Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118

RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

Corpus-dependent concept graphs using Wikipedia: A probabilistic approach

Han-joon Kim, Ga-Hui Lee

School of Electrical and Computer Engineering, University of Seoul, Seoul 02504 KOREA

Abstract. Introduction: This paper proposes a novel way of building concept graphs by probabilistically analyzing connected hyperlinks among Wikipedia articles. This paper defines a single concept using its corresponding Wikipedia article, and we use a web search engine named *Elasticsearch* to locate the concept-level Wikipedia articles for a particular document corpus. **Methodology:** The hierarchical relationships of isolated concepts are determined by computing the subsumption probabilities between incoming and outgoing hyperlinks, which are internally represented as a partial ordering matrix. To find more precise relationships among concepts, the proposed method uses the weights of concepts that result from the number of incoming and outgoing hyperlinks. Specifically, to estimate the topological weights of concepts, we devised a special weighting function that mainly uses the number of hyperlinks among Wikipedia articles. Then, the weights of concepts are used to calculate the probabilistic subsumption among concepts, which allows the generation of relatively stable concept graphs. **Results & Conclusion:** Through extensive experiments with 'OHSUMED' and 'GoogleNews' corpora, we show that our proposed method is superior to conventional methods, on average, by more than 30% in terms of the taxonomic F_1-measure. Our prototype system can produce very reasonable concept graphs that contain both noun-level and proper noun- level concepts. Such document corpus-dependent concept graphs can be used as a knowledge base for developing various text-mining applications.

Keywords: Graphs, Information Retrieval, Indexing, Probability, Text Mining, Wikipedia

1. Introduction

In the era of big data, a concept graph is a crucial data structure that is used to develop semantic search and text-mining systems [1, 2]. The nodes of the graph correspond to a universe of concept phrases and the edges correspond to the relationships between them. When building a concept graph, the most important issue is how to determine hierarchical or associative relationships among a given set of concepts [3]. In order to compose a set of concepts hierarchically, we must determine whether a particular concept node is higher or lower than other concept nodes in a concept graph. In this regard, [4] proposed a simple subsumption model in which a set of terms occurring in documents are used to describe concepts, and the hierarchical relationship between two terms is determined by computing the probability that two different sets of documents with identical terms are subsumed with each other. Another similar method of building concept graphs was proposed in [5], which used Flickr tag vocabulary for building concept graphs with a subsumption-based model. However, these conventional methods for building a concept graph are not very practical when we need to select concept-level terms manually from textual documents. Basically, the concepts that participate in a concept graph should carry semantic information suited for their titles; otherwise, some relationships among concepts are difficult to derive accurately. Here, to resolve this problem, we use the world knowledge named Wikipedia, as in [6]. In this paper, we regard each Wikipedia article as a single concept, and propose a new way of generating hierarchical relationships among concepts depending on a given document corpus. To isolate useful concepts automatically, we use the web search engine *Elasticsearch* [7] to isolate concept-level articles from large Wikipedia articles. Then, to estimate the topological weights of concepts, we devised a

special weighting function that mainly uses the number of hyperlinks among Wikipedia articles. The weighting function is effectively used for calculating the probabilistic subsumption among concepts, which allows the generation of relatively more stable concept graphs. Ultimately, a set of concept pairs with hierarchical relationships is visualized as a directed acyclic graph (DAG) [8]. We proved that the proposed method outperforms representative conventional methods and that it can automatically extract concept graphs with high accuracy. The remainder of this paper is organized as follows. Section 2 describes previous studies on concept graphs. Section 3 details our proposed probabilistic method of building concept graphs. Section 4 describes the results of experiments designed to assess the effectiveness of our method. Section 5 concludes the paper and outlines directions for future work.

2. Related Work

In [4], Sanderson *et al*. proposed a simple statistical model to determine the hierarchical relationships among terms. They called the hierarchical relation between two terms the 'term subsumption relation', which is defined as follows, based on experimental results in [4]: for two topical terms t_i and t_j, if $\Pr(t_i|t_j) \geq 0.8$ and $\Pr(t_i|t_j) > \Pr(t_j|t_i)$, then t_i is said to subsume t_j. Here, $\Pr(t_i|t_j)$ is the probability that t_i occurs in the document set in which t_j occurs. With such topical terms, term relations can be determined by the 'Document Frequency (DF) hypothesis': "*The more documents a term occurs in, the more general the term is assumed to be*". This hypothesis means that the generality and specificity of terms can be determined only by the number of documents that contain the terms. In our experiment, however, this model failed to identify the hierarchical relationships among concept-level terms occurring in Wikipedia articles. This might be because the document frequency alone is insufficient to express the semantics of terms. Nevertheless, we use the subsumption model to interpret hyperlink information among Wikipedia articles probabilistically.

In [5], Shmitz *et al*. proposed a way of producing concept graphs from Flickr image tag vocabulary by improving the subsumption-based model of [4]. Here, each tag is defined as a single concept, and the hierarchical relationships among tag concepts are computed using the following equations: if $\Pr(x|y \geq t)$ and $\Pr(y|x < t)$, $D_x \geq D_{min}$, $D_y \geq D_{min}$, $U_x \geq U_{min}$, $U_y \geq U_{min}$, then the tag term *x* is said to subsume *y*, where *t* is the co-occurrence threshold, D_x is the number of documents in which term *x* occurs, which should be greater than D_{min}, and U_x is the number of users that use *x*, which should be greater than U_{min}. The refined subsumption model focuses on removing inappropriate tag terms, such as incorrectly written words, slang, abbreviations, and Flickr-cultural curiosities. Nevertheless, the model still could not generate reasonable concept graphs in our experiment.

