PROCESS BOOK

TITLE: Analyzing and Visualizing Hillary Clinton emails dataset

Submitted by:

Kilari Murali Krishna teja

Vinitha Yaski

Rohan Kohli

Background and Motivation:

The dataset we chose for the class project is the “Hillary Clinton emails dataset” and we intend to do Topic Modelling on the data using Latent Dirichlet Allocation (LDA) to gain interesting insights and then visualize them interactively. We selected this because not only it will be quite interesting in the current political landscape of the country and in the scenario of upcoming elections but also we believe that with efficient preprocessing techniques and data mining algorithms we can try and extract some fascinating insights from this dataset.

Project Objectives:

Our primary objective is to visualize a large dataset of emails in such a way that the most relevant topics discussed in those emails are programmatically brought to the attention of the user. We have a visualization of topics and the most relevant words in that particular topic. We also expect to have a stacked area chart that describes the temporal frequency of a word within a topic over the years. For this purpose, performing topic modeling than just counting the most frequent words that appear in the emails is clearly the better option as it gives us interesting clusters of words unlike the frequency count approach. As the dataset in question is the Hillary Clinton email dataset, we expect quite a few people to be interested in our visualization and the results that data mining brings to light.

Related work:

The following websites and papers discuss exciting ways to display textual and email data .We are thankful to the professor for pointing us towards them. The first paper actually inspired us to try and display the temporal frequency relationship using the stacked area chart.

1. http://mariandoerk.de/visualbackchannel/infovis2010.pdf
2. https://flowingdata.com/2008/03/19/21-ways-to-visualize-and-explore-your-email-inbox/
3. http://www.cs171.org/2015/assets/slides/12-TextVis.pdf

Data:

The dataset we are going to use is publicly available and is hosted at www.kaggle.com/c/hillary-clinton-emails . Please note that a raw version of this dataset is released by the government initially which consists of PDF files. However, the dataset on Kaggle is a cleaned version of the government version and the entire data is distributed over different files. The file names and their description is given below:

Emails.csv

Id - unique identifier for internal reference

DocNumber - FOIA document number

MetadataSubject - Email SUBJECT field (from the FOIA metadata)

MetadataTo - Email TO field (from the FOIA metadata)

MetadataFrom - Email FROM field (from the FOIA metadata)

SenderPersonId - PersonId of the email sender (linking to Persons table)

MetadataDateSent - Date the email was sent (from the FOIA metadata)

MetadataDateReleased - Date the email was released (from the FOIA metadata)

MetadataPdfLink - Link to the original PDF document (from the FOIA metadata)

MetadataCaseNumber - Case number (from the FOIA metadata)

MetadataDocumentClass - Document class (from the FOIA metadata)

ExtractedSubject - Email SUBJECT field (extracted from the PDF)

ExtractedTo - Email TO field (extracted from the PDF)

ExtractedFrom - Email FROM field (extracted from the PDF)

ExtractedCc - Email CC field (extracted from the PDF)

ExtractedDateSent - Date the email was sent (extracted from the PDF)

ExtractedCaseNumber - Case number (extracted from the PDF)

ExtractedDocNumber - Doc number (extracted from the PDF)

ExtractedDateReleased - Date the email was released (extracted from the PDF)

ExtractedReleaseInPartOrFull - Whether the email was partially censored (extracted from the PDF)

ExtractedBodyText - Attempt to only pull out the text in the body that the email sender wrote (extracted from the PDF)

RawText - Raw email text (extracted from the PDF)

Persons.csv

Id - unique identifier for internal reference

Name - person's name

Aliases.csv

Id - unique identifier for internal reference

Alias - text in the From/To email fields that refers to the person

PersonId - person that the alias refers to

EmailReceivers.csv

Id - unique identifier for internal reference

EmailId - Id of the email

PersonId - Id of the person that received the email

database.sqlite

This SQLite database contains all of the above tables (Emails, Persons, Aliases, and EmailReceivers) with their corresponding fields. You can see the schema and ingest code under scripts/sqlImport.sql

There are in total 20220 rows in the Emails.csv prior to any data cleaning and preprocessing and it is safe to assume that the actual number of useful emails to our analysis will be very different as the topic modelling algorithm requires atleast one non-sparse word in each email to take that email into consideration and hence emails with very short length will most likely be deleted by our preprocessing.

