Proceedings of the Twelfth International AAAI Conference on Web and Social Media (ICWSM 2018)

Modeling Evolution of Topics in Large-Scale Temporal Text Corpora

Elaheh Momeni,[1] **Shanika Karunasekera,**[2]
Palash Goyal,[3] **Kristina Lerman**[3]
[1]University of Vienna, Vienna, Austria
[2]University of Melbourne, Melbourne, Australia
[3]University of Southern California, Marina del Rey, USA

Abstract

Large text temporal collections provide insights into social and cultural change over time. To quantify changes in topics in these corpora, embedding methods have been used as a diachronic tool. However, they have limited utility for modeling changes in topics due to the stochastic nature of training. We propose a new computational approach for tracking and detecting temporal evolution of topics in a large collection of texts. This approach for identifying dynamic topics and modeling their evolution combines the advantages of two methods: (1) word embeddings to learn contextual semantic representation of words from temporal snapshots of the data and (2) dynamic network analysis to identify dynamic topics by using dynamic semantic similarity networks developed using embedding models. Experimenting with two large temporal data sets from the legal and real estate domains, we show that this approach performs faster (due to parallelizing different snapshots), uncovers more coherent topics (compared to available dynamic topic modeling approaches), and effectively enables modeling evolution leveraging the network structure.

Introduction

Modeling large temporal text corpora is key to understand how topics attract attention and social and cultural norms evolve over time. Such evolutions are especially prevalent in the social systems, where the rapid exchange of ideas can quickly change the importance of topics and the attention they receive. While embedding methods are promising as a diachronic tool, they are limited to modeling only words or documents and they are not well suited to model changes due to the stochastic nature of their training. This implies that models trained on exactly the same data could produce vector spaces where words have the same nearest neighbours but do not have the same coordinates.

An alternative approach to this topic modeling task is to use the different extensions of Dynamic Topic models (DTM) (Blei and Lafferty 2006), which uses a Bayesian technique to infer the topic structure in the corpora of documents. DTM perceives a document as a mixture of a small number of topics, and topics as a (relatively sparse) distribution over word types. While documents may indeed be seen as a mixture of topics, still documents can be expected to be semantically coherent. However, this prior preference for semantic coherence is not encoded in the model, and any such observation of semantic coherence found in the inferred topic distributions is in some sense stochastic.

Moreover, all available approaches model evolution without considering different forms of topic evolution, such as: Grow - a topic can grow by integrating new words, Contract - a topic can contract by rejecting some of its words, Merge - two topics or more can merge into a single one, Split - one topic can split into two or more topics, Birth - a new topic can appear at a given time, composed of any number of words, Die- a topic can vanish at any time, and all words that belong to this topic lose their membership.

We propose a new computational approach for tracking and detecting temporal evolution of topics in a large collection of texts. This approach uses (1) a word embedding method (Mikolov et al. 2013a; 2013b) to learn contextual semantic representation of words from temporal snapshots of the collection and (2) a dynamic network clustering method (Anagnostopoulos et al. 2016) (such as dynamic community detection algorithm (İlhan and Öğüdücü 2015)) to identify dynamic topics by using a dynamic semantic similarity network developed using embedding models. Modeling topics as clusters in a dynamic network enables assessment and use of different network analysis features as well.

The main advantages of our method are: (1) instead of using embedding to model changes of words we work with clusters of words as topics, which are more meaningful and have higher interpretability, (2) for modeling the evolution we avoid the issue of the alignment of embedding models and change detection by using embedding models directly, (3) we can accelerate the modeling process by parallelizing the process in different snapshots, and (4) we can model change and evolution in different forms (grow, contract, merge, split, birth, die, and survive).

Using two historical corpora, legal data and real estate news, spanning two languages (English and German), we demonstrate that our approach with a short execution time is able to detect dynamic topics with high coherency over the years.

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Method

The system receives a set of temporal text documents, $D^1, ..., D^n$ and for each snapshot D^i, returns a set of topics $T_1^i, ..., T_m^i$. The *p*th topic of the *i*th snapshot, T_p^i, is assumed to be a semantically coherent set of words and consists of a tuple of three: $T_p^i =< e_p^i; T^{i+1}; W_p^i >$. Each topic contains an event (labeled from a set of events: Survive, Grow, Merge, Split, Contract, Die) with regard to the later snapshot, a set of relevant topics from a later snapshot, T^{i+1}, and a set of semantically related words, W. The first two attributes are useful for tracking the evolution of topics. The third attribute shows the content of the topic.

