Mining Semantic Relationships between Concepts across Documents Incorporating Wikipedia Knowledge

Peng Yan and Wei Jin

Department of Computer Science, North Dakota State University
1340 Administration Ave., Fargo, ND 58102, USA
{peng.yan,wei.jin}@ndsu.edu

Abstract. The ongoing astounding growth of text data has created an enormous need for fast and efficient text mining algorithms. Traditional approaches for document representation are mostly based on the Bag of Words (BOW) model which takes a document as an unordered collection of words. However, when applied in fine-grained information discovery tasks, such as mining semantic relationships between concepts, sorely relying on the BOW representation may not be sufficient to identify all potential relationships since the resulting associations based on the BOW approach are limited to the concepts that appear in the document collection literally. In this paper, we attempt to complement existing information in the corpus by proposing a new hybrid approach, which mines semantic associations between concepts across multiple text units through incorporating extensive knowledge from Wikipedia. The experimental evaluation demonstrates that search performance has been significantly enhanced in terms of accuracy and coverage compared with a purely BOW-based approach and alternative solutions where only the article contents of Wikipedia or category information are considered.

Keywords*:* Knowledge Discovery, Semantic Relatedness, Cross-Document knowledge Discovery, Document Representation.

1 Introduction

With the explosive growth of text data and growing demand for high-quality text mining algorithms, document representation and semantic relatedness computation approaches are increasingly crucial. Traditionally text documents are represented as a Bag of Words (BOW) and relatedness between concepts are measured based on statistical information from the corpus such as the widely used tf-idf weighting scheme [12, 14]. Recently, [5, 12] introduced an interesting text mining scenario focusing on detecting links between two concepts across multiple documents. Typically, the uncovered links involving concepts A and B have the following meaning: find the most plausible relationship between concept A and concept B assuming that one or more instances of both concepts occur in the corpus, but not necessarily in the same document. For example, both may be football lovers, but maybe mentioned in different documents. However, the techniques proposed in [5, 12] were all built under the

P. Perner (Ed.): ICDM 2013, LNAI 7987, pp. 70–84, 2013.

assumption of BOW-based representation, and thus demonstrating their inherent limitations. For example, the detected links are limited to the associations occurring in the document collection; any potential relationships not appearing in the corpus cannot be discovered even though they are closely related to two concepts of interest. The semantic relatedness computing methods used in [5, 12] were mainly based on statistical information collected from the corpus and no background knowledge has been taken into account.

To alleviate all such limitations, this work proposes Semantic Path Chaining (SPC), a new model to uncover semantic paths between concepts with a focus on taking background knowledge into consideration. The approach proposed here is based on the method proposed by Srinivasan's closed text mining algorithm [13] in the biomedical domain, but we extend it to handle a more complicated query scenario where multiple-stage semantic paths are desired and also attempt to incorporate Wikipedia knowledge to enrich document representation. Motivated by the Explicit Semantic Analysis (ESA) technique introduced by Gabrilovich et al. [2], which was able to use the space of Wikipedia articles to measure the semantic relatedness between fragments of natural language text, we develop a hybrid approach and weighting scheme that combines the advantages of ESA and content based statistical analysis. Another distinct difference from the original ESA method is that [2] only focused on document-level textual analysis through mapping a given text fragment or term to a conceptual vector space spanned by all Wikipedia articles, whereas here we extend this technique by considering other valuable evidences from Wikipedia such as categories associated with each Wiki concept to further improve the semantic relatedness estimation between concepts.

Our contribution of this paper can be summarized as follows. First, compared with traditional methods mostly based on the BOW representation, the proposed technique is able to provide a much more comprehensive knowledge repository to support various queries and effectively complements existing knowledge contained in text corpus. Over 5,000,000 Wikipedia articles and more than 700,000 Wikipedia categories are considered to help measure the semantic relatedness between concepts. Therefore the relationships revealed are not limited to those appearing in the document collection literally. Also we observe for connections between rare concepts where we have little knowledge about them, the relationships discovered are often more than one level of transitivity and most of them cannot be uncovered unless integrating the knowledge from Wikipedia. Second, besides content analysis on Wikipedia articles, the new solution also integrates other valuable information, such as Wiki categories, as an effective aid in providing a better modeling of semantic relatedness estimation (based on the assumption that two concepts that share more common categories may have a closer relationship), and thus being able to boost linking concepts that are more closely related to topics of interest to higher rankings. We envision this integration would also benefit other related tasks such as question answering and cross–document summarization. Third, the discovered potential relationships have been greatly enriched by including intermediate concepts (linking terms) derived from Wikipedia, and for these newly identified connections not appearing in the text corpus, we further introduce a pruning and validation step through an application of a sequence of devised

heuristics. Last, to demonstrate the effectiveness of our new model, a significant amount of queries covering various scenarios were conducted, evaluated, and compared against the BOW based baseline. We have also further evaluated the performance of using our adapted ESA method, the approach only incorporating the Wikipedia category information, as well as the solution combining both of the above two resources, respectively. The experiments demonstrate the significant improvement achieved by our proposed method over the original ESA method and other alternative solutions.

