Semantic Change of Words Over Time

Akhila Kotapati[1] , Vamsi Varma Kunaparaju[2] , Vamsi Krishna Reddicherla[3]

[1]Computer Science
George Mason
University,USA
akotapat@gmu.edu

[2]Data Analytics Engineering
George Mason
University,USA
vkunapar@gmu.edu

[3]Data Analytics Engineering
George Mason
University,USA
vreddich@gmu.edu

ABSTRACT

In this paper, we have implemented several models to analyze statistically significant linguistic shifts. We have analyzed how the co-occurrence changes over time, how the state of the entity is influenced by this change and how this corresponds to the events occurring in same period.

First, we have implemented word embedding models such as Word2Vec and GloVe. Through this approach, we compute the vector representation of words, from the given corpus, for each time period. Then, vector spaces are projected on to one unified coordinate system. Second, we have implemented a generative probabilistic model to find relationships among data, text documents. This is executed using a Topic modeling method called, Latent Dirichlet Allocation.

Through these analysis we would like to show that the Topic model infer word embedding trajectories are more interpretable and lead to higher predictive likelihoods than competing static models that are trained separately on time slices.

1 INTRODUCTION

Semantic change refers to change in the meaning of the word to a point that current meaning of the word is fundamentally different from the original usage. Meanings of the words are subject to constant change with time. For example, in earlier days the word plane was used to refer the geometric figure whereas it is now being used to refer to an aircraft. There are also a certain set of words which acquire additional meanings and are used to refer to other entities over time. For example, the word cell was used to refer to the smallest unit in our body, whereas after the after the advent of the mobile phones, people are using the word cell to refer to their mobile phones. Capturing change in the meaning of the words can help linguists, historians and the general public. It can prove to be game-changer when it comes to natural language processing tasks, particularly the ones dealing with news corpora. It can also help in detecting trends in

the dataset. In this paper, we are trying to model semantic change of words and choose a frame for analyzing it. We have used the popular neural word embedding techniques Word2Vec, GloVe and also used the Topics over time model to track semantic change of words over time. We have used topics over time as we believe that that many of the topics in huge datasets follow dynamic co-occurence pattern. Topics merge to form new topics and the frequency of topics rises and falls with time. We believe that analyzing documents along with their timestamps will lead us to some interesting insights and exciting results. Apart from using these techniques, we have also experimented with set of pre-processing techniques and post pruning techniques to find the best set of pre-processing techniques for topic modeling. We present our results of models using the State of Union Addresses dataset.

2 Related Work

A relative work has been done using change in the topic that surrounds an entity. They have defined and indicated the change in an entity by referencing the change in this surrounding topic. This helped them to find the direct key words which describe the change from the topics identified.

Their implementations included taking advantage of the Tf-Idf vectors, the vectors that are weighted and generated similar to one another based on the relative word's frequency of occurrence. They have utilized Google's 5-grams dataset for the quantitative analysis of this procedure. The way they have modified the Tf-Idf counts based on the occurrence of the words over time has inspired us to utilize word embeddings instead and capture the semantic change by sub dividing the dataset into equally divided time frames. We examine the change in the context of a word by observing the change in the relativeness or similarity of this word with other words over time.

3 Problem Statement

To appraise the linguistic change or shift of word meaning and its usage across time. This is shown using different models. For the word embedding models, let C be a given corpus that is created over a given time span T. We divide the model into n snapshots C_t, for each time period t. Then, for each model, a time series T(w) is constructed for each word w $\in$ V. On these time series constructed, we assess the shift that occurred to the word in its meaning and usage. In case of Topic Modeling, we use popular Topics-over-Time model to observe the evolution of topics over time.

4 Approach

4.1 Word2Vec

A Word Embedding format generally tries to map a word using a dictionary to a vector.

Word2Vec is a two-layer neural network that learns distributed representation of words and produces a word embedding model. It groups the vectors of similar words together in the vector space. Word2Vec contains two models, Continuous Bag of Words (CBOW) and Skip-gram

Model. In Continuous Bag of Words, given a word or a group of words, the model predicts the probability of a word. Here, the group of words are the context of the resultant word. In contrast, Skip-gram model takes a specific word in a corpus as its input and produces a group of words that are similar in context to the given word. For our analysis, we have implemented Word2Vec using skip-gram model.

4.1.1 SKIP-GRAM MODEL

Skip-gram model maximizes the classification of a word based on another word, that surround it within a defined window. The objective of the Skip-gram model is to maximize the average log probability of any context word, given a sequence of training words $w_1, w_2, w_3, ..., w_T$,

$$1/T \sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} \mid w_t)$$

where c is the size of the context window. At the expense of training time, we can have a higher accuracy of the model by choosing larger c.