A recent approach to building concept graphs is to use Wikipedia articles [6]. In this respect, one of our previous studies introduced a way of building concept graphs by analyzing incoming hyperlinks among Wikipedia articles [9]. Basically, this method adopts Sanderson's subsumption model, although it uses the number of incoming links instead of document frequency. That is, by regarding a Wikipedia article as a single concept, the information on term occurrence in Sanderson's subsumption model is replaced by the information on term citation from other articles. Suppose there is a hyperlink from concept c_i in a particular anchor text to another concept article, c_j. Then, we can determine the hierarchical relationship between the two concepts using the following equations:

$$\Pr(c_i|c_j) = \Pr\left(I(c_i) \supset I(c_j)\right) = \frac{|I(c_i) \cap I(c_j)|}{|I(c_j)|} \quad (1)$$

$$0.1 \leq |\Pr(c_i|c_j) - \Pr(c_j|c_i)| \leq 1.0, \qquad \Pr(c_i|c_j) > \Pr(c_j|c_i) \quad (2)$$

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

where $I(c_j)$ denotes the set of incoming links of concept c_j. If Equation (2) is satisfied, then concept c_i is evaluated as higher (or more general) than concept c_j. However, the method allows the probability $\Pr(c_i|c_j)$ to be too sensitive to the number of incoming links; for example, if concepts c_a, and c_b have 1,000 and 10,000 incoming links, respectively, with $I(c_a) \cap I(c_b) = 700$, then $\Pr(c_a|c_b) = 0.07$, $\Pr(c_b|c_a) = 0.7$. In order for a particular concept to be ranked relatively high, the concept should always have more incoming links than lower concepts. In practice, lower concept articles can sometimes have more incoming links than higher concept articles. To overcome this problem, we intend to add a new factor called 'topological weight' to Equations (1) and (2).

3. Automated Development of Concept Graphs

As mentioned above, concept graphs are built in two phases, as shown in Figure 1. First, the method extracts candidate concepts from a given document corpus. For this, it is necessary to isolate noun- or adjective-level keywords in each document and the keywords are used as query words to search for their corresponding concepts (*i.e.*, Wikipedia articles). Then, the method determines hierarchical relationships among concepts. In our work, to evaluate more accurate relationships, each concept article is given a topological weight, and subsumption relationships between two concepts are computed probabilistically using hyperlink information.

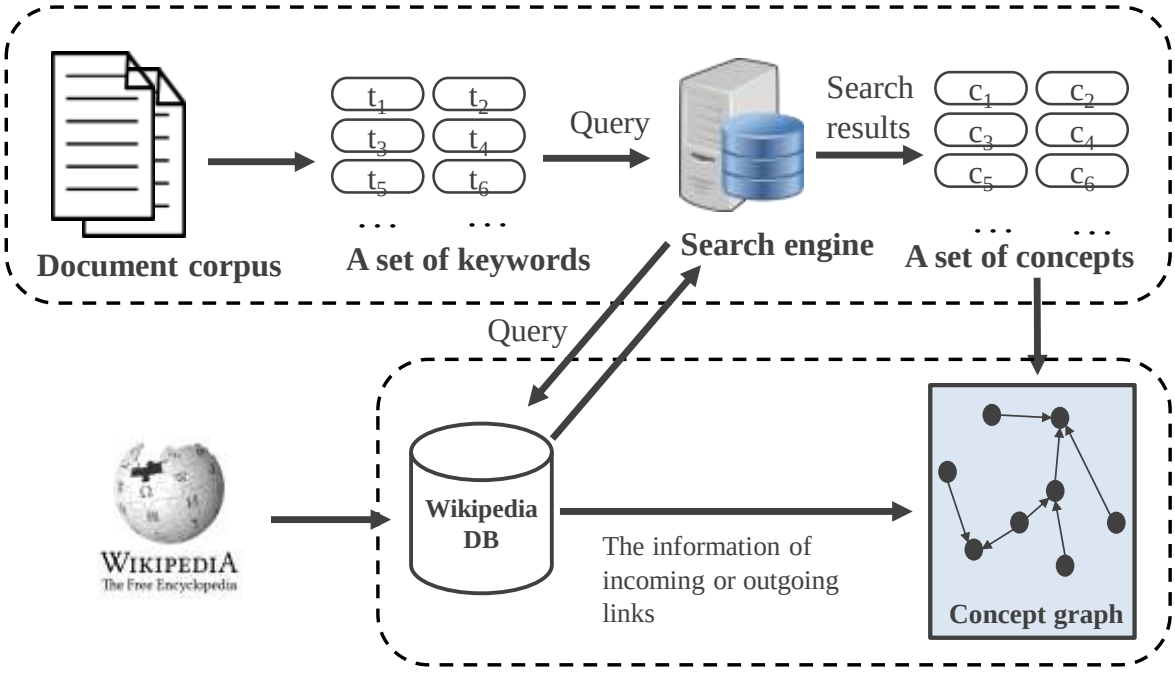

Figure. 1. The process used for building the corpus-dependent concept graphs

To build up a corpus-dependent concept graph from Wikipedia articles, we make the following assumptions:

- A single Wikipedia article defines its corresponding concept whose name is given by the title of the Wikipedia article.

- The anchor text occurring in Wikipedia articles contributes to the semantics of concepts.
- The Wikipedia article to which any anchor text points includes a high-quality description that defines a particular concept.

3.1 Conceptualization of Terms in the Document Corpus

The method used to define 'concepts' from a set of documents has a strong impact on the quality of concept graphs. Here, each concept is defined using each Wikipedia article, and its semantic quality should be reasonably high. To build a reasonable graph from a given document corpus, we perform the following two steps. First, we conceptualize the terms occurring in a given document corpus through a search engine. Then, we extract the top *N* concepts to represent the document corpus. This step corresponds to automatic concept filtering. For this, we used the search engine *Elasticsearch* [7], which is an open source search engine started by Shay Banon in 2010. Many people use it as a document database because of its distributed nature and real-time abilities [10]; that is, it is possible to use it as a Wikipedia article database including a set of concepts. Then, for each document, the search engine tries to identify its relevant concepts (*i.e.*, Wikipedia articles) by regarding a document itself as a user query. Figure 2 illustrates an actual example of term conceptualization for a given document. Specifically, to identify the concepts hidden in each document in a test dataset, we first performed a tokenization step that produces a form of text segmentation unit. Then, through POS tagging, only noun and adjective tokens are chosen, since significant phrases (*e.g.*, big data) are used as queries. All of the chosen tokens (as shown in Figure 2(b)) are submitted as query words to the *Elasticsearch* engine, which then returns their corresponding concept-level Wikipedia articles, as shown in Figure 2(c).

