Predicting Topics in Scholarly Papers

Seyed Ali Bahrainian(✉), Ida Mele, and Fabio Crestani

Faculty of Informatics, Università della Svizzera italiana (USI), Lugano, Switzerland
{bahres,ida.mele,fabio.crestani}@usi.ch

Abstract. In the last few decades, topic models have been extensively used to discover the latent topical structure of large text corpora; however, very little has been done to model the continuation of such topics in the near future. In this paper we present a novel approach for tracking topical changes over time and predicting the topics which would continue in the near future. For our experiments, we used a publicly available corpus of conference papers, since scholarly papers lead the technological advancements and represent an important source of information that can be used to make decisions regarding the funding strategies in the scientific community. The experimental results show that our model outperforms two major baselines for dynamic topic modeling in terms of predictive power.

Keywords: Topic prediction · Topic modeling
Temporal evolution of topics

1 Introduction

Scientific papers are a vehicle for advancing science and technological development. Discovering topics from scientific papers and analyzing their evolution over time is beneficial for making important decisions by governments, research organizations, funding agencies, and even researchers. As an example, research-funding organizations can adjust their granting policies based on insights produced by predictive models in order to favor topics that are trending and get increasing attention rather than those that are losing momentum and interest.

In this paper, we propose a novel evolutionary method capable of predicting the topics from the past time slices that would continue in the future time slices. Our model updates itself in an evolutionary process based on reinforcement learning which learns from the data and corrects itself over time. We use a publicly available dataset of papers from the Neural Information Processing (NIPS) conference which were published over 29 years. We compare the predictive performance of our method against two dynamic-topic-modeling baselines in terms of near-future prediction of continuing topics. The first baseline is the *Dynamic Topic Model* (DTM) by Blei et al. [4] which tracks the evolution of topics over time. DTM assumes that all topics are present in all the time slices of a sequential corpus of text. The second baseline is the *discrete Dynamic Topic*

G. Pasi et al. (Eds.): ECIR 2018, LNCS 10772, pp. 16–28, 2018.
https://doi.org/10.1007/978-3-319-76941-7_2

Model (dDTM) [1] which modifies DTM by relaxing the assumption that a topic should be present in all the time slices. Thus, dDTM tracks the evolution of intermittent topics over time, hence the word "discrete" in the name. It is noteworthy to mention that whether the topics extracted from the NIPS papers tend to be continuous or discrete, we use both DTM and dDTM as baselines to rigorously test our proposed model in the different possible scenarios.

The contributions of this paper are as follows:

– We present a novel approach for predicting the continuing topics over time.
– We conduct an analysis of the dataset of NIPS papers to show different features of our novel method, and compare it with DTM and dDTM baselines.

The remainder of this paper is organized as follows: Sect. 2 describes the related work, and Sect. 3 briefly explains the background on dynamic topic modeling. We present our topic tracking evolutionary model in Sect. 4. In Sect. 5, we show our experimental setup and results. Finally, Sect. 6 concludes the paper.

2 Related Work

In this section, we shortly review the related work regarding the use of topic models on scholarly articles.

An early work based on applying topic models to scientific articles was the one by Griffiths et al. [8]. They proposed an approach for reconstructing the official Proceedings of the National Academy of Sciences (PNAS). In a more recent work, Talley et al. [12] used topic models to create a high-level representation of scientific articles. They applied topic models to abstracts of National Institutes of Health (NIH) grant proposals to discover the research directions of various research groups and institutes. Their analysis showed unexpected overlaps in research priorities across institutes.

In 2006, Blei and Lafferty [4] introduced Dynamic Topic Model (DTM) for analyzing the evolution of topics in chronologically-ordered datasets. The model, based on Latent Dirichlet Allocation (LDA) [5], can capture the evolution of a topic over time and show various trends, for example, the changing probability of a term in a topic over time, or the popularity of that term at different time intervals. Based on DTM, Continuous-time Dynamic Topic model (cDTM) [13] and discrete Dynamic Topic Model (dDTM) [1] were introduced. cDTM can track any change in a topic and was shown to be effective for short time intervals and the changes in topics are often very small. On the other hand, dDTM provides more flexibility since it relaxes the assumption that topics need to be continuously present over all the time slices. Indeed dDTM can capture sudden variation in the topic change and also the evolution of intermittent topics over time.