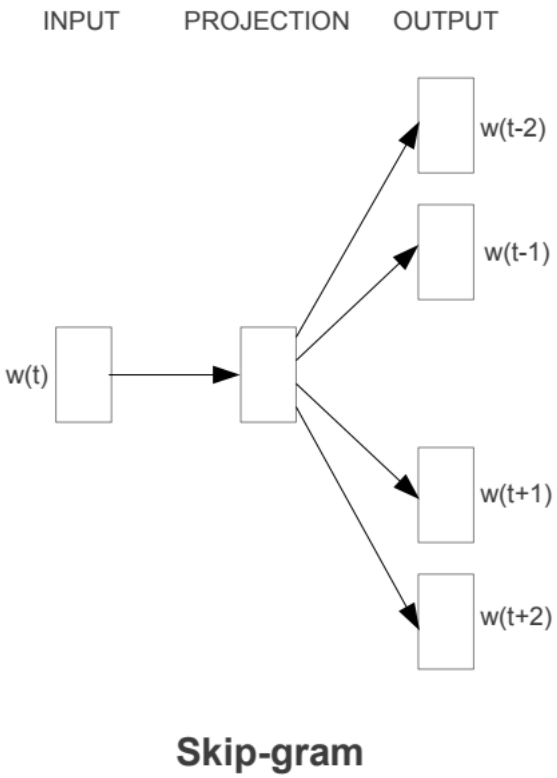
The probability can be determined using the softmax function.

$$p(w_O|w_I) = \frac{\exp\left({v'_{w_O}}^{\top} v_{w_I}\right)}{\sum_{w=1}^{W} \exp\left({v'_w}^{\top} v_{w_I}\right)}$$

where W is the number of words in the vocabulary and, vw and $v\prime w$ are the input and output vector representations of w. Due to larger cost computation, this formula is impractical. The training complexity of this architecture is proportional to

$$Q = C \times (D + D \times \log_2(V)), \quad (1)$$

where C is the maximum distance of the words. Thus, if we choose C = 5, for each training word we will select randomly a number R in range < 1; C >, and then use R words from history R words from the future of the current word as correct labels.



For each word in a training instance, probability prediction and their prediction error need to be calculated. Computing for all the words for very training instance is expensive.

In order to attenuate the cost of computing, we can limit the number of output vectors that must be updated per training instance. Two major strategies that was proposed are Hierarchial Softmax and Negative Sampling. Both the optimize only the computation of the updates

for output vectors.

4.1.2 Skip-gram model with Negative Sampling

During training, the weights of all the words are adjusted slightly to predict the training sample accurately. Depending on the size of the word vocabulary, all the weights would be slightly updated for every one of the training samples. Rather than modifying the weights of all, Negative Sampling ensures that each training sample modifies only a small percentage of the weights. In Negative Sampling, we only update the weights of a small number of negative words and, also to that of a positive word. Negative words are chosen based on the frequency of the words. Higher the frequency, more likely to be selected as negative samples.

4.2 GloVe

GloVe stands for Global Vectors, it is a word embedding model that takes advantage of the window based word co-occurrence counts and matrix factorization. The GloVe model argues that the ratio of the probabilities of the word co-occurrence counts tend to yield more meaning than the co-occurrence counts themselves. While word count based models such as LSA, HAL and PCA have faster training rates and higher efficiency in the usage of statistics, they give disproportionate importance to the words with large counts.

On the other hand, Co-occurrence based models such as Skip-gram/CBOW have improved performance and can capture complex patterns that are beyond the word similarities. But, they tend to scale with the corpus size and have inefficient usage of statistics. GloVe is proved to carry the best from both of these worlds. It has faster training rates and is scalable even for huge corpora. Also, the fact that GloVe tends to work quite well with smaller datasets made it even more easy for us in picking this model.

Glove Vectors also have an interesting property of capturing the semantic regularities along with the syntactic ones.

$$king - man + woman \approx queen$$
$$brought - bring + \text{seek} \approx sought$$

The vectors have the efficiency of capturing the meaning of the word based on the context. So, when the word king is given, they tend to capture both the male aspect and the position of the term. Therefore, when man is taken away from the context and woman is added, it results in the word queen as it has similar context but with female aspect. Similarly, the word vector that are created using the GloVe model has certain characteristics as they are described in fig1. Words with similar contextual meaning are distanced with a certain Euclidean distance count. Words with superlative relationship such as great, greater and greatest or placed in a sequential order giving them a position accordingly. These relationships in the distances allow us to extract useful relative words through vector differences i.e. the vector differences between the word pairs king-queen and brother-sister must roughly yield the same value.

It all starts with building a word co-occurrence matrix X, i.e. how many times a

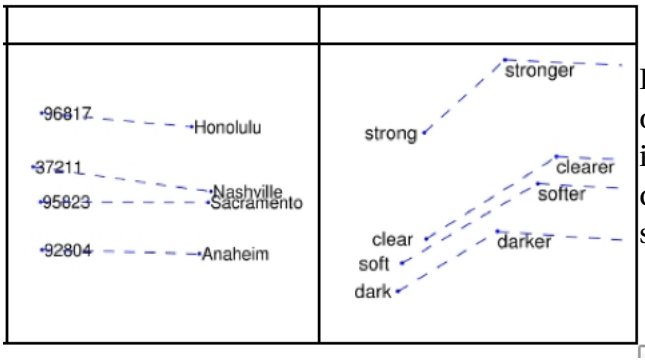


Figure 4.1: GloVe Geometry

word lets say “i” appears in the context of another word “j”. One iteration is run through the whole corpus in order to build this matrix. The cell $X_{i,j}$ in this matrix determines the strength of frequency of this co-occurrence. Rest of the model is trained solely based on this matrix. The aim of the model is to minimise the objective function J,

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

where, w_i -> vector for word i.

w_j -> separate context vector for word j,

b_i, b_j -> scalar bias terms for words i and j.

