Volume 3, Issue 4, April 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Comparing Theme Evolution Of Text Streams In Text Mining

Banupriya.M
Dept. of CSE Bharath University India.

Preethi.R
Dept. of CSE Bharath University India.

Nithya priya.B
Dept.of IT Bharath University India.

Abstract:Temporal Text Mining (TTM) is concerned with discovering temporal patterns in text information collected over time.In the existing works they have created a theme life cycle to get a temporal pattern of a particular theme. By using the theme evolutionary graph they have extracted the theme patterns. Here we are extending the same concept by comparing the theme evolutions in multiple collections, so that we can go through the vast details of more than one article at a time. This will reduce the time constraints. Here we are implementing the process by using clustering methods and algorithms. The main algorithm using here is Viterbi algorithm. Here we are using two modules such as clustering and comparing. Categories and subject descriptor: Clustering General terms: Algorithms

Keywords: temporal text mining, theme threads, clustering

I. Introduction

In many application domains, we encounter a stream of text, in which each text document has some meaningful time stamp. For example, a collection of news articles about a topic and research papers in a subject area can both be viewed as natural text streams with publication dates as time stamps. In such stream text data, there often exist in- teresting temporal patterns. For example, an event covered in news articles generally has an underlying temporal and evolutionary structure consisting of themes (i.e., subtopics) characterizing the beginning, progression, and impact of the event, among others. Similarly, in research papers, research topics may also exhibit evolutionary patterns. For example, the study of one topic in some time period may have influenced or stimulated the study of another topic after the permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. In all these cases, it would be very useful if we can discover, extract, and summarize these evolutionary theme patterns (ETP) automatically. Indeed, such patterns not only are useful by themselves, but also would facilitate organization and navigation of the information stream according to the underlying thematic structures. Consider, for example, the movie rating concept in internet and other popular social media is performed by comparing the movies which are released in the same year and by using this comparison we can nominate the best movie of the year. It is generally very difficult to rate a movie as a best movie of the year, as we should consider all the points a best movie could have . Ideally, the summary would include both the small budget and large budget films, films by the most famous director and from the new ones and any threads corresponding to the evolution of these themes. For example, the themes may include the report of the happening of the event, the statistics of victims and damage, the aids from the world, and the lessons from the tsunami. A thread can indicate when each theme starts, reaches the peak, and breaks, as well as which subsequent themes it influences. A timeline-based theme structure as shown in Figure 1 would be a veryinformative summary of the event, which also facilitatesnavigation through themes.

II. Related Work

While TTM has not been well studied, there are several lines of research related to our work. For example, in Klein- berg's work on discovering bursty and hierarchical structures in streams[10] , text streams are converted to temporal frequency data and an intimate-state automaton is used to model the stream. Detection of novel topics and trends in text streams has been studied by several researchers[3, 18, 19, 11, 13, 14], but their focus is to identify emerging trends rather than summarize the complete evolutionary theme patterns in a given text stream as we do.An interesting related work to our analysis of theme life cycles is[16], where Perkio and others used a Multinomial PCA model to extract themes from a text collection and they used a hidden theme-document weight, which is similar to d;j in Section 3, to compute the strength of a theme.The major difference between our work and theirs is that we model the theme transitions in a context-sensitive way with an HMM, which presumably captures the natural proximity of similar topics better.Text clustering is another well studied problem relevant to our work. Specifically, the aspect models studied in [9, 20, 2] are related to the mixture theme model we use to extract themes. However, these works do not consider temporal structures in text. Nallapati and others studied how to discover sub-clusters in a news event and structure them by their dependency, which could also generate a graph

structure[15]. A major difference between our work and theirs is that they perform document level clustering, while we perform theme level word clustering. Another difference is that they do not consider the variations of subtopics in different time periods while we analyze life cycles of themes. Since a theme evolution graph and theme life cycle can serve as a good summary of a collection, our work is also partially related to document summarization (eg.,[12, 1]) . Allan and others presented a news summarization method based on ranking and selecting sentences obeying temporal order[1]. However, summarization intends to retain the explicit information in text in order to maintain modelity, while we aim at extracting non-obvious implicit themes and their evolutionary patterns.

III. Algorithm

A hidden Markov model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (hidden) states. An HMM can be considered as the simplest dynamic Bayesian network. The mathematics behind the HMM was developed by L. E. Baum and coworkers. It is closely related to an earlier work on optimal nonlinear filtering problem (stochastic processes) by Ruslan L. Stratonovich, who was the first to describe the forward-backward procedure. In simpler Markov models (like a Markov chain), the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a hidden Markov model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states. Note that the adjective 'hidden' refers to the state sequence through which the model passes, not to the parameters of the model; even if the model parameters are known exactly, the model is still 'hidden'. Hidden Markov models are especially known for their application in temporal pattern recognition such as speech, handwriting, gesture recognition, part-of-speech tagging, musical score following,[8] partial discharges[9] and bioinformatics. A hidden Markov model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other.

