Improving Cross-Document Knowledge Discovery Using Explicit Semantic Analysis

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. Cross-document knowledge discovery is dedicated to exploring meaningful (but maybe unapparent) information from a large volume of textual data. The sparsity and high dimensionality of text data present great challenges for representing the semantics of natural language. Our previously introduced Concept Chain Queries (CCQ) was specifically designed to discover semantic relationships between two concepts across documents where relationships found reveal semantic paths linking two concepts across multiple text units. However, answering such queries only employed the Bag of Words (BOW) representation in our previous solution, and therefore terms not appearing in the text literally are not taken into consideration. Explicit Semantic Analysis (ESA) is a novel method proposed to represent the meaning of texts in a higher dimensional space of concepts which are derived from large-scale human built repositories such as Wikipedia. In this paper, we propose to integrate the ESA technique into our query processing, which is capable of using vast knowledge from Wikipedia to complement existing information from text corpus and alleviate the limitations resulted from the BOW representation. The experiments demonstrate the search quality has been greatly improved when incorporating ESA into answering CCQ, compared with using a BOW-based approach.

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

1 Introduction

Text is the most traditional method for information recording and knowledge representation. Text mining focuses on mining high-quality information from mass text. The widely used text representation is based on the Bag of Words (BOW) model which represents text as a collection of words, however, this representation is limited to the terms appearing in the text literally, which could lead to great semantic loss because terms that are closely related to each other will be viewed as completely irrelevant unless they are both mentioned in the text. Our previous work [1] introduced a special case of text mining focusing on detecting semantic relationships between two concepts across documents, which we refer to as Concept Chain Queries (CCQ). A concept chain query involving concepts A and B has the following meaning: find the

A. Cuzzocrea and U. Dayal (Eds.): DaWaK 2012, LNCS 7448, pp. 378–389, 2012.

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 mentioned in different documents. The previous solution used the BOW model for text representation, i.e., relationships between important terms that do not co-appear literally in the text are neglected, and thus could not contribute to the generation of the links. For instance, "Albert Gore" is closely related to "George W. Bush" since two men together produced the most controversial presidential election in 2000, which was the only time in American history that the Supreme Court has determined the outcome of a presidential election. However, "Albert Gore" will not be taken into account if it does not occur in the document collection where the concept chain queries are performed on.

In our BOW based approach for answering concept chain queries [1], the weight of each term was measured by its $TF * IDF$ based value in the document collection. In this work, we propose to further improve the model by incorporating the Explicit Semantic Analysis (ESA) technique [3]. Basically, ESA maps a given text or a term to a conceptual vector space which is spanned by all Wikipedia articles, and thus more background knowledge can be integrated into semantic representation of each term, which is able to help overcome the shortcomings resulted from the BOW representation. We also attempt to identify only the most relevant concepts generated from ESA for semantic relatedness computation. To achieve this goal, we further develop a sequence of heuristic steps for noise removal. Therefore, our method will not bring as much noise as [3] does. To validate the proposed techniques, a significant amount of queries covering different scenarios were conducted to show that we could rank those most relevant concepts to the given topics in the top positions.

Our contribution of this effort can be summarized as follows. First, compared with the solution using a BOW based approach, 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. Second, we further improve the ESA technique by providing a sequence of heuristic strategies to clean the interpretation vector which we observe contains a fair amount of noise and is not precise enough to represent the contextual clues related to topics of interest. Third, to the best of our knowledge, little work has been done to consider ESA as an effective aid in cross-document knowledge discovery. In this work, built on the traditional BOW text representation for content analysis, we successfully integrate ESA into the discovery process to help measure the semantic relatedness between concepts. We envision this integration would also benefit other related tasks such as question answering and cross–document summarization. Last, the approach presented here is able to boost concepts that are most closely related to the topics to higher rankings, compared to the widely used TF-IDF based ranking scheme.

The remainder of this paper is organized as follows: Section 2 describes the related work. Section 3 briefly introduces how concept chain queries work. In Section 4, we discuss the ESA model in detail and present our method to integrate the ESA approach into concept chain queries. Experimental results are presented and analysed in Section 5, and is followed by the conclusion and future work given in Section 6.

