Jack Kolb, Gao Xian Peh, Jong Ha Lee
INFO 290
November 26, 2016

Introduction

Given the significant negative impact of Hillary Clinton's email scandal on her election campaign, our team felt that an analysis of her emails will be an interesting choice for this assignment. In order to better understand the content of Hillary Clinton's emails, we decided to use `Word2Vec` to to compute the embeddings of the words contained in her emails. Subsequently, we conducted dimensionality reduction using t-SNE (originally PCA) and subsequently did a scatter plot to try to understand the underlying relationships in the content of Hillary Clinton's emails. In our latest work, we performed an analysis both using the standard `Word2Vec` approach and a modified approach in which certain phrases, such as "Hillary Clinton," are embedded as a single element rather than as two separate elements.

Dataset Description

We used a collection of Hillary Clinton's emails for this assignment. The data set is available from Kaggle, a general data science repository. These emails were voluntary released by the United States Department of State due to a request under the U.S. Freedom of Information Act. The data set contains 7,945 emails in total, which together form a corpus of about 2,329,240 words (1,540,809 after removing stop words). These emails were originally released as PDF files, but contributors at Kaggle processed and cleaned these files to produce a simple CSV file containing both the metadata (e.g. `From` and `To` headers) and contents of every email message. After removing stop words and performing some basic stemming, we were left with a vocabulary of 16,129 distinct words that are embedded as vectors.

Method

Preprocessing the Data

We used `Pandas` to read in the email CSV file as a data frame. As a preliminary step, we performed some basic cleaning to remove extraneous material from the emails. For example, each email was prefixed with a header giving details like its secrecy status and whether or not the text of the email was released in full or partially redacted. We removed these headers because we are primarily interested in the original content of the emails.

We decided to treat each email message as a "sentence" for the skip-gram model. For each message, we removed stop words (common and uninformative words like "the", "a", and "to") using the remove_stopwords function from the gensim.parsing.preprocessing module. Finally, we performed stemming on each sentence (e.g. reducing related words like “learning” and “learned” to its root form “learn”) using the stem_text and removed punctuation using the strip_punctuation function, both also provided in gensim’s preprocessing module.

Computing the Embeddings

We used the Word2Vec implementation provided as part of the gensim Python library to compute embeddings of the words contained in the emails. We chose to use the skip-gram model with a window size of 4 and minimum word occurrence count of 5, as both of these parameter values seem to be fairly standard. We used a vector dimensionality of 100 because it produced good plots after completing the t-SNE process. This seems reasonable because we had encountered documentation suggesting that the vector dimensionality should be roughly the square root of the total vocabulary size (which is ~117 in our case).