NOTE : Please note that even though Emails.csv contains other columns such as Case Number, Document Number and Date released, they are not of much importance to our analysis and hence will not be used in the project.

Data Cleaning and preprocessing:

This step is going to be crucial because we should first clean the dataset to remove the irregularities and to remove emails that are missing the features we care about such as From, To, Subject, Body text. Currently, since for topic models, we only consider Body text part of the emails, we describe in detail below, the steps that are performed in **R** to read, clean and preprocess the data.

- **Data reading** : The package “readr” is used in R to read the emails.csv file. Since, the size of this file is moderately large and most of the content is “string” this package will be more efficient than R’s default csv reader.
- **Data cleaning and preprocessing**: Since, this is the actual emails dataset there will be a lot of noise in the content of emails and hence the following steps are performed on the dataset to clean it and make it more compact and useful. The “tm” package of R is used for all these steps

1. Creating a corpus: The “bodytext” column is extracted from the csv and a corpus is created
2. Lowering: All characters in the corpus are converted to lower case
3. Conversion the corpus to plain text document
4. Removing punctuations from the corpus
5. Removing numbers from the corpus
6. Removing whitespace from the corpus
7. Removing stopwords from the corpus
8. The corpus is converted into document term matrix and terms with sparsity less than 0.95 are removed from the corpus. If a word is highly sparse, then it means that it occurs very infrequently.

Data mining - Topic Modelling:

Also, after this step we should do Topic Modelling (“topicmodels” package in R) to extract the topics that exist in her emails. In this step, the number of topics is basically a hyper parameter that needs to be tuned manually until we get a good mixture of topics and a good distribution of words in each topic.

We intend to take the number of topics to be 20 (NOTE: for the purpose of milestone,the code in the github repository will have the number of topics as 10 because of the huge memory and time taken by the algorithm) and then try incrementing that number until we find a decent topic model. In order to perform Topic Modelling we plan to use Latent Dirichlet Allocation (LDA) algorithm in R. Since, LDA is a bag of Words model we should also clean the body text of each email so that we remove any “stop words” (words with very high frequency) from the body as done in step 7 of data cleaning and preprocessing.

The LDA algorithm will take as input the number of Topics and a corpus of the text and then tries to find the topics. Each topic is basically a collection of coherent or similar words with each word having a probability which is the probability of that word given the topic.

For example, the word distribution of one of the topics let’s assume Topic –K will be Iran-0.34, Syria-0.21, Afghanistan- 0.17, Pakistan -0.14 etc. Then we can say that Topic-K is giving information about foreign countries and that too particularly about “muslim countries” and hence we can label Topic-K as “Foreign Countries”/ “ Muslim Countries” since the words associated to them have higher probability of appearing within the topic. In the same way if a topic, say Topic- J has the following word distribution . Clinton-0.43, Bill-0.27, Chelsea-0.15, then it is safe to assume that this topic is about her family and hence label it accordingly.

1. Interpretation of Topic models: Topic models discover “topics” that occur in a collection of text. Each topic has a different probability mass function over the set of words (all vocabulary). We want to discover which words are more relevant for which topic
2. Measure-1: Importance of words in a topic is denoted by the probability of the word given that topic. Let’s call it “**P**”
3. Disadvantage with “**P**”: common words tend to appear in every topic and hence will have high “**P**” values for every topic, thus making it difficult to efficiently and accurately discover and differentiate topics

