Keyword Annotation of Biomedical Documents with Graph-based Similarity Methods

Shuguang Wang
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, USA
Email: swang@cs.pitt.edu

Milos Hauskrecht
Department of Computer Science
University of Pittsburgh
Pittsburgh, USA
Email: milos@cs.pitt.edu

ABSTRACT

In this paper, we present a new approach that lets us extract, and represent relations among terms (concepts) in the documents and uses these relations to support various document analysis applications. Our approach works by building a graph of local co-occurrence relations among terms that are extracted directly from text and by defining a global similarity metric among these terms and sets of terms using the graph and its connectivity. We demonstrate the benefit of the approach on the problem of MeSH keyword annotation of documents based on their abstracts.

INTRODUCTION

One of the fundamental challenges of information and text analysis is to capture the similarity in between two documents. The early studies that are based on cosine metric are very sensitive to the document size and term vocabulary used in these documents. More recent document and text analysis methods rely primarily on latent semantic analysis, such as LSI [1], PLSI [2], and LDA [3]. In these latent semantic models, documents and its term components are projected into a latent space (representing for example topics or related words) and all document/text related analysis is performed in this space. The limitation of these models is how to choose the number and the structure of latent factors defining the latent space.

In this work, we study a new class of document/text similarity metrics based on the term-term graph kernels [4] that let us relate documents or their parts. Our graph-based approach and metrics are general enough to define similarity between texts of different size and they do not require one to define any latent space projection. The graph-based metrics are defined directly on the term space and can be extended to metrics on sets of terms (i.e., documents).

The graph that defines the metric is built automatically from the document corpus by analyzing, recording and combining pairs of terms that co-occur on the sentence level in these documents. The pairwise co-occurrence is then used to define links in the term-term graph. To model global term-term similarities, we study two types of graph-induced kernels (or similarity measures) that aggregate and local term-term co-occurrence relation represented in the graph: the shortest path and the resistance kernels. We evaluate both of these derived metrics on the keyword annotation problem in which we assign keywords for a new document using keywords given to documents similar to it. We use abstract of Medline documents and their MeSH keywords to show that our graph induced similarity approach perform well and outperforms the baselines in terms of predicting correct MesH labels.

PROBLEM DESCRIPTION

Several studies investigated the problem of classifying biomedical documents using MeSH labels [5], [6], [7], [8]. All together there are over 24,000 possible MeSH keywords and each document has multiple MeSH labels. The huge number of possible annotations a document may be assigned to implies that many standard classification models such as SVM, neural networks, decision trees and etc can not be easily adapted to this keyword annotation task.

In couple of recent studies[8], [9], the authors compared various methods including Medical Text Indexer [10], EAGL [11], and K-nearest neighbor (KNN) method [12]. They showed that KNN approach was the best method to classifying documents based on MeSH labels. KNN is a non-parametric method that requires to know the similarity among documents. However this study did not explicitly define the similarity among documents. Instead, it used a search engine [8] to identify similar documents for a given query document. Then the MeSH labels of the 10 most similar documents returned by the search engine were used as the labels for the query document. In this paper, we present a new nonparametric keyword assignment method that defines in a principled way a document similarity metric and uses it to infer keywords for the new unlabeled document.

METHODOLOGY

In this work we focus on and develop a non-parametric approach to assign keywords to a document by identifying

978-1-4673-2560-8/12/$31.00 ©2012 IEEE

all keyword-annotated documents that are similar to the target document and by determining a keyword label to be assigned to the target document by calculating the weight for each keyword using similar documents and their associated keywords. The weight associated with a specific keyword is simply the sum of similarities between the target document and all other documents associated with the same keyword.

More formally, let $D = \{d_1, d_2, \dots, d_N\}$ be a set of document-keyword pairs such that $d_i = (X_i, m_i)$, where X_i denotes the document text of varied length and $m_i = \{m_i^{(1)}, m_i^{(2)}, \dots m_i^{(q_i)}\}$ is a set of q_i keywords assigned to the ith document from a keyword set $\mathcal{M}$. Let K defines a similarity metric (or kernel) $K(X_i, X_j)$ over the document space. Then for a target (unlabelled) document X^* we determine the weight of a keyword m^* as:

$$w(X^*, m^*) = \sum_{d_i : m^* \subset m_i} K(X^*, X_i). \tag{1}$$

This lets us calculate the weight of each keyword $m \subset \mathcal{M}$ for the document X^* and sort the keywords according to their weights. The sorted keywords list represents the order we use to predict individual keyword labels and their sets, for example, the best N keywords.

