited. In WWW '01: Proceedings of the 10th international conference on World Wide Web, May 2001, Hong Hong, p. 406–414.

Hirst, G. and St-Onge, D, 1998, Lexical chains as representations of context for the detection and correction of malapropisms, in WordNet: An Electronic Lexical Database, MITP, p. 305–332.

Hughes, T. and Ramage, D., 2007. Lexical semantic relatedness with random graph walks. Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP–CoNLL, June 2007, Prague, Czech Republic, p.581–589.

Kartsaklis D., Pilehvar M.T. and Collier N., 2018. Mapping Text to Knowledge Graph Entities using Multi-Sense LSTMs. EMNLP, Oct. 2018, Brussels, Belgium.

Li, Y., Zuhair, A. B. and McLean, D., 2003. An approach for measuring semantic similarity between words using multiple information sources, IEEE Trans. Knowledge and Data Eng., 15 (4), 871-882.

Meng, L., Gu, J. and Zhou, Z., 2012. A New Hybrid Semantic Similarity Measure Based on WordNet. In: Lei J., Wang F.L., Li M., Luo Y. (eds) Network Computing and Information Security. Communications in Computer and Information Science. 345, 739-744. https://doi.org/10.1007/978-3-642-35211-9_93.

Miller, G. A. and Charles, W. G., 1991. Contextual Correlates of Semantic Similarity. Language and Cognitive Processes, 6 (1), 1-28.

Mikolov T., Chen K., Corrado G. and Dean J., 2013. Efficient estimation of word representations in vector space. CoRR, abs/1301.3781.

Navigli R. and Ponzetto S., 2012. BabelNet: The Automatic Construction, Evaluation, and Application of a Wide-Coverage Multilingual Semantic Network. Artificial Intelligence, 193, P. 217-250.

Pennington J., Socher R. and Christopher D.M., 2014. GloVe: Global vectors for word representation. Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014), October 2014, Doha, Qatar, 12, 1532–1543.

Pilehvar, M. T. and Navigli, R., 2015. From Senses to Texts: An All-in-one Graph-based Approach for Measuring Semantic Similarity. Artificial Intelligence, 228, 95-128.

Rubenstein, H. and Goodenough, J.B., 1965. Contextual correlates of synonymy, Communications of the ACM, 8 (10), 627–633

Siblini, R. And Kosseim, L., 2013. Using a Weighted Semantic Network for Lexical Semantic Relatedness. Proceedings of Recent Advances in Natural Language Processing, Sept 2013, Hissar, Bulgaria, p.610-618

Speer R. and Chin J., 2016. An ensemble method to produce high-quality word embeddings. arXiv preprint arXiv:1604.01692

Speer, R. and Havasi C., 2012. Representing General Relational Knowledge in ConceptNet 5,. Proc. of LREC, May 2012, Istanbul Turkey, p. 3679-3686.

Sultan, A.M., Bethard, S. and Sumner, T., 2014. Back to basics for monolingual alignment: exploiting word similarity and contextual evidence, Trans. Assoc. Comput. Linguist., 2 (1), 219–230.

Wu, Z. and Palmer, M., 1994. Verb semantics and lexical selection. the 32nd Annual Meeting of the Association for Computational Linguistics, June 1994, New Mexico USA, p.133–138.

Yang, D. and Powers, M.V., 2005. Measuring semantic similarity in the taxonomy of WordNet. Proceeding of the 28th Australasian Computer Science Conference, Jan 2005, Newcastle, Australia, p. 315-332.