All these temporal models can be used to discover latent topics discussed in sequential document collections and capture the evolution of topics by chaining the same topics over time. Although they were evaluated by computing log-likelihood on a future time slice given the data of past time slices, they were not used to predict the topics from the past that would continue in the future in terms of standard IR metrics. In this paper, we use the DTM and dDTM models as state-of-the-art baselines for temporal topic modeling and compare the performance of our topic-prediction model against them.

3 Background

3.1 Dynamic Topic Model

Topic models are defined as hierarchical Bayesian networks of discrete data where documents are distributions over topics and topics are represented as sets of words drawn from a fixed vocabulary that together represent a high-level concept [13]. These probabilistic methods can be used to have a low-dimensional representation of document corpora. Do et al. [7] state that "from an application viewpoint, topic model is a tool for extracting emergent hidden patterns from a collection of data."

LDA [5] is a well known topic model used to discover the latent topics in a document collection. Since the model is not influenced by the temporal ordering of the documents, it has the drawback of mixing together topics related to different temporal periods. To overcome this limitation, Blei and Lafferty proposed Dynamic Topic Model (DTM) [4] which divides a sequential corpus of documents into time slices. Then, it applies LDA to each of them in order to model the latent topics present in the time slices. The hyperparameters of LDA are chained together over consecutive time slices using a linear Kalman filter [9] which allows a linear evolution of the topics. To elaborate further, the parameters of each topic, $\beta_{t,k}$, are chained together in a state space model that evolves with a Gaussian noise. Subsequently, DTM draws each topic β such that:

$$\beta_{t,k}|\beta_{t-1,k} \sim \mathcal{N}(\beta_{t-1,k}, \sigma^2 I) \tag{1}$$

where $\mathcal{N}$ is a logistic normal distribution. The σ parameter, in Eq. 1, allows for variation in a topic over two subsequent time slices. By assigning small values to σ, the model ensures that one topic would not evolve to a different topic over two subsequent time slices. A similar evolution process holds for the α parameter, as α impacts the per-document topic proportions, θ, that is drawn from a Dirichlet distribution. The graphical model of DTM is illustrated in Fig. 1.

In spite of being a powerful model for statistical interpretation of a sequential corpus, DTM comes with two limitations:

(1) The assumption that topics change slowly over time which holds for some document collections (e.g., the articles from the journal *Science*) where topics evolve at a low pace, but does not hold for others (e.g., online discussions, news).

   Moreover, the topic evolutions may have skips in the timeline and it is reasonable to assume that in textual streams, different topics may emerge,

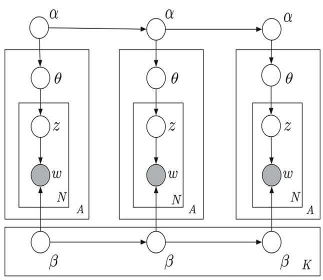

Fig. 1. Graphical model of DTM.

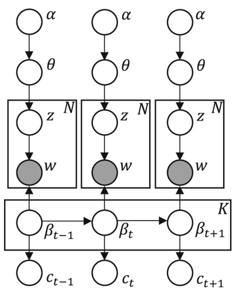

Fig. 2. Graphical model of dDTM. $C = c_0, \ldots, c_n$ is the resulting topic chain.

disappear, and appear again after some time. Hence the topics observed in one time slice might be completely different from the previous one and the assumption that each topic at time slice t has to be connected to a topic at time slice $t-1$ may be a limitation.

(2) As shown in Eq. 1, a topic evolves linearly with a Gaussian noise which allows only small variations in the topics over two consecutive time slices. The effect of such modeling is that if a new topic (i.e., it did not exist in the first time slice of the dataset) emerges in the data the model will not immediately respond to it. Only if the topic persists over a number of time slices, DTM would gradually take it into account in the evolution process and be able to model it. This causes DTM to be unable to immediately detect emerging topics.

