Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 10, October 2014, pg.895 – 902

SURVEY ARTICLE

A Survey on Anomaly Detection for Discovering Emerging Topics

S.Saranya[1], R.Rajeshkumar[2], S.Shanthi[3]
[1,2]M.E Scholar, Department of Computer Science and Engineering
[3]Assistant Professor, Department of Computer Science and Engineering
Sri Eshwar College of Engineering, Coimbatore
saranyacse41@gmail.com [1], rajeshdreams07@gmail.com [2], shan.sece@gmail.com [3]

Abstract - This paper identifies various concepts involved in social networks for finding the emerging topics. We focus on the various methods that can be applied for detecting the anomaly. The methods used are Hidden Markov Model, UMass Approach, CMU Approach, Change Finder method and Finite Mixture Model. These methods involve texts, videos, audios, URLs and mentions which are shared in the social networks. Kullback-Leibler divergence measure is used here to discover coherent themes and topics over time.

Keywords - Topic detection, temporal text mining, burst detection, anomaly detection

I. INTRODUCTION

Social data mining has some challenges like detection of topics, bursts, theme patterns from text, outlier detection and change points. To overcome these challenges there are varieties of methods and models have been proposed. Although, finding the anomaly is again the challenging task. For this, the models such as hidden markov model and finite mixture model are used to discover the topics. In Hidden Markov Models for segmentation and detection the tasks are involved with the dragon's approach and UMass approach to find the new topics. Temporal Text Mining (TTM) is concerned with discovering temporal patterns in text information collected over time. It uses Kullback-Leibler divergence to Measure and Expectation Maximization (EM) algorithm to detect outliers and change points. Topic dynamics framework uses Kleinberg's Burst Model and ShaSha's Burst Model to detect the bursts. The new framework is designed from a unifying viewpoint that a topic structure in a text stream is modelled using a finite mixture model.

A. Topic Detection and Tracking

The Topic Detection and Tracking Study is concerned with the detection and tracking of events. The input to this process is a stream of stories. This stream may or may not be pre-segmented into stories, and the system may or may not be trained to recognize specific events. The three technical tasks involved here are the tracking of known events, the detection of unknown events, and the segmentation of a news source into stories[1]. The segmentation task is to correctly locate the boundaries between adjacent stories, for all stories in the corpus. The detection task is characterized by the lack of knowledge of the event to be detected.

The retrospective detection task is defined to be the task of identifying all of the events in a corpus of stories. The on-line new event detection task is defined to be the task of identifying new events in a stream of stories. The tracking task is defined to be the task of associating incoming stories with events known to the system. Segmentation will be evaluated in two ways namely Direct Evaluation of Segmentation and Indirect Evaluation of Segmentation.

Dragon's approach:

Dragon's approach to segmentation is to treat a story as an instance of some underlying topic, and to model an unbroken text stream as an unlabeled sequence of these topics. In this model, finding story boundaries is equivalent to finding topic transitions. At a certain level of abstraction, identifying topics in a text stream is similar to recognizing speech in an acoustic stream identifying the sequence of topics in an unbroken transcript therefore corresponds to recognizing phonemes in a continuous speech stream. The classic Hidden Markov Model (HMM) technique [1] is used, in which the hidden states are topics and the observations are words or sentences. The topics used by the segmenter are referred to as *background* topics, were constructed by automatically clustering news stories from this training set. The clustering was done using a multi-pass k-means algorithm.

UMass Approach:

In UMass approach **Content Based LCA Segmentation** which makes use of the technique of local context analysis (LCA). It is an expansion of ad hoc queries for information retrieval. LCA can be thought of as an association thesaurus which will return words and phrases which are semantically related to the query text and are determined based on collection-wide co-occurrence as well as similarity to the original sentence. It is somewhat similar to the topic models used in Dragon's method and to the relevance features in CMU's method.[1] The strong point of the LCA method is that, other than the length model estimation, it is completely unsupervised. One weakness of this method is that the current implementation is somewhat slow since it requires a database query per sentence.

Discourse Based HMM Segmentation method uses a Hidden Markov Model to model "marker words," or words which predict a topic change. The HMM segmenter is relying on words which predict the beginning or end of a segment without regard to content. This is somewhat similar to CMU's use of vocabulary features. The model is trained using segmented data. Unknown word probabilities were handled with a very simple smoothing method.

