Dirichlet-Hawkes Processes with Applications to Clustering Continuous-Time Document Streams: A Simulation Study

Project Report

---

by

Yijun Chen

yijunc@clemson.edu

David Kwietniewski

dkwietn@clemson.edu

April 2016

Abstract

Our project is based around a paper called Dirichlet-Hawkes Processes with Applications to Clustering Continuous-Time Document Streams. The majority of the project deals with cluster analysis. Cluster analysis is the organization of a collection of patterns into clusters based on their similarity. Intuitively, patterns within valid clusters are more similar to each other than they are to a pattern belonging to a different cluster. The applications of clustering can be found in various areas, marketing, insurance, city planning, and many other areas.

In order to make sure clustering can help group and analyze the properties of the dataset, the clustering method must be good. The major clustering approaches are partitioning algorithms, hierarchy algorithms, model-based algorithms, and density-based algorithms. This project uses density based clustering to cluster news articles. The density based clustering algorithm selected uses Gibbs sampling. The Gibbs sampling algorithm leverages both the textual contents and temporal dynamics of the document streams to obtain better clustering. A random process, known as the Dirichlet-Hawkes Process was used to take into account both textual and temporal information of the data in a unified framework. This model establishes a connection between Bayesian Nonparametrics and temporal point processes. As mentioned in the paper, the model leads to better predictive performance in relation to content complexity and temporal components of the data.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

There are already lots of text mining methods focused on text content analysis. This project used the Dirichlet-Hawkes process to combine both content and temporal information for use in clustering [1]. Using the clusters, the present information, dependent on past events, can be used to predict future trends. Such a method is implemented by Latent Dirichlet Allocation in this project. For the data set, a set of online news articles were collected and used. The collected documents were transformed into a corpus and then a matrix for further analysis. The Latent Dirichlet Allocation is a straightforward implementation of the Dirichlet-Hawkes process. The Latent Dirichlet Allocation can also be treated as a simplified version of Dirichlet-Hawkes process by fixing the random parameter. The difference between the two models will be explained in further detail later.

For the temporal analysis part of this project, the nature of online documents can be used to fix the arrival time as small intervals by discretizing the time-line into bins and still guarantee that the actual counts of events are nonuniform over time. The probability that each document drops into the small bins are completely randomized. This results in a self-exciting and mutual exciting intensity measure for the processing of each topic.

More precisely, the project realized the following:

- A connection between Bayesian Nonparametrics and Stochastic Point Processes allows the number of clusters to accommodate the complexity of online streaming contents while at the same time learning the latent dynamics governing the respective continuous arrival patterns inherently. The results show that Dirichlet-Hawkes processes can recover meaningful topics rather than simply cluster words.

- The combination of Dirichlet processes and Hawkes processes have implications beyond clustering document streams. The processes can also be used to predict the future probability for the occurrence of a certain topic. By using visualization tools in R, like Wordcloud and Shiny, it is possible to analysis the interaction and similarity between each topic and give a 3D visualization result of the entire process.

- The conditional density distribution is used to simulate the intensity function for the Hawkes process and also calculate the perplexity of the clustering process to create a model and do a performance analysis in order to make sure the fitted model has performed well.

Chapter 2

Preliminaries

The following sections describe the most important definitions and theorems used by this project, which aims help with the understanding of the Dirichlet-Hawkes Processes[1].

2.1 Density based clustering for mixed model

A cluster is defined as a maximal set of density connected points. The words in each document follow a distribution as determined by the document contents, but it is assumed that there is a hidden distribution, which is the posterior distribution in the mixed model for words distributed over the topic. The entire clustering process is to create the set connected by those points. Each cluster represents a certain topic and has its own distribution and distribution parameter. This clustering process was implemented using the Gibbs sampling method. This cluster process not only gathered the density related words together, but also the words in the same cluster can form a meaningful story.

2.2 Dirichlet Process

The Dirichlet process (DP) is one of the most prominent random probability measures due to its richness,computational ease, and interpretability[7]. It can be used to model the uncertainty about the functional form of the distribution for parameters in a model.The Dirichlet Distribution is a distribution over the $K-1$ probability simplex. Let p be a K dimensional vector s.t. $\forall j : , p_j \geq 0$ and $\sum_{j=1}^{K} p_j = 1$, then

$$P(p|\alpha) = Dir(\alpha_1, \cdots, \alpha_K) = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_{j=1}^{K} p_j^{\alpha_j - 1}$$

Thus the first term in the above equation is just a normalization constant, the Dirichlet Distribution is conjugate to the multinomial distribution, i.e. if

$$c|p \backsim Multinoimial(\bullet|p)$$

then the posterior also follow Dirichlet Distribution

$$P(p|c = j, \alpha) = \frac{P(c = j|p)P(p|\alpha)}{P(c = j|\alpha)} = Dir(\alpha')$$

The main feature for Dirichlet Process is it defines a distribution over distribution, which means:

$G \backsim DP(\bullet|G_0, \alpha)$, where $\alpha > 0$ is a scaling parameter, and G_0 is the base distribution.

Let Θ be a measurable space, G_0 be a probability measure on Θ, then for all $(A_1, \cdots, A_K)$ finite partitions of Θ,

$$G \backsim DP(\bullet|G_0, \alpha)$$

$$G(A_1), \cdots, G(A_K)) \backsim Dir(\alpha_0 G_0(A_1), \cdots, \alpha_0 G_0(A_K))$$

The Dirichlet Process shows that its prior and conditional posterior distributions all follow the Dirichlet distribution. This is an application of the Bayesian Theorem. If the prior parameter is known first, then the given parameter can be used along with the distribution to generate data sets. It is assumed that this data set has a posterior distribution. Applying the clustering algorithm to fit the dataset into a model can yield the estimated parameter for the posterior distribution, which is actually a Bayesian nonparametric inference model. The main algorithm used in the clustering process is Gibbs sampling by maximizing the likelihood function for the posterior parameter.