This paper is organized as follows: Section 2 describes related work. Section 3 introduces our new semantic relatedness computation measures. In Section 4, we discuss the new model of mining semantic relationships between concepts incorporating Wikipedia knowledge in detail. Experimental results are presented and analyzed in Section 5. Section 6 concludes this work and describes future directions.

2 Related Work

Most of existing text mining algorithms for capturing relationships between concepts have built on the traditional Bag-of-Words representation and significant efforts have been paid to content analysis of document collections with no or little background knowledge being taken into account [12, 14, 15], thus resulting in a limited discovery scope. To alleviate such problems, there has been work recently reported on exploring methods of utilizing external knowledge to assist in the discovery tasks. Bollegara et al. [1] developed an approach for semantic relatedness calculation using returned page counts and text snippets produced by a Web search engine. Mehmet Ali Salahli [9] proposed another Web oriented method that calculated semantic relatedness between terms using a set of determiners (special words that are supposed to be highly related to terms of interest). However, the performance of these approaches highly relies on the generated outputs from search engines and has not reached the satisfying level. WordNet based approaches are another direction to approach this problem, especially in handling synonym, hyponymy/hypernymy relations. Hotho et al. [4] exploited WordNet to improve the BOW text representation and Martin [6] developed a method for transforming the noun-related portions of WordNet into a lexical ontology to enhance knowledge representation. Scott and Matwin [10] proposed a new representation of text based on WordNet hypernyms. These WordNet-based techniques have shown their advantages of improving the tradition BOW based representation to some degree but suffer from relatively limited coverage of Wordnet compared to Wikipedia, the world's largest knowledge base to date. Gabrilovich et al [3] applied machine learning techniques to Wikipedia and proposed a new method to enrich document representation from this huge knowledge repository. Specifically, they built a feature generator to identify most relevant Wikipedia articles for each document, and then used concepts corresponding to these articles to create new features. The experimental evaluation showed great improvements across a diverse collection of datasets. However, with the process of feature generation so complicated, a considerable computational effort is required.

In terms of improving semantic relatedness computation using Wikipedia, Milne [7] proposed a Wikipedia Link Vector Model (WLVM) for this purpose. However, only the hyperlink structure of Wikipedia and article titles were extracted to compute semantic relatedness between query terms, without any analysis of textual contents of Wikipedia articles. Gabrilovich et al. [2] presented a novel method, Explicit Semantic Analysis (ESA), for fine-grained semantic representation of unrestricted natural language texts. Using this approach, the meaning of any text can be represented as a weighted mixture of Wikipedia-based concepts (articles), called an interpretation vector [2]. [2] also discussed the problem of possibly containing noise concepts in the vector, especially for text fragments containing multi-word phrases (e.g., multi-word names like George Bush). Our proposed solution is motivated by [2, 7] and to tackle the above problems, we adapt the ESA technique to better suit our task and further develop a sequence of heuristic strategies to filter out irrelevant terms and retain only top-k most relevant concepts to the given topics. Moreover, other than content-based statistical information of Wikipedia articles being incorporated, other valuable evidence sources provided by Wikipedia, such as categories associated with each concept, are also combined into our final concept ranking scheme. The detailed experimental results and comparisons will be presented in section 5.

3 Semantic Path Chaining

Semantic Path Chaining (SPC) is attempting to mine semantic paths between two concepts (e.g., two person names) across documents incorporating Wikipedia knowledge. We propose to use the features extracted from text corpus, as well as the relationships discovered from Wikipedia to construct semantic paths which stand for potential conceptual connections between them.

3.1 Ontology Mapping and Semantic Profile Representation

To detect semantic relationships between topics of interest, we first represent each topic as a semantic profile which is essentially a set of highly related concepts to the given topic in the corpus. To further differentiate between the concepts, semantic type (ontological information) is employed in profile generation. Table 1 illustrates part of semantic type - concept mappings. Thus each profile is defined as a vector composed of a number of semantic types.

Table 1. Semantic Type - Concept Mapping

Semantic Type	Instances
Human Action	attack, killing, covert action, international terrorism
Leader	vice president, chief, governor
Country	Iraq, Afghanistan, Pakistan, Kuwait
Diplomatic Building	consulate, pentagon, UAE Embassy
Government	Bush administration, white house, national security council
Person	deputy national security adviser, chairman, executive director

$$profile(T_i) = \{ST_1, ST_2, ..., ST_n\} \quad (1)$$

Where ST_i represents a semantic type to which the concepts appearing in the topic-related text snippets belong. We used sentence as window size to measure relevance of appearing concepts to the topic term. Under this representation each semantic type is again referred to as an additional level of vector composed of a number of terms that belong to this semantic type.

$$ST_i = \{w_{i,1}m_1, w_{i,2}m_2, ..., w_{i,n}m_n\} \quad (2)$$

Where m_j represents a concept belonging to semantic type ST_i, and $w_{i,j}$ represents its weight under the context of ST_i and sentence level closeness. When generating the profile we replace each semantic type in (1) with (2). In (2), to compute the weight of each concept, we employ a variation of the *TF*IDF* weighting scheme and then normalize the weights:

$$w_{i,j} = s_{i,j} / highest(s_{i,l}) \quad (3)$$

Where $l = 1, 2, ..., r$ and there are totally r concepts for ST_i, $s_{i,j} = df_{i,j} * Log(N / df_j)$, where N is the number of sentences in the collection, df_j is the number of sentences concept m_j occurs, and $df_{i,j}$ is the number of sentences in which topic T and concept m_j co-occur and m_j belongs to semantic type ST_i. By using the above three formulae we can build the corresponding profile representing any given topic.

