Keyphrase-Driven Document Visualization Tool

Gábor Berend and Richárd Farkas
Department of Informatics,
University of Szeged
Árpád tér 2., Szeged, 6720, Hungary
{berendg,rfarkas}@inf.u-szeged.hu

Abstract

The need to navigate through massive document sets is getting common due to the abundant data available around us. To alleviate navigation, tools that are able to grasp the most relevant aspects of document subsets and their relations to other parts of the corpus can be highly beneficial. In this paper, we shall introduce an application[1] that processes and visualizes corpora to reveal the main topics and their relative roles to each other. Our suggested solution combines natural language processing and graph theoretic techniques for the visualization of documents based on their automatically detected keyphrases. Furthermore keyphrases that describe thematically related subcorpora are also extracted based on information-theoretic grounds. As for demonstration purposes our application currently deals with papers published at ACL workshops.

1 Introduction

The abundance of textual data that surrounds us often poses difficulties when we are looking for relevant documents in some field. For instance, when a researcher faces a new problem to be solved, she often has to process large amounts of academic papers that are not necessarily directly related to her field of expertise. Difficulties can arise simply from the amount of data to be processed as well as from the absence of knowledge about which articles to regard as relevant. Various solutions exist that try to alleviate data management, such as assigning keyphrases or generating summaries to documents and document subsets, but probably the most useful way of doing so is to support these approaches with some kind of visualization.

Our application constructs a similarity graph of documents and performs a force-directed layout implementation on that graph. Document similarity can be measured based on multiple criteria, i.e. bibliographic similarity – based on co-authorships and citations – or contextual similarity – based on the shared vocabulary or proper keyphrases that documents have in common. In our demonstration – which provides a visualization framework for ACL workshop papers – we present a contextual similarity-based visualization which is based on **keyphrases**.

In order to determine the kepyhrases of articles our state-of-the-art keyphrase extraction module was utilized. Then similarity between pairs of documents is calculated based on the extracted keyphrases and a similarity graph of the documents is formed.

As a subsequent step document communities, i.e. subcorpora of thematically related articles are formed. The most representative keyphrases are then assigned to the identified document subsets which are determined relying on information theoretic grounds. Using a few keyphrases to describe a cluster can help the users in identifying topics they are interested in.

Representing documents in a bag-of-keyphrases fashion – instead of a bag-of-word one – had multiple benefits, i.e.

1. our representation is less influenced by the problem of measuring similarities in high dimensional spaces and
2. we naturally enjoyed computational benefits by representing documents with their most relevant terms only.

For some empirical support regarding these observations see our previous studies (2013).

[1] available at http://www.inf.u-szeged.hu/~berendg/keyphraseViz

The Companion Volume of the Proceedings of IJCNLP 2013: System Demonstrations, pages 17–20,
Nagoya, Japan, 14-18 October 2013.

2 Related work

Recently, several methods have been suggested for a more effective handling of large document sets.

Quazvinian et al. (2013) proposed a graph-based approach – utilizing the so-called *Citation Summary Network* built from sentences citing a particular paper – to create extractive summaries of scientific articles. They employed their approach not only for single documents but for scientific topics, i.e. multiple documents from the same area as well. Even though their suggested methodology is appealing, it treated the topics to be summarized and the assignment of documents to those topics to be known in advance. Also, summaries can be beneficial in getting to know a topic from a glance, however, it can hardly be utilized to reveal the intra-topic document relations, nor the relatedness of different topics to each other.

Topic models such as Latent Dirichlet Allocation (Blei et al., 2003) provide an efficient way to analyse document sets. In their model, documents are treated as a mixture of topics where each topic has a distribution over the vocabulary of words. Although topic models are able to reveal general trends and identify topics based on word usage of documents, it does not really make it clear how documents are organized within each topic and it is also unclear how different topics connect to each other.

Eisenstein et al. (2012) introduced *TopicViz*, an LDA-based document visualization system, which can be regarded as a visually-aided information retrieval system. There are two basic differences between their approach and ours. First, they relied on topic models, whereas we employed graph partitioning in order to automatically determine document subtopics. Second, in their work they manually identified the topics determined by LDA whereas we let the automatically detected communities "speak for themselves", i.e. the most informative sets of keyphrases of size 3 were determined based on information theoretic grounds. Our proposed method did not need to know the number of topics to be identified in advance and its time requirements are also more favourable compared to the training of topic models, i.e. it can be performed on the fly during the initialization of our application. A further possible advantage of our approach compared to other LDA models as topic models tend to be trained on the single tokens level, whereas our approach can easily extract informative noun phrases and multi-word expressions as it operates on n-gram level.

