Explicit Semantic Path Mining via Wikipedia Knowledge Tree

Tian Xia
School of Information Resource Management
Renmin University of China
xiat@ruc.edu.cn

Miao Chen
Data To Insight Center
Indiana University
Bloomington
miaochen@indiana.edu

Xiaozhong Liu
Department of Information and Library Science
Indiana University Bloomington
liu237@indiana.edu

ABSTRACT

While classical bag-of-word (BoG) approaches represent text content in the word level, recent studies show that knowledge-based concept indexation is a promising approach to further enhance the text search and mining performance. In this study, we propose a new knowledge indexation/extraction method, Explicit Semantic Path Mining (ESPM), for knowledge-base text mining. It has roots in a concept-based vector constructing method, Explicit Semantic Analysis (ESA), which has shown success in text mining tasks. For this new method, given an input piece of text, ESPM can efficiently identify the independent and optimized semantic path(s) on a concept map, which is, in this study, the Wikipedia category tree. Unlike earlier studies focusing on BoG based vector space, ESPM is a semantic path mining algorithm, which generates the top down semantic categories of a given text by leveraging the rich link information between Wikipedia categories and articles. Preliminary experiment based on ODP data shows ESPM delivers high quality independent semantic paths from both precision and ranking viewpoints.

Keywords

Explicit Semantic Path Mining, Explicit Semantic Analysis, Wikipedia, Semantic Category, Text Mining

INTRODUCTION

The success of modern text information retrieval/recommendation systems is based on the Bag-of-Word (BoG) hypothesis, which assumes the word is the basic semantic unit of the given document. However, recent studies begin to question the weakness of BoG approach, while challenging the traditional TFIDF based text index and retrieval methods. For instance, (Burke 2000) tried to identify the knowledge map of the corpus. (Liu et al., 2014), on the contrary, positioned each document on the heterogeneous graph for information retrieval.

Explicit Semantic Analysis (ESA) is another important effort to achieve this goal. The ESA algorithm can generate concept-level features for a given word or text document. It enriches feature set of text by adding concept-based features to the feature space. By concept feature, it means higher-level semantic units can be considered orthogonal to each other, and the concepts (text) may not explicitly exist in the given text. Experiments (Anderka & Stein, 2009; Hotho, Staab & Stumme, 2003; Scholl, 2010) have shown that ESA, when combining word and concept features, can enhance text categorization performance on standard corpus over the classical bag-of-words approach (Gabrilovich & Markovitch, 2007).

Unfortunately, while a number of studies successfully employed ESA to enhance the text mining performance, the accuracy, or to say the quality, of ESA vector is still problematic. For instance, given the text "***Iraq's Top Shiite Cleric Calls for New Government***", the top ranked ESA concepts include "***John Flower***", "***Iraqi National List***", "***Hammadi Ahmad***", and "***Promised Day Brigades***", which can hardly represent the semantics of the given text, even though they may be statistically useful (for text mining purpose).

In order to cope with this problem, in this study, we propose a new method Explicit Semantic Path Mining (ESPM) by leveraging the rich linkage and categorical relationships of Wikipedia. For each given text, instead of generating the semantic concept vector, ESPM identifies an optimized *semantic path* on the Wikipedia category tree. For instance, for the same input above, by using ESPM, we got the following semantic path: "***Politics → Politics by country → Politics of Iraq → Iraqi nationalism***", which makes more sense comparing with the ESA concept vector. As another example, for input "***Construction of world's biggest optical telescope starts with a bang***", ESPM finds the path: "***Science→Scientific instruments→Astronomical***" on the Wikipedia category tree.

Evaluation with ODP (Open Directory Project) data shows ESPM is an efficient algorithm to explore the deep

ASIST 2014,November 1-4, 2014, Seattle, WA, USA.

categorical knowledge of the given text, which can be potentially important for text mining and retrieval tasks.

