

Evolution of Movie Topics Over Time

Cong Meng, Mian Zhang, Wenqiong Guo

December 14, 2012

1 Introduction

Topic modeling has emerged as a powerful tool for understanding and managing large electronic archives. [1, 2] It provides the methods to discover the hidden themes that pervade the collection; annotate the documents according to those themes; and then use annotations to organize, summarize and search the texts. Besides discovering topics from corpus, topic modeling algorithm has also been used to model the evolution of topics over time, as well as the connections/hierarchies of topics by treating documents as time series data. In this project, we are particularly interested in applying the topic modeling method to explore the dynamics in the movie topics evolution.

The contents of a journal article, specifically the words or terminologies used, can indicate the topics this article focuses on. Similarly, the synopsis/storyline of a movie can serve as a good indication of what topics it relates to. By treating the storyline of each movie as a document, we can analyze the words used in the storyline to get the information of topics. The most frequent movie topics in one time period will give us some idea about the trend of movies at that time. Putting the topics of movies in a time series will hopefully reveal some interesting dynamics in theater. By using the data from IMDB, especially the story lines, we are hoping to show how the movie topics evolve over time.

2 Model Build-up

We build up our initial model based on the LDA in probabilistic modeling. The data are assumed to be observed from a generative probabilistic process that includes hidden variables (the hidden thematic structure in movie); we will infer the hidden structure using posterior inference (the topics that describe a movie collection); and try to situate new data into the estimated model (fit a new movie into the topic structure).

2.1 Latent Dirichlet Allocation

The intuition of Latent Dirichlet Allocation (LDA) is that documents exhibit multiple topics. For example, a movie can exhibit a combination of different topics as 'computer hacker', 'car chase' and so on. The generative model for LDA assumes that each topic is a distribution over words; each document is a mixture of corpus-wide topics; each word is draw from one of those topics. In reality, we only observe the documents, while the other structures are hidden variables. Our goal is to infer the hidden variables, i.e., compute the conditional probability distribution $P(\text{topics}, \text{proportions}, \text{assignments} | \text{movies})$. Let K be a specific number of topics, V the size of the vocabulary, $\vec{\alpha}$ a positive K -vector, and η a scalar. We let $Dir_V(\vec{\alpha})$ denote a V -dimensional Dirichlet with vector parameter $\vec{\alpha}$ and $Dir_K(\eta)$ denote a K dimensional symmetric Dirichlet with scalar parameter η

- (1) For each topic,
 - (a) Draw a distribution over words $\vec{\beta}_k \sim Dir_V(\eta)$
- (2) For each document,
 - (a) Draw a vector of topic proportions $\vec{\theta}_k \sim Dir(\vec{\alpha})$
 - (b) For each word,
 - (i) Draw a topic assignment $Z_{d,n} \sim Mult(\vec{\theta}), Z_{d,n} \in 1, 2, \dots, K$
 - (ii) Draw a word $W_{d,n} \sim Mult(\vec{\beta}_{Z_{d,n}}), W_{d,n} \in 1, 2, \dots, V$

The graphical model for LDA is shown in Figure 1.

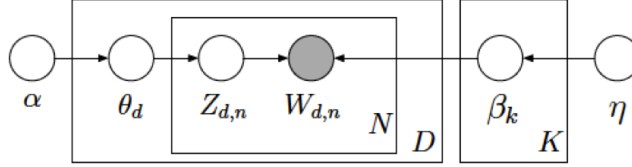


Figure 1: This figure is taken from [1]. A graphical model representation of the Latent Dirichlet Allocation (LDA). Nodes denote random variables; edges denote dependence between random variables. Shaded nodes denote observed random variables; unshaded nodes denote hidden random variables. The rectangular boxes are "plate notation", which denote the replication

LDA provides a joint distribution over the observed and hidden variables. The hidden topic decomposition of a particular corpus arises from the corresponding posterior distribution of the hidden variables given the D observed documents $\vec{\omega}_{1:D}$.

$$p(\vec{\theta}_{1:D}, z_{1:D,1:N}, \vec{\beta}_{1:K} | \omega_{1:D,1:N,\alpha,\beta}) = \frac{p(\vec{\theta}_{1:D}, z_{1:D}, \vec{\beta}_{1:K} | \omega_{1:D,\alpha,\beta})}{\int_{\vec{\beta}_{1:K}} \int_{\vec{\theta}_{1:K}} \sum_{\vec{z}} p(\vec{\theta}_{1:D}, z_{1:D}, \vec{\beta}_{1:K} | \omega_{1:D,\alpha,\beta})} \quad (1)$$

This posterior can be thought of the 'reversal' of the generative process described above. The quantities needed for exploring a corpus are the posterior expectations of the hidden variables.