The prior distribution is treated as the words distribution in each document, and the posterior distribution as the topics over the documents. By sampling on all the words in the document, the cluster can also create its own distribution, which is the words distributed in each topic. The parameter for the initial prior distribution comes from a Poisson Point Process, which also uses the temporal information for each document.

2.3 Hawkes process

The Hawkes processes is a doubly stochastic point process that is able to model reciprocity between groups of individuals that also makes events associated with edges co-dependent through time[4]. The Hawkes Process applied in the project is a Poisson process-based model, where the base Poisson process counts the arrival time of each document[3]. The Dirichlet Process is used to transform the temporal information into the birth-and-death time for each topic of posterior distribution.

A general definition for a self-exciting process process N has the following definition:

$$\lambda(t) = \lambda_0(t) + \int_{-\infty}^{t} v(t-s)dN_s = \lambda_0(t) + \sum_{t_i<t} v(t-t_i),$$

where $\lambda_0 : \mathbb{R} \mapsto \mathbb{R}_+$ is a deterministic base intensity and $v : \mathbb{R}_+ \mapsto \mathbb{R}_+$ expresses the positive influence of the past events t_i on the current value of the intensity process. Use the basic definition of intensity process, Hawkes Process can be easily introduced. The Hawkes Process proposes an exponential kernel $v(t) = \sum_{j=1}^{P} \alpha_j e^{-\beta_j t} 1_{\mathbb{R}_+}$, so that the intensity of the model becomes :

$$\lambda(t) = \lambda_0(t) + \int_0^t \sum_{j=1}^{P} \alpha_j e^{-\beta_j (t-s)} dN_s = \lambda_0(t) + \sum_{t_i<t} \sum_{j=1}^{P} \alpha_j e^{-\beta_j (t-t_i)},$$

The counting process was used to analyze the trend for each topic with respect to time. The Bayesian nonparametric model used on this time-series data is generative and accounts for the rate of events between clusters of individuals. It is built upon mutually-exciting Hawkes processes. Pairs of mutually-exciting Hawkes processes are able to capture the causal nature of reciprocal interactions. Here the processes excite one another through their actualized events. Since Poisson processes are a special case of Hawkes processes, the Poisson Process can be applied to simplify the model fitting[2].

Chapter 3

Data Processing

3.1 Data Collection

The dataset used for this project was obtained from the One America News Network website. All of the U.S. news article pages between September 12, 2015 and January 14, 2016 on the website were downloaded to be used for this project's dataset. Over 1200 news articles were obtained from oann.com. Data was obtained from the One America News Website because in comparison to other news sources like CNN, Fox, and the NY times, the One America News websites provides a much simpler design making the automated download of news articles much easier. Data for this project was obtained by writing a web crawler to go to the page for past news articles on oann.com, extract the links of all of the articles, and then download those articles as html pages. The webcrawler was written in Java, and it employed a library known as jsoup to parse the html of the pages containing lists of previous articles in order to determine what links to download.

3.2 Data Preprocessing

Once all of the news article web pages from the One America New Network website were downloaded, they had to be converted into a format that could be processed before clustering and analysis could be performed. A Java library known as Boilerpipe was used to extract the main article text from the webpage's html. Boilerpipe uses computer vision and machine learning techniques to determine the important content on a webpage and translates it to plain text. Boilerpipe is able to filter out extraneous information such as page headers, pictures, and sidebars from the actual page content. The simplicity

of the OANN article pages website allowed for an accurate extraction of the news articles from all of the extraneous information on the web page. Starting with the extracted text, it still needed further preprocessing, and Python scripts for all additional preprocessing of the articles. First, the title and publication date were extracted from the article to be used later. All of the article information was combined into one large tab-separated text file that contained an article's publication date, title, and text separated by tabs on a single line.

The article's text was further processed by removing stop words and special characters, since neither provides additional information that is useful when clustering using a bag of words approach. Then, Python's natural language processing toolkit was used to transform all the words to their base forms. This was a necessary step because it made it so that similar words would be represented as having the same meaning during clustering. The naive solution to this problem is a regular expression based approach that removes ending characters from each word. For example, the words "running", "runs", and "run" relate to the same thing, and by removing the ending with regular expressions, only the base word "run" is left. This approach works for most cases, but there is already a problem when the word "ran" is introduced, and simply changing the ending does not relate "ran" and "run" together. Python's natural language processing toolkit fixes this problem through the use of a wordnet to look at the base meaning of the words. The natural language processing toolkit handles the case of "ran" and "run", and it also handles more complicated cases, like "children" and "child" by realizing that they both relate to the same base word of "child." The final step was to convert our tab-separated file into a format that could be used by R, which was the language that was used to perform the project's actual computation. In order for the articles to be used by the R algorithm, each article had to be contained in its own file before it could be processed. To fix this problem, another Python script was written to take the tab separated collection of articles and convert each line into a file with the publication date embedded into the file's metadata to be used by the R program.