RELATED WORK

The vector space model, along with other related approaches, is a widely used representation for text documents. It assumes independence of words and uses a vector of words to represent documents. Similarly, the ESA approach also represents text in a vector, while by using a vector of concepts as opposed to words. The assumption of independence between words usually does not hold in the real world, and ESA, which uses concept-level units from large knowledge bases, has a better chance of holding the orthogonal assumption. It indexes a relatively large corpus that contains concepts and text for concepts, e.g. Wikipedia, to obtain the associations between terms and concepts. For example, for the Wikipedia case, it considers Wikipedia article titles as concepts (also called "concepts" in this paper) and uses the article titles as vector dimensions for representing a text document. It scans through article body of Wikipedia concepts and indexes information of in which Wikipedia article a term occurs.

While the ESA is a general methodology that can be applied on any corpus with concept-level titles or categories, we focus on the Wikipedia use here following several other studies (Minier, Bodo, & Csato, 2007; Scholl, Böhnstedt, García, Rensing & Steinmetz, 2010). Given a Wikipedia dump, which contains m Wikipedia articles (concepts) $\underline{C_j}$ ($j \in [1,\ldots,m]$) and n terms in the text descriptions of the concepts, an $m*n$ index matrix M can be established based on information of what terms are used to describe a concept. In M^T, the transpose of matrix M and thus an $n*m$ matrix, each row represents the concept vector for a term: row i in M^T is the concept vector ($[C_{1i}, C_{2i}, \ldots, C_{mi}]$)for t_i. Feature selection happens after deriving the concept vector for a document and plays an important role in ESA. It involves processing at 2 stages in (Gabrilovich & Markovitch, 2006, 2007): first, at the stage of building term vector for concepts, ESA selects words important to the concepts using information gain, the process of which is also called "attribute selection"; second, ESA employs feature selection on the generated feature again using information gain.

Several studies have been conducted to understand or enhance ESA performance (Anderka & Stein, 2009; Hotho, Staab & Stumme, 2003; Scholl, 2010). These studies mostly use Wikipedia corpus to generate concept vectors, and therefore the resulted vector is a vector of Wikipedia concepts given a text document. For example, Scholl et al. (2010) altered the ESA concept vector scores by integrating article links and categories information with the original ESA index matrix. However, to the best of our knowledge, mining semantic category path given a text is an innovative research question.

METHODOLOGY

Unlike most existing corpus, Wikipedia provides high-quality user-oriented hierarchical category definition. The top levels of categories, in most cases, are defined by professional editors. For instance, the first-level includes 26 general categories, i.e., *Culture*, *Education*, *Environment*, *Politics*, and *Science*, while Wikipedia page authors and contributors define most of the bottom level categories, i.e., *American military personnel killed in the War of 1812*. In this section, we propose the method for mining the ranked important semantic category paths given a free text. Each path connects a number of linked categories from first level to the lower level categories.

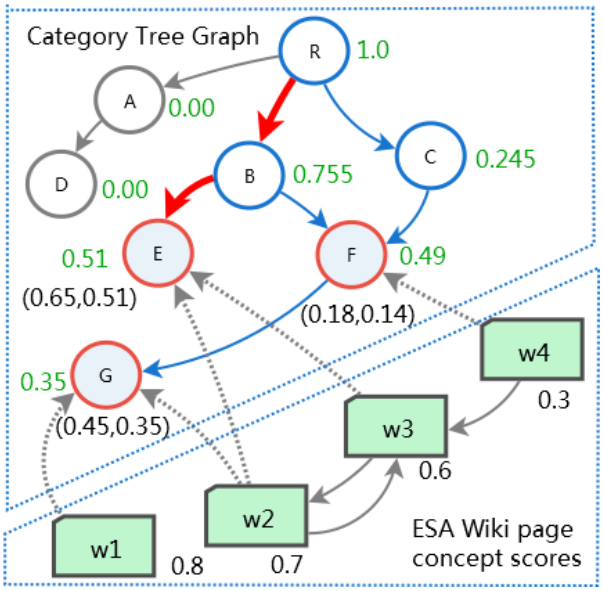

Figure 1. ESPM Process (related Wikipedia pages "vote" for the important category path on the tree

Figure 1 visualizes the ESPM generation progress. Given a text, the related Wikipedia pages vote for the significant semantic path(s) on the Wikipedia category tree graph. In this section, we describe this process in detail.