2 Related Work

There has been work on discovering connections between concepts across documents using social network graphs, where nodes represent documents and links represent connections (typically URL links) between documents. However, much of the work on social network analysis has focused on special problems, such as detecting communities [7] [12]. Our previous work [1] introduced Concept Chain Queries (CCQ), a special case of text mining focusing on detecting cross-document links between concepts in general document collections (without hyperlinks). This was motivated by Srinivsan's closed text mining algorithm which was built within the discovery framework established by Swanson and Smalheiser [4]. Specifically, the solution proposed attempted to generate concept chains based on the "Bag of Words" (BOW) representation and extended the technique in [2] by considering multiple levels of interesting concepts instead of just one level as in the original method. Each document in [1] was represented as a vector containing all the words appearing in the relevant text snippets in the corpus but did not take any auxiliary knowledge into consideration, whereas in this new solution, in addition to content analysis, we further examine the potential of integrating the Explicit Semantic Analysis (ESA) technique to better serve this task which effectively incorporates more comprehensive knowledge from Wikipedia. Related works attempting to overcome the limitations of BOW approach and integrate the background knowledge into text representation have also been reported in categorization and knowledge discovery applications. For example, WordNet was utilized in [6] to improve the BOW text representation and Scott et al [8] proposed a new representation of text based on WordNet hypernyms. These WordNet-based approaches were shown to alleviate the problems of BOW model but are subject to relatively limited coverage of Wordnet compared to Wikipedia, the world's largest knowledge base to date. Gabrilovich et al [5] 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, Gabrilovich et al also [3] 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 vector of Wikipedia-based concepts (articles), called an interpretation vector [3]. [3] 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 this work and to tackle the above problem we further develop a sequence of heuristic strategies to filter out irrelevant concepts and clean the vector. Another interesting work is an application of ESA in a cross-lingual information retrieval setting to allow retrieval across languages [9]. In that effort the authors performed article selection to filter out those irrelevant Wikipedia articles (concepts). However, we observe the selection process resulted in the loss of many dimensions in the following mapping process, whereas in our proposed approach, the process of article selection is postponed until two semantic profiles have been merged so that the semantic loss could be possibly reduced to the minimum.

3 Concept Chain Queries

As described earlier, concept chain query (CCQ) is attempting to detect links between two concepts (e.g., two person names) across documents. A concept chain query involving concept A and concept B intends to find the best path linking concept A to concept B. The paths found stand for potential conceptual connections between them.

3.1 Semantic Profile for Topic Representation

A semantic profile is essentially a set of concepts that together represent the corresponding topic. 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.

$$profile(T) = \{ST_1, ST_2, ..., ST_n\} \tag{1}$$

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.

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

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 $TF * IDF$ weighting scheme and then normalize the weights:

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

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.

Table 1. Semantic Type - Concept Mapping

Semantic Type	**Instances**
Religion	Islam, Muslim
Human Action	attack, killing, covert action, international terrorism
Leader	vice president, chief, governor
Country	Iraq, Afghanistan, Pakistan, Kuwait
Infrastructure	World Trade Centre
Diplomatic Building	consulate, pentagon, UAE Embassy

3.2 Concept Chain Generation

We adapt Srinivasan's closed discovery algorithm [2] to build concept chains for any two given topics. Each concept chain generated reveals a plausible path from concept A to concept C (suppose A and C are two given topics of interest). The algorithm of generating concept chains connecting A to C is composed of the following three 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 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.
3. Expand the concept chain using the created BP profile together with the topics to build additional levels of intermediate concept lists 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 Utilizing Wikipedia Knowledge in Concept Chain Queries

Wikipedia is currently the largest human-built repository in the world. In this effort, we are attempting to improve our query model through integrating the Explicit Semantic Analysis (ESA) [3] technique that uses the space of Wikipedia articles to compute semantic relatedness between texts. In ESA, each term is represented by a vector storing the term's association strengths to Wikipedia articles and each text fragment is mapped to a weighted vector of Wikipedia concepts called an interpretation vector. Therefore, computing semantic relatedness between any two text fragments is naturally transformed into computing the Cosine similarity between interpretation vectors of two texts.