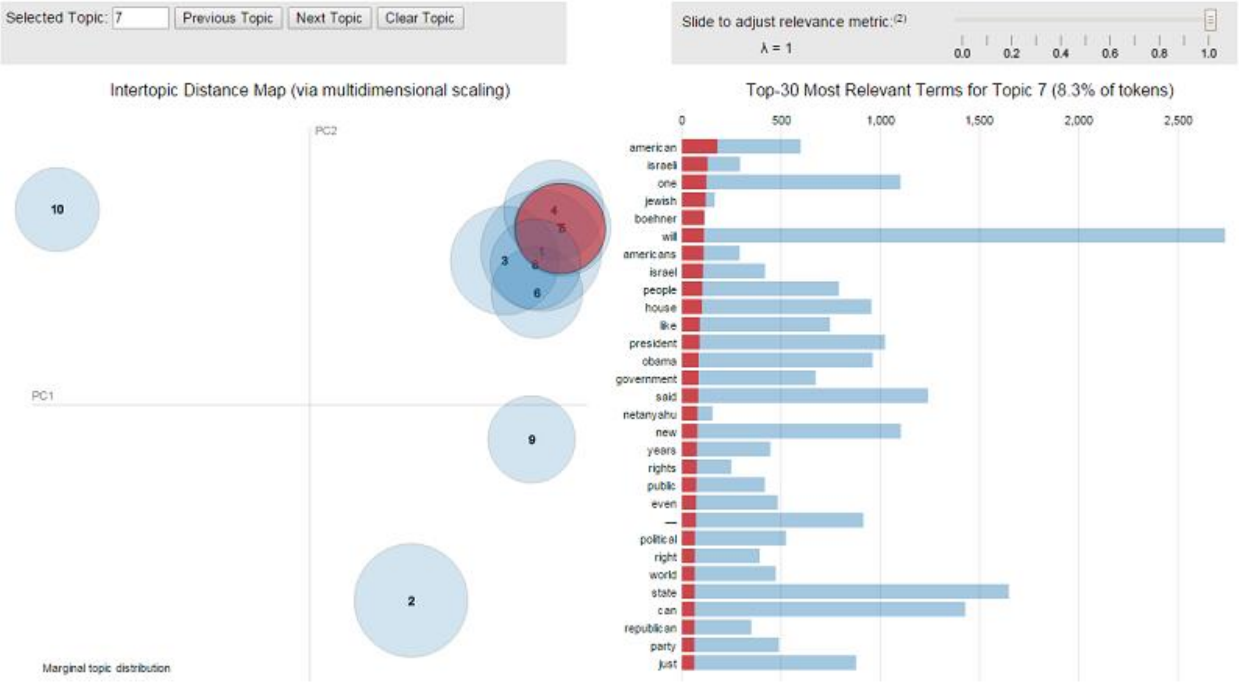
4. Measure-2: We can modify “**P**” by dividing “**P**” with overall probability of the word. Let’s name is “**L**”. “**L**” takes care of the above disadvantage.But,
5. Disadvantage with “**L**”: In the case of rare words (whose probability of occurrence is very low) “**L**” will have high values since we will be dividing “**P**” with very small values.
6. **RELEVANCE PARAMETER**: The relevance parameter **λ** is a tradeoff parameter between ”P” and “R”.

Relevance = **λ*P + (1- λ)*L**

Design:

The following three visualizations show the word distributions for topic “7” which is talking about American government, president Obama and israel for different values of **λ**