CMU Approach:

This approach is mainly used for the context of multimedia information retrieval applications. The CMU approach was designed around the idea that various "features" of these multiple media sources should be extracted and then combined into a statistical model that appropriately weighs the evidence, and then decides where to place segment boundaries. In the CMU approach the relative behaviour of an *adaptive* language model is compared to a *static* trigram language model in an on-line manner. Clustering algorithms and online detection algorithms are involved in this approach.

B. Temporal Text Mining

Temporal Text Mining (TTM) is concerned with discovering temporal patterns in text information collected over time. Since most text information bears some time stamps, TTM has many applications in multiple domains, such as summarizing events in news articles and revealing research trends in scientific literature. 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.[2] In such stream text data, there often exist interesting temporal patterns. It would be very useful if we can discover, extract, and summarize these evolutionary theme patterns (ETP) automatically.

In this work, propose general probabilistic approaches to discover evolutionary theme patterns from text streams in a completely unsupervised way. To discover the evolutionary theme graph, this method 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[2]. 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. In addition to the theme structure, revealing the strength of a theme at different time periods, or the life cycle of a theme, is also very useful. Since it is often very useful to discover the temporal patterns that may exist in a stream of text articles, a task which we refer to as Temporal Text Mining (TTM).

We define the problem of discovering and summarizing the ETPs in a text stream. And present general probabilistic methods for solving the problem through (1) discovering latent themes from text, which includes both interesting global themes and salient local themes in a given time period; (2) discovering theme evolutionary relations and constructing an evolution graph of themes; and (3) modelling theme strength over time and analyzing the life cycles of themes.

Theme Extraction using Expectation Maximization (EM) algorithm:

We extract themes from each sub collection Ci using a simple probabilistic mixture model. In this method, words are regarded as data drawn from a mixture model with component models for the theme word distributions and a background word distribution. Words in the same document share the same mixing weights. The model can be estimated using the Expectation Maximization (EM) algorithm [2] to obtain the theme word distributions. To discover any evolutionary transition between two theme spans, we use the Kullback-Leibler divergence to measure their evolution distance.
Here 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.

C. Detecting Outliers and change Points from time series

In the area of data mining, there have been increased interests in these issues since outlier detection is related to fraud detection, rare event discovery, etc., while change-point detection is related to event/trend change detection, and activity monitoring. The issue of detecting change points in time-series data has extensively been addressed in statistics and has become one of the issues receiving vast attention in data mining, which is recognized as *event change detection* and closely related to *activity monitoring*. Here, by the term *change point*, we mean a time point at which the data properties suddenly change.[3] Conventionally, outlier detection and change point detection have been addressed independently. We developed algorithms for online discounting learning of these models, where they can track time-varying data sources adaptively by gradually forgetting out-of-date statistics. Then, for a new input data, its score is calculated in terms of its deviation from the learned model, with a high score indicating a high possibility of being a statistical outlier. The two directions involved here are 1) dealing with time series and 2) detecting change points in it.

As for 1), here consider time-series models such as AR (auto regression) model in place of independent models such as Gaussian mixtures. Here employ an algorithm for online discounting learning of a time-series model and apply it to detecting outliers from time series. A score for any given data is calculated in terms of its deviation from the learned model, with a higher score indicating a high possibility of being an outlier.

As for 2), here present a new two-stage time-series learning scheme, which connects change point detection to outlier detection, the process of which is summarized as follows: First, by taking an average of the scores over a window of a fixed length and sliding the window, we obtain a new time series consisting of moving-averaged scores. Change point detection is then reduced to the issue of detecting outliers in that time series. We name this scheme ChangeFinder.

ChangeFinder:

The change point detection scheme employs two-stage time-series learning, which we named ChangeFinder.[3] A remarkable aspect of ChangeFinder is that it repeats the learning process twice, where the outlier detection is first done using the model learned in the first stage and change point detection is done using the learned model in the second one.