To summarize, the procedure of building semantic profiles for a given topic T of interest is composed of the following four steps:

1. Concept Extraction: extract all potential concepts from the document collection which co-occur with the topic T in the sentence level.
2. Semantic Type Employment: each concept will be associated with and grouped under one or more semantic types (e.g., Human Action, Country, Person) which it belongs to.
3. Weight Calculation: for each concept, a variation of the *TF*IDF* scheme is used to calculate its weight (as shown in Formula 2).
4. Weight Normalization: within each semantic type, the concept weights are further normalized by the highest concept weight observed for the semantic type as given in Formula 3, and then ranked according to the normalized weights.

3.2 Chaining Semantic Paths

In this step, we search potential conceptual connections in different levels, and use them to construct semantic paths linking two given topics (concepts). Suppose A and C are two given topics of interest, the algorithm of generating semantic paths connecting A to C from the text corpus is composed of the following sequential steps:

1. Conduct independent searches for A and C. Build the A and C profiles. Call these profiles AP and CP respectively.
2. Compute a B profile (BP) composed of terms in common between AP and CP. The corpus-level weight of a concept in BP is the sum of its weights in AP and CP. This is the first level of intermediate potential concepts generated from the text corpus.
3. Expand the semantic paths using the created BP profile together with the topics to build additional levels of intermediate concept lists DP and EP which (i) connect the topics to each concept in BP profile in the sentence level within each semantic type, and (ii) also normalize and rank them (as detailed in section 3.1).

4 Mining Semantic Relationships between Concepts Incorporating Wikipedia Knowledge

4.1 Wiki-article Content Based Measure

To utilize Wikipedia knowledge to complement existing information in the document collection, we adapt the Explicit Semantic Analysis (ESA) technique proposed by Gabrilovich et al. [2] as our underlying content-based measure for analyzing Wikipedia articles relevant to the given topics of interest. Under this measure, each article in Wikipedia is treated as a concept, and each document is represented by an interpretation vector containing related Wikipedia concepts (articles) to the document.

$$\phi(d) =< as(d,a_1),...,as(d,a_n) > \tag{4}$$

Where $as(d,a_i)$ represents the association strength between document d and Wikipedia article a_i. Suppose d is spanned by all words appearing in it, i.e., $d =< w_1, w_2, ..., w_j >$, the association strength $as(d,a_i)$ is computed as follows:

$$as(d,a_i) = \sum_{w_j \in d} tf_d(w_j) tf \cdot idf_{a_i}(w_j) \tag{5}$$

Where $tf_d(w_j)$ is the frequency of word w_j in document d, and $tf \bullet idf_{a_i}(w_j)$ is the $tf \bullet idf$ value of word w_j in Wikipedia article a_i. As a result, the vector for a document is represented by a list of real values indicating the association strength of a given document with respect to Wikipedia articles. By using efficient indexing strategies such as single-pass in memory indexing, the computational cost of building these vectors for a given term (or text fragments containing multiple terms) can be reduced to within 200-300 ms.

As discussed above, the original ESA method [2] is subject to the noise concepts introduced, especially when dealing with multi-word phrases. For example, when the input is "George Bush", the generated interpretation vector will contain a fair amount of noise concepts such as "That's My Bush", which is actually an American comedy television series. This Wikipedia concept (article) is selected and ranked in the second

place in the list because "Bush" occurs many times in the article "That's My Bush", but obviously this article is irrelevant to the given topic "George Bush". In order to make the interpretation vector more precise and relevant to the topic, a sequence of heuristics is devised to clean the vector as shown in Figure 1. More specifically, a modified Levenshtein Distance algorithm is devised to measure the relevance of the given topic to each Wikipedia concept generated in the interpretation vector with a single word as a unit for allowable edit operations, which allows the adapted algorithm to be used to compute the similarity between any two text snippets. If the topic contains only one word, then the number of its occurrences in the corresponding Wikipedia article will be used for judgement. If it occurs more than three times, this article is viewed as relevant to the given topic and is kept in the interpretation vector. If the topic contains multiple words, we will view each word as if it were a character and employ our adapted version of the Levenshtein distance algorithm to evaluate the relevance of the topic to the article text. If their Levenshtein distance is under the defined threshold, the article is viewed as relevant. Otherwise, it will be removed from the interpretation vector.

Input: a topic T of interest
an interpretation vector V representing the topic T

Output: a cleaned Wikipedia-based concept vector V' representing the topic T

1. If T is a single word topic, then count the number of occurrences of T in the article texts represented by each concept v_i in V, respectively. If T occurs more than 3 times, then keep v_i in V, otherwise, remove v_i from V.

2. If T is a multi-word topic, then the adapted Levenshtein distance algorithm applies to measure the relevance of each Wikipedia concept (article) v_i in V to topic T.