1.λ =1

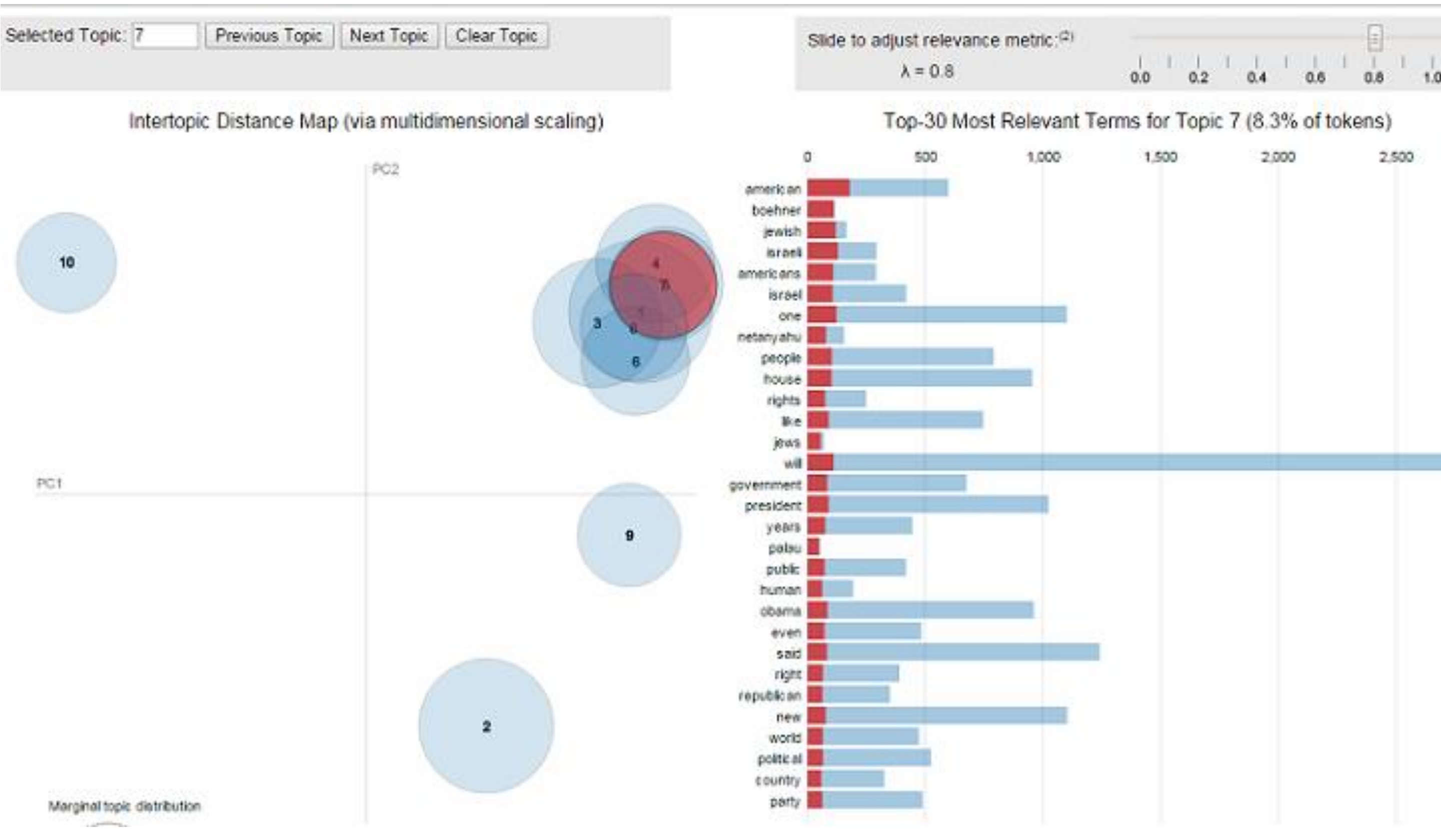


Selected Topic: 7

Slide to adjust relevance metric:(2)