3.3 Data Processing

Upon completion of the data preprocessing, the next step was to perform the actual experiment. The R programming language was chosen for the computational part of the project because it proved to be more suitable than Python for such large scale clustering. The original algorithm was implemented in Python, but problems were encountered using Python that made it easier to use the R programming

language instead of trying to find workarounds for the encountered problems. The main feature of the project was the Dirichlet-Hawkes Process. The Dirichlet-Hawkes Process fixes deficiencies of the the Recurrent Chinese Restaurant Process for clustering text, and it connects Bayesian Nonparametrics and Temporal Point Processes. The main problem for the Recurrent Chinese Restaurant Process was the way that it handles time. The process divides time into a series of episodes that is not dynamically adjusted by the the algorithm, and the Dirichlet-Hawkes Process solves this problems by accounting for the fact that smaller time periods may provide a better measure for clustering. The Dirichlet-Hawkes Process starts with a time sorted collection of documents. The first phase of the Dirichlet-Hawkes Process is known as the Dirichlet Process. During the Dirichlet Process, the documents are clustered using a process known as Gibbs Sampling. Once the documents are initially clustered, the results of those clusters are given to the Hawkes Process. The Hawkes Process accounts for the temporal components of each cluster and is able to determine how the clusters relate to each other.

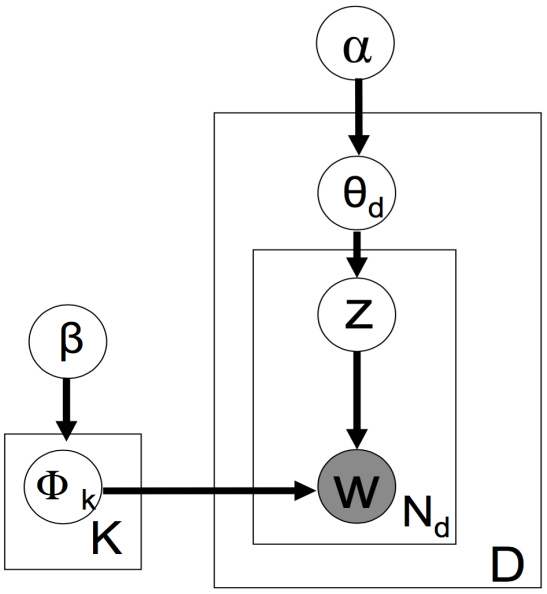
Chapter 4

Simulation Design

4.1 Latent Dirichlet Allocation

After retrieving the news articles from the Internet and cleaning the text, the first step was feed the text into the Dirichlet Process. For this project, Latent Dirichlet Allocation was used to implement the Dirichlet Process clustering[6]. The below flowchart describes the method.

The below flowchart describes the method used.



K - Number of the latent topics (number of clusters).

D - Number of documents.

N_d - Number of words in the document d.

β - Dirichlet prior on ϕ_k is the distribution for words over a topic and also is the distribution generated by Gibbs sampling. ϕ_k - Distribution of words generated from topic k.

α - Dirichlet prior on hidden parameter θ_d

z - hidden variable, which is a latent topic

w - Observation object in this process, which is a word.

The first part was to use the Gibbs sampling method to sample the words to create the clusters and also get the estimated value for two hidden variable of the posterior distribution, which are ϕ_k and θ_d. The generative process has some key features. The parameter $\theta_d \sim Dir(\alpha)$ follows a Dirichlet distribution. Then for the observed words in document d given, the topic parameter was chosen from $z_{dn} \sim Mult(\theta_d)$, and the words follow $w_{dn} \sim Mult(\phi z_{dn})$. Using the given information, the generalized mixture expression for each document exhibits multiple topics:

$$P(\phi_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}|\alpha, \beta) = \prod_{k=1}^{K} P(\phi_k|)\beta \prod_{d=1}^{D}[P(\theta_d\ \alpha) \prod_{n=1}^{N}(P(z_{d,n}|\theta_d)P(w_{d,n}|z_{d,n}, \phi_{1:K}))]$$

The posterior over hidden variables are given as following:

$$P(z, \phi, \theta|w, \alpha, \beta) = \frac{P(z, \phi, \theta, w|\alpha, \beta)}{P(w|\alpha, \beta)}$$

The numerator is traceable (because of conjugacy), but the denominator is not traceable, since it involves summing over all z:

$$P(w|\alpha, \beta) = \int P(\theta|\alpha)(\prod_{n=1}^{N} P(w_n|\theta, \beta))d\theta.$$

Gibbs sampling can help solve the problem for parameter estimation. The desired result from Gibbs sampling is listed as follows:

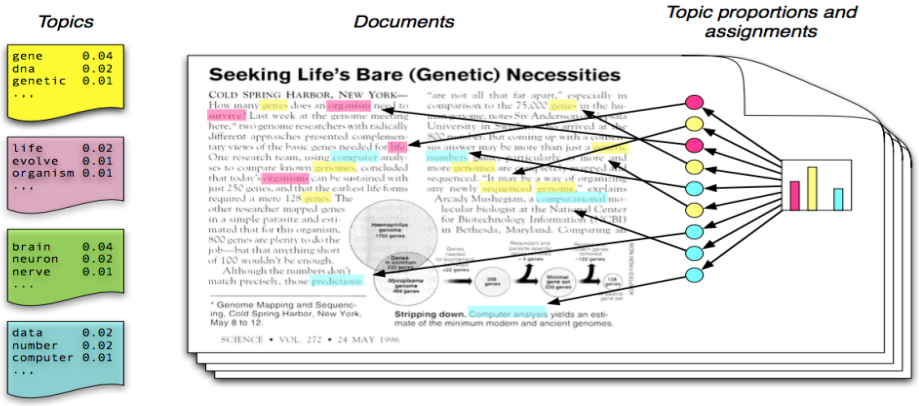
$$\text{probability of term } v \text{ in topic } k: \hat{\phi_{kv}} = E[\phi_{kv}|w_{1:D,1:N}]$$

$$\text{term probabilities in topic: } \hat{\phi_{kv}} = \frac{\lambda_{kv}}{\sum_{v'} \lambda_{kv'}}$$

$$\text{term-score (like tf-idf): } \hat{\phi_{kv}} log(\frac{\hat{\phi_{kv}}}{(\prod_{k'} \hat{\phi_{k'v}})^{1/K}})$$

$$\text{topic proportions: } \hat{\theta_{dk}} = \frac{\hat{\gamma_{dk}}}{\sum_{k'} \gamma_{dk'}}$$