Learning Semantic Space: The distributional methods learn a semantic space that maps words to a continuous vector space $\mathbb{R}^s$, where s is the dimension of vector space. A family of neural language methods embeds words in a fixed-dimension vector space, in such a way that words in similar contexts tend to produce similar representations in vector space. These methods project words in a lower-dimensional vector space, so that each word w_i is represented by a s-dimensional vector v_i. We use the word2vec method (Mikolov et al. 2013a; 2013b) to learn word vector representation (word embeddings) that we track across time.

Modeling Dynamic Topics: For modeling topics we utilize a clustering-based approach on a semantic representation network, extracted from trained embedding models.

Formally, a network N(W, E) denotes the sets of words W and edges E in a network. Edges show semantic similarity between words. Evolving network N is described over a time period and it will be decomposed into a sequence of static snapshots $N^1, ..., N^n$.

For each snapshot, t we list a set of all words W^t and develop N^t by assessing semantic similarity between a pair of words leveraging trained embedded vectors. The edge between two words is created when the similarity score between their embedded vectors are higher than the *semantic similarity threshold,* φ. The semantic similarity is calculated based on the cosine similarity score, formally:

$$simSemantic(v_i^t, v_j^t) = \frac{v_i^t, v_j^t}{||v_i^t||_2 ||v_j^t||_2} > \varphi \quad (1)$$

Next, topic detection leverages community detection algorithms (i.e. network clustering approach). While N^t represents the *t*th snapshot of the network, $T^t = \{T_1^t, T_2^t, ..., T_m^t\}$ represents the set of topics in the snapshot t.

Finally, according to available approach (Anagnostopoulos et al. 2016; İlhan and Öğüdücü 2015) for dynamic community detection, we use a matching metric in order to track evolution of a topic from one snapshot to the following snapshots and measure the similarity between two clusters in successive time steps. Two topics match if their similarity value $sim(T_i^t, T_j^{t+1})$ exceeds a user set *similarity threshold* θ, formally:

$$sim(T_i^t, T_j^{t+1}) = min\left(\frac{|T_i^t \cap T_j^{t+1}|}{|T_i^t|}, \frac{|T_i^t \cap T_j^{t+1}|}{|T_j^{t+1}|}\right) > \theta \quad (2)$$

Being able to reuse traditional community detection techniques without having to modify them for detecting clusters in each snapshot separately is one of the main advantages of this described solution. Furthermore, this method results in parallelizing and accelerating dramatically community detection on all snapshots which reduces running time. However, this solution is not perfect due to the instability of community detection algorithms. The majority of community detection algorithms that work well are stochastic, two runs on the same network do not necessarily provide the same partition. Sometimes, for two networks which are the same apart from tiny modifications, the solution can provide different results. An alternative approach with higher stability is to study all snapshots simultaneously (Tantipathananandh, Berger-Wolf, and Kempe 2007).

Modeling Topic Evolution: In order to model evolution of topics, we use an existing metric, namely *Instability*, in order to compute the percentage of increase/decrease in the number of words in a topic (İlhan and Öğüdücü 2015). More formally, given a topic T_i^t has n_i^t members at time snapshot t and a successor cluster T_j^{t+1} has n_j^{t+1} members at time snapshot $t+1$, the *Instability* is defined as: $inst(T_i^t, T_j^{t+1}) = (n_j^{t+1}/n_i^t) - 1$.

Next, considering a user-defined *instability threshold, (ϕ)*, six events proposed by Ilhan et al. (2015), including Grow, Survive, Contract, Split, Merge and Die are identified to model evolution of a topics. More precisely, after positive matching between T_i^t and T_j^{t+1}, we could then label the similar topic as being the:

- **Contract**: If $inst(T_i^t, T_j^{t+1}) < -\phi$. The topic has been contracted (i.e. there is a substantial percentage decrease in the number of members).
- **Survive**: If $-\phi < inst(T_i^t, T_j^{t+1}) < \phi$. The topic has been survived (i.e. there is a negligible increase/decrease in the number of members).
- **Grow**: If $inst(T_i^t, T_j^{t+1}) > \phi$. The topic has been grown (i.e. there is a substantial percentage increase in the number of members).
- **Split**: A topic T_i^t at time t may match with a set of topics $T_*^{t+1} = \{T_i^t ... T_j^{t+1}\}$ in a later snapshot in split case.
- **Merge**: A set of topics $T_*^t = \{T_1^t ... T_i^t\}$ may match to a topic T_j^{t+1} in the subsequent snapshot $t+1$ in merge case.
- **Die**: If there is no similar topic at a later snapshot, this means θ is not exceeded. Then it is assumed that the topic dies.

Experimental Evaluation

Dataset

We evaluate the performance of our proposed approach on two corpora: a Legal Dataset and News articles related to Real Estate.