2.1. If NumOfWords(T) $\leq$ 2, then extract all text snippets TS_j within the window size NumOfWords(T) from the article text of v_i. If there exists a j such that LevenshteinDistance(T, TS_j) = 0, then keep v_i in V, otherwise, remove v_i from V.

2.2. If NumOfWords(T) > 2, then extract all text snippets TS_j within the window size NumOfWords(T)+1 from the article text of v_i. If there exists j such that LevenshteinDistance(T, TS_j) $\leq$ 1, then keep v_i in V, otherwise, remove v_i from V.

Fig. 1. Interpretation vector cleaning procedure

After the cleaning step, we are able to use the resulting interpretation vectors for computing similarities between any two concepts. In our context of mining associations between two topics, say A and C, we compute the Cosine similarity between the interpretation vectors of topic A and each concept V_i in the intermediate BP profile, as well as between topic C and each concept V_i, and take the average of two Cosine similarities as the overall similarity for each concept V_i in BP profile.

4.2 Wiki-Category Based Measure

Human edited categories associated with each Wiki concept (article), another valuable resource provided by Wikipedia, have also been integrated to better serve this task. Based on the assumption that those concepts (articles) sharing similar categories may be closer to each other in terms of semantic relatedness, a Wikipedia category

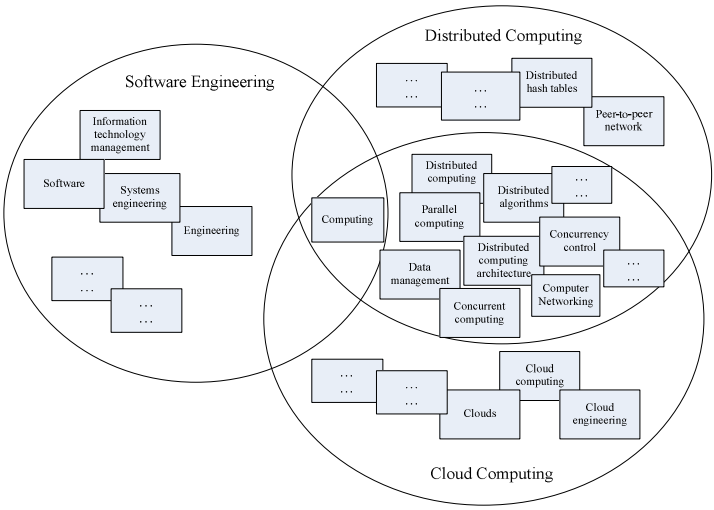

Fig. 2. Category Overlaps of the Concepts in the Interpretation Vectors of "Distributed Computing," "Cloud Computing" and "Software Engineering"

interpretation vector has been built for each desired Wiki concept and the semantic relatedness between two concepts of interest is determined by the percentage of common categories shared by the two corresponding category interpretation vectors.

Formally, suppose the interpretation vector for article a_i is $V_i =< p_1, p_2, ..., p_m >$, where p_i in V_i represents a Wiki page (or article) that is relevant to a_i, then article a_i can be further represented as a *Category Space Vector (CSV)* as follows spanning the Wikipedia category space.

$$CSV(a_i) =<< w_{i,1,1} c_{1,1}, w_{i,2,1} c_{2,1}, ... >, ..., < w_{i,1,m} c_{1,m}, w_{i,2,m} c_{2,m}, ... >> \quad (6)$$

Where $c_{x,y}$ represents category c_x that p_y in V_i belongs to, and $w_{i,x,y}$ is the weight for $c_{x,y}$. To calculate $w_{i,x,y}$, we count the number of sub-vectors within $CSV(a_i)$ in which $c_{x,y}$ appears, and then normalize it:

$$w_{i,x,y} = \frac{w_{i,x,y}}{highest(w_{i,d,y})} \quad (7)$$

Where $d = 1, 2, ..., r$ and there are totally r categories in Wikipedia. The semantic relatedness between two Wikipedia concepts (articles) can then be computed by the Cosine similarity between their corresponding CSVs. Figure 2 shows the categories built for three concepts: "Distributed Computing," "Cloud Computing" and "Software Engineering." The produced semantic relatedness between "Distributed Computing" and "Cloud Computing" is 0.715, 0.094 between "Distributed Computing" and

"Software Engineering", and 0.151 between "Cloud Computing" and "Software Engineering". This is consistent with our understanding that "Distributed Computing" and "Cloud Computing" are more semantically closely related while both further away from "Software Engineering.

4.3 Final Weighting Scheme

A final ranking for each concept generated in the intermediate profiles is calculated by linearly combining its TFIDF-based similarity, content-based similarity and category-based similarity together as below:

$$S_{overall} = \lambda_1 \cdot S_{TFIDF} + \lambda_2 \cdot S_{content} + (1 - \lambda_1 - \lambda_2) \cdot S_{category} \quad (8)$$

Where λ_1 and λ_2 are two tuning parameters that can be adjusted based on the preference on the two similarity schemes in the experiments. S_{TFIDF} refers to the similarity computed using the traditional BOW model, and $S_{content}$ and $S_{category}$ refer to the similarities computed using the content based measure and category based measure respectively.