Thus the process for topic clustering can be described by the following graph:



After the results of clustering and parameter estimation, the temporal model can be fitted to get the value for the intensity measure of each topic.

Here the definition of intensity and the intensity function of a Poisson Process are used, to use the density distribution of the posterior topic to fit an intensity process.

4.2 Inhomogeneous Poisson point process

If a Poisson point process has a constant parameter, such as λ, then it is called a homogeneous or stationary Poisson point process. The parameter, called rate or intensity, is related to the expected (or average) number of Poisson points existing in some bounded region. In fact, the parameter λ can be interpreted as the average number of points per some unit of extent.
So by Law of large numbers:

$$\lim_{t\to\infty} \frac{N(t)}{t} = \lambda$$

This is the constant intensity rate. For a Inhomogeneous Poisson, the conditional intensity function of a point process on the real half-line is a function $\lambda(t|Ht)$ defined as:

$$\lambda(t|Ht) = \lim_{\delta t\to 0} \frac{1}{\delta t} P(\text{one event occurs in the time interval } (t, t+\delta t)|Ht)$$
$$\Lambda(s,u) = \int_s^u \lambda(t|Ht)dt$$

Since there is a density distribution of P, and the time interval can be fixed at the beginning when the document arrives, so the intensity process can be calculated to fit the mode. An Inhomogeneous Poisson point process is actually a case for Hawkes process. Also by Gibbs sampling, the kernel for each cluster can be recovered to verify the self-decaying feature for the intensity. Moreover, integrated graph can be drawn to present the cluster partition, which is how the kernel triggering the next cluster is an interaction between different point process.

After the content and temporal information analysis, further information about the model was desired, so R's shiny library was used to create a 3D web application to show the interaction between topics and words and how the topic model was influenced by the concentration parameter chosen at the very beginning.

Chapter 5

Experimental Results and Analysis

5.1 Content Analysis

The first part of the experimental results analysis was the content analysis for each cluster. Each cluster represents a topic which contain a meaningful story. It was discovered that the data used most frequently clustered into nine clusters while using the randomized Gibbs sampling algorithm. The experiment was repeated many times, and nine clusters seemed to provide the most optimal results for continued content and temporal analysis.

The content of some of the clusters is shown, and some of the topics of the clusters were selected for deeper analysis, such as the intensity process and triggering kernel analysis.

When implementing Gibbs Sampling to do the clustering, it is not a constantly stable process based on the nature of Markov Chain, so we chose the most frequently occurring number of clusters.

Figure 5.1: Topic1

Figure 5.2: Topic2

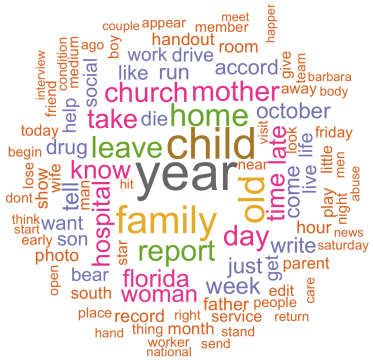


Figure 5.3: Topic3

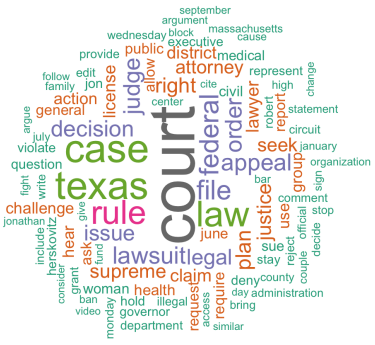


Figure 5.4: Topic4

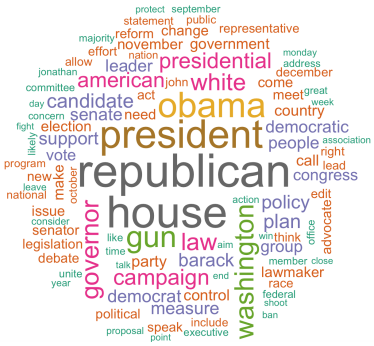


Figure 5.5: Topic5

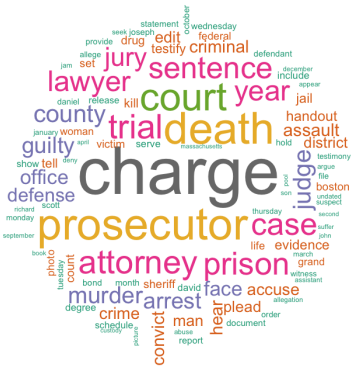


Figure 5.6: Topic6

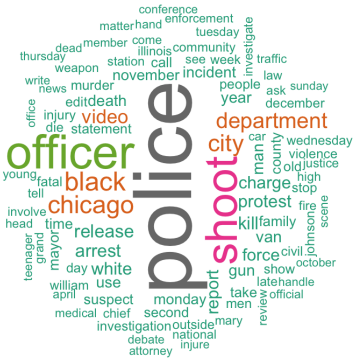


Figure 5.7: Topic7

Figure 5.8: Topic8

Figure 5.9: Topic9

Words in certain topics are not simply gathered together, and the combination can tell a meaningful story. For example, for topic 1, the most frequent words are California, student, school, and so on. We can tell this topic relates issues relating to college campuses. Topic 4 included some things related to court and legal issues. Some topics share some of the frequent words, which will explained later by a 3D visualization web application.