λ = 1

2. λ = 0.8



Selected Topic: 7

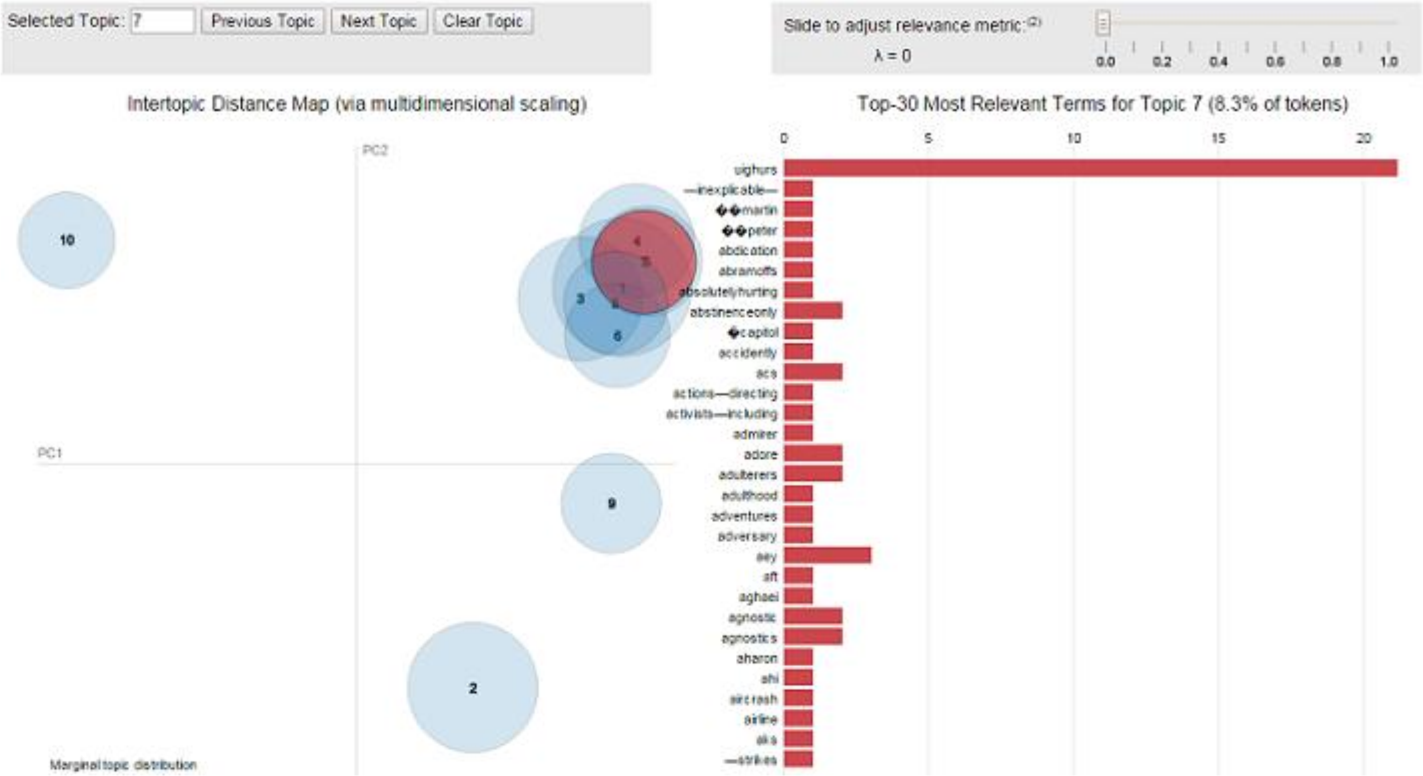
Slide to adjust relevance metric:(2)

λ = 0.8

Intertopic Distance Map (via multidimensional scaling)

Top-30 Most Relevant Terms for Topic 7 (8.3% of tokens)

3. λ=0



Selected Topic: 7

Previous Topic | Next Topic | Clear Topic

Slide to adjust relevance metric:(2)

λ = 0

Intertopic Distance Map (via multidimensional scaling)

Top-30 Most Relevant Terms for Topic 7 (8.3% of tokens)

IMPLEMENTATION OF MILESTONE:

In the project github repository, only the file named **"LDA Hillary.R"** is to be executed. All the other folders and files belong to another library named "LDAvis" which we are trying to edit**. Also make sure that both the "LDA Hillary.R" file and the data folder named "Hillary Clinton public emails dataset" are in the same directory and that this directory is the working directory of "R".** Please follow the below steps to run the file in the order they are given. Make sure that you have 64 bit R installed as there have been issues with 32 bit version as it cannot allocate a memory of greater than 1.8gb to a vector.

1. **Packages required:** Run the following commands in R console to install the required packages before running **"LDA Hillary.R"** file. You can copy/paste the following lines to install them.
   ```
   install.packages("readr")
   install.packages("topicmodels")
   install.packages("dplyr")
   install.packages("stringi")
   install.packages("tm")
   install.packages("LDAvis")
   install.packages("servr")
   ```