4.1 Document Representation with ESA

In ESA, each article in Wikipedia is treated as a Wikipedia concept (the title of an article is used as a representative concept to represent the article content), and each document given is represented by an interpretation vector containing related Wikipedia concepts (articles) with regard to this document. Formally, a document d can be represented as follows:

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

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

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

Where $tf_d(w_j)$ is the occurrence 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 can be reduced to within 200-300 ms. In concept chain queries, the topic input is always a single concept (a single term or phrase), and thus equation (5) can be simplified as below as $tf_d(w_j)$ always equals to 1:

$$as(d,a_i) = \sum_{w_j \in d} tf \bullet idf_{a_i}(w_j) \tag{6}$$

4.2 Interpretation Vector Cleaning

As discussed above, the original ESA method is subject to the noise concepts introduced, especially when dealing with multi-word phases. 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, we have developed a sequence of heuristics to clean the vector. Basically, we use a modified Levenshtein Distance algorithm to measure the relevance of the given topic to each Wikipedia concept generated in the interpretation vector. Instead of using allowable edit operations of a single character to measure the similarity between two strings as in the original Levenshtein Distance algorithm, we view a single word as a

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) ≤ 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 thatLevenshteinDistance(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) ≤ 1, then keep v_i in V, otherwise, remove v_i from V.

Fig. 1. The Interpretation Vector Cleaning Procedure

unit for edit operations, and thus the adapted algorithm can be used to compute the similarity between any two text snippets. The heuristic steps used to remove noise concepts are illustrated in Figure 1.

4.3 Integrating ESA into Concept Chain Queries

Given the advantages of using ESA as a semantic representation method, we integrate the kernel of ESA into our concept chain queries, aiming to improve search quality. Specifically, we build interpretation vectors for both of the two given topics as well as each intermediate concept in the merged BP profile, and then apply the cleaning procedure on these vectors to remove noise concepts. Finally, we compute Cosine similarities between interpretation vectors for the topics and each concept in the BP profile. The new model of answering concept chain queries is illustrated in Figure 2.

A combination of techniques of BOW representation and ESA method is considered in this new solution, and therefore two types of ranking schemes are integrated as follows.

TF*IDF-Based Similarity. As the most widely used document representation, the BOW representation has demonstrated its advantages. It is simple to compute and strictly sticking to the terms occurring in the document, thereby preventing outside noise concepts that do not appear in the document from flowing into the feature space of the representation. Given these benefits, a variation of *TF * IDF* weighting scheme under the context of BOW representation and semantic types (detailed in Section 3.1) is incorporated into our final ranking where a sentence window is employed to further filter the noise that may incur. We call this kind of similarity TF*IDF-based similarity.