5.2 Temporal Analysis: Intensity Process

Only some of the temporal analysis result is shown by selecting topics 3, 4 and 9 to present the temporal results. The first part of temporal analysis is the intensity process for each topic, which is a realization of the Hawkes Process. The intensity function and intensity measure were simulated based on the Law of Large Numbers for the intensity at a single interval for a point process. Below are the Intensity Processes for the three topics:

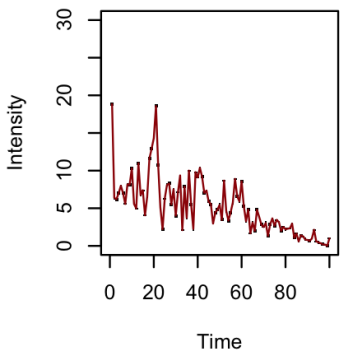


Figure 5.10: Topic3

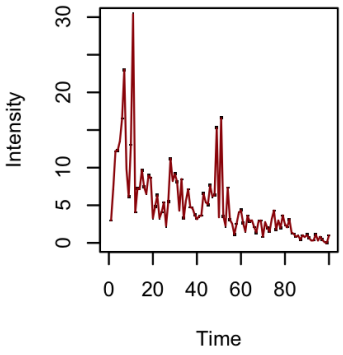


Figure 5.11: Topic4

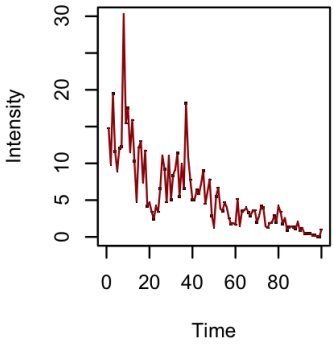


Figure 5.12: Topic9

The time length is from 2015 August to 2016 January. It can be seen that all three topics are self-decaying. They eventually will have a decreasing trend for the intensity. The intensity represents the expected value of the number of topics in the thinned time interval. It is also possible to treat the temporal intensity as a group of time series data. This intensity process describes the features of the topic. Because of the nature of the popular topic, it will only remain popular for a certain period of time. From the content analysis, it is known what the most popular stories are, and from the intensity process, the intensity rate for each popular topic can be compared.

The features for the Inhomogeneous Poisson Process are self-decaying with mutual interaction mainly because the process follows the distribution with the triggering kernel. The triggering kernel is used to verity such mutual interaction. The single kernel for each topic is described by the intensity for the cluster in Gibbs Sampling. This intensity is different from the previous topic intensity. The topic intensity follows the distribution of topics over the documents, which is a hidden posterior distribution for Dirichlet Process. The kernel intensity is the intensity for each distribution within the cluster, which is the distribution of words over the topic. It is an independent distribution. However, the distribution of topics is conditional on the baseline Poisson Process and Dirichlet Prior. The cluster partition shows the mutual relation between the kernels because the words in each topic are not exclusive. Some topics will share common words. The order of the clusters is decided by the Markov Chain of Gibbs Sampling, where when one cluster intensity decays, the Gibbs Sampling collapses at the same time until no more clusters can satisfy the density distribution. The Gibbs Sampling will eventually stop. The last scaled intensity describes the shape of the posterior multi-variable distribution.

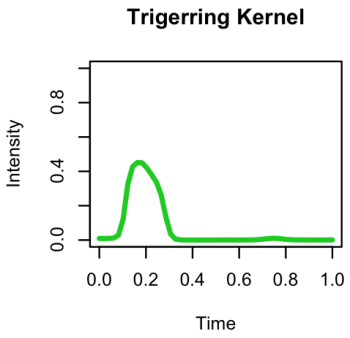


Figure 5.13: Topic3

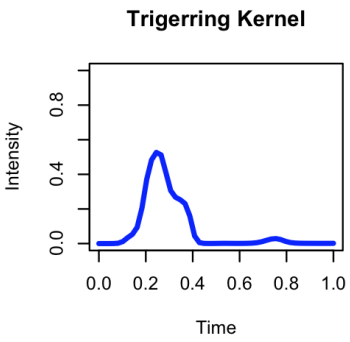


Figure 5.14: Topic4

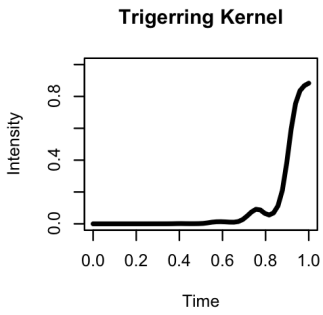


Figure 5.15: Topic9

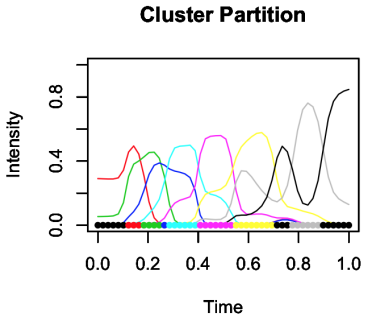


Figure 5.16: Cluster Partition

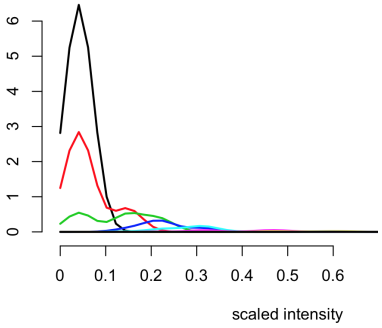


Figure 5.17: Scaled Intensity

5.3 Performance Analysis

The model was fit, and the parameters were estimated, however the performance of the model needed to be analyzed. The value of the maximized log likelihood function was used to do the evaluation of Gibbs sampling by 10-fold Cross Validation.

With the increase number of topics, the value of log likelihood is less like to be optimal, we replicated this cross valudation process for several times then find that the reasonable range for the number of topics should around 8-15. Also we list the estimated parameter for the posterior distribution of each cluster.

Where α is the common prior they share, and the θ_d is the distribution parameter for each topic.We have $\alpha < \theta_d$, which proves our model is reasonable.

The plot for Gibbs sampling estimates the parameter of the posterior distribution by maximizing the log likelihood, so the optimal value appears about 8-15 times as introduced before, in this case, the optimal number of topics is 11, which is very close to the most frequent number 9 tested during the Dirichlet Process Mixed model. Also the detail value of the log likelihood function was listed to show

Table 5.1: Estimated Parameter

Cluster Number	α	θ_d
1	0.0015	0.108
2	0.0015	0.952
3	0.0015	0.143
4	0.0015	0.143
5	0.0015	0.099
6	0.0015	0.123
7	0.0015	0.109
8	0.0014	0.112
9	0.0015	0.09

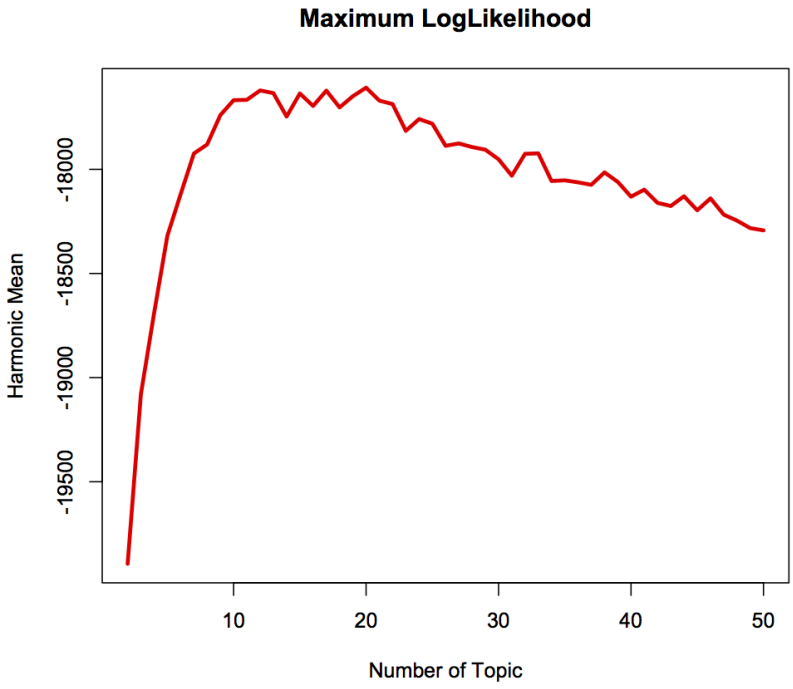


Figure 5.18: Log Likelihood

Table 5.2: Maximun Log Likelihood

Number of Topic	2	5	8	11	14	17	20
Log Likelihood	-19893.92	-18318.99	-17666.29	-17880.74	7746.53	-17622.11	-17607.99
Number of Topic	23	25	30	35	40	50	
Log Likelihood	-17815.19	-17780.84	-17951.69	-18052.97	-18131.27	-18293.02	

the trend for the value of the likelihood function. With the number of topic increasing, the value also decreasing, thus the model will become less efficiency and less accuracy.

5.4 3D Visualization

A tool known as Shiny for R was used to create a visualization web application which shows a deeper analysis for the relations between topics and the influences of changing parameters for the Dirichlet process. The left side of the pictures show the similarity between clusters. Overlapping clusters mean that they share some of the same content, and the distance between clusters indicates the dissimilarity between two clusters. For example, two clusters related to politics may overlap, but one cluster related to sports and another cluster related to politics may be very far apart. Changing the lambda parameter, which is the concentration parameter, changes the ordering of the most frequent words based density. This is the link for our web application to view the 3D Visualization of each topic, with the chage of Dirichlet concentration parameter.