In the next section we present another topic model called discrete Dynamic Topic Model (dDTM) which aims at addressing the above limitations. As demonstrated in previous works [1,10], it can model the above-mentioned real-world applications for users' posts and news articles more effectively. Differently from DTM, it uses a non-linear evolution process for the topics and does not rely on the approximations of the evolution of the topic proportions over time.

3.2 Discrete Dynamic Topic Model

The dDTM [1] estimates the topical chains based on the multinomial distributions over words (i.e., topics) in a non-linear fashion as opposed to DTM where topics evolve linearly over all time slices.

The graphical model of dDTM is illustrated in Fig. 2. As we can see, similarly to DTM, the model is based on LDA. Given a sequential dataset (i.e., stream of documents) divided into different time slices, dDTM applies LDA for computing topics in each time slice. As in DTM, the model chains together the multinomial distributions over observed words (i.e., topics), β, to estimate the latent topical chains over time. Differently from DTM, it does not estimate the α parameter which influences the topic proportions over subsequent time slices. The reason is that in the case of streaming data, the data over two subsequent time slices might not always be similar enough, hence an estimation of topic proportions based on the first time slice might not be correct. Indeed, in dDTM the assumption that a topic would always be present in all the time slices is relaxed. Thus, it is not reasonable to adjust topic proportions of one time slice based on estimations from the previous time slice. In settings where a topic is not present at the time slice t_0, but it is present at t_1, the topic proportions of t_0 are not a reliable approximation for the topic proportions of t_1.

For chaining the topics over different time slices, dDTM uses a Hidden Markov Model (HMM) [11]. Similarly to the Kalman filter, it implements an Expectation Maximization (EM) algorithm, named Baum-Welch [6]. However, DTM using a Kalman filter makes the assumption that the evolution of a topic in the state space model is linear, and this is suitable for continuous-time settings. Whereas, the HMM used by dDTM enables to relax the assumption that every topic should be present in all time slices allowing the discovery of latent topic chains in discrete-state settings.

4 Our Model

In this section, we present a novel approach for predicting topics that will continue in the future. In our research on how to effectively predict the continuing topics over time, we devise various strategies which consider the *recency* and the *establishment* effects. The former is captured by increasing the weight of the most recent topics, whereas the latter assigns higher weights to the more established topics. Based on these two effects, we developed a dynamic method that combines recency and establishment measurements.

4.1 The K2RE Method

We call our methodology *Kalman combination of Recency and Establishment (K2RE)*. It combines the two effects of recency and establishment using a Kalman filter. In the following we first explain the two effects and then further elaborate on other details of our model.

Recency. The recency effect ranks the topics by assigning higher weights to most recent topics. Then, as an energy function it computes the correlation of each topic (whose continuation is to be predicted) with the vector of newly computed weights. We formally define the recency effect as follows: given the topics of the last n consecutive time slices of a sequential dataset, we would like to predict which topics continue in the $(n+1)_{th}$ time slice. Let V be the vocabulary of all words occurring in the first n time slices. We construct a word vector containing probability scores corresponding to each word in V. The assigned probability scores are higher for the words appearing in the most recent topics. Thus, first we compute LDA topics (this is explained in Sect. 5.1) from the first n time slices and then we compute the average probability of each word present in all topics according to the recency effect using the following equation:

$$P_{ref,Rec} = \sum_{n=1}^{N}\sum_{t=1}^{T}\sum\nolimits_{w_i \in t} \frac{P(w_i) * 2^n}{(n * t)} \tag{2}$$

where n is the sequence number of the time slice, t is the number of topics derived from each time slice, and w_i is a word from that topic. The 2^n is the rate with which higher weights are assigned to recent topics. The resulting word vector is an average representation of the probability of all the words present in all the n time slices.

Therefore, this effect assigns higher weight to a word which has occurred in the most recent time slice of a sequential corpus. We refer to the word vector where the probability of each word is computed with Eq. 2 as the recency-reference vector.

Establishment. As for the establishment effect, given a vocabulary V made of all the words occurring in the first n time slices, we create a word vector containing probability scores corresponding to each word in V. In this case, the assigned probability scores are higher for the words which have persisted over time. For this purpose, we compute LDA topics (this is explained in Sect. 5.1) from the first n time slices and compute the average probability of each word present in all the topics according to the establishment effect using the following equation:

$$P_{ref,Est} = \sum_{n=1}^{N}\sum_{t=1}^{T}\sum\nolimits_{w_i \in t} \frac{P(w_i) * 2^{-n}}{(n * t)} \tag{3}$$

where n is the time-slice sequence number, t is the number of topics derived from each time slice, and w_i is a word from that topic. The 2^{-n} is the rate with which higher weights are assigned to established topics and, as we can see, it is the opposite of Eq. 2. It weighs the words opposite to that of the recency effect. Therefore, the word vector constructed by averaging all topics in all n previous time slices based on the establishment effect will be a representation of the average occurrence of each word in V, where the most established (i.e., persisting in occurrence) words have higher weights. We refer to the word vector where the probability of each word is computed using Eq. 3 as the establishment-reference vector.

Combining Recency and Establishment. Now that we defined the recency and establishment effects, we explain how to combine them. Our model dynamically estimates the weights of each of the effects for each time slice and corrects itself over time by learning from the data. We refer to this method as a *Kalman combination of Recency and Establishment (K2RE)*. It utilizes the Kalman filter [9] to estimate the recency and establishment weights over time. By measuring the dataset changing behavior in terms of establishment and recency, K2RE adapts itself to the dynamics of the data over time.

In the following, we first present a general overview of K2RE and then elaborate on its details. Figure 3 illustrates the components of the K2RE method for integrating the scores from the recency and the establishment effects.

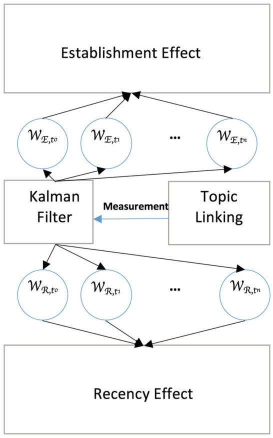

Fig. 3. Components of the K2RE method.

The linear interpolation for a time slice t is defined as:

$$K2RE_t = w_{E,t} * Score_{establishment} + w_{R,t} * Score_{recency} \tag{4}$$

where $Score_{establishment}$ and $Score_{recency}$ are computed by the establishment and the recency effects, respectively. Furthermore, $w_{E,t}$ and $w_{R,t}$ are establishment weights and recency weights computed by the Kalman filter at time t, such that $w_{E,t} + w_{R,t} = 1$. This means that at each time slice t, each of the two effects will be given a weight that reflects its contribution to the future topics. The weights can be either equally assigned or they reflect the effect that dominates. The Kalman filter is always initialized by assigning equal probability of 0.5 to both $w_{E,t}$ and $w_{R,t}$ and then it dynamically updates the weights based on the learning from the data, according to the following system of equations:

$$f(n) = \begin{cases} X^{\mathrm{G}} = A^{\mathrm{G}} f_{t-1} + \varepsilon_t^{\mathrm{G}} & t = 2, \ldots, T \\ z_t = H^{\mathrm{G}} f_t + w_t^{\mathrm{G}} & t = 1, \ldots, T \end{cases}$$

where f_t is the system state at time t, A^{G} denotes the transition of the dynamic system from $t-1$ to t, H^{G} describes how to map state f_t to an observation, z_t (i.e., measurement), and both $\varepsilon_t^{\mathrm{G}}$ and w_t^{G} are mutually independent Gaussian noise variables with co-variances R_t and Q_t, respectively. The dynamic system evolves over time and updates itself proportional to the Kalman gain.

Now we describe the topic-linking module which is based on the dDTM [1]. As described in Sect. 3.2, dDTM tracks the evolution of intermittent topics that may occur discretely over time, such that a topic does not need to be necessarily present over all the time slices. In the topic-linking module shown in Fig. 3, we use a component of dDTM which links together similar topics over time. Such linking can be discrete or continuous under our model (i.e., a topic may be present over all time slices or it may skip some time slices). In particular, we use a Gaussian random walk in a Markovian state space model. The Markov assumption enforces probabilities of a hidden state at time t to be computed merely depending on the previous time slice and not based on all the previous states. We utilized this assumption to compute topic chains that capture the evolution of a topic discretely over time, so that two topics will be linked over two different time slices if they are similar according to the following criterion:

$$\beta_{t,k}|\beta_{t-m,1..k} \sim \mathcal{N}(\beta_{t-1}, \sigma^2 I) \tag{5}$$

where $\beta_{t,k}$ is topic k at time slice t, and $m \in \{1, 2, 3, \ldots, n\}$ with n being the number of previous time slices and σ is the maximum variance allowed from the mean of a topic in the previous time slice. By assigning a small value to σ, our model links two topics that are highly similar. Furthermore, we use the Baum-Welch [3] algorithm to learn the forward and backward probabilities of the transitions among the topics. The model takes as input the topics from the first n time slices and computes their continuation in the $(n+1)_{th}$ time slice.

After linking similar topics over every two consecutive time slices, the topic-linking module computes the recency rate of the topics for time slice n which is the number of topics in the time slice $n-1$ that have been present in the time slice n divided by the total number of topics in the same time slice. This measurement is given as the observation matrix to the Kalman filter for each time slice. Subsequently, the Kalman filter computes the evolution of recency and establishment weights using the Kalman filter system of equations and, using Eq. 4, a K2RE reference vector is generated.

For the purpose of computing correlation between each topic and the K2RE reference vector, we use the Pearson correlation metric that is $P_{X,Y} = \frac{COV(X,Y)}{\sigma_X \sigma_Y}$, where $P_{X,Y}$ is the Pearson correlation of two populations X and Y, COV is the covariance, and σ is the standard deviation. In our use case, the Pearson correlation indicates the level of correlation of each topic with the K2RE reference vector. The intuition behind using correlation as an energy function is that, topics are by definition a set of words that depend on one another and a change in one word may cause changes in the probabilities of other words. According to this intuition we chose the Pearson correlation as an energy function for comparing word vectors. Finally, we require a threshold for distinguishing continued and discontinued topics. To determine an effective threshold we use *10-fold cross validation*. Hence, we split the dataset in 10 folds, iteratively leaving out a chunk of the data, and compute the threshold which minimizes the Mean Squared Error (MSE) of prediction on the remaining folds. Then, using the computed threshold, we evaluate the left-out fold.

5 Experimental Setup

In this section we present our experimental setup, including a description of the dataset followed by the evaluation of our approach.

5.1 Dataset Description

Our dataset consists of all the papers of the Neural Information Processing Systems (NIPS) conference published between years 1987 and 2015 [1]. Therefore, our dataset is spread over 29 years. The total number of papers is 5993.

Topic extraction: The dataset is sorted chronologically and divided into time slices of fixed size (one year). We treat every paper as a document and applied LDA to extract the latent topics from each time slice. Since the number of topics (K) discussed in two different time slices might vary, it is important to estimate the number of topics per time slice. For this purpose, similar to the method proposed in [8], we went through a model selection process. This consists in keeping the LDA Dirichlet parameters (commonly known as α and η) fixed and assigning several values to K. We computed an LDA model for each assignment and subsequently we picked the model that satisfies:

$$\underset{K}{\operatorname{argmin}}\ logP(W|K)$$

where W indicates all the words in the corpus. We repeated this process for each time slice to find the optimal number of topics for all the time slices.

Labeling: In our prediction tasks we assume that given the topics of the first n time slices (the first 28 years of the dataset) we would like to predict those that will persist in the $(n+1)_{th}$ time slice (the $29th$ year of the dataset). We carried out the labeling process semi-automatically. At first, using a k-nearest-neighbors implementation, for each of the topics from the first 28 time slices we identify the top 5 neighboring topics in the $29th$ time slice to simplify the annotation task. Then, given a topic from the first 28 time slices and its top 5 neighbors in the last time slice, we asked three human assessors (who were domain experts) to determine whether the topic was a continuation or not. This was done for all the topics of the last 28 time slices to have a ground truth for the prediction task. By aggregating the votes of the 3 human assessors each topic was labeled.

The assessors were given instructions on how to label the topics as 'continued' and 'not continued'. These instructions include putting more emphasis on the top 20 words in each topic to take a decision. That is due to the fact that the users of our system would look at the top words of each topic to understand it. As a result, our dataset consists of 839 topics out of which 305 topics are labeled as continued and 534 topics are labeled as discontinued.

[1] The dataset was downloaded from https://www.kaggle.com/.

5.2 Evaluation

In this section we present the evaluation of our method against the state-of-the-art dynamic topic modeling approaches.

First experiment. We evaluate our method using standard IR evaluation metrics, namely, precision, recall, F_1 measure and Mean Average Precision (MAP).

For performing the continuing-topic prediction task using DTM and dDTM we compute the log likelihood of each topic (whose continuation has to be predicted) under the corresponding model. Then we normalize the log likelihood scores to compute a score between 0 and 1. Furthermore, similar to the K2RE case we use 10-fold cross validation to compute the performance of all the models.

The dDTM model estimates the number of topic chains automatically. However, both dDTM and DTM require that we manually set the number of topics per each time slice. As we explained in Sect. 5.1, we estimated the number of topics in each time slice using model selection when building our dataset. Since the average number of estimated topics per time slice was very close to 30 we initialize both DTM and dDTM by setting their number of topics to 30.

Figure 4 shows an interpolated precision-recall diagram. The figure shows that our novel model outperforms the baselines in terms of the precision-recall curves and MAP.

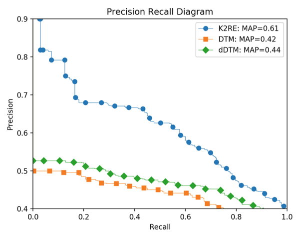

Fig. 4. Precision and recall of our model against DTM and dDTM.

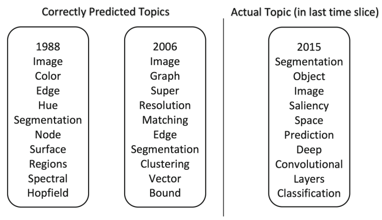

Fig. 5. An example of a continued topic against the ground truth

Furthermore, Table 1 shows a comparison between all the methods with respect to precision, recall, F_1 measure, and MAP. Our results show that the K2RE method outperforms the two other state-of-the-art dynamic topic models used as baselines.

Second experiment. As a second experiment, we use a more objective evaluation without relying on human annotated data. For this purpose, we compute the average Euclidean distance between each topic that was predicted by our model as "continued" and its closest neighboring topic in the last time slice. By doing so, we can compare our model against the other models in terms of how correctly they could distinguish between those topics that continued and the ones that they did not. In this case the lower the distance measure the better each model has performed. Furthermore, by repeating the same procedure for

Table 1. Comparison of approaches: precision, recall, F_1 measure, and MAP.

	Precision(%)	Recall(%)	$\mathbf{F_1}$(%)	MAP(%)
K2RE	61.56	84.78	71.33	61.53
DTM	50.03	85.66	63.17	42.84
dDTM	49.95	89.51	64.12	44.38

the topics that did not continue we can again compute an average Euclidean distance for how similar they are to the actual topics of the last time slice. In this case the higher the distance the better a model has performed.

We present the results of this experiment in Table 2.