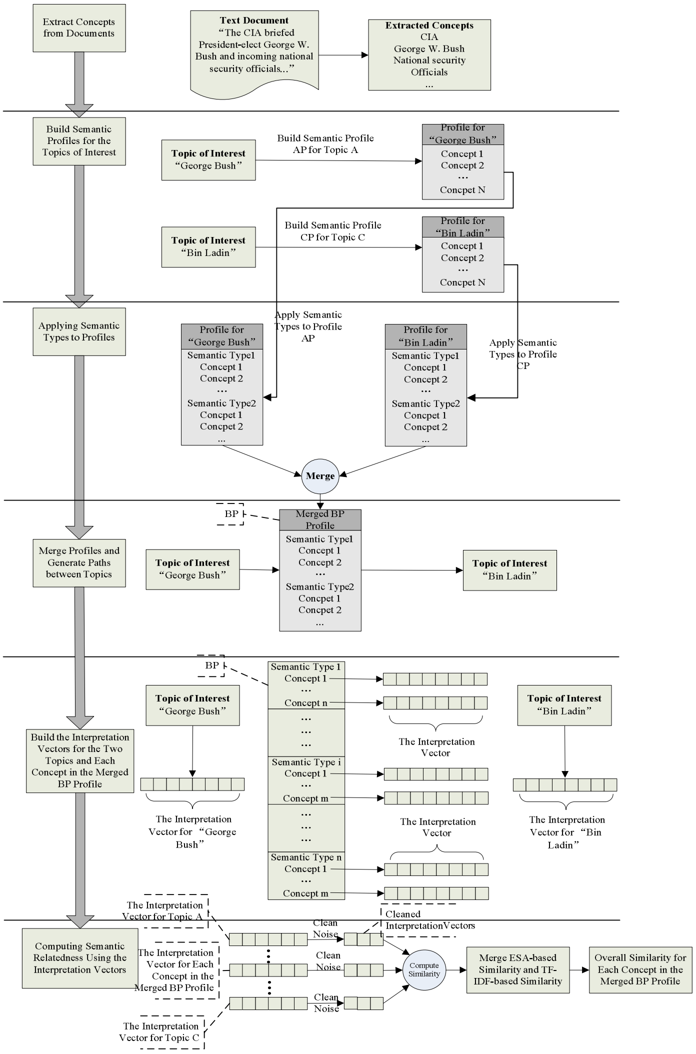
Extract Concepts from Documents
Text Document
“The CIA briefed President-elect George W. Bush and incoming national security officials...”
Extracted Concepts
CIA
George W. Bush
National security
Officials
...
Build Semantic Profiles for the Topics of Interest
Topic of Interest
“George Bush”
Build Semantic Profile AP for Topic A
Profile for “George Bush”
Concept 1
Concept 2
...
Concpet N
Topic of Interest
“Bin Ladin”
Build Semantic Profile CP for Topic C
Profile for “Bin Ladin”
Concept 1
Concept 2
...
Concpet N
Applying Semantic Types to Profiles
Apply Semantic Types to Profile AP
Profile for “George Bush”
Semantic Type1
Concept 1
Concept 2
...
Semantic Type2
Concept 1
Concpet 2
...
Profile for “Bin Ladin”
Semantic Type1
Concept 1
Concept 2
...
Semantic Type2
Concpet 1
Concpet 2
...
Apply Semantic Types to Profile CP
Merge
Merge Profiles and Generate Paths between Topics
BP
Merged BP Profile
Semantic Type1
Concept 1
Concept 2
...
Semantic Type2
Concpet 1
Concpet 2
...
Topic of Interest
“George Bush”
Topic of Interest
“Bin Ladin”
Build the Interpretation Vectors for the Two Topics and Each Concept in the Merged BP Profile
BP
Semantic Type 1
Concept 1
...
Concept n
...
Semantic Type i
Concept 1
...
Concept m
...
Semantic Type n
Concept 1
...
Concept m
Topic of Interest
“George Bush”
Topic of Interest
“Bin Ladin”
The Interpretation Vector
The Interpretation Vector for “George Bush”
The Interpretation Vector
The Interpretation Vector for “Bin Ladin”
Computing Semantic Relatedness Using the Interpretation Vectors
The Interpretation Vector for Topic A
The Interpretation Vector for Each Concept in the Merged BP Profile
The Interpretation Vector for Topic C
Clean Noise
Clean Noise
Clean Noise
Cleaned InterpretationVectors
Compute Similarity
Merge ESA-based Similarity and TF-IDF-based Similarity
Overall Similarity for Each Concept in the Merged BP Profile

Fig. 2. The new model of answering concept chain queries

ESA-Based Similarity. Unlike the BOW model, ESA makes use of the knowledge outside the documents themselves to compute semantic relatedness. It well compensates for the semantic loss resulted from the BOW technique. The relatedness between two concepts in ESA is computed using their corresponding interpretation vectors containing related concepts derived from Wikipedia. In the context of concept chain queries, we compute the Cosine similarity between 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. We call this kind of similarity ESA-based Similarity.

Integrating TF*IDF-Based Similarity and ESA-Based Similarity into the Final Ranking. The TF*IDF-based similarity and ESA-based similarity are finally linearly combined to form a final ranking for concepts generated in the intermediate profiles:

$$S_{overall} = \lambda \bullet S_{TFIDF} + (1-\lambda) S_{ESA} \tag{7}$$

Where λ is a tuning parameter that can be adjusted based on the preference on the two similarity schemes in the experiments. S_{TFIDF} refers to the TF*IDF-based similarity and S_{ESA} the ESA-based similarity.

5 Empirical Evaluation

We performed our evaluation using the 9/11 counterterrorism corpus. The Wikipedia snapshot used in the experiments was dumped on April 05, 2011.

5.1 Processing Wikipedia Dumps

As an open source project, the entire content of Wikipedia is easily obtainable. All the information from Wikipedia is available in the form of database dumps that are released periodically, from several days to several weeks apart. The version used in this work was released on April 05, 2011, which was separated into 15 compressed XML files and totally occupies 29.5 GB after decompression, containing articles, templates, image descriptions, and primary meta-pages. We leveraged MWDumper [13] to import the XML dumps into our MediaWiki database, and after the parsing process, we identified 5,553,542 articles.