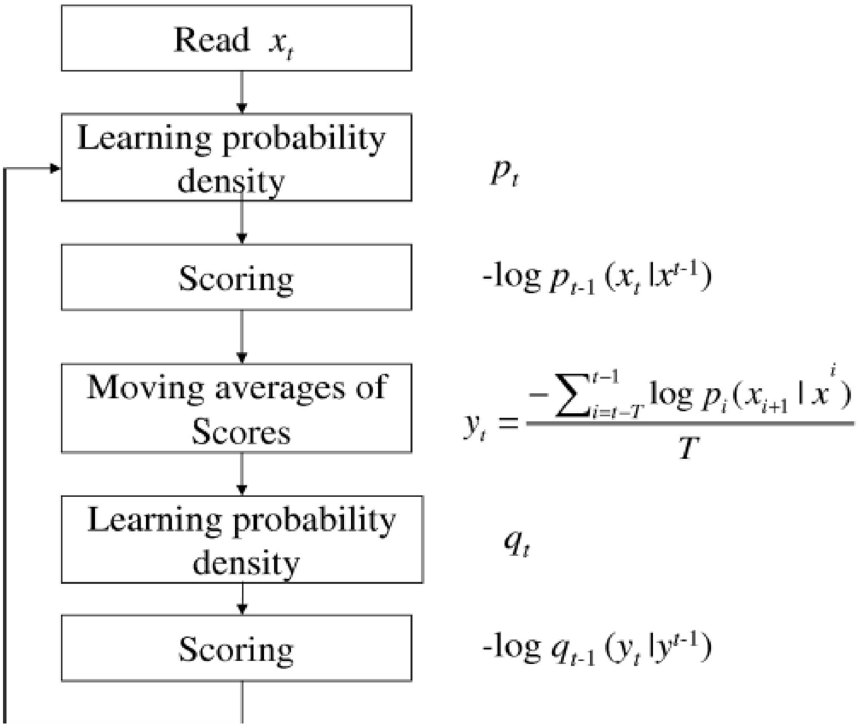
Read x_t
Learning probability density
p_t
Scoring
$-\log p_{t-1}(x_t|x^{t-1})$
Moving averages of Scores
$y_t = \frac{-\sum_{i=t-T}^{t-1} \log p_i(x_{i+1}|x^i)}{T}$
Learning probability density
q_t
Scoring
$-\log q_{t-1}(y_t|y^{t-1})$

Fig: Flow of Change Finder

The change finder first reads the data sequences x_t and constructs the sequence of probability density functions. Then the score is calculated using the given formula:

Score (xt) = _log pt_1 (xt|x^{t_1})

The average score is calculated and the learning probability density is calculated for the scores and the same steps are repeated to find the outlier. The feature of ChangeFinder with AR model is discussed here and the reason is that efficient learning algorithms such as **SDAR** (Sequential Discounting AR learning) **algorithm [3]** are available, and that the AR model is a typical time-series model and natural knowledge representation for time-series data.

D. Topic Dynamics

The hot topics or shifts in trends are often manifested in `bursts' of publications, and these are difficult to stay on top of. It is natural to hope that we might be able to better manage some parts of this continuing challenge by automating the identification of new trends and the detection and tracking of topic bursts. The essence of this work is to view bursts as intervals of increasing `momentum'. Here allow topics to take different attributes such as position, velocity, mass that can have any underlying meaning we like. Then track changes of sign in velocity, and therefore changes of sign in momentum (mass . velocity). This is a simple and very general notion of burst, and here believe it is useful. The emphasis on momentum is a stock market view of bursts, and thus also a stock market view of topic streams. Technical stock market analysis focuses heavily on monitoring momentum, since it can be a natural measure of human sentiment.

Increases in momentum are viewed as bursts of sentiment, and swings in momentum (shifts in its sign) as changes in larger trends. In this work, highlight advantages of this approach for scientific topics. A market perspective can be appropriate if we view science as a marketplace of ideas. This approach can offer a number of advantages. An immediate advantage is that we can adapt existing machinery for monitoring trends, identifying burst periods, and measuring burst strength. This instantly provides many familiar tools for analyzing and visualizing patterns, backed by decades of experience. Here presented a Topic Dynamics framework [4] as an alternative to detection and analysis of bursts. This framework rests on physical intuition, modelling bursts as intervals of increasing momentum, which can be applied to many `trend' quantities of interest, such as changes of `value' in the stock market and changes of `impact' in the scientific literature. The Kleinberg's burst model and ShaSha's burst model are discussed here to identify the bursts.

Kleinberg's Burst Model:

This model is motivated originally by a problem of representing bursts of email messages. Kleinberg's burst algorithm[4] models bursts with an infinite state automaton in which each state represents a message arrival rate (of a Poisson arrival process).The higher the state, the smaller the expected time gap between messages. `Word bursts' can then be defined as having arrival rates defined by the number of messages containing a particular word. Kleinberg's model on topic tracking in the news formulates `memes' as patterns of words, using the model of bursts. A major contribution is to propose scalable clustering approaches for identifying short distinctive phrases travelling intact through on-line text.