Dataset1-Legal Data: A collection of legal opinions from court websites and from data donations between 1920 to 2014 in English[1]. It contains $2,902,806$ documents and $1,721,377,159$ word tokens. We divided this collection to 19 snapshots. Each snapshot is related to a period of 5 years (such as 1920-1925 or 1925-1930).

[1] Available at https://www.courtlistener.com/opinion/

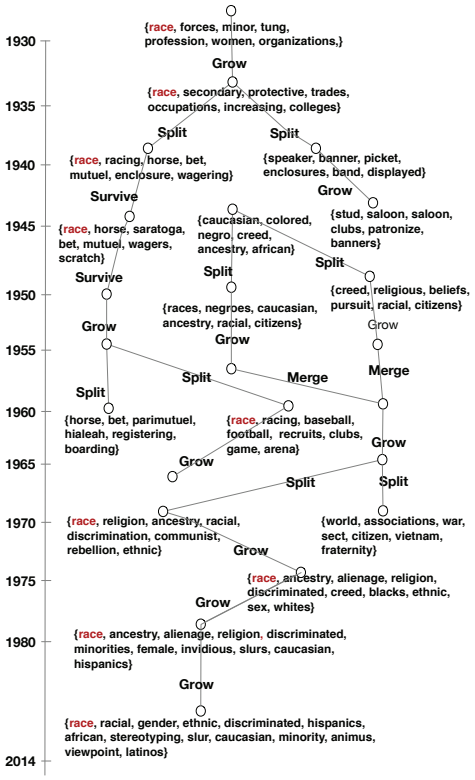

Figure 1: Demonstration of topics related to "race", proposed by our system and their evolution events

Dataset2-Real Estate Data: We crawled collection of daily news items related to real estate between 2000 to 2017 in German (collected using Bing and Google news). It contains $204,521$ documents and $46,237,197$ word tokens.

Set-up

For each snapshot of each dataset, first, we trained embedding models using the skip-gram implementation of the Gensim library. Second, using trained models, we created semantic similarity networks for each snapshot with a satisfactory density level by setting $\varphi = 0.5$. Third, we used Louvain, a well-known community detection algorithm to obtain clusters of words. Finally, starting from the earliest snapshot of each dataset, we identified dynamic topics, and their related evolution events. In order to demonstrate the effectiveness and accuracy of our proposed approach, we conducted three experimental evaluations: the first evaluation compares coherency of the topics proposed by our system to a traditional dynamic topic model algorithm (DTM), taking into account human perception of the topics, the second study shows evidence of our system's capability to identify dynamic topics. This study considering a word, "race", with high contextual evolution, we show how our system is capable of modeling the evolution of the relevant topic, and third, we reported the execution time of our system and compared it with the running time of the other approach. Furthermore, in order to compare the performance of our approach against available topic modeling approaches, we trained DTM models on both datasets using DTM Gensim implementation (choosing the default hyper-parameters)[2].

Results

Topic Coherency Evaluation: An evaluation of topic models is usually based on estimated likelihood or perplexity. However, they are inadequate proxies for how topics are perceived by humans. According to Newman et al. (Newman, Karimi, and Cavedon), a topic has frequently some odd-words-out in the list of top words. This leads to the idea of a scoring model based on word association between pairs of words for top-5 word pairs in a topic. For measuring word association, instead of using the collection itself, a large external text data source should be used (in order to avoid reinforcing noise or unusual word statistics). PMI is seen as the measure of word association. Therefore, for considering human perception of topics we used pointwise mutual information (PMI) based on an external data source, Google 5-grams data.

Using Google 5-gram data as an external source[3], we counted a co-occurrence of the topic's five highest ranked words in any of the 5-gram and scored a topic's coherence by averaging the pairwise PMI score of words. Higher average PMI implies a more coherent topic. We assessed average PMI score for 400 randomly selected topics (100 proposed by our approach for each dataset and 100 proposed by DTM for each dataset). We found that the median coherency of all topics proposed by our approach (median-Legal = 8.1 and median-Real-Estate = 7.4) is higher than the coherency of all topics proposed by DTM approach (median-Legal = 6.3 and median-Real-Estate = 6.2). However, the median coherency of all topics for the Real-Estate dataset, with German language is lower than for the Legal dataset. This is due to the lower accuracy of tokenization of German documents (e.g., "fortbildungsangebotweiter" that be should tokenized as "fortbildungs", "angebot", "weiter").

Furthermore, we selected 25 topics each from the real estate and legal data. Then we asked five human subjects (who can speak both languages) to score each of the 50 topics on a three-point scale where 3="useful" and 1="useless". Our human scoring of the these topics has high inter-rater reliability measuring using Fleiss' kappa score (0.71). Finally, we see broad agreement between the average PMI score and the human scoring. The correlation between the average PMI score and the mean human score is $\rho = 0.71$ for Real Estate and $\rho = 0.74$ for the Legal data (we define correlation ρ as the Pearson correlation coefficient).

Case Evaluation: In order to test the ability of our approach to identify dynamic topics, we investigated a case study related to a word, "race", in the Legal dataset. It appears to have a high contextual evolution. Figure 1 shows

[2]As the execution time of DTM for all snapshots is very long (see Execution Time Evaluation Section), we trained the models using only the latest four snapshots of each dataset.

[3]We use the Google 5-gram data as the external data source as it supports temporal comparison of words. The result obtained is similar to an evaluation using Wikipedia (Newman, Karimi, and Cavedon)

Figure 2: Execution time for different sizes and numbers of snapshots in 2 various set ups: sequential and parallel.

how our system models the evolution of relevant topics related to this word. First, for each snapshot (in our model the word appeared first in 1930) the figure shows the proposed cluster, which contains the word "race" (shown in bold red). For each cluster, only top semantically similar words to the word "race" are shown. Furthermore, in order to demonstrate the evolution of the topic the figure shows other relevant topics if in previous or later snapshots the topic merges with or splits from other topics containing the word "race". For topics that do not contain the word "race", we also show top semantically similar words to the word "race". We only show the evolution of topic until 1980 as the topic in later snapshots (until 2014) only continues to grow. Second, it shows similar topics in different snapshots (dynamic topics). The lines connect each topic in a snapshot to the top-2 similar topics in the later snapshot using equation 2. Finally, we labeled the matching topic to each similar topic by setting instability threshold, ϕ, as 0.5 and using the definitions given for Birth, Grow, Merge, Split, Contract, and Die in the previous section.

The results proposed by our system clearly correspond to how topics surrounding the word "race" in fact evolved. Figure 1 demonstrates these results. Before 1955, the word appeared in topics connected to horse racing. Then, between 1945 and 1960, we can observe the evolution of other topics, related to religion and caucasian & negro. It was only around 1960 alongside the Civil Rights Movement that the word race began to take on the topic associations we know today for example "religion, ancestry, racial, discrimination". Later, the topic grew and more words (such as gender, ethnic, discriminated, hispanics, african, race stereotyping, slur, minority) appeared.

Execution Time Evaluation: Finally, we investigate how fast our approach performs in different set ups and compare it to other approaches. As our approach can perform each step independently on each snapshot, it can run the whole process in parallel (multi-thread processing) to reduce the running time. More precisely, Figure 2 shows execution time for different sizes of snapshots. Size of snapshots in both datasets range between 100MB-5GB (smaller size in earlier snapshots). Therefore, the figure shows execution time for average snapshot size of 500MB and 1GB. The Figure also shows the effect of different numbers of snapshots in different set-ups: sequential (run the process sequentially for each snapshot) and parallel (run the process in parallel for all snapshots) running. Finally, the last plot shows comparison of the execution time of DTM approach with the execution time of our approach. As execution time of DTM was long (almost 5 days for four snapshots with avrage size of 1GB) we just trained the models for the latest 3 snapshots of each dataset. We executed the process on a machine (i7 core processor and 64GB RAM space). Figure 2 clearly demonstrates the scalability of our approach for detecting and modeling evolution of dynamic topics in a large-scale text corpora. It executes for an average snapshot size of 1GB for all 19 snapshots around 30 hours in the parallel set up.

For future work, we will explore the effect of different thresholds and usage of different clustering methods on the performance and execution time and effectiveness of our approach.

References

Anagnostopoulos, A.; Lacki, J.; Lattanzi, S.; Leonardi, S.; and Mahdian, M. 2016. Community detection on evolving graphs. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 3522–3530.

Blei, D. M., and Lafferty, J. D. 2006. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, 113–120. New York, NY, USA: ACM.

İlhan, N., and Öğüdücü, c. G. 2015. Predicting community evolution based on time series modeling. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, 1509–1516. New York, NY, USA: ACM.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013a. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; and Dean, J. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, 3111–3119. USA: Curran Associates Inc.

Newman, D.; Karimi, S.; and Cavedon, L. External evaluation of topic models. In *in Australasian Doc. Comp. Symp., 2009*, 11–18.

Tantipathananandh, C.; Berger-Wolf, T.; and Kempe, D. 2007. A framework for community identification in dynamic social networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, 717–726. New York, NY, USA: ACM.