5.2 Evaluation Data

We performed concept chain queries on the 9/11 counterterrorism corpus. This involves processing a large open source document collection pertaining to the 9/11 attack, including the publicly available 9/11 commission report. 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. The evaluation data generated includes 1,346 chains of length 1, 6,709 chains of length 2, 6,036 chains of length 3, and 400 chains of length 4.

5.3 Experimental Results

Parameter Settings. As mentioned in Section 4.3, we use a combination of TF*IDF-based similarity and ESA-based similarity to rank the links detected by our system. λ in Equation 7 is a parameter that needs to be tuned so that 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 value of λ was tuned in the range of [0.1, 1] and the best performance was obtained when λ was set to 0.2.

Query Results. The effectiveness of our approach is measured by precision and recall of the concept chains the system generated. Table 2 makes a comparison between the searching results of concept chain queries using a BOW-based approach (CCQ-BOW) and concept chain queries with ESA integrated (CCQ-BOW-ESA). In Table 2, 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 of Table 2 stand for the precision and recall values (P for precision, while R for recall).

Table 2. Searching Results of Top N Concepts

		CCQ-BOW/CCQ-BOW-ESA					
		S1	**S2**	**S5**	**S10**	**S20**	**S30**
L1	P	0.664/0.710	0.659/0.686	0.656/0.663	0.664/0.672	0.662/0.675	0.668/0.674
	R	0.340/0.339	0.474/0.474	0.627/0.643	0.767/0.782	0.848/0.878	0.902/0.918
L2	P	0.741/0.762	0.733/0.755	0.714/0.749	0.714/0.744	0.709/0.742	0.710/0.784
	R	0.269/0.285	0.390/0.410	0.547/0.591	0 660/0.720	0.746/0.825	0.784/0.866
L3	P	0.754/0.769	0.745/0.765	0.723/0.761	0.721/0.754	0.716/0.751	0.716/0.748
	R	0.261/0.279	0.380/0.403	0.538/0.585	0.649/0.713	0.734/0.819	0.771/0.860
L4	P	0.261/0.432	0.296/0.369	0.578/0.693	0.252/0.286	0.261/0.279	0.260/0.268
	R	0.233/0.340	0.450/0.480	0.670/0.700	0.630/0.810	0.940/0.940	0.970/0.970

Table 3. Searching Results of Query Pair "Abdel Rahman :: Blind Sheikh"

BP Term	**Semantic Type**	**Term Rank**					
		CCQ-BOW			**CCQ-BOW-ESA**		
		P=5	P=10	P=20	P=5	P=10	P=20
Islamic Group	Corporation	4	4	4	2	2	2
Saudi Arabia	Country	3	3	3	1	1	1
CIA	Government	-	7	7	3	3	3
President Clinton	Man	-	9	9	-	8	8
Jihad	Organization	-	-	-	-	-	15
Khifa Refugee Center	Organization	-	7	7	-	7	7
Omar Abdel Rahman	Man	1	1	1	1	1	1
Terrorist	Person	-	10	10	2	2	2
Islamist	Religion	-	9	9	3	3	3
Abdullah	Man	-	-	11	-	7	7
Muslim	Religion	4	4	4	2	2	2
Government	Government	-	8	8	2	2	2
Bin Ladin	Person	3	3	3	2	2	2
Attack	Human Action	3	3	3	2	2	2

Table 3 shows an example of the improvement of concept rankings of key BP terms by integrating ESA into answering concept chain queries. The terms in this table were produced by running a query: "Abdel Rahman" and "Blind Sheikh". P=5 means we keep the top 5 concepts within each semantic type of BP. Each entry is the ranking position of the corresponding key concept in BP. The entry value "-" means the concept cannot be found in the results. It is obvious that for most of the key BP terms, the ranks are boosted. The concepts in Table 3 are strongly related to Abdel Rahman who is also known as "The Blind Sheikh". For instance, Abdel Rahman was a blind Egyptian Muslim leader and accused of being the leader of "The Islamic Group" which is considered as a terrorist organization by the United States government. Therefore, the concepts "Islamic Group", "Islamist", "Muslim" and "Terrorist" typically characterize his identity.