Table 2. Comparison of approaches based on distance with ground-truth topics

	K2RE	DTM	dDTM
Ave. Euclidean dist. (Continued)	0.0305	0.0543	0.0493
Ave. Euclidean dist. (Discontinued)	0.1026	0.0744	0.0850

As we can see from the results presented in Table 2, our evolutionary K2RE model achieves higher similarity scores to the actual topics in the last time slice compared with the two baselines. As shown in the table, the average Euclidean distance to the ground truth (topics) in the case of continued topics is lower than that of the baselines. Moreover, the distance is higher in the case of discontinued topics as predicted by the K2RE method. This experiment confirms that the predictions made by the K2RE method are objectively closer to the ground truth as compared with the two baselines.

Qualitative example. Finally, we show a qualitative example of two topics which were predicted by K2RE to continue in the last time slice against the actual topic appearing in that time slice. We chose a topic about "image processing" (shown in Fig. 5) and we observed that *image processing* is an important topic for the NIPS conference over the years. Image segmentation based on color changes to image super resolution and then ten years later into deep convolutional neural networks, object detection, and image segmentation. As a further example on how our model correctly predicted a discontinued topic, we present the top-10 words of a randomly chosen topic from the year 1999: "spatial, temporal, localization, space, vector, belief, sequence, probability, robot, state". As we can see, this topic is mostly related to the navigation and localization of a robot. Our analysis of the data shows that the word "localization" did not occur in the 2015 time slice, hence such topic disappeared. A funding agency or a research organization with access to such insights (e.g., which topics continue and what are the main themes of a continuing topic over time) can make informed decisions and planning. Indeed, by looking at the image processing topic in 2014 we might be able to come up with more specific and detailed directions of research about 2015 (e.g., use of convolutional neural networks) but knowing the general trend and how past research is evolving into the future is also of strong importance.

6 Conclusion

In this paper, we introduced an evolutionary model capable of predicting topics that continue in the next time slice in a sequential corpus of documents. Our results showed that our method outperforms the state-of-the-art dynamic topic models in the prediction task. Our evolutionary K2RE model can learn the changes in the data over time and adapt itself to the changes. We used a corpus of scholarly papers to show the effectiveness of our model.

As a future work we plan to extend our model to other domains. As an example, our model could be adapted to the meetings scenario to prepare users for a future meeting [2] or be used in recommender systems for tracking user intent and context over time in order to anticipate users' information needs.

References

1. Bahrainian, S.A., Mele, I., Crestani, F.: Modeling discrete dynamic topics. In: Proceedings of the Symposium on Applied Computing, SAC 2017, pp. 858–865 (2017)
2. Bahreinian, S.A., Crestani, F.: Towards the next generation of personal assistants: systems that know when you forget. In: Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2017, pp. 169–176 (2017)
3. Baum, L.E., Sell, G.R.: Growth transformations for functions on manifolds. Pacific J. Math. **27**(2), 211–227 (1968)
4. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd International Conference on Machine Learning, ICML 2006, pp. 113–120 (2006)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
6. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc. B **39**(1), 1–38 (1977)
7. Do, T.-M.-T., Gatica-Perez, D.: By their apps you shall understand them: mining large-scale patterns of mobile phone usage. In: Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia, MUM 2010, pp. 27:1–27:10 (2010)
8. Griffiths, T.L., Steyvers, M.: Finding scientific topics. In: Proceedings of the National Academy of Sciences (2004)
9. Kalman, R.E.: A new approach to linear filtering and prediction problems. Trans. ASME-J. Basic Eng. **82**(Series D), 35–45 (1960)
10. Mele, I., Bahrainian, S.A., Crestani, F.: Linking news across multiple streams for timeliness analysis. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, pp. 767–776 (2017)
11. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. In: Readings in Speech Recognition (1990)

12. Talley, E.M., Newman, D., Mimno, D., Herr II, B.W., Wallach, H.M., Burns, G.A., Leenders, A.M., McCallum, A.: Database of NIH grants using machine-learned categories and graphical clustering. Nat. Methods **8**(6), 443–444 (2011)
13. Wang, C., Blei, D., Heckerman, D.: Continuous time dynamic topic models. In: Proceedings of Uncertainty in Artificial Intelligence (2008)