Here, f is a weighted function that helps in preventing the most common word pairs from skewing the objective function. If the common word pairs occur with very high frequency the function just returns 1 or else, it returns a weight in the range of 0-1.

4.3 Topics Over Time

Latent Dirchlet Allocation is popular method of Topic Modelling. Latent Dirchlet Allocation is a generative probabilistic model that relates documents through latent topics. The core assumptions made by LDA are :

- Each document exhibits multiple topics in different proportion.
- Each topic is a distribution over words.
- Each word is drawn from one of those topics.

The only input we give to the model are documents. The other structures (topic distributions, per-document-per word topic assignment) are latent variables which are inferred from the documents. The hidden variables are inferred by computing the posterior distribution, conditional distribution given the documents. Let K be a specific number of topics, V the size of the vocabulary, α a positive K-vector, and η a scalar. We let Dir V (α) denote a V-dimensional Dirichlet with vector parameter Îś and Dir K(η) denote a K dimensional symmetric Dirichlet with scalar parameter η

1. For each topic,
    - Draw a distribution over words B˜Div V (η)
2. For each document,
    - Draw a vector of topic proportions
    - Φ K˜Dir (α)
3. For each word,

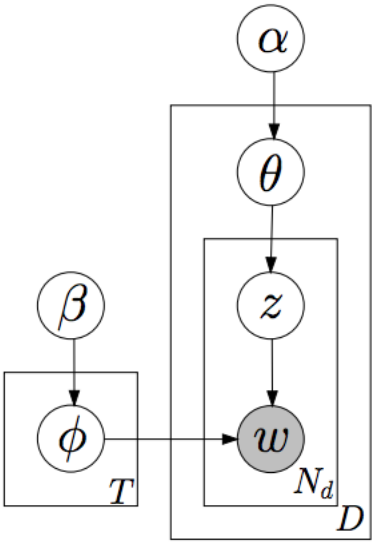


Figure 4.2: Basic LDA model

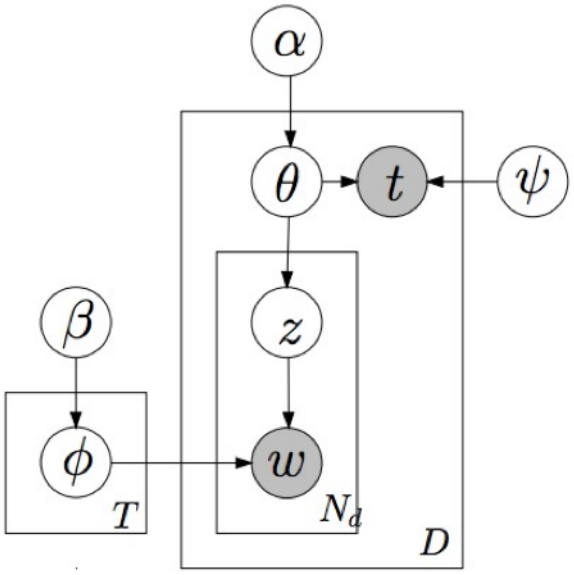


Figure 4.3: TOT model

- Draw a topic assignment
  - Z_d, n˜Mult (θ), Z_d„n ∈ 1, 2, ..., K
- Draw a word
  - W_d,n˜Mult (β Z_d„n), W_d„n ∈ 1, 2, ..., V

Latent Dirchlet Allocation provides a joint distribution of over the observed (documents) random variables and latent variables. The topic composition of the documents emerges from the corresponding posterior distribution of the hidden variables given the topics D.

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

Topics over time is a generative [3] probabilistic model with time stamped documents. The topic discovery is not only influenced by word occurrences but also by the temporal information of the documents. The TOT parameters for TOT model are as follows :

- θd|α âĹij Dirichlet(α)
- Φz|β âĹij Dirichlet(β)
- zdi|θd âĹij Multinomial(θd)
- wdi|Φzdi âĹij Multinomial(Φzdi)
- tdi|Ψzdi âĹij Beta(Ψzdi).

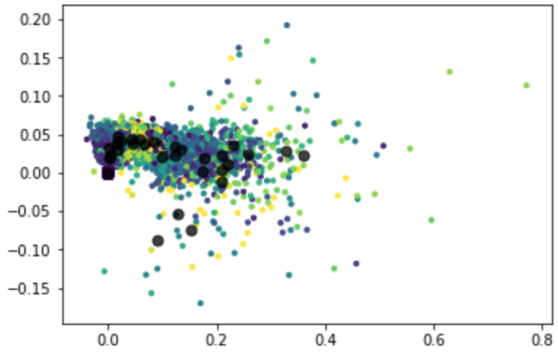
A timestamp associated with a document is generated by importance sampling, from a mixture of per-topic beta distributions over time. This allocation is ultimately dependent on the beta distributions. The topics generated will have both words and the timestamps.

5 EXPERIMENTS

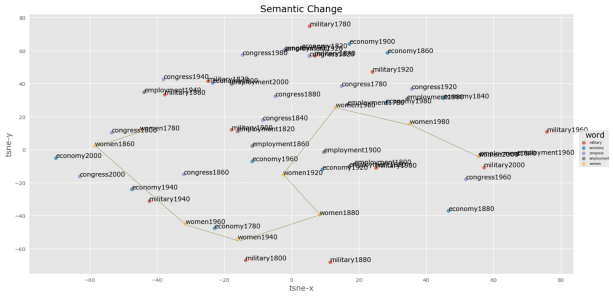
5.1 WORD2VEC

For the experiments, we have divided the data into a period of twenty years. Before training each individual time series, we have applied few pre-processing techniques. First, we have converted the entire text into lower case. Second, techniques like punctuation mark removal, numeric removal and special character removal are implemented, which is not very helpful in constructing a model. Now, we need to break the text down into individual words, so that, even the word window consists of individual words. In order to obtain this, we have applied tokenization technique on the data. After obtaining the tokenized words, we started training the models individually for all the time series. The resultant models are the vector representations of words for each time period.

We have also run the Word2Vec model on the whole corpus and implemented k-means clustering. Through this we have observed that most of the related words are clustered together, which implies that majority of the words held their meaning since decades and haven't changed even in their context.

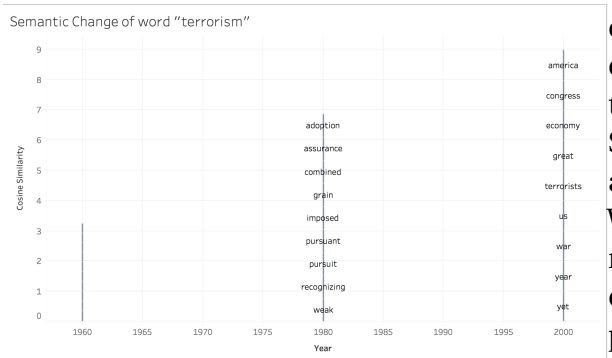


Now, to observe the words that have changed during time, we have reduced the dimensionality of word vectors. Here, we choose the size of dimensions to be 100. So, to reduce the 100 dimensions vectors into a 2-dimensional vector, we have used t-stochastic neural embedding (t-sne) on the models. Vectors in each model are reduced to a 2-dimensional vectors and are appended with their time periods. Among these, we randomly choose few words which we feel are used differently at different period. These words along with their time period are plotted with their t-sne coordinates. Here, we can observe that words that are used together in a particular time period have ended up close to each other.

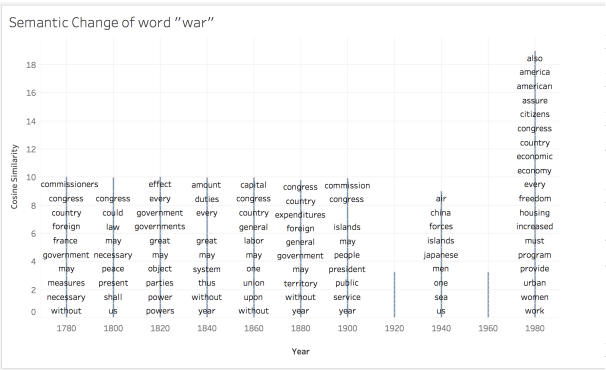


As one can observe, economy and military in 1940s have ended up together, giving the context that during world-war 2, most of the economy was spent on military. Similarly, there are few other co-occurrences like congress and economy in 2000s.

Finally, we have also analyzed how the words change during time. To observe this, we have taken word-similarity for a word over the time period and noticed that the words similar to that particular word are different for different time periods. This gives the intuition that those words are used in different context in that periods. Few such words are war and terrorism.



As, seen in the figures there are different words at different time stamps corresponding to that particular word. Words at a certain period of time gives the scenario of incidences that one can relate. For example, the words associated to "war" in 1940s relate to the war that happened in that decade by having the similarity with words like Japanese, Air, Forces, US.



5.2 GLOVE

5.2.1 PREPROCESSING

Initially the dataset consisted of all the speeches of different presidents of America over the years from 1790 all the way through 2017. Initial preprocessing steps included cleaning the text of useless punctuations and stripping of unnecessary white spaces. We decided to split this dataset in to sub-datasets of decades i.e. each dataset consisted of speeches that were recorded in the time span of 10 years. So, first dataset had speeches from 1790-1800 and the second one had 1800-1810 and so on. We have decided to divide the dataset for glove models with 10-year threshold instead of 20 because we wanted to test the ability of GloVe to perform well on small datasets.

5.2.2 IMPLEMENTATION

The Idea is to implement and train the model(GloVe) on each of these sub-divided datasets and form word vectors using these as individual corpora. Once the word vectors are created, we decided to calculate and observe the most similar words for some words-of-interest such as "government", "people", "military", "community" etc. After observing the most similar words, we wanted to calculate the similarity in between certain word pairs using a similarity measure. Since, we had about 20 different glove models in our hands each trained with a different corpus that has been sub-divided from the dataset, the similarity measures tend to change. Since the difference between the corpora is that they have been captured at different and almost equally divided time frames, the change in this similarity can be referred to as the "shift in similarity over time" or "semantic shift over time".

Now, let us go through the implementation of the above described thoughts step by step. First, we have implemented the glove model on the whole dataset. This step although needed some time due to low computational power has been essential to understand the working of the

model and its implementation algorithm. Once this implementation is done, we have used a toy implementation of GloVe that allows us to find the top most similar words in the corpus. This toy implementation had a default pre-trained vector set. This implementation had been modified to take in any type of corpus file and produce word vectors from it. Using this implementation, we have successfully built about 24 models each trained with datasets of different time frames as previously stated. Now, with a dataset like state of speech union, it is highly unlikely to observe the shift in the meaning of a word over time. All that the dataset consists of is the speeches from the presidents and the terms that are used in the speeches are carefully brushed in a way that the audience would be able to understand and relate to. So, let us see how the most similar words (capturing relativeness instead of meaning) for the word "government" changed over time.

1850	1900	2000
peace	country	business
laws	war	politics
state	time	spend
future	state	source
people	damage	supremacy

Fig2: most similar words with "government"

During the time of peace (1800) we can observe that the words are completely related to the same while, during the times of war the similar list of words change. It is highly surprising how the word "supremacy" ends up in the recent years. Now, that we have tried to observe the change in relativeness, let us try and find out the shift in similarity measure. Cosine similarity is used to calculate the distances between some word pairs. The word pairs chosen (obviously because they demonstrated great shift) are "people" - "government" and "economy" - "congress". Below is a plot (Fig-3) of the shift in similarity measure between these word pairs calculated in python, generated in Tableau.

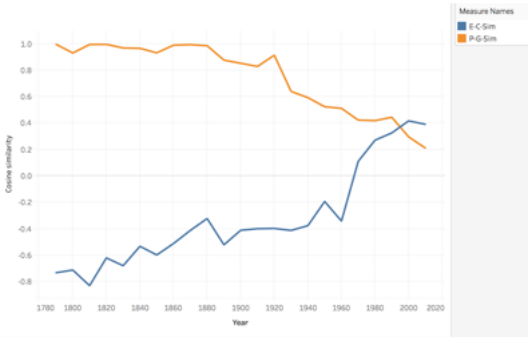


Fig-3: change in similarity over time

The line "E-C-Sim" represents the similarity between the words "economy and congress" which, as can be observed, increases significantly over the period of time. This represents that the words are coming closer in terms of similarity. They seem to gain high similarity starting from 1940, all the way to the present. The other line "P-G-Sim" represents the similarity between the words "people" and "goverment". This line keeps decreasing indicating that the words are going farther apart as time passes by. Now, the harder point to understand is whether these similarities are really decreasing or increasing in the real-world context or is it as because the frequencies of the words i.e. the no. of time these words are being used is

decreasing. As time passes by it is hard for a word to maintain a constant frequency rate. In order to study this another plot has been generated in Tableau representing the frequency of the word over time.

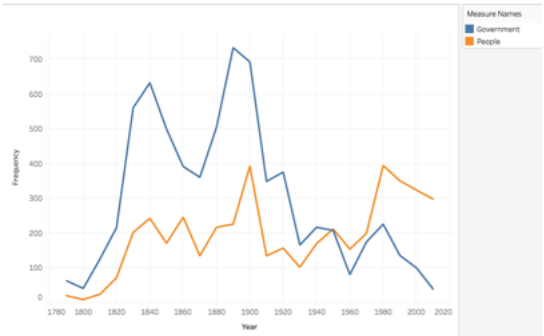


Fig-4: change in frequency over time.

As can be observed from the above plot, both these words used to be rare in the early 1800's whereas they started picking rapid pace since then and has significantly increased over the years up until 1940. From the beginning of 1940 the frequency of the word government started depreciating while that of people has increased even more. This might possibly be because of the way the audience to the speeches have changed over the year. In the beginning, there was only a mall of audience and the government officials listening to the speech and so it has been addressed to them as the target. But, as time passed by and technology has improved, people have started listening to it in radios and later in the television. So, the shift in the target audience might have caused the change in these word frequencies. But, the frequency seems to closely follow one another representing some authentication to the similarity plot.

5.2.3 TSNE FOR GLOVE VISUALS

In order to visualize these word embeddings, t-distributed stochastic neighbour embedding (TSNE) has been used. TSNE is essentially a technique that is used to visualise high dimensional data in the means of lesser dimensions. In our dataset, the word vectors produced has 50 dimensions per each word which makes it impossible to visualise. By applying this method, we brought down those 50 dimensions to two dimensions and have visualised the vectors.

Initial TSNE visualizations included the whole dataset without sub-division. The reason for this is to observe the performance of GloVe model on the dataset and see if it is capturing the similar words. Below is an example plotting of word vectors thus produced.

Fig-5: word embedding through TSNE

The above plot shows how the words related to the field of education have all been grouped

together. This shows us the power of the word vectors created through GloVe model. Now let us see the shift in the words as they are visualized in different time frames. The procedure for this is the same as to that of the procedure that has been done above. The plots have been created for different datasets from different time frames and the positions have been observed.

fig-6: word embedding (1790-1800)

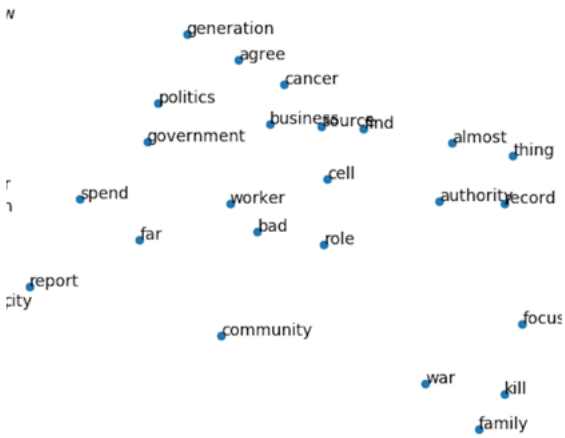


Fig-7: word embedding (2000-2010)

After, observing the above two plots (Fig-6 and Fig-7) it is quite evident how the word business came in to context with the word government as time passed by from 1800's to 2000's. Does this relate to more activities of lobbying that are happening with in the government? But, people in the government stay highly cautious and polish the speeches as much they can to hide such facts.

5.3 LDA

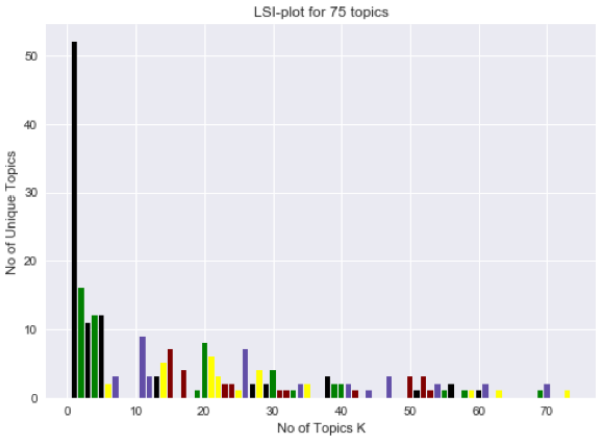
5.3.1 Topics Over Time

We have considered speech from each year as a separate document. We started off the by cleaning the text data using regex. We converted the documents into lower case and removed the punctuation marks. We have also removed certain common words which were making all the documents look similar. Subsequently, text preprocessing techniques such tokenization, stop words removal, Lemmatization were applied.

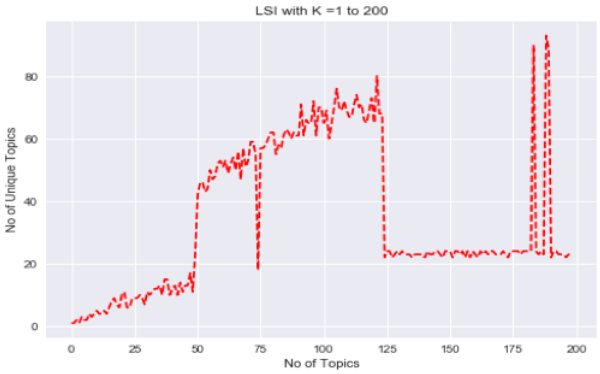
We chose lemmatization over stemming because stemming being too aggressive replaced most of the words with its roots making it difficult for us to identify the topics. After the preprocessing, we created dictionary and bag of words model from the corpus. We built a TF-IDF model of the corpus to make sure that the stop words have lower weight compared to the important words.

5.3.2 Choosing the number of topics

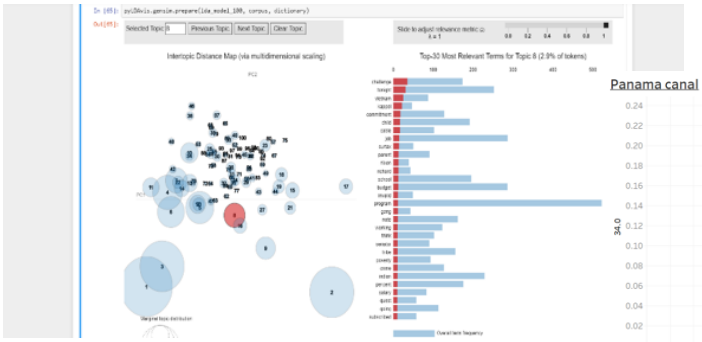
Choosing the number of topics is one of the core challenges when it comes to Latent Dirchlet Allocation. For finding the optimal number of topics, we experimented with other topic modeling technique which is Latent Semantic Indexing. Latent Semantic Indexing is topic modeling algorithm based on singular value decomposition. LSI transforms data into completely different space such that two closely related topics end up being close to each other. We experimented with number of topics in LSI and ran 3 iterations with different value for K in each iteration. Number of unique topics per model were plotted.