ShaSha's Burst Model:

Shasha and co-workers have developed this model based on the hierarchies of fixed-length time intervals. This method was motivated originally by a problem of modelling bursts of gamma rays.[4] These wavelet-like hierarchies have intervals of different scale defined by powers of 2. Bursts occur in intervals in which event frequency of occurrence exceeds a given threshold. The main limitation of these methods is they were defined within the context of a single topic and these are computationally expensive models.

Reconsidering Bursts in terms of momentum:

Instead of defining bursts in terms of arrival rate in a single topic stream, our topic dynamic model adapts basic notions of dynamics from physics and models topic bursts as momentum change of the topic. The momentum is the product of mass and velocity, which is the rate of position change. It is hard to measure the change of momentum from these values directly; we adapt popular stock market trend analysis techniques such as EMA (Exponential Moving Average) and MACD (Moving Average Convergence/Divergence) in our model. Since all the trend analysis indicators are computable in an online fashion, our model can be very efficient, avoiding implementation complexity of existing burst models. The topic dynamic model also leads to a clear definition of burst strength as the MACD histogram value. This benefits not only detection of bursts, but also momentum based prediction of bursts. A hierarchical topic structure is naturally integrated into the model, so that bursts are accumulated along the hierarchical structure. This allows exploration of semantic information involving multiple topics.

Some of the useful measures included in these methods are:

- mass
- position
- velocity
- momentum
- acceleration
- force

Formalizing Bursts:

A traditional scheme for identifying trends in stock market analysis is to use moving averages to smooth out noise, and then estimate derivatives (rates of change, hence trends) using differences of smoothed values (moving averages) that have been computed in mildly different ways.

EMA:

For a variable x = x (t) which has a corresponding discrete time series x = {xt | t = 0, 1, ...}, the n-day EMA (Exponential Moving Average) with smoothing factor:

$$\begin{aligned} EMA[x]_t &= \alpha\, x_t \;+\; (1-\alpha)\, EMA[x]_{t-1} \\ &= \sum_{k\geq 0} \alpha\, (1-\alpha)^k\, x_{t-k}. \end{aligned}$$

MACD:

In technical stock market analysis [4], the MACD (Moving Average Convergence/Divergence) of a variable xt (usually price) is defined by the difference of its n1- and n2-day moving averages:

$$MACD(n_1, n_2) = EMA(n_1) - EMA(n_2)$$

MACD histogram:

MACD histogram is an estimate of the derivative of the MACD :

$$\text{signal}(n_1, n_2, n_3) = EMA(n_3)[MACD(n_1, n_2)]$$
$$\text{histogram}(n_1, n_2, n_3) = MACD(n_1, n_2) - \text{signal}(n_1, n_2, n_3)$$

The topic dynamics framework not only detects burst periods, and burst strength, but also can be used for forecasting oncoming bursts Furthermore; this framework embraces hierarchical structure of bursts, and takes into account semantic links between topics that are missed by single-steam analysis.

E. Finite Mixture Model

In a wide range of business areas dealing with text streams, including CRM, knowledge management, and Web monitoring services, it is an important issue to discover topic trends and analyze their dynamics in *real-time* [5]. A topic is here defined as a seminal event or activity. Specifically we consider the following three tasks in topic analysis:

1) *Topic Structure Identification*; learning a *topic structure* in a text stream, in other words, identifying what kinds of main topics exist and how important they are.
2) *Topic Emergence Detection*; detecting the emergence of a new topic and recognizing how rapidly it grows, similarly, detecting the disappearance of an existing topic.
3) *Topic Characterization*; identifying the characteristics for each of main topics. Our framework is designed from a unifying viewpoint that a topic structure in a text stream is modelled using a finite mixture model (a model of the form of a weighted average of a number of probabilistic models) and that any change of a topic trend is tracked by learning the finite mixture model dynamically.

The main purpose of this work is to propose a new topic analysis framework that satisfies the requirement as above, and to demonstrate its effectiveness through its experimental evaluations for real data sets. This framework is designed from a unifying viewpoint that a topic structure in a text stream is modelled using a finite mixture model (a model of the form of a weighted average of a number of probabilistic models) and that any change of a topic trend is tracked by learning the finite mixture model dynamically [5]. Here each topic corresponds to a single mixture component in the model.

All of the tasks 1)-3) are formalized in terms of a finite mixture model as follows: As for the task 1), the topic structure is identified by statistical parameters of a finite mixture model. They are learned using our original *time-stamp based discounting learning algorithm*, which incrementally and adaptively estimates statistical parameters of the model by gradually forgetting out-of-date statistics, making use of time-stamps of data. This makes the learning procedure adaptive to changes of the nature of text streams.