Dimensionality Reduction

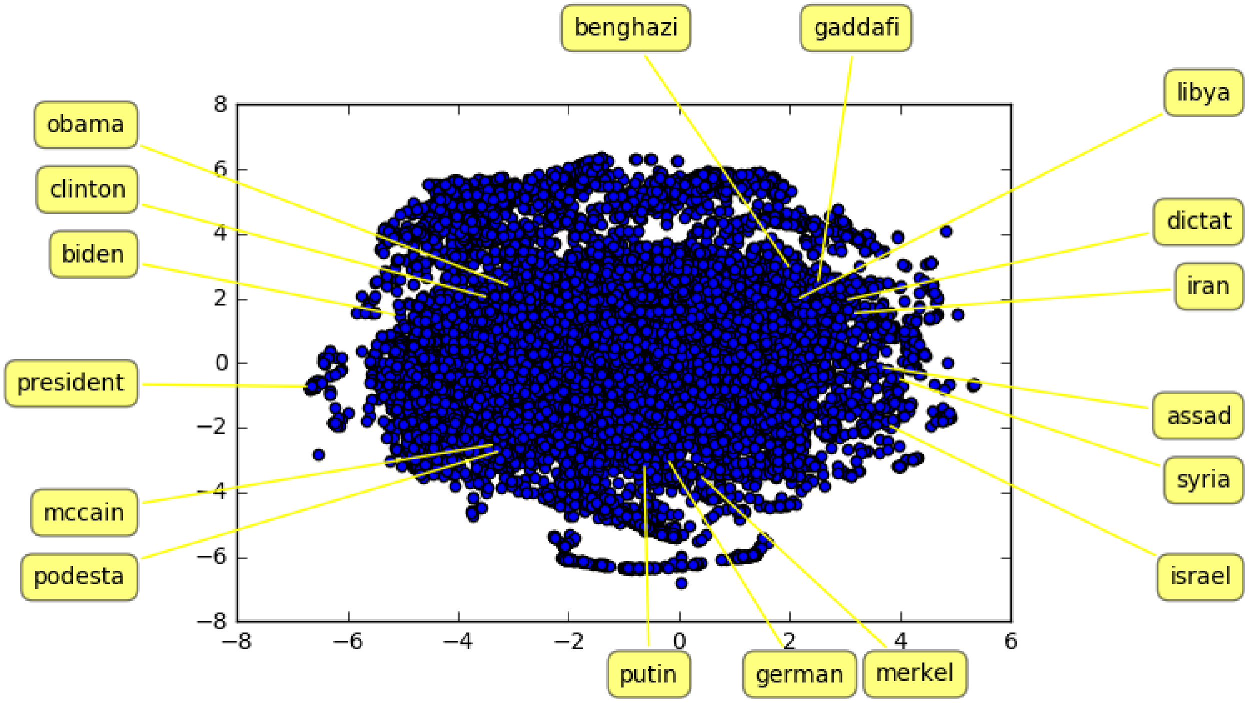
To reduce the 100-element vector embeddings of each word down to 2 dimensions, we used a Python implementation of t-SNE from the scikit learn library. We changed to t-SNE from PCA so that we could look for clustering behaviors in our visualization. This required instantiating a new t-SNE model with the number of components set to 2 and using its fit and transform functions. The model can be further specialized with several optional parameters, but we chose to leave them at their default values given our limited knowledge of of the underlying techniques involved.

Plotting

As the last step, we constructed a scatter plot where each word is depicted at the *x* and *y* coordinates corresponding to the two elements of its embedding after dimensionality reduction by t-SNE. We used the matplotlib library and its scatter function for this purpose. We also picked a few words to explicitly label on the plot in order to try to interpret what kinds of inter-word relationships the skip-gram model may have surfaced. We completed this using matplotlib’s annotate function.