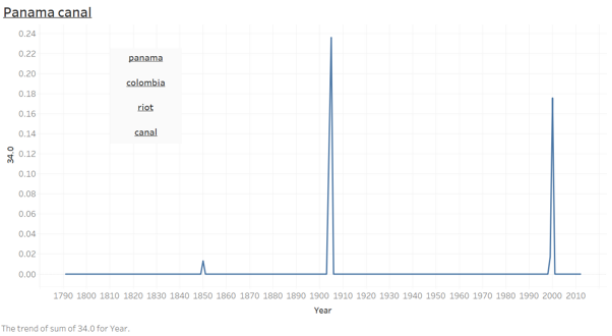
We got 58 unique topics for a K value of 75 in LSI. We ran 200 iterations of lsi by setting to find the optimal number of topics.



The number of unique topics increase with increase in number of topics and the curve hits the peak at 120 before falling. We have chosen 100, 120 possible values for number of topics when we run LDA based on the results of LSI. We ran multiple iterations of LDA with different K value (no of topics) each time. The topic models were evaluated based on the topic coherence and we also used the visualizations to look at how spread out the topics were. We compared the LDA models with no of topics ranging from 25-125. The models have been evaluated using coherence. The model with 100 topics had the highest average coherence amongst all the topics Therefore we choose 100 as the number of topics for our document set. The image placed below shows the pyLDAvis of topic model.



PANAMA CANAL

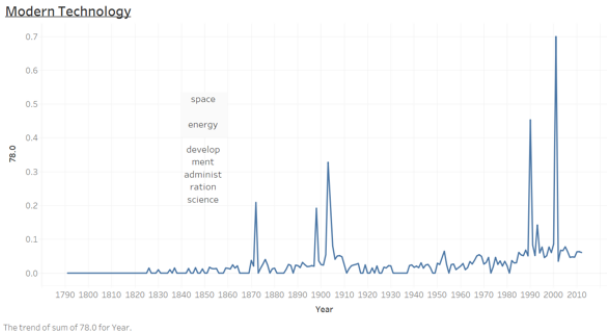


The topic about the panama canal issue has risen rightly between 1900-1910. It is another example of the ability of TOT to exactly localize the topic.

5.3.3 RESULTS

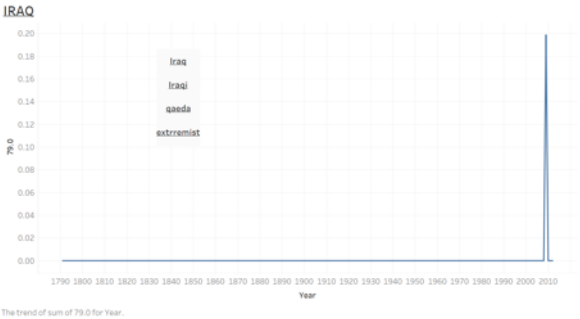
TOPICS DISCOVERED

MODERN TECHNOLOGY AND SCIENCE



The topic related to modern technology containing words like space, energy, science has risen in the 21st century it hits its peak in 2000s. This topic has been addressed previously in the speeches in 1900s but one can clearly notice the upsurge in 2000's due to the technical advancements in 2000s.

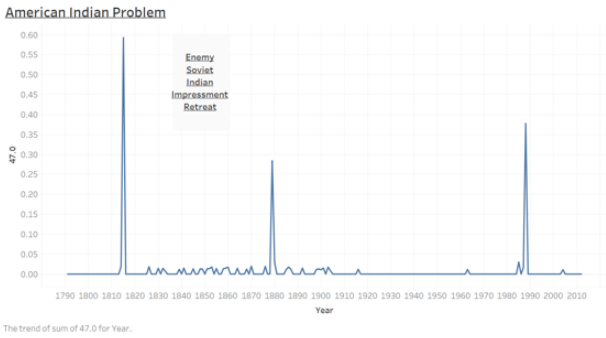
IRAQ

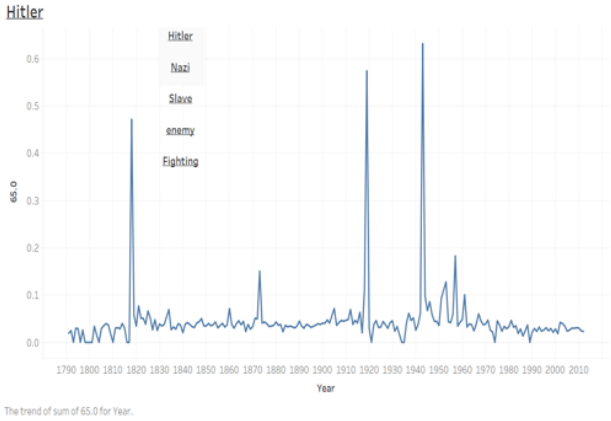


We can clearly see that the topic gained prominence only in the 2000s. Presidents started taking extremism, qaeda, Iraqi only after 9/11.