Given the input free *text*, we utilized ESA algorithm to generate the concept vector, and we assume the top n concepts in the vector, $V_T=\{P(w_1|text), P(w_2|text)\ldots P(w_n|text)\}$, could be semantically related to the given text (even though the noisy concepts can pollute the ESA vector). $P(w_t|text)$ is the ESA inference score (Gabrilovich & Markovitch, 2007), the probability that text is relevant to the Wikipedia concept w_t.

Then, we use all the n concepts in V_T to "vote" for the category (node) importance and probability on the Wikipedia category tree. Note that one concept (Wikipedia page) can belong to multiple Wikipedia categories. In this step, we have two premises. **First**, all the concepts in V_T can be connected via incoming and outgoing page hyperlinks, and the links can be important to help us filter the noisy concepts in the concept vector. As the following formula shows, the probability (importance) of a Wikipedia category

for given *text*, $P(C_i|text)$, is calculated by all the pages belong to it, $w_j \in C_i$, and all the pages linked to those pages and belongs to the same category, $w_k \leftrightarrow w_j (w_k, w_j \in C_i)$. $|w_j \in C_i|$ is the total number of Wikipedia pages in the target category.

$$P(C_i|text) = \frac{\sum_{w_j \in C_i} (\lambda \cdot P(w_j|text) + (1-\lambda) \cdot \frac{\sum_{w_k \leftrightarrow w_j} P(w_k|text)}{|w_k \leftrightarrow w_j|})}{|w_j \in C_i|}$$

In another word, based on this formula, if a number of highly ranked concepts are interconnected in the ESA vector, and they all belong to a specific category, this category can be important. For instance, as Figure 1 shows, because w_2 and w_3 are well-connected, the probability that category E and its related paths to be selected is higher than other categories on the tree.

Once we get the category probability distribution for the given text, we need to normalize category probability, $P'(C_i|text)$, on the tree-like category graph. Because the parent or child nodes may affect the probability that the category is relevant to the text for key path identification. First, we normalize the category probabilities to ensure that the root node probability will always equals to 1.0, which means any text must belong to "something" defined by Wikipedia. Second, we transfer every node's probability to their parents iteratively, $\sum_{C_i \to C_{child\_k}} P(C_{child_k}|text)$ (bottom up); all the nodes' probabilities will be transferred to the root node through all possible paths.

$$P'(C_i|text) = \frac{\sum_{C_i \to C_{child_k}} P(C_{child_k}|text)}{|C_i \to C_{child_k}|}$$

Through bottom-up method, all the possible nodes would be assigned values. Then, we use top-down method to find all possible paths from root node to seed nodes. Define the path weight as the sum of all the category nodes on the path:

$$P(path_k|text) = \frac{\sum_{C_i \in path_k} P'(C_i|text)}{|C_i \in path_k|}$$

As another premise, for ESPM, we need to find a number of *independent paths* on the Wikipedia category tree. For instance, if we find $A \to B \to C$ we don't want find another path $A \to B \to C \to D$, as these two paths are dependent and provide very similar information. To characterize this assumption, we use greedy algorithm to identify the top *k independent* important paths on the tree.