While the keyword selection process is relatively straightforward, different orders of keywords can be induced by different document similarity metrics $K(X_i, X_j)$. So the main challenge is how to design an appropriate document similarity metric.

Modeling text similarities

Our objective is to derive the similarity among documents. We achieve this in three steps. First, we extract a weighted term-term association network using term-term co-concurrences in the text corpus. After that we use the term co-occurrence graph to define the term-term similarities with the help of various graph kernels. Finally, we extend the pair-wise term-term similarities to similarity among two arbitrary sets of terms (i.e., documents).

Term-term association graph

We propose to build the term-term association network from the corpus of documents. We extract pair-wise co-occurrences (associations) among terms at the sentence level. We ignore common stop words such as 'a', and 'the', and use Porter Stemmer [13] to normalize spelling variations. If two co-occurred terms are connected by an undirected edge in the network. The assumption here is that two co-occurred terms tend to be directly associated via some relation. Of course co-occurrences of terms in a sentence may not always imply the presence of a direct relation among them. To alleviate the problem, each link is assigned a weight W_{ij} that reflects how frequent the co-occurrence of terms on the sentence level really is. The intuition is that is if two terms co-occur in many different sentences/documents, they are very likely to be related.

By aggregating all pair-wise (local) term-term associations and their frequency we construct the complete term-term association graph. One of the important features of the graph construction method is that the network can be constructed from a large corpus of unlabelled documents very efficiently without any human expert input.

Shortest Path Kernel

The term-term association graph and its links reflect direct relations among different terms and their strength. Our first solution is to define the similarity with the help of the 'strongest' (i.e., shortest) association path connecting the two terms.

Given the lengths of all paths, the length of the shortest path for any pair of nodes is the minimum of all (direct and indirect) path distances. A standard way to compute the shortest paths between all pairs of nodes in a graph is the Dijkstra's algorithm[14]. The computational cost of the solution is $O(N^3)$. However, we note that this metric can be calculated offline so it is not necessary to compute it at the time of the inference.

Resistance Kernel

The shortest distance similarity metric for any pair of terms is defined by the best possible path between their nodes. However, this may not be the best metric for a more complex graph with many different paths connecting two nodes. Intuitively, if more parallel paths between two nodes exist, the strength of association in between terms should be higher than the strength induced by any of these individual paths. The shortest path metric ignores the presence of multiple paths and considers only the best path.

To account for both the serial and parallel association graph connections in defining the global term-term similarity metric, we propose to interpret the association graph as a resistance network. Figure 1 illustrates the resistance network one would obtain from a weighted association network. In this case, the links and their weights in the graph are replaced with connections with resistances corresponding to their weights. More specifically, a weight W_{ij} between nodes i, j in the original weighted graph defines the electric conductance C_{ij} of the connection that is the reciprocal of its electric resistance $R_{ij} = 1/C_{ij} = 1/W_{ij}$.

In order to compute the pair-wise resistance distances for all pairs of nodes in the graph, we need to solve pseudo inverse of the Laplacian matrix of the graph [15]. The computational cost of it is $O(N^3)$. Again, We can always define the kernel offline for the efficiency purpose.

Set-to-set similarities

From pair-wise term distance defined using graph kernels, we can extend it to support more useful set-to-set similarity

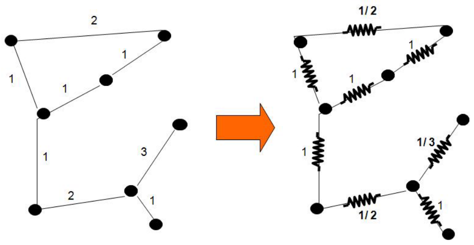

Figure 1. Building a resistance network from an association network

inference to compute document similarity. In order to infer the similarity between two sets of terms, S and T, we collapse all the elements in S and T to two auxiliary nodes S' and T' in the graph. All the edges inside S and T are removed and the edge weights are aggregated from S and T to the other nodes. Thus the set-to-set similarity inference is transformed into a pair-wise similarity inference on these two new auxiliary nodes. This transformation implies that the structure of the association graph is changed and the kernels have to be re-computed. To avoid expensive re-computation, we adopted Nyström approximation [16] to efficiently estimate the similarity between the new auxiliary nodes. For more details, please refer to Amizadeh et al. 2011.