2.2 Algorithms

The central computational problem for topic modeling with LDA is approximating the posterior distribution in above. We choose to use mean field variational inference to do this, which is implemented in the package provided by professor David Blei. This allow us to quickly get our preliminary result. The basic idea behind variational inference is to approximate an intractable posterior distribution over hidden variables, such as (1) with a simpler distribution containing free variational parameters. These parameters are then fit so that the approximation is close to the true posterior. Variational inference is chosen because it can be faster than sampling-based approaches such as Gibbs sampling. We introduce a variational distribution over the latent variables $q(\beta, \theta, z)$, where $q(\beta, \theta, z) = \prod_{k=1}^K q(\beta_k | \lambda_k) \prod_{d=1}^D q(\theta_d | \gamma_d) \prod_{n=1}^N q(z_{d,n} | \phi_{d,n})$. This indicates that the instance of each variable has its own distribution. And each component is in the same family as the model conditional.

$$p(\beta | z, \omega) = h(\beta) \exp(\eta_g(z, \omega)^T \beta - a(\eta_g(z, \omega))) \quad (2)$$

$$q(\beta | \lambda) = h(\beta) \exp(\lambda^T \beta - a(\lambda)) \quad (3)$$

The object to be optimized is the evidence lower bound (ELBO) with respect to q .

$$\mathcal{L}(\lambda, \phi_{1:n}) = E_q[\log p(\beta, Z, \omega)] - E_q[\log q(\beta, Z)] \quad (4)$$

This is equivalent to finding the $q(\beta, z)$ that is closest in KL divergence to $p(\beta, z | \omega)$. And the update rule can be summarized as:

Initialize λ randomly

Repeat until the ELBO converges

- (1) For each data point, update the local variational parameters: $\phi_i^t = E_{\lambda^{t-1}}[\eta(\beta, \omega_i)]$ for $i \in 1, \dots, n$
- (2) Update the global variational parameters: $\lambda^t = E_{\phi^t}[\eta_g(Z_{1:n}, \omega_{1:n})]$

2.3 Dynamic Topic Models

In the LDA model, movie synopsizes are exchangeable no matter which year the movie was from. However, this assumption is not appropriate. Because the movies span tens of years, during which time the language as well as the style of a topic are very likely to have changed, , and certain movies from earlier time are

likely to have impact on later movies. Furthermore, we want to track how these changes happen over time. Dynamic topic models (DTM) captures the evolution of topics in a sequentially organized movies. In the DTM, we divide the data by time slice, e.g., by year. We model the movies of each slice with a K-component topic model, where the topics associated with slice t evolve from the topics associated with slice $t-1$. The time-series topics is modeled by a logistic normal distribution of $\beta_{k,1} \rightarrow \beta_{k,2} \rightarrow \dots \rightarrow \beta_{k,T}$. Specifically, we assume $\beta_{t,k}|\beta_{t-1,k} \sim \mathcal{N}(\beta_{t-1,k}, I\sigma^2)$, and $p(\omega|\beta_{t,k}) \propto \exp(\beta_{t,k})$. we can approximate the posterior over the topic decomposition with variational methods explained in previous section. At the topic level, each topic is now a sequence of distributions over words. Thus, for each topic and year, we can calculate the probability of the words and visualize the topic as a whole with its top words over time. This gives a global sense of how the important words of a topic have changed through the span of the collection. For individual words of interest, we can examine their score over time within each topic. We can also examine the overall popularity of each topic from year to year.