First, we calculate all the relevant paths' weight with aforementioned method. Then, we generate a graph, where each path is conceptualized as a node on the graph (with the node weight = path importance), and if any two paths are dependent, there will be an edge connecting these two nodes. For dependence measure, we utilize the similarity between two paths, i.e., if $Sim(p_i, p_j) > \Lambda$, path p_i and p_j are dependent. On this graph, we first pick the node with the largest weight. Then, we will remove all the nodes (paths) connect to this node. We will repeat this process until all the nodes on the graph are removed and picked. Note that, after this step, we will get a list of ranked paths, which are all independent to others. This greedy algorithm has been proved useful in prior tree mining and feature selection studies (Chakrabarti & Mehta 2010).

EXPERIMENT

In order to verify the ESPM performance, in this experiment, we use ODP (Open Directory Project) data to evaluate the algorithm performance. The aim of the evaluation is to test, 1. the accuracy of the highly ranked semantic paths (ranking evaluation), and 2. how much effort users are expected to spend to find the first correct path in the results. For the first, we use *BestMatch@k* as the evaluation metric, while, for the latter, *MRR* (Mean Reciprocal Rank) is employed.

Each directory, in ODP, has a text snippet (description) and a category path. In this experiment, we utilize the directory text snippet as the text input, and we use ESPM to estimate the semantic paths, $\{p_i\}$. The ODP directory category path, *cp*, is employed as the truth. The similarity between each ESPM path and ODP path, $sim(p_i, cp)$, is calculated by the overlapped nodes on the path *cp* divided by the total number of nodes on *cp*. If $sim(p_i, cp) > \beta$, then, we assume the estimated path is correct, otherwise, the path could be wrong. Note that this judgment data could be biased, as ODP only provide one path. Other relevant paths could be missing in ODP data.

For experiment, we use Wikipedia English dump dated on 2014-03-04 with 10,836,523 articles. 12,894 ODP directories with description snippets are used for ESPM evaluation. For each ODP directory, we use ESA to calculate the top 100 concepts for ESPM ranking generation. For MRR, for different β value, the evaluation results are presented in Table 1.

β value	0.20	0.25	0.30	0.35	0.40
Arts	33.09%	31.68%	30.19%	22.20%	13.80%
Business	17.50%	17.15%	16.49%	15.58%	10.12%
Health	20.58%	14.31%	13.43%	12.55%	7.30%
Science	19.03%	13.64%	9.85%	7.39%	2.99%
Society	18.06%	13.72%	11.74%	8.56%	5.00%
Sports	63.70%	63.57%	63.53%	51.97%	26.44%
Avg	28.87%	26.17%	24.66%	19.73%	10.97%

Table 1. MRR for ESPM Evaluation

For *BestMatch@k*, we also use nDCG to calculate path similarity between original cp_i and generated path p_j. Since cp_i is the target semantic path, we take it as the ideal ordering result, and then calculate the matching degree as follows:

$$MatchScore(cp_i, p_j) = \frac{DCG_{p_j|cp_i}}{IDCG_{p_j|cp_i}} = \frac{\sum_{k=1}^{|p_j|} \frac{rel(cp_i, c_{jk})}{log_2(1+k)}}{\sum_{k=1}^{|cp_i|} \frac{1}{log_2(1+k)}}$$

if the k^{th} category $c_{jk} \in p_j$ occurred in path cp_i, then $rel(cp_i, c_{jk}) = 1$, otherwise is *0*.

If BestMatch score is high, users are more likely to find the needed semantic path(s) from the top ranked result collection. In another word, with a higher BestMatch score,

ESPM is able to prioritize the important paths over other paths in the results.

Name/Count	BM@1	BM@2	BM@3	BM@4	BM@5	BM@6
Arts/2203	13.72%	19.99%	24.33%	27.08%	29.06%	30.69%
Business/1977	7.55%	12.22%	15.76%	18.13%	20.63%	21.73%
Health/685	7.65%	12.12%	17.12%	21.96%	23.83%	25.94%
Science/1503	6.52%	11.01%	15.61%	18.77%	21.58%	24.22%
Society/4242	6.93%	11.16%	13.97%	16.52%	18.23%	20.53%
Sports/2284	26.39%	29.70%	32.22%	33.44%	34.69%	36.19%
Avg/12894	11.62%	16.15%	19.61%	22.12%	24.05%	25.94%