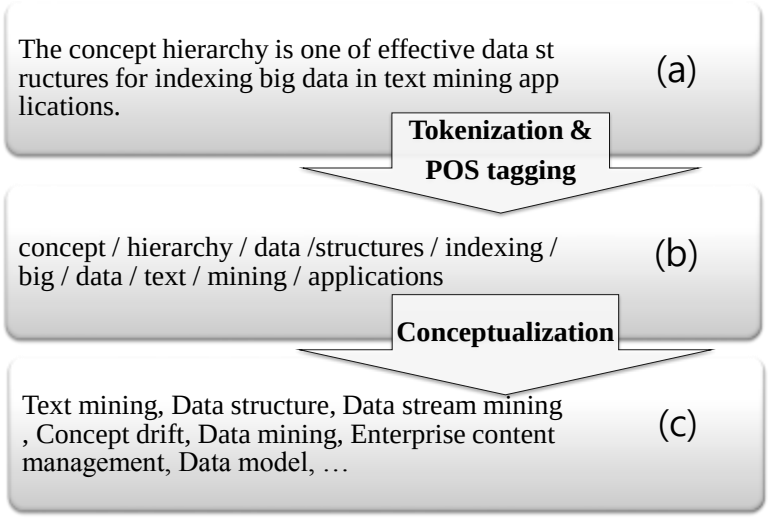

Figure. 2. An example of term conceptualization

Figure 3 shows the search result of the *Elasticsearch* engine for term conceptualization. The figure indicates that 62,775 concepts (*i.e.*, Wikipedia articles) were derived from the given query, as shown in the 'total' section. In addition, an extracted Wikipedia concept and its relevance to the query are given in the 'title' and 'score' sections. The score is determined using the *TF-IDF*

similarity scoring formula of Lucene [11], which is used in our proposed scoring function for isolating significant quality concepts.


```
{
  "took" : 29,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "failed" : 0
  },
  "hits" : {
    "total" : 62775,
    "max_score" : 0.6689206,
    "hits" : [ {
      "_index" : "wikisubsets",
      "_type" : "wikisubset",
      "_id" : "a8gaglBzQZCeZhCiC4owrw",
      "_score" : 0.6689206,
      "fields" : {
        "title" : [ "Text Mining" ]
      }
    },  {
      "_index" : "wikisubsets",
      "_type" : "wikisubset",
      "_id" : "3cVKIttCRqGmYYm9yOThlQ",
      "_score" : 0.5359975,
      "fields" : {
        "title" :  [ "Data structure" ]
      }
    },  {
```


Figure. 3. An example of a search result using *Elasticsearch*

Building a good concept graph requires that only useful concepts in a given corpus be detected. However, it is difficult to determine good-quality concepts directly using conceptualized terms. Therefore, we propose a variant of the *TF-IDF* weight function called *CF-ICDF* to identify meaningful concepts relevant to each document. The *CF-ICDF* value of concept *c* in document *d* is defined as follows:

$$CF - ICDF(c,d) = log(1 + CF(c,d)) \cdot log\left(\frac{N}{DF(c)}\right) \cdot score(c,d) \quad (3)$$

where *CF*(*c*, *d*) denotes the frequency of concept *c* in document *d* and *N*/*DF*(*c*) the inverse document frequency considering concept *c* in *N* documents. Another factor, *score*(*c*,*d*), accounts for the weight of concept *c* for the document *d*, which corresponds to the similarity value appearing in the 'score' section of the search result in Figure 3. Concepts with larger *CF-ICDF* values in a given document are increasingly major concepts. As many *CF-ICDF* values are calculated as there are concepts. Sometimes a single concept has more than one *CF-ICDF* value. The *CF-ICDF* value of a concept with multiple *CF-ICDF* values is set to the average of its *CF-ICDF* values. In (2), we show that this *CF-ICDF* value serves as the weight of concept *c*.

3.2 Building Concept Graphs using Wikipedia Links

After isolating major concepts from a given document corpus, we define the hierarchical relationships among concepts. First, we consider the characteristics of inter-connected Wikipedia articles that are each defined as a single concept. The Wikipedia concept with more incoming links is rated as more important, as in the PageRank algorithm [12]. To compose hierarchical relationships

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

promptly, we use two types of link: incoming and outgoing links. An incoming link is any link that is received by the target article from other articles and an outgoing link is any link that is referenced by the target article to other articles. Simply, we may rank all of the concepts with the equation |I(*c*)|–|O(*c*)| for concept *c*. Here, to build more robust hierarchical relationships, we weight concept *c* as follows:

$$w(\mathrm{c}) = \frac{log\ (||\mathrm{I(c)}| - |\mathrm{O(c)}||)}{\max_{\mathrm{c}\in\mathrm{T}} log(||\mathrm{I(c)}| - |\mathrm{O(c)}||)} \cdot log(|\mathrm{I(c)}|) \quad (4)$$

where I(*c*) (or O(*c*)) denotes the set of incoming (or outgoing, respectively) hyperlinks, and *T* the set of candidate concepts derived from the given corpus. Here, multiplying *log* (||I(*c*)|-|O(*c*)||) with *log*(|I(*c*)|) is for determining more robust hierarchical positions of concepts. Now we describe what hierarchical relationships the extracted concepts have by using the weight *w*(*c*). Still, Equation (4) accounts for only the absolute weight value of each concept. With the concepts derived from Equation (4), we thus develop a special function to calculate the probabilistic subsumption between two concepts. To evaluate the hierarchical relationship between two concepts c_i and c_j, we have devised the following probabilistic equation so as to obtain clearer hierarchical relationships.

$$Hr(c_i|c_j) = Pr(c_i|c_j) \cdot w(c_i) \quad (5)$$

$$|Hr(\mathrm{c_i}|\mathrm{c_j}) - Hr(\mathrm{c_j}|\mathrm{c_i})| \geq \delta, Hr(\mathrm{c_i}|\mathrm{c_j}) \geq Hr(\mathrm{c_j}|\mathrm{c_i}) \quad (6)$$

where $Pr(c_i|c_j)$ is the probability that concept c_i subsumes concept c_j. We consider the hierarchical relationship of a concept pair $<c_i, c_j>$ with weight $w(c_i)$, as well as the probability $Pr(c_i|c_j)$. Equation 6 compares $Hr(c_i, c_j)$ with $Hr(c_j, c_i)$ to determine whether a given concept pair becomes a hierarchy relationship; that is, if $Hr(c_i, c_j) > Hr(c_j, c_i)$, then c_i is specified as an upper concept of c_j. Here, the difference between $Hr(c_i, c_j)$ and $Hr(c_j, c_i)$ should exceed a threshold value, δ. Note that a larger difference in the *Hr* values of two concepts means that they have a stronger hierarchical relationship. Ultimately, the *Hr* values among concepts are internally represented as a partial ordering matrix. Moreover, to visualize the structure of concepts, we used a DAG in which cycles of concepts do not occur and a concept may have one or more higher concepts. Here, concepts with a hierarchical relationship are expressed as 'higher concept ← lower concept'.

4. Experiments

4.1 Empirical Setup

This section addresses the performance evaluation and visualization of concept graphs for the proposed method. As a source of concepts, we used the English Wikipedia as of March 2017, which contains about 5.3 million articles. Specifically, we isolated 1,600,000 concept-level Wikipedia articles and extracted various properties that can be used to construct concept graphs from Wikipedia, which was stored in the *Elasticsearch* engine. As document corpora for building the corpus-dependent concept graphs, we adopted the 'OHSUMED' and 'GoogleNews' corpora. The OHSUMED corpus comes from the on-line medical information database MEDLINE, which contains titles and abstracts from 270 medical journals; it has been used for testing text-classification techniques [13]. We built the GoogleNews corpus to evaluate our proposed method. For convenience, for the OHSUMED corpus, we selected only the set of documents belonging to categories 'C02' (about virus diseases) and 'C11' (about eye diseases). For the GoogleNews corpus, we used only the set of news about the categories 'Google' and 'IoT'. Table 1 shows the details of these test datasets.

Table 1. Test datasets

Document corpus	Categories	Number of documents	Number of concepts
OHSUMED	C02 (virus diseases)	1,166	300
	C11 (eye diseases)	993	300
GoogleNews	Google	500	50
	IoT	500	50

Next, the concepts found in Table 2 were generated through term conceptualization (See Section 3.1) with each corpus. Finally, we generated the hierarchical relationships of the top N (say 50 or 300) concepts from the Wikipedia corpus; consequently, we built a concept graph dependent on the given document corpus.

Table 2. The list of concepts extracted with the OHSUMED and GoogleNews corpora

Document corpus	Categories	Concepts extracted
OHSUMED	C02 (virus diseases)	Hyperprolactinaemia, Gardasil, HPV vaccine, Cervarix, Autoimmune hepatitis, Influenza vaccine, Measles vaccine, Corneal endothelium, MMR vaccine, Hepatitis B vaccine, HIV vaccine, HDV, Varicella vaccine, Hepatitis B, Common cold, Hepatitis D, *etc*.
	C11 (eye diseases)	Glaucoma, Peroxisome, Dialysis, Ganglioside, Gastrin, Plasmin, Vitamin K, Hypothyroidism, Galactose, Magnesium chloride, Eye, *etc*.
GoogleNews	Google	Google, Online advertising, Web search engine, Cloud computing, Software, Larry Page, Gmail, PageRank, Google Search, Algorithm, World Wide Web, Chrome OS, Android Dev Phone, *etc*.
	IoT	Internet of Things, Embedded system, Internet, Machine to machine, Smart objects, Smart grid, Biochip, Ambient intelligence, Buzzword, Emerging technologies, Computer science, Computer engineering, Smartphone, Home automation, Web service, Web of Things, *etc*.

4.2 Performance Measures

We attempted to evaluate how well the generated concept graphs emulated some of the properties of the manually constructed concept graphs, although their quality is difficult to measure objectively. To this end, we use *precision* and *recall* (commonly used in information retrieval), which are called taxonomic precision (TP) and taxonomic recall (TR) in our work, respectively. When we denote the set of discovered relationships by $T' \subset C \times C$ (where *C* is a set of concepts) and $T \subset C \times C$ is the set of true hierarchical relationships, the two measures are defined as follows:

$$TP = \frac{|T' \cap T|}{|T'|}, \tag{7}$$

$$TR = \frac{|T' \cap T|}{|T|}, \tag{8}$$

Finally, we compute their combined measure, called the taxonomic F_1-measure (TF), which gives equal weight to both recall and precision; this measure ranges from 0 to 1, and is proportional to the effectiveness of the constructed concept graph.

$$\mathrm{TF} = \frac{2\times \mathrm{TP}\times \mathrm{TR}}{\mathrm{TP}+\mathrm{TR}}, \tag{9}$$

Here, the true hierarchical relationships are obtained from MeSH Tree Structures (*https://www.nlm.nih.gov/mesh/trees.html/)* and the Open Directory Project (ODP) (http://dmoz.org/). The MeSH Tree Structures contain hierarchical relations for approximately 15,000 medical terms created by the National Library of Medicine (NLM), and were used to evaluate the concept graphs for the OHSUMED corpus. The ODP has been rated an exemplary, comprehensive Web directory, in which more than one million categories (*i.e.*, concepts) are hierarchically organized under 16 top-level categories. Note that ODP has a DAG structure, since one concept node can have more than one parent concept node using symbolic links.

4.3 Visualization Results

For performance evaluation, it is difficult to evaluate the accuracy of the concept graphs produced since there are no standard test data for evaluating concept graphs. This is especially true because our proposed method considers proper-noun-level concepts. In this section, we evaluate the concept graphs produced in a qualitative manner by inspecting the visualization results of the concept graphs.

Figures 4 and 5 show the concept graphs for the OHSUMED corpus. Figure 4 depicts the concept graph with the hierarchical relationships among a number of concepts about virus diseases, in which the concepts about ‘Respiratory viruses’, ‘Helminths’, ‘Brain viruses’, ‘Digestive viruses’, and ‘Antigens’ form separate clusters. By contrast, Figure 5 depicts the concept graph for eye diseases, in which the concepts about specific eye-related diseases are connected, focusing on the concepts ‘Eye’ and ‘Glaucoma’. This is because this concept graph contains much more specific concepts than in the case of Figure 4.

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

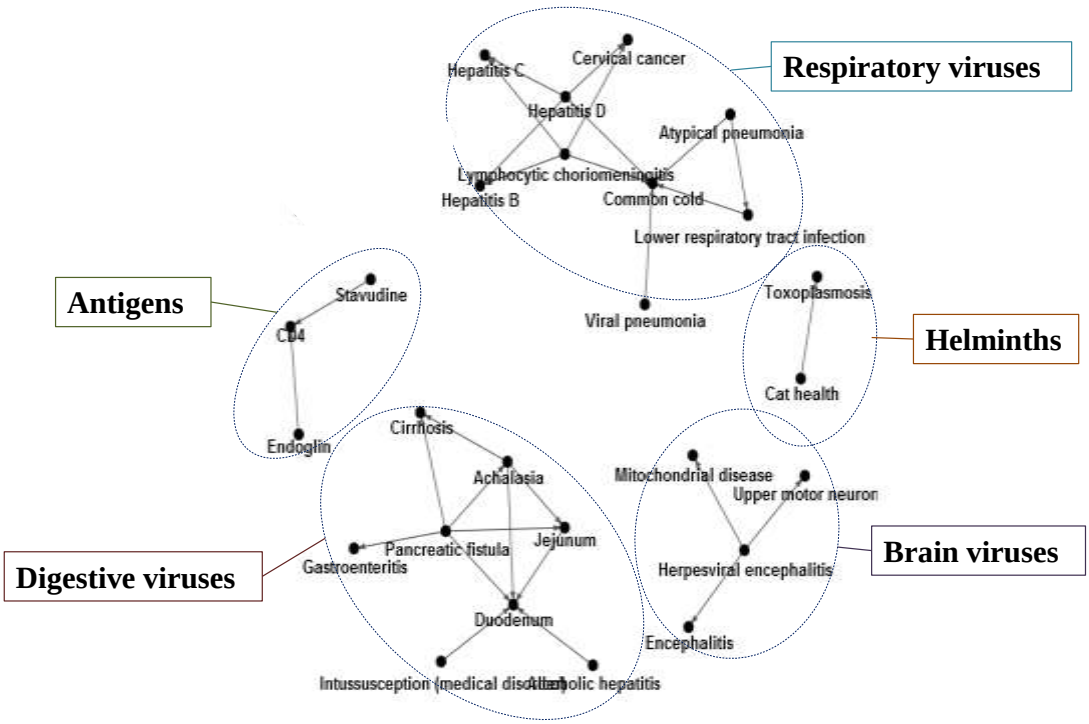

Figure. 4. Part of the concept graph for the category 'C02' (about virus diseases) in the OHSUMED corpus (when δ =1.9)

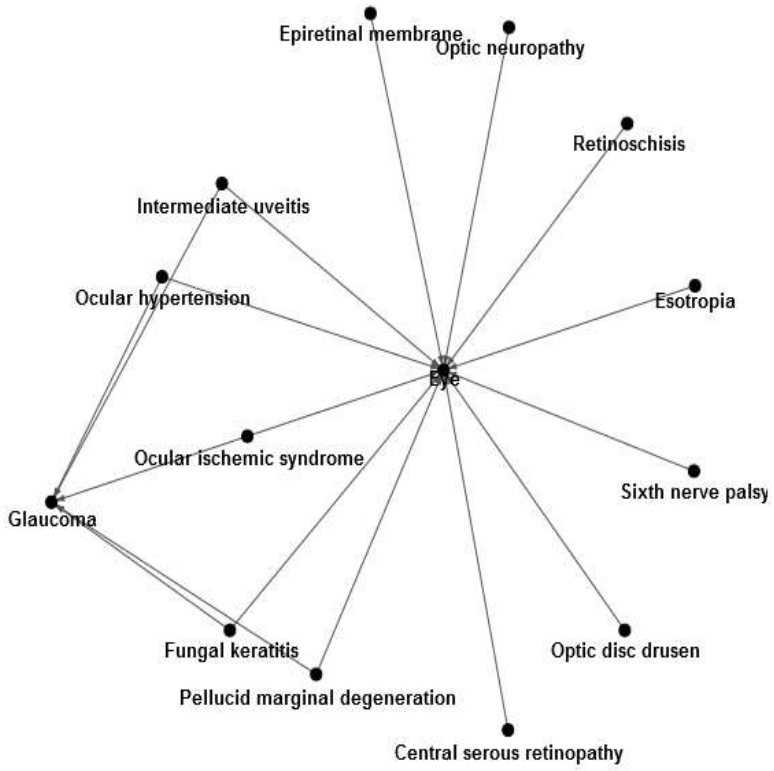

Figure. 5. Part of the concept graph for the category 'C11' (about eye diseases) in the OHSUMED corpus (when δ =1.9)