In scientific document set visualization, citation analysis is often taken into consideration. The explicit relations among documents like citations can be naturally taken into consideration in our graph-based approach (which is not straightforward in LDA-based solutions). However, citation-based methods have the limitation that they are mostly useful for scientific document sets where citations exist.

3 Document set representation

Our representation used for the visualization of document collections is based on a weighted, undirected graph having individual documents as its nodes. In our case study – when our purpose was to visualize a document set that is clearly interpretable for computational linguists and which is comprised of easily distinguishable, thematically related subcorpora – we relied on the workshop papers present in the ACL Anthology Corpus (Schäfer et al., 2012).

3.1 Single-Document Keyphrase Extraction System

To have an efficient representation of the documents we first used our single-document keyphrase extraction system. Keyphrase extraction was treated as a supervised learning task where successive n-grams extracted from a document (i.e. keyphrase candidates) have to be classified as proper and improper keyphrases.

We utilized the NUS Keyphrase Corpus (2007) and the database of the SemEval-2 shared task on scientific keyphrase extraction (Kim et al., 2010) as training data for our supervised keyphrase candidate ranker. Our keyphrase ranking solution was based on the posterior probability of a "keyphrase or not" binary MaxEnt model trained within the MALLET (2002) framework and using a combination of our feature sets from our previous works on keyphrase extraction as described in (2010) and (2011).

3.2 Visualizing and partitioning the document set

Keyphrases extracted from the individual papers were used next as an input for the construction of a similarity graph which served as the basis of visualization.

3.2.1 Similarity graph

$G_{n,t} = (V, E_n, w_t)$ was defined as a weighted graph of documents, where $E_n = \{(u, v) : v \in neigh(u, n) \vee u \in neigh(v, n)\}$ and $neigh(u, n)$ is a function which returns the set of the n vertices that are closest to vertex u based on the similarity measure w_t.

The similarity measure $w_t(u, v)$ assigns a positive similarity score to documents u and v comparing the overlap between their top-t keyphrases that best describe them. Values n and t are thus hyperparameters that can be adjusted in our application to see their effects on the connectedness of the document graph.

Since a pair of documents can have multiple keyphrases in common, the weight assigned to a pair of nodes can be determined in multiple ways (which can be adjusted in the applet). For a similarity graph $G_{n,t}$, the similarity of documents u and v is 0 if the two documents have no keyphrases in common, otherwise it is aggregated due to one of the following strategies, via calculating

1. the Jaccard or Dice similarity between them, accordingly to the formulae $\frac{A \cap B}{A \cup B}$ and $\frac{2|A \cap B|}{|A|+|B|}$,
2. the cosine similarity of the two documents based on their top-t ranked keyphrases
3. $\sum_{k \in A \cap B} p(k, u)p(k, v)$
4. $\min_{k \in A \cap B}(p(k, u), p(k, v))$
5. $\max_{k \in A \cap B}(p(k, u), p(k, v))$

where sets A and B consist of the top-t ranked keyphrases of documents u and v, respectively and $p(k, u)$ is the probability that is assigned to the event that phrase k is a proper keyphrase of document u.

3.2.2 Visualization of the similarity graph

A force-directed layout visualization is employed based on the publicly available Java implementation of TouchGraph.[2] Their source code was extended and modified to our special needs, e.g. the awt windowing scheme was replaced by the more standardized Swing technology and various input fields for user interaction were added to the user interface. User interactions supported by the current version of the program are

1. Filter documents for keyphrases
2. Filter documents for some kind of metadata (such as the date or authors of a publication)
3. Hide/unhide entire document communities.

To illustrate the importance of nodes within the document graph, PageRank values are determined for each vertex. Since the similarity graph created is designed to be sparse – influenced by the parameter of maximal neighbours n – these calculations can be efficiently calculated with sparse matrix multiplication during the initialization phase of the programme.

Our application supports two kinds of partitioning of the document set to be visualized. If the user has a reliable partitioning of the document set in advance, it can be employed directly during the visualization, otherwise an automatic community detection is to be performed.