4.4 The New Model of Mining Semantic Relationships

After defining the semantic relatedness measures between concepts, we are presenting now the new solution for building semantic paths between concepts. Suppose A and C are two given topics of interest, with Wikipedia knowledge incorporated in our model, we are able to leverage Wiki concepts to enrich the relationships (i.e., not limited to those occurring in the document collection literally). Thus the generated links would be an integration of relationships identified from the text corpus as well as from Wikipedia knowledge. The process can be summarized as the following major steps and is further illustrated in Figure 3.

1. Build ESA-based interpretation vectors for A and C. Employ the cleaning procedure illustrated in Figure 1 to remove noise concepts in the generated interpretation vectors. The concepts that survived after cleaning are ordered according to their association strength as described in Section 4.1, and will be serving as potentially novel connections between topics A and C.
2. Enrich the generated BP profile with newly identified Wiki concepts (represented by the corresponding Wikipedia article titles) by merging the cleaned interpretation vectors for topics A and C. The weight of each newly identified Wiki concept in BP is the sum of its association strengths in the cleaned interpretation vectors for topics A and C.
3. Go through the same procedures as in the above two steps to enrich DP and EP profiles that contain the intermediate concepts connecting the topics to each concept in BP profile.

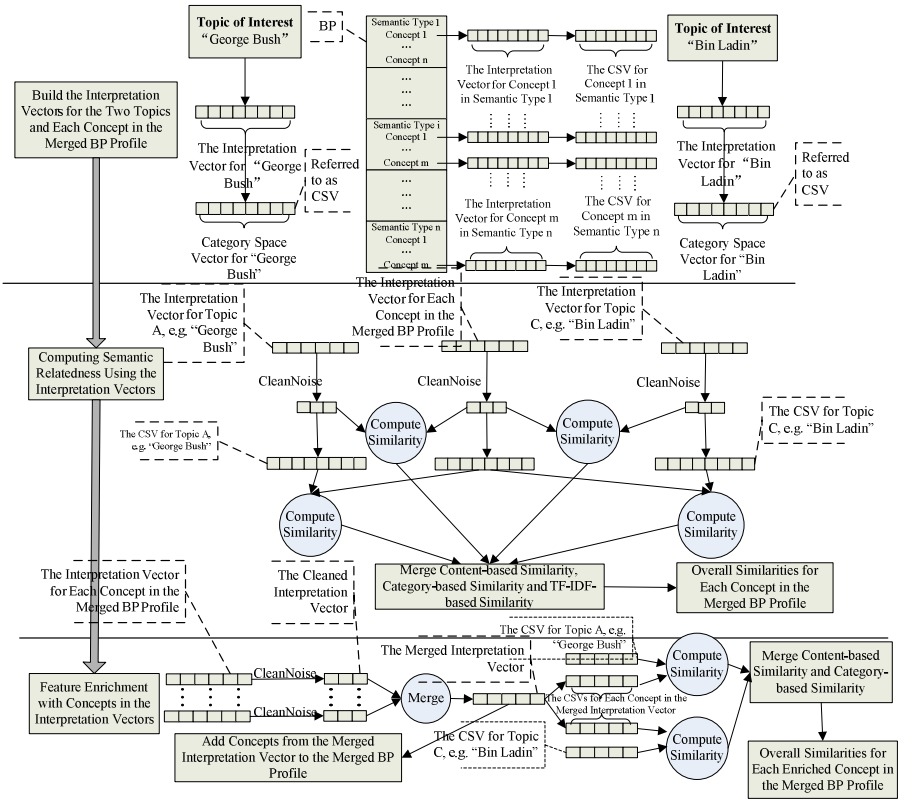

Fig. 3. The new model of mining semantic relationships

4. The BP profile is further enriched by considering relevant Wiki categories that the newly identified Wiki concepts (articles) belong to. The weight of each newly identified Wiki category in BP is the same as that of the corresponding Wiki concept.
5. Go through the same procedure as in Step 4 to enrich DP and EP profiles with the newly identified relevant Wiki categories.

5 Empirical Evaluation

5.1 Processing Wikipedia Dumps

Wikipedia offers free copies of the entire content in the form of XML files. It is an ever-updating knowledge base, and releases the latest dumps to interested users regularly. The version used in this work was released on April 05, 2011, which was separated into 15 compressed XML files and altogether occupied 29.5 GB after de-compression. An open source tool MWDumper [8] was used to import the XML dumps into our MediaWiki database, and after the parsing process, we identified 5,553,542 articles.

5.2 Evaluation Data

An open source document collection pertaining to the 9/11 attack, including the publicly available 9/11 commission report was used in our evaluation. The report consists of Executive Summary, Preface, 13 chapters, Appendix and Notes. Each of them was considered as a separate document resulting in 337 documents. The whole collection was processed using Semantex[11] and concepts were extracted and selected as shown in Table 1. A significant amount of query pairs selected by the assessors covering various scenarios [16] (e.g., ranging from popular entities to rare entities) were conducted and used as our evaluation data.

5.3 Experimental Results

Parameter Settings. As mentioned in Section 4.2, a combination of corpus-level TF*IDF-based similarity, Wiki-article content based similarity and category-based similarity is used to rank the intermediate concepts detected by the system. λ_1 and λ_2 are two parameters that need to be tuned so that the similarities between concepts best match the judgements from our assessors. To accomplish this, we first built a set of training data composed of 10 query pairs randomly selected from the evaluation set, and then generated B profiles for each of them using our proposed method. Among each B profile, we selected the top 5 concepts (links) within each semantic type, and compared their rankings with the assessors' judgements. The values of λ_1 and λ_2 were tuned in the range of [0.1, 1] and we observe the best performance was achieved with $\lambda_1 = 0.1$ and $\lambda_2 = 0.3$. This setting was also used in our later experiments.

Query Results. Tables 2 through 4 make a comparison between the search results of our baseline where the corpus-level TF-IDF based statistical information is used to generate chains without the involvement of Wikipedia and various Wiki-enabled models proposed in this work. Specifically, Table 2 shows the improvement achieved by integrating the Wiki-article content based measure over the baseline; Table 3 presents the result when the relevant Wiki categories are used to improve the discovery

Table 2. The Effect of Integrating the Adapted ESA Technique (original ESA+ Vector Cleaning)

		Baseline/Wiki-ESA					
		S_5	**S_{10}**	**S_{15}**	**S_{20}**	**S_{30}**	**S_{40}**
L_1	P	0.756/0.788	0.764/0.789	0.763/0.786	0.759/0.787	0.759/0.787	0.761/0.789
	R	0.440/0.618	0.538/0.721	0.576/0.763	0.593/0.793	0.624/0.826	0.644/0.849
L_2	P	0.845/0.855	0.844/0.855	0.843/0.853	0.843/0.852	0.842/0.850	0.841/0.849
	R	0.528/0.575	0.573/0.622	0.608/0.659	0.633/0.683	0.657/0.706	0.676/0.723
L_3	P	0.846/0.856	0.845/0.856	0.844/0.854	0.844/0.853	0.843/0.851	0.842/0.850
	R	0.530/0.575	0.573/0.620	0.608/0.658	0.634/0.681	0.657/0.705	0.676/0.722
L_4	P	0.691/0.699	0.689/0.695	0.687/0.692	0.686/0.691	0.684/0.689	0.684/0.689
	R	0.392/0.413	0.513/0.534	0.587/0.610	0.638/0.661	0.690/0.713	0.720/0.744

Table 3. The Effect of Integrating Wikipedia Categories

		Baseline/Wiki-CSV					
		S_5	**S_{10}**	**S_{15}**	**S_{20}**	**S_{30}**	**S_{40}**
	P	0.756/0.767	0.764/0.773	0.763/0.770	0.759/0.767	0.759/0.767	0.761/0.769
L_1	R	0.440/0.589	0.538/0.694	0.576/0.738	0.593/0.759	0.624/0.793	0.644/0.816
	P	0.845/0.856	0.844/0.855	0.843/0.853	0.843/0.852	0.842/0.851	0.841/0.850
L_2	R	0.528/0.580	0.573/0.628	0.608/0.663	0.633/0.687	0.657/0.710	0.676/0.728
	P	0.846/0.857	0.845/0.857	0.844/0.855	0.844/0.854	0.843/0.853	0.842/0.851
L_3	R	0.530/0.580	0.573/0.627	0.608/0.662	0.634/0.686	0.657/0.709	0.676/0.727
	P	0.691/0.702	0.689/0.699	0.687/0.696	0.686/0.694	0.684/0.692	0.684/0.691
L_4	R	0.392/0.422	0.513/0.547	0.587/0.622	0.638/0.673	0.690/0.725	0.720/0.755

Table 4. The Effect of Integrating both ESA and Wikipedia Categories

		Baseline/Wiki-ESA-CSV					
		S_5	**S_{10}**	**S_{15}**	**S_{20}**	**S_{30}**	**S_{40}**
	P	0.756/0.798	0.764/0.818	0.763/0.814	0.759/0.810	0.759/0.809	0.761/0.809
L_1	R	0.440/0.648	0.538/0.840	0.576/0.880	0.593/0.898	0.624/0.929	0.644/0.949
	P	0.845/0.864	0.844/0.865	0.843/0.862	0.843/0.861	0.842/0.859	0.841/0.865
L_2	R	0.528/0.625	0.573/0.679	0.608/0.713	0.633/0.736	0.657/0.758	0.676/0.727
	P	0.846/0.866	0.845/0.865	0.844/0.863	0.844/0.862	0.843/0.860	0.842/0.858
L_3	R	0.530/0.625	0.573/0.676	0.608/0.710	0.634/0.734	0.657/0.756	0.676/0.772
	P	0.691/0.709	0.689/0.705	0.687/0.701	0.686/0.699	0.684/0.696	0.684/0.695
L_4	R	0.392/0.443	0.513/0.570	0.587/0.645	0.638/0.696	0.690/0.748	0.720/0.778

model; Table 4 demonstrates the overall benefit when both the Wiki article contents and Wiki categories are incorporated. The table entries can be read as follows: S_N means we only keep the top N concepts within each semantic type in the searching results and L_N indicates the resulting chains of length N. The entries in the three tables stand for the precision and recall values (P for precision and R for recall). It is easy to observe that the search performance has been significantly improved with the integration of Wikipedia knowledge, and the best performance is observed when both the Wiki article contents and categories are involved.

We further use $F-measure$ to interpret the query results as a harmonic mean of the precision and recall. Figures 4 through 7 compare the search results graphically between the baseline and our new models in terms of *F-scores* for chains of different lengths. The X-axis indicates the number of concepts kept in each semantic type in the search results (S_N means the top N are kept), while the Y-axis indicates the *F-score.* We can see that the achieved F-score continues to rise as we increase the number of top concepts kept in the search results, and the most significant upward trend was observed when the number of top concepts kept increased from 5 to 10. It is also obvious that our new model consistently achieves better performances for different lengths than the baseline solution, and the approach that integrates both the Wiki article contents and categories shows the most improvement.

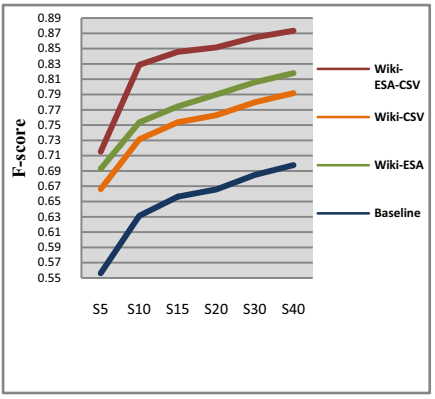

Fig. 4. The result of length 1

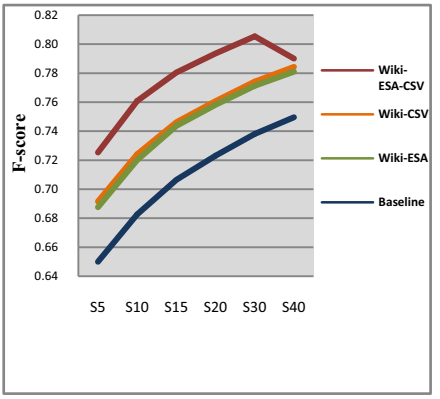

Fig. 5. The result of length 2

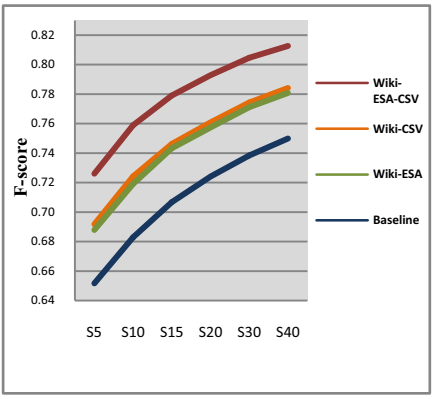

Fig. 6. The result of length 3

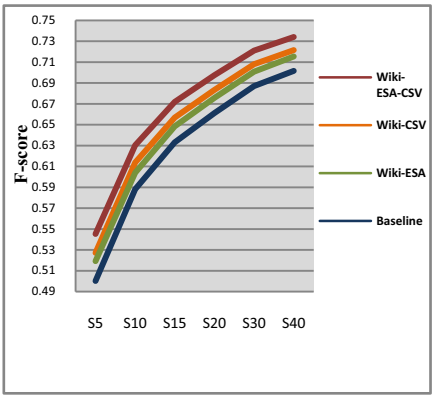

Fig. 7. The result of length 4

Table 5 shows newly discovered semantic relationships where linking concepts can only be acquired by integrating information from multiple documents or from Wikipedia knowledge (i.e., not contained in the existing document collection). For instance, for the query pair: "Atta :: dekkers," two intermediate important persons connecting them were identified: "Mohammed_Atta_al_Sayed" was an Egyptian hijacker and one of the ringleaders of the September 11 attacks and "Marwan_al-Shehhi" was the hijacker-pilot of United Airlines Flight 175, crashing the plane into the South Tower of the World Trade Center as part of the September 11 attacks.

Table 5. Instances of Enriched Semantic Relationships

Query Pair	Resulting Chain
L2 (Length 2)	
abdel_rahman :: blind_sheikh	abdel_rahman →ballistic_missile_threat_unite_state → blind_sheikh
bush :: bin_ladin	bush → east_africa_embassy_bombings →bin_ladin
adel :: ffi	adel → afghanistan → ffi
marty_miller :: oakley	marty_miller → unocal → oakley
gore :: stephen_hadley	gore → clarke → stephen_hadley
alexis :: lloyd_salvetti	alexis → janice_kephart_roberts → lloyd_salvetti
donovan :: wall_street	donovan → intelligence_group → wall_street
L3 (Length 3)	
atta :: dekkers	atta → mohammed_atta_al_sayed → marwan_al-shehhi → dekkers
amal :: sudanese	amal → bahrain → cia → sudanese
karachi :: usama_asmurai	karachi → june_14_terrorist_attack_outside_us_consulate_in_karachi → may_8_bus_attack_in_karachi → usama_asmurai
binalshibh :: pistole	binalshibh → fbi → minneapolis → pistole
martha_stewart :: saudi_arabia	martha_stewart → al-jawf,_saudi_arabia → khaled_of_saudi_arabia → saudi_arabia
L4 (Length 4)	
kenya :: mohamed	kenya → mihdhar_hazmi → afghanistan → shanksville → mohamed
gore :: stephen_hadley	gore → suicide_hijackings → white house → national_security_council → stephen_hadley
crawford :: khalilzad	crawford → bill_clinton → afghan → deputy_secretary_state_richard_armitage → khalilzad