Table 2. NDCG@k for ESPM Evaluation

Evaluation result shows that:

1. From semantic path accuracy viewpoint, ESPM can generate high quality path ranking with regard to ODP judgment data. Take MRR as a case, while $\beta < 0.4$, most categories can find the correct result in the top 10 paths, which is promising.
2. Some categories' performance outperforms others. In the MRR evaluation, for example, we find Arts, Business, and Sports performance is superior to Science and Society categories.
3. From ranking perspective, BestMatch score stably increases while k increasing, and most categories reach 25% when $k = 6$. The result indicates that users can expect to find the more accurate results in the top-ranked path collection.

CONCLUSION AND FUTURE WORK

In this paper, we propose a novel problem, Explicit Semantic Path Mining (ESPM) for a given text. Unlike prior studies in ESA, three kinds of relations are taken into consideration: in/outgoing links between Wikipedia pages, category-concept relationship, and the hierarchical relations between Wikipedia categories. Semantic category path mining can be important for text mining and retrieval studies.

Evaluation based on ODP data indicates that ESPM is a promising approach to find the semantic categorical path of the given text. As ESPM employed independent path identification algorithms, selected paths in the result collection are independent to others. Therefore, from ranking perspective, the important paths can be prioritized from noisy data, and the result collection can be more informative than other similar methods.

Next step, we will use the ESPM algorithm to solve other tasks, for instance, text classification or information retrieval problems. Meanwhile, we will use more sophisticated probability models, i.e., supervised topic modeling algorithms (Ramage et al., 2009), to characterize the word-category information. However, very large number of categories and the complex relationships between articles and categories will challenge the performance of the topic modeling algorithms.

ACKNOWLEDGEMENTS

This work is funded by the Beijing Higher Education Young Elite Teacher Project, National Social Science Foundation of China (Grant No. 12&ZD220 and 09CTQ027).

REFERENCES

Anderka, M., & Stein, B. (2009, July). The ESA retrieval model revisited. In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retriev-al (pp. 670-671). ACM.

Burke, R. (2000). Knowledge-based recommender systems. Encyclopedia of library and information systems, 69(Supplement 32), 175-186.

Chakrabarti, D., & Mehta, R. (2010). The paths more taken: matching DOM trees to search logs for accurate webpage clustering. In Proceedings of the 19th international conference on World wide web (pp. 211-220). ACM.

Gabrilovich, E., & Markovitch, S. (2007). Harnessing the Expertise of 70, 000 Human Editors: Knowledge-Based Feature Generation for Text Categorization.Journal of Machine Learning Re-search, 8, 2297-2345.

Gabrilovich, E., & Markovitch, S. (2006, July). Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. In AAAI (Vol. 6, pp. 1301-1306).

Hotho, A., Staab, S., & Stumme, G. (2003a). Ontologies improve text document clustering. Paper presented at the Third IEEE International Conference on Data Mining (ICDM'03).

Liu, X., Yu, Y., Guo, C., Sun Y., & Gao L. (2014). A Comparative Study of Academic impact and Wikipedia Ranking, ACM/IEEE Joint Conference on Digital Libraries (JCDL).

Minier, Z., Bodo, Z., & Csato, L. (2007, September). Wikipedia-based kernels for text categorization. In Symbolic and Numeric Algorithms for Scientific Computing, 2007. SYNASC. International Symposium on (pp. 157-164). IEEE.

Scholl, P., Böhnstedt, D., García, R. D., Rensing, C., & Steinmetz, R. (2010). Extended explicit se-mantic analysis for calculating semantic related-ness of web resources. In Sustaining TEL: From Innovation to Learning and Practice (pp. 324-339). Springer Berlin Heidelberg.

Ramage, D., Hall, D., Nallapati, R. & Manning, C.D. (2009). Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In EMNLP '09 Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics.