Topics and Trends in States of the Union from the Beginning of the Nation

Alexandra Gates, Aleister Montfort, Mario Moreno

Introduction

On January 9, 1790, George Washington went before Congress and delivered the first State of the Union (SOTU henceforth). In what's become an annual tradition, every President since has delivered his message to the country. Within those speeches, presidents reflect the challenges and opportunities facing the Nation, outline their policy priorities and, along the way, provide a window into the prevailing national sentiment.

Using the text of every SOTU address, our project will attempt to map United States' priorities over time and predict how a given President would talk about a topic in aggregate for his SOTU. In order to do so, this project will use topic modeling to track topics over time, and then deploy deep learning to draft a new paragraph given an input of a president and a topic. Lastly, time permitting, we will also spend time visualizing our findings in an easy-to-digest fashion.

Broadly speaking, our project will be divided into four parts: cleaning and preparing the data, topic modeling, deep learning text generation, and visualization. We will write python scripts that clean, explore the data, tokenize, break down the data to paragraphs, and use common algorithms such as Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA), and Non-Negative Matrix Factorization (NMF) to do the topic modeling for each paragraph. Once each paragraph is given its topic area, another set of python scripts will break this new dataset into train, test, and validation splits in order to generate a sample paragraph from a given president on a given topic.

Literature Review

There's an extensive amount of research related to the fields of topic modeling, included papers and articles already published that are directly related to our dataset and choice of project. In an academic paper analyzing lexical changes in SOTU addresses over time, Rule, Cointet and Bearman (2015) developed a strategy for identifying categories in texts despite changes in lexicology over time. In doing so, they provide a blueprint for two critical questions in our own analysis: how do we account for changes in the use of language and words in the last 230 years; and how do we identify common themes in these texts in spite of those changes?

To answer those questions, the authors deploy a method known as co-occurrence, which identifies categories based on terms co-appearing over a unit of text. This analysis involves discrete steps. First, the authors find frequently occurring nouns and paired terms such as 'national security,' and 'local government.' Next, they induce semantic categories by understanding the patterns in co-occurrence of terms in a period of time by identifying how many times joint terms appear together in a given paragraph, compute a proximity score that measures the relatedness of similar terms, and then employ a community detection algorithm to

identify cohesive subsets. They first applied these steps to all the SOTU speeches to develop a global semantic network, and then broke it down into local semantic networks by doing the same steps over discrete periods of time. Finally, the relationship between local semantic networks was determined using a river network.

The authors provide three results, two of which are directly applicable to our work. First, they are able to identify a set of clustered points that directly relate to one of nine topics covered by SOTU speeches over time. Second, they are able to track the importance of these nine topics over time based on how often they are mentioned in each speech.

Other papers use different methods to identify topic clusters in SOTU speeches, though without the significant time component differences explored in the prior paper. Crockett and Lee (2012) use 23 recent SOTU speeches to identify topic clusters. To do so, they employ two different approaches: text mining and topic approach. The difference between these methods is that the former creates an agnostic set of clusters, determined by an algorithm, whereas the latter involves some domain knowledge which defines the “major topics of interests”. These authors only provide an analysis of what are the most important topic clusters for this subset of texts.

Schmidt and Fraas (2015), in an article for *The Atlantic*, developed an interactive tool to analyze the frequency of words mentioned by each president, as well as the evolution of relevant topics or values that are important in United States discourse. For instance, they track the evolution of the words “Liberty” or “War”, and discuss the inclusion of gender topics by tracking the evolution of the word “her”. They also count the mention of foreign countries as a proxy for understanding isolationist tendencies of presidencies.

Beyond multiple forms of detecting clusters and given our interest in writing new text given a topic and a president, we also reviewed papers or research relevant to deep learning networks and machine learning speech creation. One of the goals of this project is to predict what any president among the past 45 presidents would have said about a specific topic. For example, what would a paragraph of President Wilson's SOTU would look like if we provide a model with the name of the President and a topic of interest? With this goal in mind, we have reviewed some of the works or projects that have used neural networks to predict speeches, scripts or any other type of texts.

There are plenty of examples on the web that use Recurrent Neural Networks to perform this task. As pointed out by Le, with the use of Recurrent Neural Networks (RNN) it is possible to make use of sequential information to generate text predictions. For example, to predict a word in a sentence, RNN makes use of the information given by the relations of the previous words to predict the following word. An application of this to political speeches is Karpathy's software, which uses this method to generate a machine generated speech as if President Obama were speaking.

This project will use three different approaches to predict political speeches: RNN, Bidirectional RNN and Long Short-Term Memory models. We will tune the parameters, evaluate the performance of these models, and pick the one that better predicts a specific president's speech, according to an evaluation metric to be defined.

Plan of Action

Overall, we will start by cleaning and preprocessing the data into a format machines can use. Specifically, we will tokenize the data and vectorize it using one-hot encoding by paragraph. Each paragraph will be assigned a topic using common algorithms such as Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA), and Non-Negative Matrix Factorization (NMF). The output of this analysis will add multiple dummy variable to the dataset for each topic identified by the topic modelling analysis.

After topic modelling is complete, we will then split the data into training, validation, and testing data sets. The training dataset will be used to tune parameters such as determining the appropriate number of nodes and layers necessary. We will also choose loss and activation functions for each of the methods we have chosen to run. We will then run RNN, bidirectional RNN, and Long Short-Term Memory (LSTM) to write possible paragraphs of text based on the president and topic, as identified above. We will train the models of the training data, tune the parameters using the validation data, then test the accuracy of our models on the testing data.

The final product will be a simple website where the user inputs a president and a topic, and the website returns a machine generated paragraph learned on what the president had said about the topic in his SOTU addresses.

For a more detailed breakdown of when each component will be done, see Table 1 below.

Evaluation

Evaluating an unsupervised model that clusters topics using algorithms like LDA, LSA, or NMF might be challenging, given that unsupervised learning models don't require train, test, or validation splits to create their clusters. Therefore, in order to truly evaluate our model, we'll need to hold out a random subset of paragraphs in our analysis as a potential test set and measure a value such as perplexity -- how confused our model was in determining the topics in the hold-out set when compared to its confusion in determining clusters in the training data. Other methods for evaluating unsupervised topic modeling problems exist, and we will likely explore them throughout the course of this work. It's important to note, however, that these methods often require a hold-out test set for unsupervised learning evaluation.

To evaluate the models that predict political speeches, we need to define at least two metrics: a similarity score to evaluate how the predicted text compared to other (unseen) texts from the

same author, and also if it is grammarly correct and follows a logical structure. Further evaluations metrics will be defined.

Teamwork Makes the Dream Work

We will meet twice a week on Saturdays and Wednesdays. In Saturday's meeting we will determine what work needs to get done that week and divide that work up between the three of us. In Wednesday's meeting goal we will update each other on our progress thus far and adjust work divisions as necessary. Overall, we anticipate splitting the work evenly among the three of us and meeting our goals for each week. See table 1 for a detailed breakdown of the tasks to be performed.

Table 1: Calendar and Work Divisions.

Week	Tasks To Get Done That Week	Who Does What	Deliverable
Week 3	Administrative: establish github; read in data; decide on models and methods to use; conduct literature review; decide on structure of data. Pipeline: clean data. Paper: write up section of final paper related to cleaning and preprocessing data.	Mario: establish github and help with cleaning data. Alix: read in data and help with cleaning data. Aleister: start code for cleaning data Group: decide on models and methods to use; conduct literature review; submit project proposal.	Project Proposal
Week 4	Exploratory analysis: Most common words by president by in general and by speech; most common words by party; most common words by century/half century/different time cuts; most common topics. Pipeline: finish cleaning & preprocessing data; start topic modelling. Paper: write up section of final paper related to cleaning and preprocessing data & exploratory analysis.	Mario: wrap up preprocessing code & start topic modelling code. Alix: exploratory analysis doing most common words by century/half century/different time cuts; most common topics. Aleister: exploratory analysis doing most common words by president by in general and by speech; most common words by party.	
Week 5	Pipeline: continue topic modelling. Paper: write up section of final paper related to topic modelling.	Mario: work on topic model code using Latent Dirichlet Allocation (LDA) method.	May 1: Mid-quarter Presentation

		Alix: work on topic model code using Latent Semantic Analysis (LSA) method. Aleister: work on topic model code using Non-Negative Matrix Factorization (NMF) method.	
Week 6	Pipeline: Finish code for topic modelling; split data into train, validate, test datasets; start code for tuning hyper-parameters for paragraph predicting. Paper: write up section of final paper related to topic modelling and parameter tuning performed for text prediction.	Mario: finish topic model code using Latent Dirichlet Allocation (LDA) method; start tuning code. Alix: finish topic model code using Latent Semantic Analysis (LSA) method; split data and start tuning code. Aleister: finish topic model code using Non-Negative Matrix Factorization (NMF) method; start tuning code. Group: meet to discuss findings from separate topic modelings & choose what parameters to tune.	
Week 7	Pipeline: finish tuning code, start paragraph predicting modelling code. Paper: write up section of final paper related to parameter tuning and models performed.	Mario: finish tuning parameters; start code to do paragraph predicting using Bidirectional RNN. Alix: finish tuning parameters; start code to do paragraph predicting using RNN. Aleister: finish tuning parameters; start code to do paragraph predicting using LSTM.	
Week 8	Pipeline: finalize code for neural networks with tuned parameters; evaluate model and adjust as necessary. Paper: write up section of final paper related to models performed,	Mario: work on code to do paragraph predicting using Bidirectional RNN. Alix: work on code to do paragraph predicting using RNN.	

	evaluation done, and adjustments made.	Aleister: work on code to do paragraph predicting using LSTM. Group: meet to look over results of each model and determine what needs to be adjusted.	
Week 9	Pipeline: develop final website and visualizations. Paper: write up section of final paper related to website and visualizations.	Mario & Aleister: work on website. Alix: create trends visualizations.	
Week 10	Pipeline: Finalize code for submission. Paper: Edit paper and put into final format.	Group: split remaining work evenly.	June 4-5: Final Presentation & Project Due

References

Crocket and Lee. “Does it Matter What They Said? A Text Mining Analysis of the State of the Union Addresses of USA Presidents.” *13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*. (2012). https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6299261&tag=1

Fraas and Schmidt. *The Language of the State of the Union.* The Atlantic Magazine Online. January 18, 2015. https://www.theatlantic.com/politics/archive/2015/01/the-language-of-the-state-of-the-union/384575/

Karpathy. *The Unreasonable Effectiveness of Recurrent Neural Networks.* Personal Blog. (May 2015). https://karpathy.github.io/2015/05/21/rnn-effectiveness/

Le. *Recurrent Language Networks: The Powerhouse of Language Modeling.* Medium Towards Data Science. (Sept 2018). https://towardsdatascience.com/recurrent-neural-networks-the-powerhouse-of-language-modeling-d45acc50444f

Rule, Cointet, and Bearman. “Lexical Shifts, substantive changes, and continuity in State of the Union discourse, 1790-2014.” *Proceedings of the National Academy of Sciences of the United States of America.* Vol 112, No. 35 (Sept 2015), pp 10837-10844. https://www.jstor.org/stable/pdf/26464078.pdf?ab_segments=0%2Fdefault-2%2Fcontrol&refreqid=search%3A9dd2c687070fee011c528616f40828b7

Wallach, Murray, Salakhutdinov, Mimno. “Evaluation Methods for Topic Models.” *GitHub Paper*. http://dirichlet.net/pdf/wallach09evaluation.pdf