EVALUATION

We perform two experiments to evaluate our method. The first one is a MeSH label prediction task in which we predict a list of MeSH labels for a given document. This experiment is very similar to the evaluation presented in [8]. The second experiment compares our method with previous methods at the document similarity level. We believe the second experiment is more informative as the results of the evaluation reflect more closely how well different methods can derive similarity between documents.

MeSH Prediction

In this experiment, we implement the KNN method that is similar to method described in [8]. The method works as follows. Each document we want to label with MesH labels is fed into the state of the art search engine, Lemur, to find 10 documents most similar to it. The MeSH labels in these 10 documents are then ordered based on the relevance score. The score of a MeSH label is determined by summing the retrieval score of the 10 most similar documents that have been assigned that label.

We randomly selected about 50,000 non empty documents in MEDLINE Baseline 2008 distribution to extract the association network and define the pair-wise resistance distance kernel. We also randomly selected another 500 testing documents to predict their MeSH labels. Following [8], we use an intuitive document perspective metric, Precision at 10 (P10), to measure how well we can predict the MeSH labels on the test documents. This metric measures how many of the first 10 suggested labels correspond to manual annotations.

Table I summarizes the comparison between the KNN method and our resistance kernel method. Apparently, the proposed resistance kernel method is a better alternative and the difference is statistically significant.

Methods	P10
KNN with Lemur	0.40
Resistance	**0.45**

Table I
P10 OF MESH PREDICTION. THE BEST SCORE IS IN BOLD.

Document Similarity

In this experiment, we demonstrate how well our methods can estimate document similarities. We randomly selected 3000 non-empty abstracts from MEDLINE Baseline 2008 distribution for the evaluation. We determine the "real" similarities among these 3000 documents based on their common MeSH labels, i.e., the more common labels between a pair of documents, the more similar these two documents are. We adopt a standard metric, DICE coefficient [17], to compute the document similarities.

We conduct the evaluation using the leave one out setting. For each of the 3000 documents, we use various methods to estimate the similarities between the document to the rest and order them. All together, for each method we had a list of 3000 ordered documents. In addition, we constructed the 'true' ordering of these documents based on DICE similarity generated for their MeSH keywords. A good similarity metric should generate very similar orderings as the true ordering. We compare the ordered list of 3000 documents generated by our methods to the true ordering and evaluate how well our methods are able to estimate the document similarities.

We assess quality of the similarity ordering by measuring how many misplacements (in %) are introduced using various methods when compared to the true ordering induced using MeSH labels. More specifically, we count the number of swaps that are necessary to make a candidate ordering the same as the true ordering. This method can be seen as a generalized version of the Area Under ROC Curve measure proposed in [18].

As a baseline we use of the standard scoring a standard scoring method, TFIDF[19], that is widely used statistical measure in many search engines to determine the document similarities. We compare this baseline to two proposed graph kernel methods. We extract the term-term association network from the these 3000 documents without using any MeSH labels. We parse these documents and identify associations (or frequent co-occurrences) among terms at sentence level. Then we define the shortest path and resistance kernels over the extracted association network as we have described in Methodology Section. These kernels allow us to compute the pair-wise distances among all documents.

Table II summarizes the comparison of various methods to determine the document similarities. Each entry in the table shows the mean % of misplacements obtained for 3000 test documents and their 95% confidence intervals (CI). The resistance kernel is clearly the best method. The TFIDF method is a very strong baseline and it is one of the best weighting schemes used in search engines to estimate the document similarities. The shortest path kernel does not perform as well as the other methods due to its limitation of using single paths between terms to estimate the similarities.

A more important finding in this experiment is that we can combine the resistance kernel with the TFIDF approach and achieve the best performance. We expand each test document with the five most relevant terms using the resistance kernel and compute the TFIDF scores and document similarities. The resulting combination gives us the best performance.