3.2.3 Modularity-driven community detection

The community detection employed here maximizes Newman's modularity (2004) in order to obtain a partitioning of the documents. Intuitively, what modularity measures for a given partitioning of a graph is the difference between the fraction of intra-community edges and the expected fraction of intra-community edges in the graph with the same number of vertices and edges but with its edges rewired randomly.

As our intention was to be able to deal with possibly massive data collections, it was a key aspect to keep computation requirements relatively low. Blondel et al. (2008) introduced a method which greedily approximates that partitioning of a graph which has the highest modularity. The proposed iterative method works in a bottom-up manner, starting from the state when all the vertices of the graph form a separate community. In the following steps vertices are moved into a community in such a way that their replacement should yield a best increase locally in the modularity.

3.3 Multi-Document Keyphrase Extraction System

Keyphrases are not only useful in automatically determining thematically related subgroups in document sets, but can be also applied to characterize those subgroups found in some corpus. For this reason our application assigns representative phrases to each community determined according to Section 3.2.3.

[2] http://sourceforge.net/projects/touchgraph/

The keyphrases of a cluster are those top-ranked keyphrases of the individual documents comprising the cluster which had the highest information gain metric, i.e. using the numbers of occurrences of a cluster-level keyphrase candidate inside and outside the particular cluster.

Then, for a subset of the document collection, the top-3 highest ranked candidates based on their information gain – which had at least a high relative frequency within the documents in the particular cluster as the relative frequency of the phrase outside the cluster – were treated as the keyphrases of the given cluster.

4 Conclusions

Since it is crucial in information-rich environments to be able to navigate quickly and effectively within document sets, we created a framework which alleviates it by implementing a visualization tool. Based on the keyphrases that can be automatically determined for individual documents of a document collection, useful and computationally efficient visualization can be built.

The fact that the visualization module only waits for a plain text input makes it possible to easily visualize datasets other than the one comprising of ACL workshop papers. Even though we favored keyphrase-based calculation of document similarities, the simple structure of the input file makes it also possible to perform visualizations based on other (e.g. bibliographic) criteria as well.

Besides providing a visualization tool for document sets it is also made easy to obtain more information from it via the automatic detection of document subgroups and the multi-document keyphrases assigned to each of them which makes the identification of topics possible.

Acknowledgments

This work was in part supported by the European Union and the European Social Fund through the project FuturICT.hu (TÁMOP-4.2.2.C-11/1/KONV-2012-0013) and "Hungarian National Excellence Program" (TÁMOP 4.2.4.A/2-11-1-2012-0001).

References

Gábor Berend and Richárd Farkas. 2010. Sztergak: Feature engineering for keyphrase extraction. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 186–189, Stroudsburg, PA, USA. Association for Computational Linguistics.

Gábor Berend and Richárd Farkas. 2013. Extracción de palabras clave de documentos individuales para extracción de palabras clave de documentos múltiples. *Computación y Sistemas*, 17(2).

Gábor Berend. 2011. Opinion expression mining by exploiting keyphrase extraction. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1162–1170, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March.

Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008+, July.

Jacob Eisenstein, Duen Horng Chau, Aniket Kittur, and Eric Xing. 2012. Topicviz: interactive topic exploration in document collections. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '12, pages 2177–2182, New York, NY, USA. ACM.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 21–26, Morristown, NJ, USA. ACL.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

M. E. J. Newman and M. Girvan. 2004. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113+, February.

Thuy Dung Nguyen and Min-Yen Kan. 2007. Keyphrase extraction in scientific publications. In *Proceedings of the 10th international conference on Asian digital libraries: looking back 10 years and forging new frontiers*, ICADL'07, pages 317–326, Berlin, Heidelberg. Springer-Verlag.

Vahed Qazvinian, Dragomir R. Radev, S. M. Mohammad, Bonnie J. Dorr, David M. Zajic, M. Whidby, and T. Moon. 2013. Generating extractive summaries of scientific paradigms. *J. Artif. Intell. Res. (JAIR)*, 46:165–201.

Ulrich Schäfer, Jonathon, and Stephan Oepen. 2012. Towards an acl anthology corpus with logical document structure. an overview of the acl 2012 contributed task. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 88–97, Jeju Island, Korea, July. Association for Computational Linguistics.