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

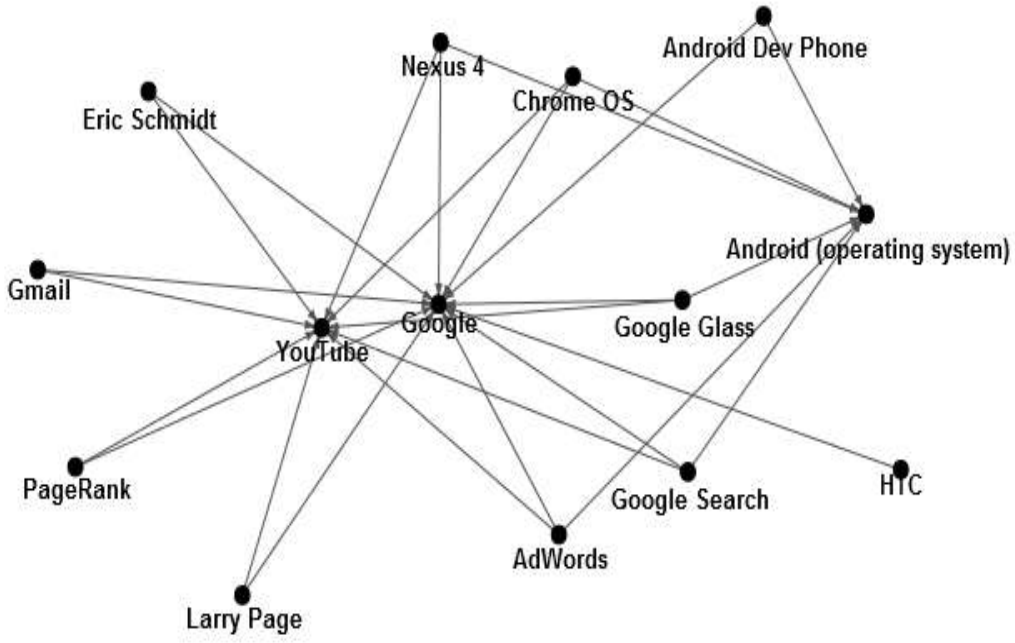

Figure. 6. Part of the concept graph for the category 'Google' in the GoogleNews corpus (when δ =1.4)

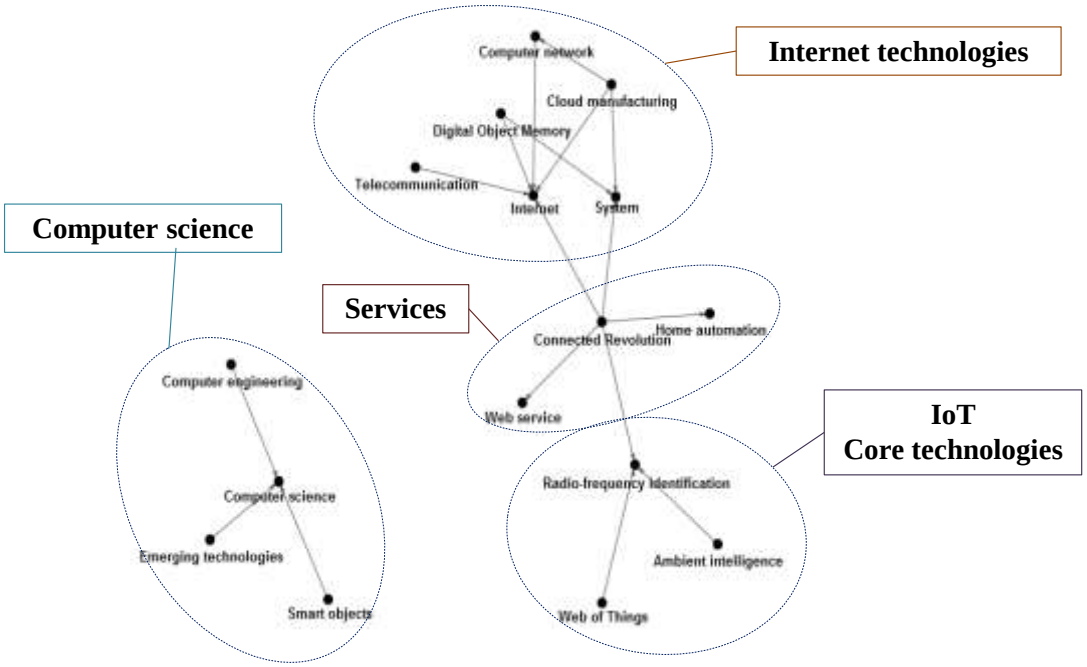

Figure. 7. Part of the concept graph for the category 'IoT' in the GoogleNews corpus (when δ =1.2)

Figures 6 and 7 show the concept graphs for the GoogleNews corpus about the company Google and IoT. In the concept graph in Figure 6, several concepts are connected focusing on the concepts 'Google', 'Android', and 'YouTube'. In Figure 7, several concepts about IoT are connected, while forming a set of clusters called 'Computer science', 'Internet technologies', 'IoT services', and 'IoT core technologies', similar to the concept graph in Figure 4.

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

4.4 Performance Evaluation

Table 3. The probabilistic values for concept pairs

Corpus	Sanderson's method				Proposed method			
	Hypernym	Hyponym	*Diff*	*Hr*	Hypernym	Hyponym	*Diff*	*Hr*
OHSUMED (C02)	Cervical cancer	Vulvar intraepithelial neoplasia	0.788	0.920	Cervical cancer	Vulvar cancer	2.450	2.510
	Cervical cancer	Vulvar cancer	0.771	0.964	Toxoplasmosis	Cat health	2.343	2.368
	Gastroenteritis	Lymphocytic choriomeningitis	0.750	0.898	Duodenum	Pancreatic fistula	2.326	2.723
	Common cold	Lymphocytic choriomeningitis	0.750	0.898	Cervical cancer	Lymphocytic choriomeningitis	2.253	2.316
	Cervical cancer	Lymphocytic choriomeningitis	0.739	0.890	Hepatitis B	Hepatitis D	2.207	2.212
	Hepatitis C	Hepatitis D	0.737	0.908	Common cold	Lymphocytic choriomeningitis	2.110	2.173
	Hepatitis C	Lymphocytic choriomeningitis	0.732	0.881	Encephalitis	Herpesviral encephalitis	2.208	2.141
	Hepatocellular carcinoma	Lymphocytic choriomeningitis	0.726	0.881	Common cold	Atypical pneumonia	2.048	2.165
OHSUMED (C11)	Glaucoma	Pellucid marginal degeneration	0.747	0.995	Eye	Pellucid marginal degeneration	2.453	2.659
	Glaucoma	Intermediate uveitis	0.737	0.985	Eye	Sixth nerve palsy	2.400	2.582
	Glaucoma	Fungal keratitis	0.726	0.976	Eye	Ocular ischemic syndrome	2.393	2.594
	Glaucoma	Ocular ischemic syndrome	0.722	0.971	Eye	Optic disc drusen	2.383	2.569
	Glaucoma	Optic disc drusen	0.717	0.966	Eye	Central serous retinopathy	2.346	2.545
	Glaucoma	Sixth nerve palsy	0.713	0.961	Eye	Intermediate uveitis	2.232	2.633
	Glaucoma	Ocular hypertension	0.708	0.973	Glaucoma	Pellucid marginal degeneration	2.321	2.463
	Glaucoma	Central serous retinopathy	0.708	0.957	Eye	Fungal keratitis	2.318	2.594