Visualization

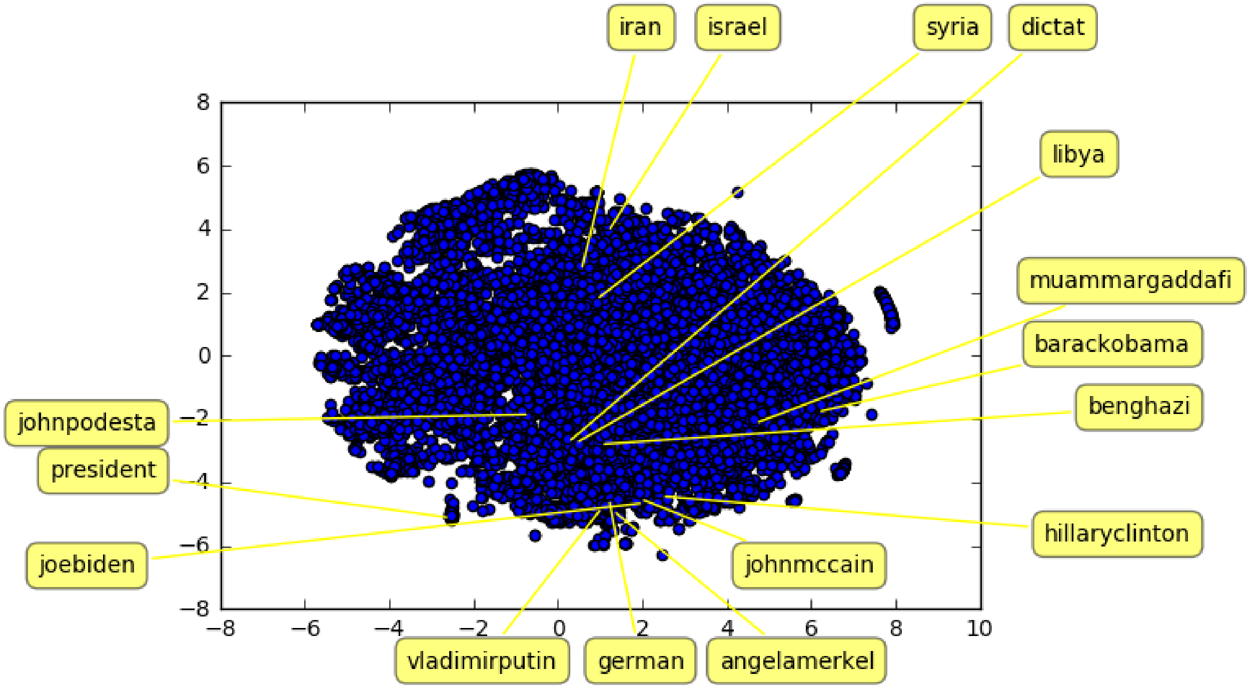
Without Phrases



The above plot is a visualization of the t-SNE reduced embeddings of several interesting words seen in the Clinton email dataset. Because we used t-SNE rather than PCA for our dimensionality reduction, we are looking for clustering behavior rather than vector relationships in this visualization. The first feature of note is the fact that the words appear to be grouped roughly into geographical clusters. American political figures like Barack Obama, Hillary Clinton, and John McCain all appear on the left side of the plot. Even within this group, there appears to be a distinction between key political leaders (Obama, Clinton, Biden) and individuals who play supporting roles (McCain, Podesta). The second major cluster is in the top right of the figure. Here, we see the names of Middle Eastern states and their leaders. Benghazi, Gaddafi, and Libya all appear near to each other, as do Syria and Assad. Iran is somewhere between Syria and Libya, and Israel is arguably still in this cluster but farther away from the other members, which makes sense from a geopolitical perspective as well. The final cluster is in the bottom center of the figure and includes European leaders and countries. It was encouraging to see that, like Assad/Syria and Gaddafi/Libya, the embedding for Angela Merkel appears close to the embedding for “German” (the stemmed form of Germany).

As a final exercise, we looked at the embeddings for “President” and “Dictat” (the stemmed form of Dictator). Interestingly, the embedding for president is isolated in a cluster on the left extreme of the figure, while the embedding for dictator appears to be part of the Middle East cluster we observed earlier. It is unclear what this means, as “President” is often a title applied to leaders of non-democratic regimes, and dictatorships are hardly exclusive to the Middle East.

With Phrases



In a second analysis, we forced the `Word2Vec` model to treat certain two-word phrases as a single element to be embedded. We accomplished this simply by removing the whitespace between the two words where they occurred within the Clinton email corpus before feeding the text into `gensim` for processing. For example, “hillary clinton” becomes “hillaryclinton.” The above plot is the result of computing the embeddings and plotting them after reduction to two dimensions via t-SNE. Unfortunately, our results here are probably less insightful than those we saw before adding the notion of phrases to our analysis. Hillary Clinton and John McCain now seem to fall into the European leader/nation cluster, and the embedding for Barack Obama has somehow ended up near Muammar Gaddafi. Libya and Benghazi are clustered near each other but are relatively far away from Syria, Iran, and Israel. The embedding for “president” is still in an isolated cluster, and the embedding for “dictator” is closest to the embedding for Libya.

We think that our phrase-based analysis may not have worked well because we focused mainly on the names of important political figures, but these may not always be used in full form in email. For example, we could commonly see people refer to Russian leader Vladimir Putin as simply “Putin.” The situation is further complicated by the use of titles in front of last names. It is also correct to refer to Vladimir Putin as President Putin. In fact, the full phrase “Hillary Clinton” appears surprisingly infrequently in this corpus of emails. We suspect this is because most people would refer to her simply as Hillary if they have a close relationship or more formally as Secretary Clinton or Madam Secretary if the situation is more formal. In any case, computing a single embedding for the same person that covers the many ways they can be referred to presents an interesting opportunity for future work.

References/Citations:

Dataset Source: https://www.kaggle.com/kaggle/hillary-clinton-emails