Methods	**Avg % of misplacements**	**95% CI**
TFIDF	0.2277	[0.2264, 0.2289]
ShortestPath	0.2371	[0.2365, 0.2375]
Resistance	**0.2024**	[0.2021, 0.2028]
TFIDF+Resistance	**0.1960**	[0.1944, 0.1975]

Table II
MISPLACEMENTS(%) FOR VARIOUS DOCUMENT SIMILARITY ESTIMATES. THE BEST SCORE IS IN BOLD.

CONCLUSION AND FUTURE WORK

In this paper, we have presented and tested methods for defining similarity between text using graph induced kernels. Our experiments on a keyword annotation task showed that the resistance distance is particularly suitable for this task. We also demonstrate the possibility of incorporating our method into existing methods and improving the induced similarity metric.

In the future, we plan to investigate how to refine the association network to better represent the relations among terms. We expect that the similarity metric learned over the refined association network will further improve the similarity estimations.

REFERENCES

1. T. K. Landauer, P. W. Foltz, and D. Laham, "Introduction to latent semantic analysis," *Discourse Processes*, vol. 25, pp. 259–284, 1998.
2. T. Hofmann, "Probabilistic latent semantic indexing," in *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval*. ACM Press, August 1999, pp. 50–57.
3. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
4. S. Amizadeh, S. Wang, and M. Hauskrecht, "An efficient framework for constructing generalized locally-induced text metrics," in *IJCAI*, 2011, pp. 1159–1164.
5. S. Sohn, W. Kim, D. C. Comeau, and W. J. Wilbur, "optimal training sets for bayesian prediction of mesh assignment," *Journal of American Medeical Informatics Association*, vol. 15, pp. 546–553, 2008.
6. M. E. Ruiz and P. Srinivasan, "Hierarchical text categorization using neural networks," *Information Retrieval*, vol. 5, pp. 87–118, 2002.
7. R. Rak, L. A. Kurgan, and M. Reformat, "Multilabel associative classification categorization of MEDLINE articles into MeSH keywords," *IEEE Engineering in Medicine and Biology Magazine*, vol. 26, no. 2, pp. 47–55, 2007.
8. D. Trieschnigg, P. Pezik, V. Lee, F. D. Jong, and D. Rebholz-schuhmann, "Mesh up: effective mesh text classification for improved document retrieval," *Bioinformatics*, 2009.
9. M. Huang, A. Névéol, and Z. Lu, "Recommending MeSH terms for annotating biomedical articles." *Journal of the American Medical Informatics Association : JAMIA*, vol. 18, no. 5, pp. 660–667, May 2011.
10. A. Aronson, A. R. Aronson, J. Mork, J. G. Mork, C. Gay, C. W. Gay, S. Humphrey, S. M. Humphrey, W. Rogers, and W. J. Rogers, "The nlm indexing initiative's medical text indexer," in *In Proceedings of the 11th World Congress on Medical Informatics Demner-Fushman and Lin Answering Clinical Questions (MEDINFO 2004)*, 2004, pp. 268–272.
11. P. Ruch, "Automatic assignment of biomedical categories: toward a generic approach," *Bioinformatics*, vol. 22, no. 6, pp. 658–664, 2006.
12. J. Lin and J. W. Wilbur, "PubMed related articles: a probabilistic topic-based model for content similarity," *BMC Bioinformatics*, vol. 8, no. 1, 2007.
13. M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.
14. E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.
15. D. J. Klein and M. Randić, "Resistance distance," *Journal of Mathematical Chemistry*, vol. 12, pp. 81–95, 1993.
16. M. M. S. Kumar and A. Talwalkar, "Sampling techniques for the nyström method," in *AISTATS '09: International Conference on Artificial Intelligence and Statistics 2009*, 2009.
17. A. R. Webb, *Statistical Pattern Recognition*. John Wiley and Sons Ltd., 2002.
18. T. Joachims, "A support vector method for multivariate performance measures," in *ICML '05: Proceedings of the 22nd international conference on Machine learning*. ACM, 2005, pp. 377–384.
19. G. Salton and M. J. McGill, *Introduction to modern information retrieval*. McGraw-Hill, 1983.