3 Data and Preprocessing

We have modified the python script provided by Professor Chris Potts to scrape the data from IMDB website. In order to get useful informations, we only collect the data of movies that have synopsis. This criteria, we assume, will ensure that the movies we analyze have good size of audience and represent the mainstream of the film industry. The information of a movie recorded is: the html index, name, screen time, number of reviews and synopsis. This dataset was preprocessed in R to generate data in the standard format. All the story lines were gone through to generate the "dictionary". The non-informative words and the numbers indicating years are filtered out in this process. We construct a sparse matrix for this collection of movie story lines based on the dictionary. The movies in the matrix were reorder by time for the purpose of dynamic modeling. Then the data is ready to be trained and tested.

4 Result

4.1 Topic Identification

To test our data pre-processing correctness, and to pre-identify topics in our data set, we first modeled the entire movie synopsis data set with a 100 topics LDA model (Figure 2) (if time permits, we would also like to compare LDA model fitting with DTM model, to see the potential advantages and disadvantages of treating movies data as time series). Out of the 100 topics, around 10 topics were found more probable than others. Thus in Dynamic Topic Modeling (DTM), we picked 20 topics to fit our movie data. By treating movie data as a time-series text data, DTM could catch the trend of movie topics evolution. Five out of the twenty topics along with top key words within the topics are listed below (Figure 3).

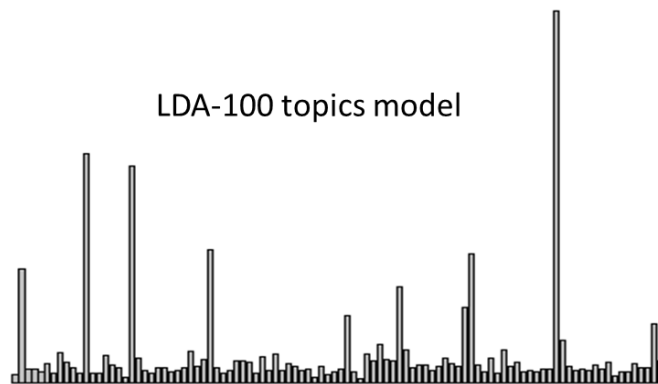


Figure 2: Topic proportion histogram from LDA model. Movie synopsis data (N=1034 was used for test) were modeled through a 100 topics LDA model. The text data were stemmed to word roots. And high and low frequency words , numbers, punctuations and stop words were filtered out of the vocabulary.

Top 10 most probable words from the most probable topics*

"People Name"	"Home"	"School"	"Action"	"Horror"
david	love	Helen	kill	vampire
chris	family	woman	car	blood
john	father	look	leave	dawn
paul	marry	leave	west	blade
malcolm	friend	house	call	love
jane	some	time	run	Leave
kevin	sister	meet	terri	Murder
leave	house	friend	gun	father
house	day	talk	police	night
friend	princess	school	home	sudden

* A total of 20 topics were inferred in dtm. The top 100 words with the highest probability in each topic were scanned and filtered to select the top 10 words.

Figure 3: Dynamic Topic Modeling (DTM) 20 topics model results. Time series of movie synopsis data (N=2878) were run through DTM. Here shows 5 topics out of the 20, along with top 10 words within the topic. Topic names were assigned based on top key words.

4.2 Trending of topics

If we pick out certain topics and monitor their proportion change through time, we could extract information in how certain movie topics evolve (Figure 4) and potentially observe how they were affected by certain social events at particular time in history. In Figure 4, the proportions of action and Sci-fi movies were drawn through time. As shown in the figure, action movies were more popular in earlier days, and became a bit less popular in 21st century, and Sci-fi movies have been popular ever since. Interestingly, we saw a large spike in the year of 1994 for both action and Sci-fi movies.