3D Visualization Link:

http://bl.ocks.org/joannachenyijun/raw/be3f870122a7b693442411007796b963/#topic=9&lambda=undefined&term=

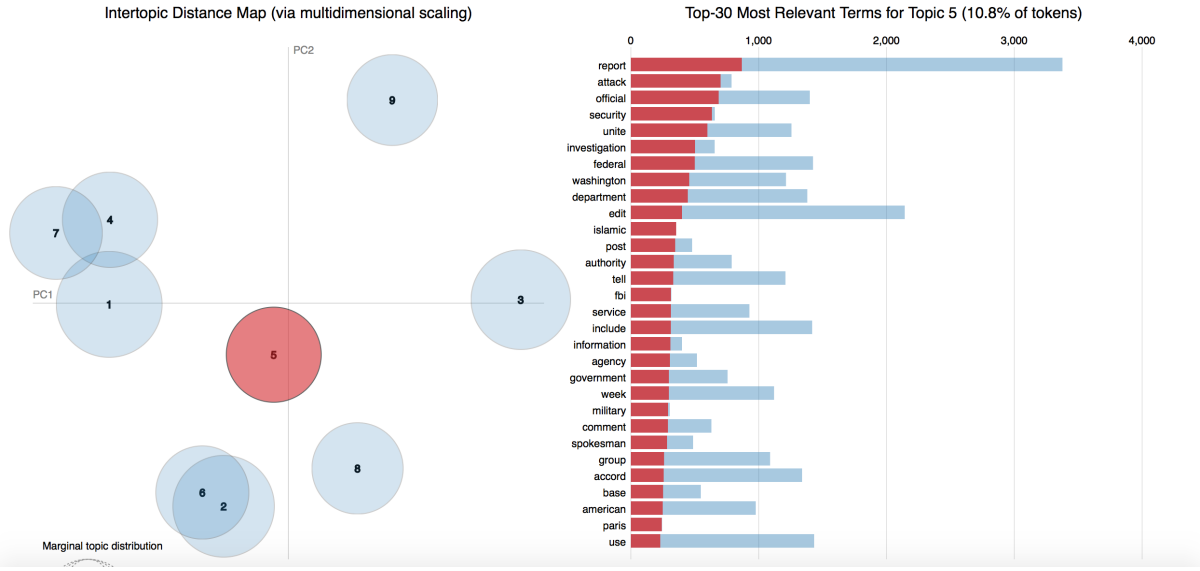


Figure 5.19: Topic 5, lamba=1

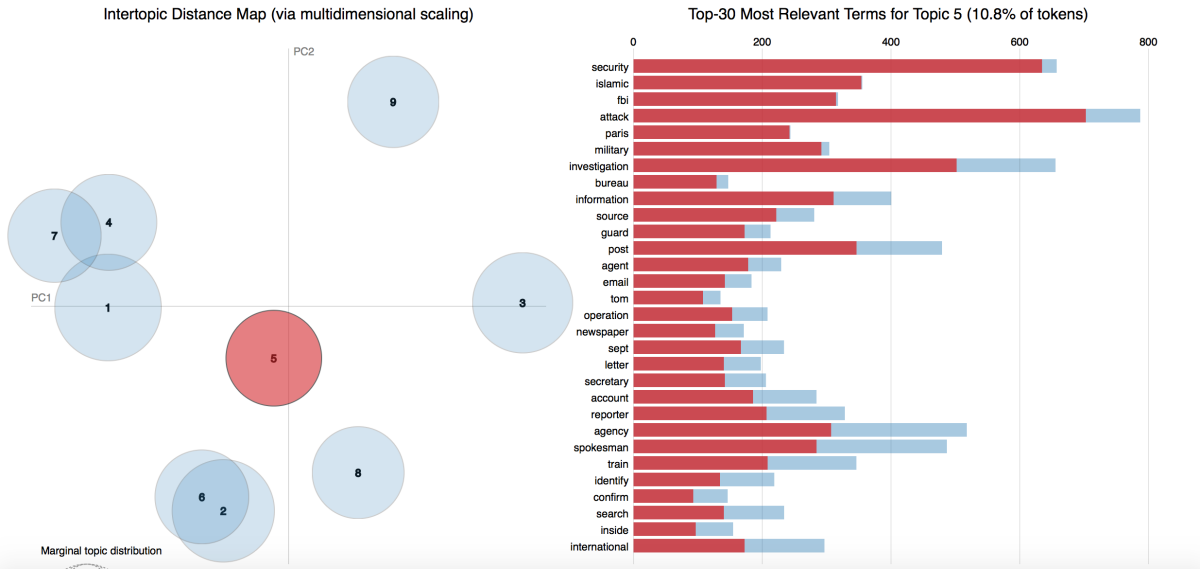


Figure 5.20: Topic 5, lamba=0.1

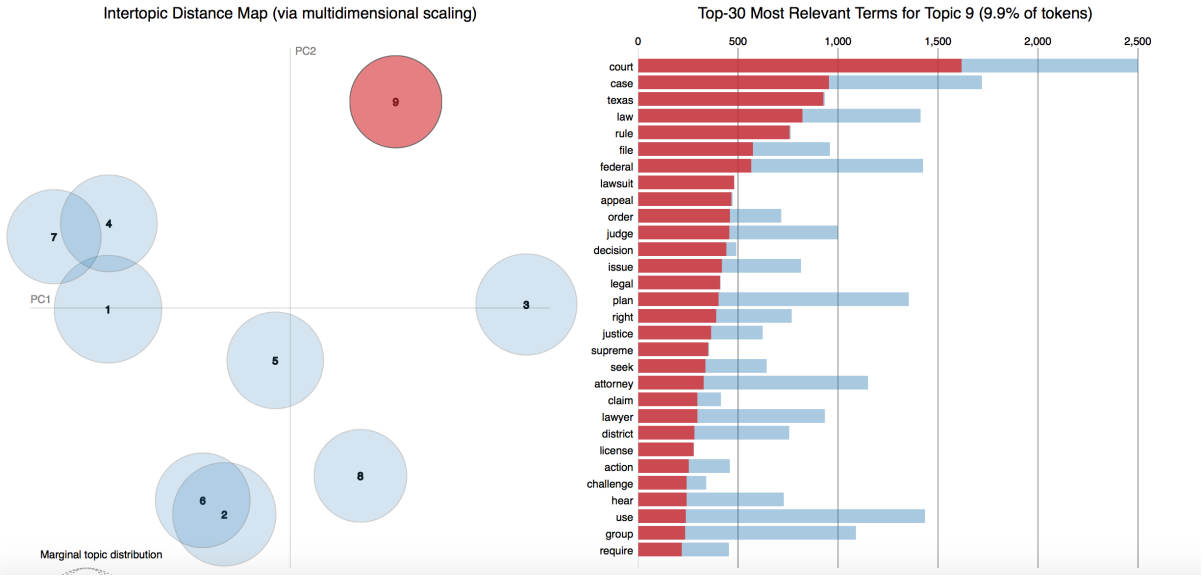


Figure 5.21: Topic 9, lamba=1

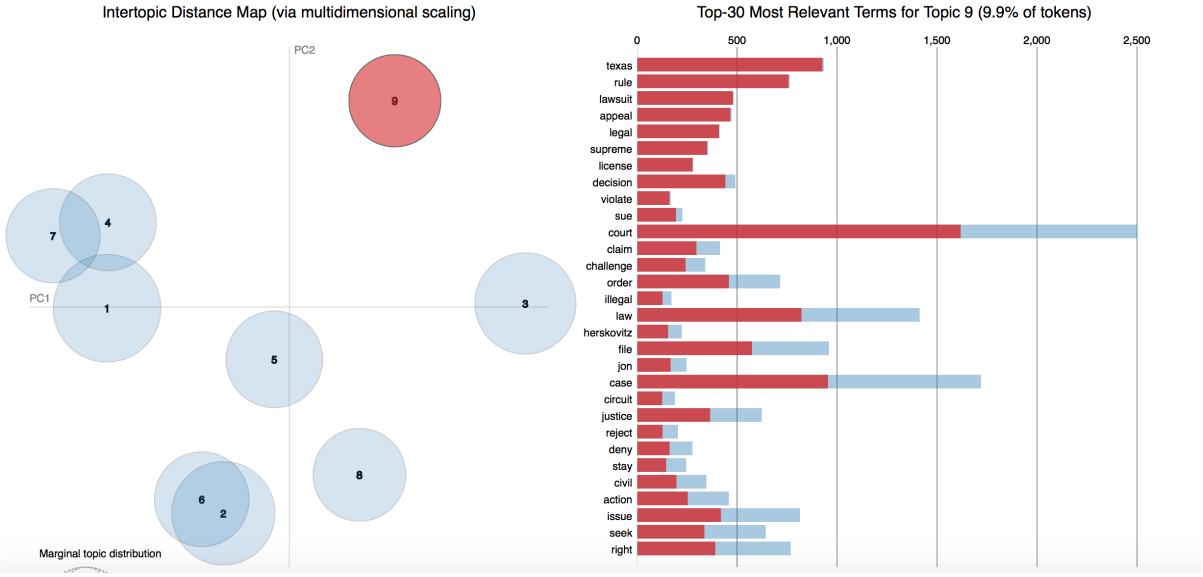


Figure 5.22: Topic 9, lamba=0.1

Chapter 6

Discussion

This project, which was based on Dirichlet-Hawkes Processes with Applications to Clustering Continuous-Time Document Streams, proved to be successful. This project required a significant amount of work in order to get it to successfully cluster a collection of news articles using the Dirichlet-Hawkes Process. The results were as expected, and it will be interesting to see what other applications the Dirichlet-Hawkes Process will have.

There are numerous extensions that can come from this project. While the project accomplished all of the goals that were laid out for the scope of this project, there are many other interesting applications that can stem from this project. It would be interesting to apply the algorithms used to a larger data set in order to look for greater trends. Different parameters within the algorithm can be tweaked to attempt to obtain improved results. It is believed that clustering longer news articles would provide even better clusters, but it would be more interesting to see the limitations