IV. Proposed Algorithm

The Viterbi algorithm is a dynamic programming algorithm for finding the most likely sequence of hidden states – called the Viterbi path – that results in a sequence of observed events, especially in the context of Markov information sources and hidden Markov models. Viterbi algorithm is usually used as an expectation step in Expectation–maximization algorithm. The terms Viterbi path and Viterbi algorithm are also applied to related dynamic programming algorithms that discover the single most likely explanation for an observation. For example, in statistical parsing a dynamic programming algorithm can be used to discover the single most likely context-free derivation (parse) of a string, which is sometimes called the Viterbi parse. The Viterbi algorithm was proposed by Andrew Viterbi in 1967 as a decoding algorithm for convolutional codes over noisy digital communication links.The algorithm has found universal application in decoding the convolutional codes used in both CDMA and GSM digital cellular, dial-up modems, satellite, deep-space communications, and 802.11 wireless LANs. It is now also commonly used in speech recognition, keyword spotting, computational linguistics, and bioinformatics. For example, in speech-to-text (speech recognition), the acoustic signal is treated as the observed sequence of events, and a string of text is considered to be the "hidden cause" of the acoustic signal. The Viterbi algorithm finds the most likely string of text given the acoustic signal.

V. Conclusion

Text streams often contain latent temporal theme structures which reflect how different themes influence each other and evolve over time. Discovering such evolutionary theme patterns can not only reveal the hidden topic structures, but also facilitate navigation and digestion of information based on meaningful thematic threads. In this paper, we propose general probabilistic approaches to discover evolutionary theme patterns from text streams in a completely unsupervised way. To discover the evolutionary theme graph, ourmethod would first generate word clusters (i.e., themes) for each time period and then use the Kullback-Leibler divergence measure to discover coherent themes over time. Such an evolution graph can reveal how themes change over time and how one theme in one time period has influenced other themes in later periods. We also propose a method based on hidden Markov models for analyzing the life cycle of each theme. This method would first discover the globally interesting themes and then compute the strength of a theme in each time period. This allows us to not only see the trends of strength variations of themes, but also compare the relative strengths of different themes over time.

References:

[1] J. Allan, R. Gupta, and V. Khandelwal. Temporal summaries of news topics. In Proceedings of ACM SIGIR 2001, pages 10{18, 2001.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993{1022, 2003.

[3] S. Boykin and A. Merlino. Machine learning of event segmentation for news on demand. Commun. ACM, 43(2):35{41, 2000.

[4] T. M. Cover and J. A. Thomas. Elements of Information Theory.Wiley, 1991.

[5] W. B. Croft and J. Laerty, editors. Language Modeling and Information Retrieval. Kluwer Academic Publishers, 2003.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Journal of Royal Statist. Soc. B, 39:1{38, 1977.

[7] R. Feldman and I. Dagan. Knowledge discovery in textual databases (kdt). In KDD, pages 112{117, 1995.

[8] M. A. Hearst. Untangling text data mining. In Proceedings of the 37th conference on Association for Computational Linguistics (ACL 1999), pages 3{10, 1999.

[9] T. Hofmann. Probabilistic latent semantic indexing. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pages 50{57, 1999.

[10] J. Kleinberg. Bursty and hierarchical structure in streams. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 91{101, 2002.

11] A. Kontostathis, L. Galitsky, W. M. Pottenger, S. Roy, and D. J. Phelps. A survey of emerging trend detection in textual data mining. Survey of Text Mining, pages 185{224, 2003.

[12] R. Kumar, U. Mahadevan, and D. Sivakumar. A graph-theoretic approach to extract storylines from Search results. In Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining, pages 216{225, 2004.

[13] J. Ma and S. Perkins. Online novelty detection on temporal sequences. In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 613{618, 2003.

[14] S. Morinaga and K. Yamanishi. Tracking dynamics of topic trends using a _nite mixture model. In Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining, pages 811{816, 2004

[15] R. Nallapati, A. Feng, F. Peng, and J. Allan. Event threading within news topics. In Proceedings of the Thirteenth ACM conference on Information and knowledge management, pages 446{453, 2004.

[16] J. Perkio, W. Buntine, and S. Perttu. Exploring independent trends in a topic-based search engine. In Proceedings of the Web Intelligence, IEEE/WIC/ACM International Conference on (WI'04), pages 664{668, 2004.

[17] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. Proc. Of the IEEE, 77(2):257{285, Feb. 1989.

[18] K. Rajaraman and A.-H. Tan. Topic detection, tracking, and trend analysis using self-organizing neural networks. In PAKDD, pages 102{107, 2001.

[19] S. Roy, D. Gevry, and W. M. Pottenger. Methodologies for trend detection in textual data mining. In the Textmine '02 Workshop, Second SIAM International Conference on Data Mining, 2002.

[20] C. Zhai, A. Velivelli, and B. Yu. A cross-collection mixture model for comparative text mining. In Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining, pages 743{748, 2004.