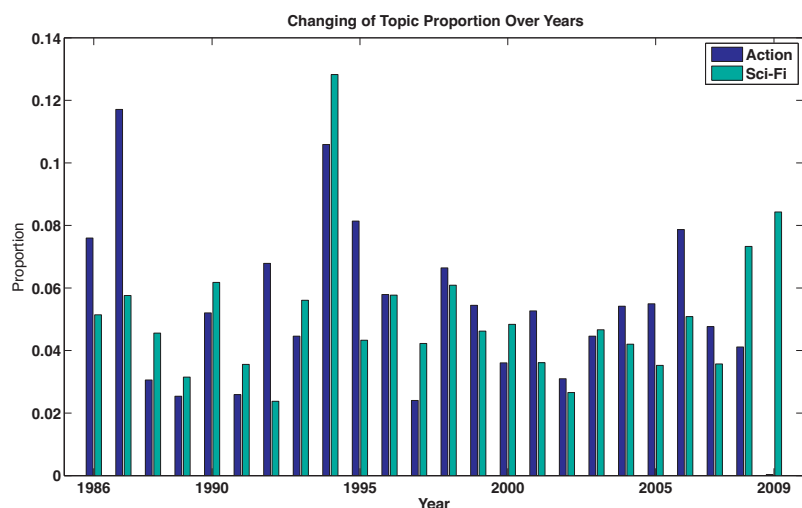


Figure 4: . Action and Sci-Fi movies proportion evolve through time. In general action movies had a higher proportion in early 90s, and Sci-fi movies have been popular ever since.

4.3 The evolution of each topic

Top key words were picked out to look at how multinomial distribution evolves through time. Figure 5 shows top key words from topic action and Sci-fi. We found in action movies, gang has become less popular through the years, while superman and monster are becoming more popular. And gun has been a top key

word in action movies since the early days. In Sci-fi movie, wolverine and mutant have become more popular since the 90s, possibly due to the appearance of X-Man movie series.

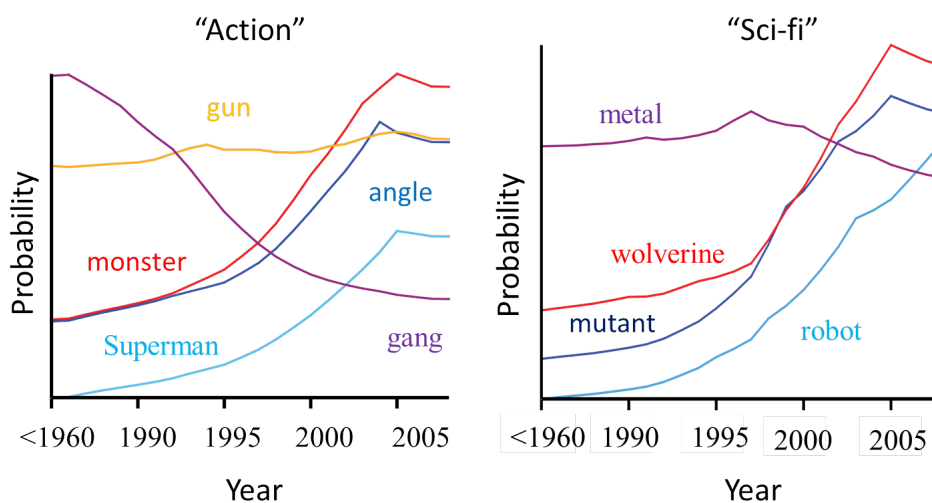


Figure 5: Top key words evolution in action and Sci-fi movies.

5 Conclusion and future improvement

By applying DTM on movies synopsis data set, we were able to extract information on main movie topics, as well as how they evolve over time. By looking at top key words probability from each topic, we could monitor the popularity of them and observe them changing over time, and their correlation with other key words within the same topic.

There exists much more information from DTM modeling for us to explore given enough time. For example, the two topics, action and sci-fi, had a peak at the year of 1994. One potential explanation is to look back into the history, a lot of excellent movies such as Forrest Gump, Pulp Fiction, The Shawshank Redemption, Quiz show, The stand, Four wedding and a funeral (all from IMDB TOP 250 movies) came to the market in 1994, in one aspect verifies the popularity of several topics in 1994. We could also instead of dividing movie synopsis data in a unit of years, divide movies monthly or quarterly to see if there is seasonal effects on movie topics evolution.

DTM is a versatile and general model, but there exist a few simplified assumptions. For example, all the topics and vocabulary within the model are assumed to remain existing over the years, but almost certainly some of the topics or vocabulary may fade away from movies that were made more recently. On the contrary, some new topics may emerge as well. Thus, if we could incorporate topics that are more flexible instead of being fixed and never disappear, we could potentially fit the data better.

References

- [1] David M. Blei, *Probabilistic Topic Models*. COMMUNICATION OF THE ACM,2012
- [2] David M. Blei, John D. Lafferty *Dynamic Topic Models* ICML,2006