of the clustering algorithm when applied to shorter text fragments. The methods used during this project can be applied to other types of text in addition to solely news articles. The applicability of the algorithms can be tested on other data types, such as numeric data, in addition to text. Another extension for this project would be to incorporate the algorithm into other machine learning algorithms.

Bibliography

[1] Nan Du,Mehrdad Farajtabar, Amr Ahmed,Alexander J. Smola,Le Song(2015) Dirichlet-Hawkes Processes with Applications to Clustering Continuous-Time Document Streams *ACM*. KDD '15, August 10-13, 2015, Sydney, NSW, Australia.

[2] J. F. C. Kingman. Poisson processes *Oxford university press* 1992

[3] Charles Blundell,Katherine A. Heller,Jeffrey M. Beck(2012)Modelling Reciprocating Relationships with Hawkes Processes *Duke University*

[4] John Carlsson, Mao-Ching Foo, Hui-Huang Lee, Howard Shek (2007) High Frequency Trade Prediction with Bivariate Hawkes Process. *Stanford University*, 60(3):770-783.

[5] KAI KOPPERSCHMIDT and WINFRIED STUTE(2011) The Statistical Analysis of Self-Exciting Point Processes . *Statistica Sinica*

[6] Bettina Grun Kurt Hornik (2011) topicmodels: An R Package for Fitting Topic Models *Journal of Statistical Software*, May 2, 2010

[7] Carl Edward Rasmussen (2010) Dirichlet Process Gaussian Mixture Models: Choice of the Base Distribution *JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY*, May , 2010