6 Conclusion and Future Work

This paper proposes a new solution for mining semantic relationships between concepts across multiple documents by taking extensive background knowledge from Wikipedia into consideration. Specifically, we focus on detecting cross-document semantic relationships between concepts where most of them cannot be uncovered by the traditional paradigm. We also go one step further by incorporating the knowledge from Wikipedia to help identify more potential relationships that do not occur literally in the existing document corpus. The experiments were conducted using a large set of queries covering various scenarios, and compared with a purely BOW-based representation model, the original ESA method, and the approach only incorporating the Wikipedia category information. The results demonstrate the effectiveness of our proposed new hybrid solution combing all valuable resources and show the much broader and well-rounded coverage of significant relationships between concepts.

Wikipedia provides some other valuable information resources which were not used in this study. For instance, each Wikipedia article contains plenty of anchor text links which may imply potential relationships between different articles. Also, the "redirect" links pointing to a specific article may indicate synonymy and be further helpful to semantic

relatedness computing. Moreover, the infobox templates provide a good chance to increase the data quality using the ontology mapping technique. We will be exploring the usage of these resources and evaluating their performance in our future work.

References

1. Bollegara, D., Matsuo, Y., Isizuka, M.: Measuring Semantic Similarity between Words Using Web Search Engines. In: 16th International World Wide Web Conference, pp. 757–766. ACM, New York (2007)
2. Gabrilovich, E., Markovitch, S.: Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In: 20th International Joint Conference on Artificial Intelligence, pp. 1606–1611. Morgan Kaufmann, San Francisco (2007)
3. Gabrilovich, E., Markovitch, S.: Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. In: 21st National Conference on Artificial Intelligence, vol. 2, pp. 1301–1306. AAAI Press, Menlo Park (2006)
4. Hotho, A., Staab, S., Stumme, G.: Wordnet improves Text Document Clustering. In: SIGIR 2003 Semantic Web Workshop, pp. 541–544. Citeseer (2003)
5. Jin, W., Srihari, R.: Knowledge Discovery across Documents through Concept Chain Queries. In: 6th IEEE International Conference on Data Mining Workshops, pp. 448–452. IEEE Computer Society, Washington (2006)
6. Martin, P.A.: Correction and Extension of WordNet 1.7. In: Ganter, B., de Moor, A., Lex, W. (eds.) ICCS 2003. LNCS, vol. 2746, pp. 160–173. Springer, Heidelberg (2003)
7. Milne, D.: Computing Semantic Relatedness using Wikipedia Link Structure. In: The New Zealand Computer Science Research Student Conference. Hamilton, New Zealand (2007)
8. MWDumper Software, `http://www.mediawiki.org/wiki/Manual:MWDumper`
9. Salahli, M.A.: An Approach for Measuring Semantic Relatedness between Words via Related Terms. Journal of Mathematical and Computational Applications 14(1), 55–63 (2009)
10. Scott, S., Matwin, S.: Text Classification Using WordNet Hypernyms. In: Workshop on Usage of WordNet in Natural Language Processing Systems, pp. 45–52. Association for Computational Linguistics (1998)
11. Srihari, R.K., Li, W., Niu, C., Cornell, T.: InfoXtract: A Customizable Intermediate Level Information Extraction Engine. In: HLT-NAACL 2003 Workshop on Software Engineering and Architecture of Language Technology Systems, vol. 8, pp. 51–58. Association for Computational Linguistics, Stroudsburg (2003)
12. Jin, W., Srihari, R., Ho, H.H., Wu, X.: Improving Knowledge Discovery in Document Collections through Combining Text Retrieval and Link Analysis Techniques. In: Seventh IEEE International Conference on Data Mining, pp. 193–202. IEEE Computer Society, Washington (2007)
13. Srinivasan, P.: Text Mining: Generating hypotheses from Medline. Journal of the American Society for Information Science and Technology 55(5), 396–413 (2004)
14. Swason, D.R., Smalheiser, N.R.: Implicit Text Linkage between Medline Records: Using Arrowsmith as an Aid to Scientific Discovery. Library Trends 48(1), 48–59 (1999)
15. Srihari, R.K., Lamkhede, S., Bhasin, A.: Unapparent Information Revelation: A Concept Chain Graph Approach. In: 14th ACM International Conference on Information and Knowledge Management, pp. 329–330. ACM, New York (2005)
16. Yan, P., Jin, W.: Improving Cross-Document Knowledge Discovery Using Explicit Semantic Analysis. In: Cuzzocrea, A., Dayal, U. (eds.) DaWaK 2012. LNCS, vol. 7448, pp. 378–389. Springer, Heidelberg (2012)