Table 3 shows the result of using the conventional method in [4] and our proposed method with the OHSUMED corpus, in which the identified concepts are directly related to the given corpus; for convenience, only the top eight concept pairs are given in the table. Here, a pair of concepts with a hierarchical relationship is expressed in the columns 'Hypernym' and 'Hyponym'. The column *Diff* means the value of $\left|Hr\left(c_i\middle|c_j\right)-Hr\left(c_j\middle|c_i\right)\right|$ in Equation (6), which is used as an indicator of how much two concepts differ. In other words, this means that as the value of *Diff* increases, the hierarchical relationship becomes clearer. Table 3 shows that the baseline method using the 'OHSUMED -C02' documents produces a biased distribution of concepts in which the concepts are sensitively affected by the number of hyperlinks; in particular, there is a highly biased distribution in 'OHSUMED -C11'. The problem with the baseline method is that relevant concepts actually have a low degree of relationship with each other, although the differences in the number of links are larger. However, the proposed method overcomes this problem because it considers both the topological weight of concepts and the difference in the numbers of incoming and outgoing hyperlinks.

Figure 8 shows the changes in the quality of the automatically generated concept graphs on varying the threshold value δ in Equation (6). This figure includes the number of discovered hierarchical relationships |T'| and the taxonomic F_1-measure (*i.e.*, TF). From this figure, we can see that the proposed method can recover the original hierarchical structure of manually constructed concept graphs with reasonably high quality, although it is not perfect. When using our proposed method, the degree of recovery of the predetermined hierarchical structure approaches 80% in terms of F_1-measure. Note that an appropriate threshold value of δ should be selected for building effective concept graphs. Regardless of document corpora, when the value of δ is set to 0.5, our method performs the best in terms of the F_1-measure; at this time, the taxonomic precision increases sharply, and reaches 100% in the case of the GoogleNews corpus.

Figure 9 compares our method with the conventional methods described in [4], [5], and [9] in terms of the average F_1-measure for the concept graphs constructed in all cases; in this figure, the conventional methods are denoted 'Sanderson', 'Schmitz', and 'Lee', respectively. Here, we found that our proposed method significantly outperformed the conventional methods; specifically, our method is superior to Lee's second-ranked method by about 30% on average, which implies that the topological weight of concepts significantly contributes to producing more accurate concept graphs. In comparison, Sanderson's method generated low-quality concept graphs because it depends only on the document frequency, without any semantic information.