As for the task 2), any change of a topic structure is recognized by tracking the change of main components in a mixture model. Here apply the theory of *dynamic model selection* to detecting changes of the optimal number of main components and their organization in the finite mixture model. Here may recognize that a new topic has emerged if a new mixture component is detected in the model and remains for a while. Unlike conventional approaches to statistical model selection under the stationary environment, dynamic model selection is performed under the non-stationary one in which the optimal model may change over time. Further note that we deal with a complicated situation where the dimension of input data, i.e., the number of features of a text vector, may increase as time goes by.

As for the task 3), here classify every text into the cluster for which the posterior probability is largest, and then here characterize each topic using feature terms characterizing texts classified into its corresponding cluster. These feature terms are extracted as those of highest information gain, which are computed in real-time.

S.No	Methods	Usage
1	Hidden Markov Model	Detection and tracking of events.
2	Temporal Text Mining	Discover, Extract, and Summarize evolutionary theme patterns automatically.
3	Change Finder	Detecting change points in time-series data
4	Topic Dynamics	Detection and analysis of bursts
5	Finite Mixture Model	Discover topic trends and analyze their dynamics

Table: comparison of various methods used in the detection of emerging topics

II. CONCLUSION

This paper describes the comparison and analysis between various methods involved in the detection of emerging topics. It also illustrates that there are many techniques that can be followed for detecting the topics, bursts and theme patterns. This kind of comparison reflects that the efficiency differs from each method. This paper shows the usage of various models used in the detection of emerging topics such as finite mixture model and Hidden Markov Models.

REFERENCES

[1] J. Allan et al., *Topic Detection and Tracking Pilot Study: Final Report*, Proc. DARPA Broadcast News Transcription and Understanding Workshop, 1998.

[2] Q. Mei and C. Zhai, *Discovering Evolutionary Theme Patterns from Text: An Exploration of Temporal Text Mining*, Proc. 11th ACM SIGKDD Int’l Conf. Knowledge Discovery in Data Mining, pp. 198-207, 2005.

[3] J. Takeuchi and K. Yamanishi, *A Unifying Framework for Detecting Outliers and Change Points from Time Series*, IEEE Trans. Knowledge Data Eng., vol. 18, no. 4, pp. 482-492, Apr. 2006.

[4] D. He and D.S. Parker, *Topic Dynamics: An Alternative Model of Bursts in Streams of Topics*, Proc. 16th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining, pp. 443-452, 2010.

[5] S. Morinaga and K. Yamanishi, *Tracking Dynamics of Topic Trends Using a Finite Mixture Model*, Proc. 10th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining, pp. 811-816, 2004.

[6] Toshimitsu Takahashi, Ryota Tomioka, and Kenji Yamanishi, *Discovering Emerging Topics in Social Streams via Link-Anomaly Detection* , ieee transactions on knowledge and data engineering, vol. 26, no. 1, january 2014

[7] Andreas Krause, Jure Leskovec and Carlos Guestrin, *Data Association for Topic Intensity Tracking*, proceedings of the 23 rd International Conference on Machine Learning, Pittsburgh, PA, 2006.

[8] Teemu Roos and Jorma Rissanen,*On sequentially normalized maximum likelihood models*.

[9] Jon Kleinberg, *Bursty and Hierarchical Structure in Streams*, Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002.

[10] Y. Urabe, K. Yamanishi, R. Tomioka, and H. Iwai, *Real-Time Change-Point Detection Using Sequentially Discounting Normalized Maximum Likelihood Coding*, Proc. 15th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD' 11),2011.

[11] A. Krause, J. Leskovec, and C. Guestrin, *Data Association for Topic Intensity Tracking*, Proc. 23rd Int'l Conf. Machine Learning (ICML' 06), pp. 497-504, 2006.

[12] H. Small, *Visualizing Science by Citation Mapping*, J. Am. Soc. Information Science, vol. 50, no. 9, pp. 799-813, 1999.

[13] D. Aldous, *Exchangeability and Related Topics*, Ecole d' Ete´ de Probabilite´s de Saint-Flour XIII—1983, pp. 1-198, Springer, 1985.

[14] D. Lewis, *Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval*, Proc. 10th European Conf. Machine Learning (ECML' 98), pp. 4-15, 1998.

[15] C. Giurc_aneanu, S. Razavi, and A. Liski, *Variable Selection in Linear Regression: Several Approaches Based on Normalized Maximum Likelihood*, Signal Processing, vol. 91, pp. 1671-1692, 2011.