AMERICAN INDIAN MOVEMENT

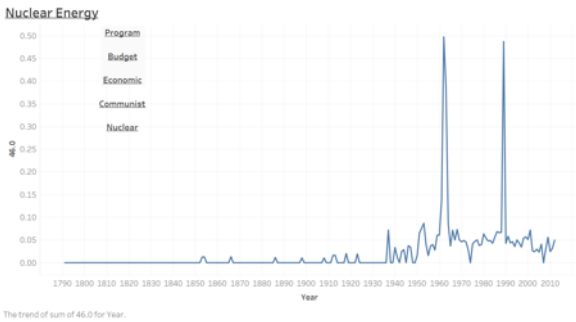
This topic is related to the American indian movement in 1800s . The topic hits its peak in 1800s.





NUCLEAR ENERGY

The topic of nuclear energy has only started from the 1950. Until then the topic of nuclear energy was never mentioned.

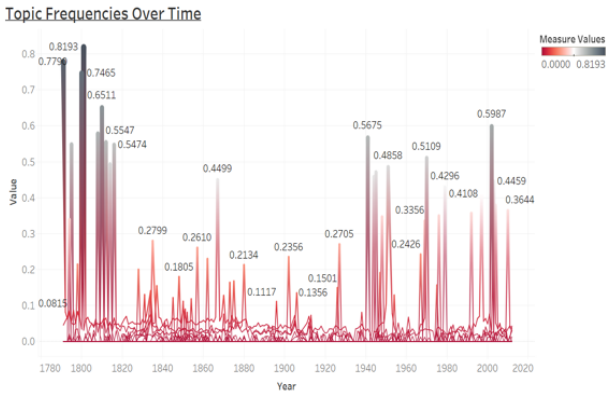


HITLER

This topic containing the word hitler hits its peak between 1930-1950. We can see that this model can be used to find the age in which prominent figures were really famous.

6 CONCLUSION

Understanding the semantic change of words and topics can help linguists, librarians in a huge way. It can also help NLP applications by solving the problem polysemy which has been a thorn in side of NLP. However, identifying words from a corpus is a difficult because such change occurs slowly. In this project, we tried to identify semantic change of words using word embeddings and Topics over time model. These methods work well and do a great job in tracking the sematic change. However, we believe that the results would have been better if we choose a larger corpus such as google-n gram dataset. The Topics over time model does a great job in identifying the events over the course of time. It faithfully points towards an event by increasing topic frequency of the words related to that event. In the example of Panama Canal issue it increases the frequency of the topic related to panama canal exactly between 1901 and 1910. This kind of accuracy can make the news classification easier.



7 Future Work

For our future work we will trying to implement dynamic word embeddings to model semantic change. We will also trying to model semantic change using Dynamic topic models. We would also want to look for a reliable metric to evaluate a topic model.

8 Contributions

- Akhila Kotapati
  - Data Acquisition, Data Preprocessing, Word2Vec
  - implementation, K-means Clustering, TSNE for plots.
  - Tools: Tableau, Jupyter.
- Vamsi Krishna Reddicherla
  - Data Acquisition, Data Preprocessing, LDA (Topics over time).
  - Tools: Tableau, Jupyter.
- Vamsi Varma Kunaparaju
  - Data Acquisition, Data Preprocessing, GloVe implementation, TSNE for plots.
  - Tools: Tableau, Jupyter.

References

[1] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. *GloVe: Global Vectors for Word Representation.*

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. *Distributed Representations of Words and Phrases and their Compositionality,Advances in Neural Information Processing Systems* . 26 (NIPS 2013).

[3] Manish Chablani, *Word2Vec (skip-gram model): PART 1 - Intuition(2017,June 14th)* `https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b`

[4] Jan Bussieck. *Demystifying Word2Vec(2017,August 22).* `https://www.deeplearningweekly.com/blog/demystifying-word2vec`

[5] Xin Rong. *Zur Elektrodynamik bewegter Körper*. (German) [*On the electrodynamics of moving bodies*]. Annalen der Physik, 322(10):891âĂŞ921, 1905.

[6] *word2vec Parameter Learning Explained(2016 June 5th)* `https://arxiv.org/pdf/1411.2738.pdf`

[7] Robert Bamler, Stephan Mandt. *Dynamic Word Embeddings(2017, Jul 2017).* https://arxiv.org/pdf/1702.08359.pdf

[8] Chris McCormick. *Word2Vec Tutorial Part 2 - Negative Sampling(2017, Jan 11).* http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/

[9] Vivek Kulkarni . *Statistically Significant Detection of Linguistic Change (2014, Nov 12th)* https://arxiv.org/pdf/1411.3315.pdf

[10] Derry Tanti Wijaya . *Understanding Semantic Change of Words Over Centuries(2011, Oct 24th).* http://rtw.ml.cmu.edu/papers/wijaya-detect11.pdf

[11] X. Wang and A. McCallum. *Topics over time: a non-markov continuous-time model of topical trends.* . In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining,KDD âĂŹ06, New York, NY, USA, 2006. ACM.

[12] Ashok N. Srivastava, Mehran Sahami . *Text Mining: Classification, Clustering, and Applications(2009)*

[13] Chris Manning, Richard Socher . *GloVe: Global Vectors for Word Representation, Natural Language Processing with Deep Learning, Stanford.*

[14] JonGauthierâĂŤCambridge,Massachusetts. *A GloVe implementation in Python.* .

[15] maciejkula/glove-python . *https://github.com/maciejkula/glove-python/blob/master/readme.md*