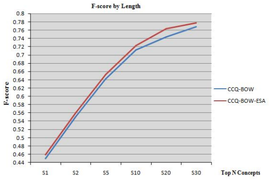

Fig. 3. Result of Chains of Length 1

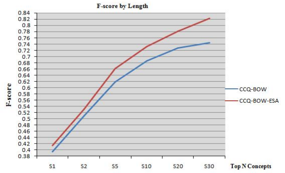

Fig. 4. Result of Chains of Length 2

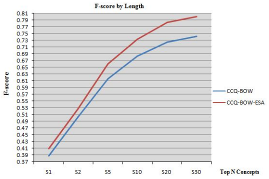

Fig. 5. Result of Chains of Length 3

We further use $F-measure$ to interpret the query results as a weighted average of the precision and recall. Figures 3 through 5 compare the searching results graphically between concept chain queries with BOW (CCQ-BOW) and concept chain queries with ESA integrated (CCQ-BOW-ESA) in terms of how the integrated solution would improve the query model for chains of different lengths. The X-axis indicates the number of concepts kept in each semantic type in the searching results (S_N means we keep the top N), 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 increases from 1 and 5. It is also obvious that our new model consistently achieves better performances for different lengths than the solution based on a BOW approach.

6 Conclusion and Future Work

This paper proposes a new solution for improving cross-document knowledge discovery through our previously introduced concept chain queries, which focus on detecting semantic relationships between topics across documents where revealed semantic paths may lead to early discovery of hypotheses. In this effort, we propose a hybrid approach that integrates Wikipedia knowledge into the traditional BOW model, which complements existing knowledge in text corpus and further improves search quality. We present experiments that demonstrate the effectiveness of this new approach. Specifically, the key terms representing significant relationships between topics are greatly boosted, compared with the method using the $TF * IDF$ ranking scheme.

Future direction includes exploration of other potential resources provided by Wikipedia to further improve query processing, such as categories that relevant Wiki articles belong to and the underlying category hierarchy. These valuable information resources may be combined with our defined semantic types to further contribute to ontology modeling. As a cross language knowledge base, we also plan to combine Wiki knowledge in a cross-lingual setting to better serve different query purposes.

Acknowledgments. We would like to thank the anonymous reviewers for their comments and suggestions. This publication was made possible by Grant Number EPS-0814442 from NSF ND EPSCoR Infrastructure Improvement Programs (IIP).

References

1. Jin, W., Srihari, R.K.: Knowledge Discovery across Documents through Concept Chain Queries. In: Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006) Workshop on Foundation of Data Mining and Novel Techniques in High Dimensional Structural and Unstructured Data, Hong Kong, China (2006)
2. Srinivasan, P.: Text Mining: Generating hypothesis from Medline. JASIST 55, 396–413 (2004)
3. Gabrilovich, E., Markovitch, S.: Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), pp. 1606–1611 (2007)
4. 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)
5. Gabrilovich, E., Markovitch, S.: Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. In: AAAI 2006 (2006)
6. Hotho, A., Staab, S., Stumme, G.: Wordnet improves text document clustering. In: Proceedings of the Semantic Web Workshop at SIGIR 2003 (2003)
7. Gibson, D., Kleinberg, J., Raghavan: Inferring web communities from link topology. In: Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia, pp. 225–234
8. Scott, S., Matwin, S.: Text Classification Using WordNet Hypernyms, pp. 45–52. Association for Computational Linguistics (1998)
9. Sorg, P., Cimiano, P.: Cross-lingual Information Retrieval with Explicit Semantic Analysis. In: CLEF Workshop (2008)
10. Wang, P., Hu, J., Zeng, H.-J., Chen, L., Chen, Z.: Improving Text Classification by Using Encyclopedia Knowledge. In: Seventh IEEE International Conference on Data Mining (ICDM) (2007)
11. Srihari, R.K., Li, W., Niu, C., Cornell, T.: InfoXtract: A Customizable Intermediate Level Information Extraction Engine. Journal of Natural Language Engineering (2006)
12. Faloutsos, C., McCurley, K., Tomkins, A.: Fast discovery of connection subgraphs. In: Proceeding of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004, pp. 118–127 (2004)
13. MWDumper. Software, `http://www.mediawiki.org/wiki/Manual:MWDumper`