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118

RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

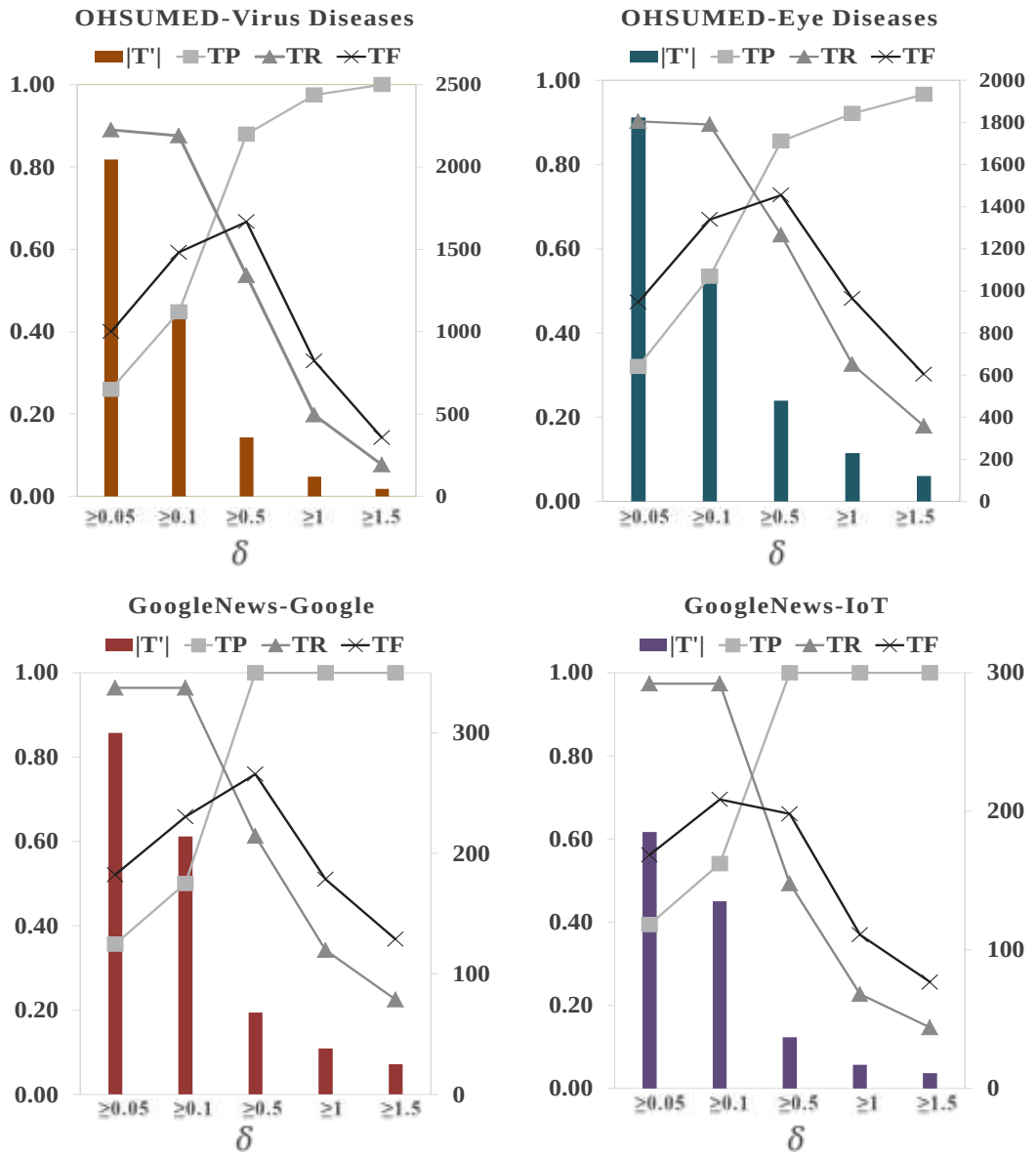

Figure. 8. Changes in the taxonomic F_1-measure and the number of discovered relationships by varying the threshold value δ

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 104-118
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

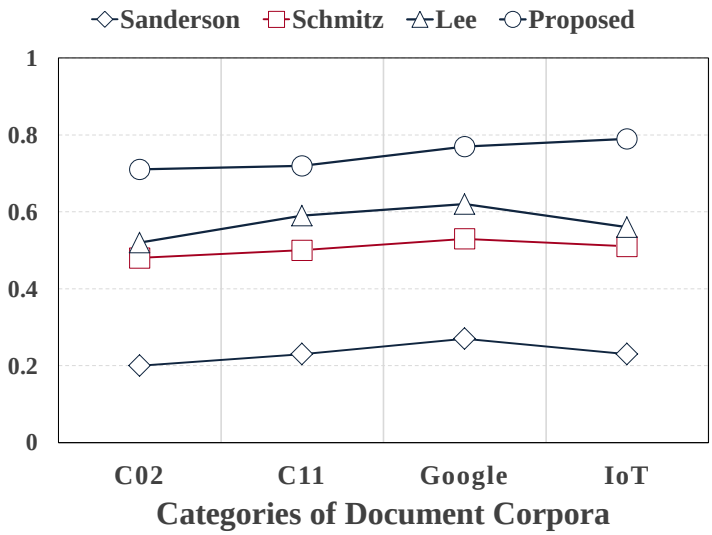

Figure. 9. Comparison with conventional methods in terms of the average taxonomic F_1-measure (TF)

5 Conclusions

In this paper, we proposed a novel way of automatically building a concept graph containing hierarchical relationships by probabilistically analyzing the information of connected hyperlinks within Wikipedia articles. To identify concept-level Wikipedia articles automatically, we used the *Elasticsearch* engine, which can find a subset of concept-level Wikipedia articles for a submitted document corpus. Then, the topological weights that mainly use the number of hyperlinks among the chosen concepts are used for calculating the probabilistic subsumption among concepts, which contributes to generating more stable concept graphs. We confirmed that reasonable concept graphs including noun-level and pronoun-level concepts can be derived only by analyzing hyperlinks in Wikipedia articles. We believe that the concept graphs produced with each of the given corpora can be used effectively as knowledge bases for organizing big text data and improving various text-mining applications. In the near future, we plan to utilize the proposed method to develop a tensor-based text-mining system [14].

Acknowledgements

This work was supported by the 2015 Research Fund of the University of Seoul.

References

[1]. Gulrandhe. C, Bawankar. C, "Concept Graph Preserving Semantic Relationship for Biomedical Text Categorization", International Journal of Computer Science And Applications 2015; 3(1): 9-12.

[2]. Ni. Y, Xu. QK, Cao. F, Mass. Y, Sheinwald. D, Zhu. HJ, Cao. SS, "Semantic documents relatedness using concept graph representation", in Proc. Ninth ACM International Conference on Web Search and Data Mining, 2016, 635-644.

[3]. McAfee. A, Brynjolfsson. E, Davenport. TH, Patil. DJ, Barton. D, "Big data: The management revolution", Harvard Business Review 2012; 90(10): 61–67.

[4]. Sanderson. M, Croft. B, "Deriving concept hierarchies from text", in Proc. 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 1999, 206–213.

[5]. Schmitz. P, "Inducing ontology from Flickr tags", in Collaborative Web Tagging Workshop at WWW2006, 2006.

[6]. Medelyan. O, Milne. D, Legg. C, Witten. IH, "Mining meaning from Wikipedia", International Journal of Human-Computer Studies 2009; 67(9): 716-754.

[7]. Vidhya. R, Vadivu. G, "Research document search using Elasticsearch", Indian Journal of Science and Technology 2016; 9(37): 1-4.

[8]. Aumann. Y, Feldman. R, Yehuda. YB, Landau. D, Liphstat. O, Schler. Y, "Circle graphs: New visualization tools for text-mining", in Proc. European Conference on Principles of Data Mining and Knowledge Discovery, 1999, 277–282.

[9]. Lee. GH, Kim. HJ, "Automated Development of Concept Hierarchy Tree using Backlink Information of Wikipedia", KIISE Database Research 2015; 31(1): 40-49.

[10]. Gormley. C, Zachary. T, "Elasticsearch: The Definitive Guide", O'Reilly Media, 2015.

[11]. Gospodnetic. O, Hatcher. E, "Lucene in Action", Manning Publications Co., 2005.

[12]. Langville. AN, Meyer. CD, "Deeper inside PageRank", Internet Mathematics 2004; 1(3): 335–380.

[13]. Sebastiani. F, "Machine learning in automated text categorization", ACM computing surveys (CSUR) 2002; 34(1): 1-47.

[14]. Kim. HJ, Hong. KH, Chang. JY. "Semantically enriching text representation model for document clustering", in Proc. 30th Annual ACM Symposium on Applied Computing, 2015, 922–925.