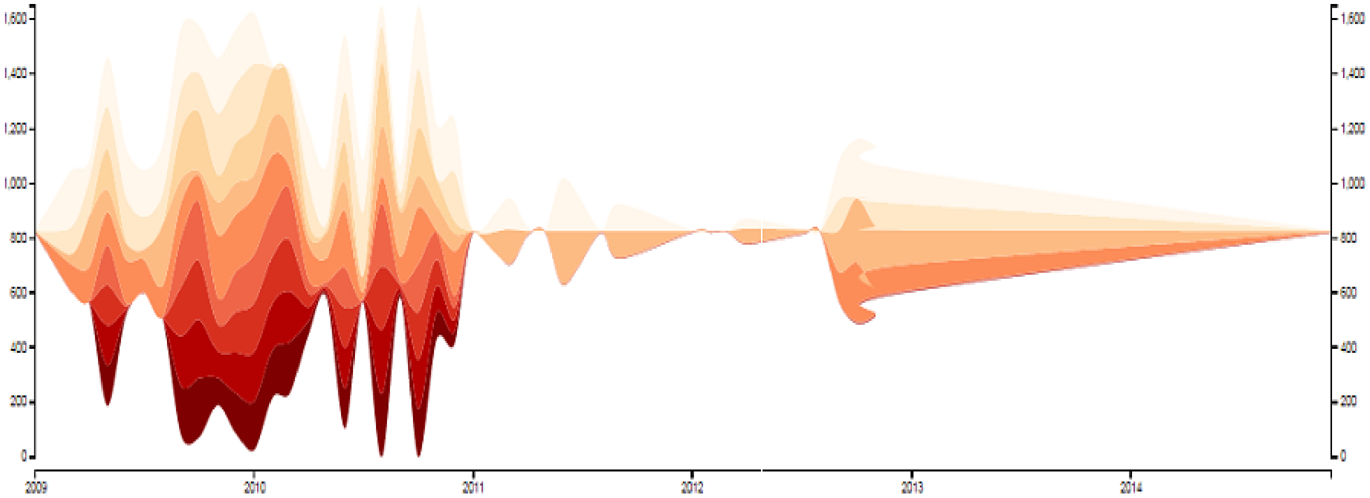
2. **Run the file by copy/pasting the entire file into the R console.**

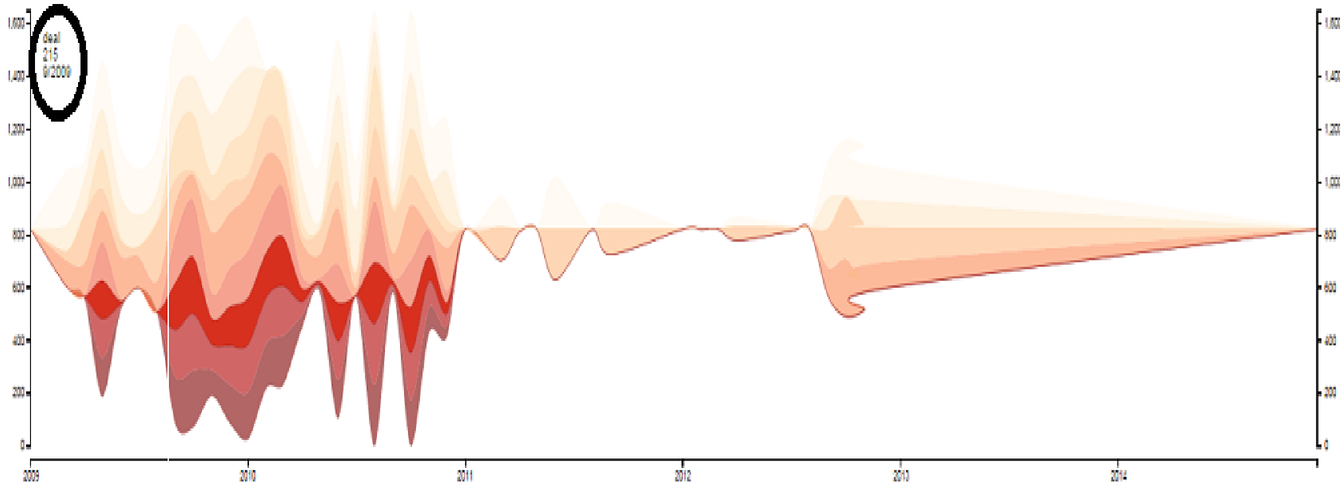
DESIGN EVOLUTION:

- **Migrating from R to d3.js:** We decided to fully migrate from R to d3.js because of the following reasons.
    - The R program we wrote that does the Topic Modelling LDA and produces the visualization is too memory intensive and hence it is not ideal for general use. Moreover, there are specifics that are needed to be met when any uses is running the program and hence we decided to pre compute the topics and terms in each topics using R and use d3.js to visualize the topics which will take far lesser resources.
    - The Bubble chart visualization we get using R is too cluttered and there is overlapping which prevents the user to select any topic of his choice easily. As the number of topics increases, it will get more difficult for the user to select one particular topics. We overcame these disadvantages using d3 and we still displayed the commonalities and contrast among the topics effectively.
    - The final visualization that is displayed by using the R library seems to require too much domain knowledge about the working of topic modelling and the LDA algorithm and hence it won't be effective for the general audience. We addressed this issue using d3.js
    - We were unable to proceed with our original plan of modifying the R library "LDAVis" code because of the inherent complexity of the code and also the

time constraint. Hence, we redid the same visualizations with d3 in more readable and understandable code.

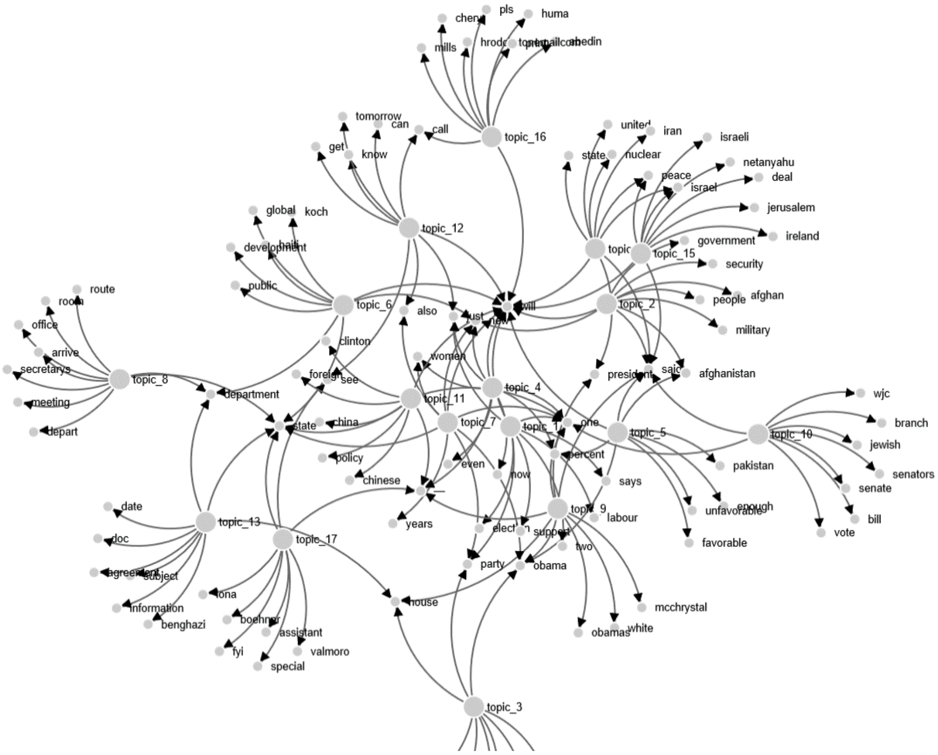
- **Evolution of visualizations in d3.js:**
    - **Stream Graph:** The initial version of stream graph had the time on x-axis. Since it is not feasible to display the frequencies for each individual day for all the years and also there is also a high chance of words being not available on many days, we choose to sum the frequencies per month and we display the data for each month. Tooltip was implemented in stream graph such that on mouse hover, we display the particular word and its frequency for the specific month of the year. The code is inspired from mike bostocks stream graph code. Data preprocessing was done in R in order to get the data in the format required by the stream graph.





- **Labeled - Force directed graph:**

We needed a visualization for the overview of the whole data that we are going to visualize. Considering the multiple topics that are spoken about in Hillary Clinton's emails and the most frequent terms in each topic, force directed graph seemed to be the best choice of visualization. In this visualization, all the categories and the most frequent terms in each topic are represented as nodes. The paths run from each topic node to the nodes that represent the frequent terms in that topic. There are some terms that are common to two different categories which helps us understand the interlinked categories.



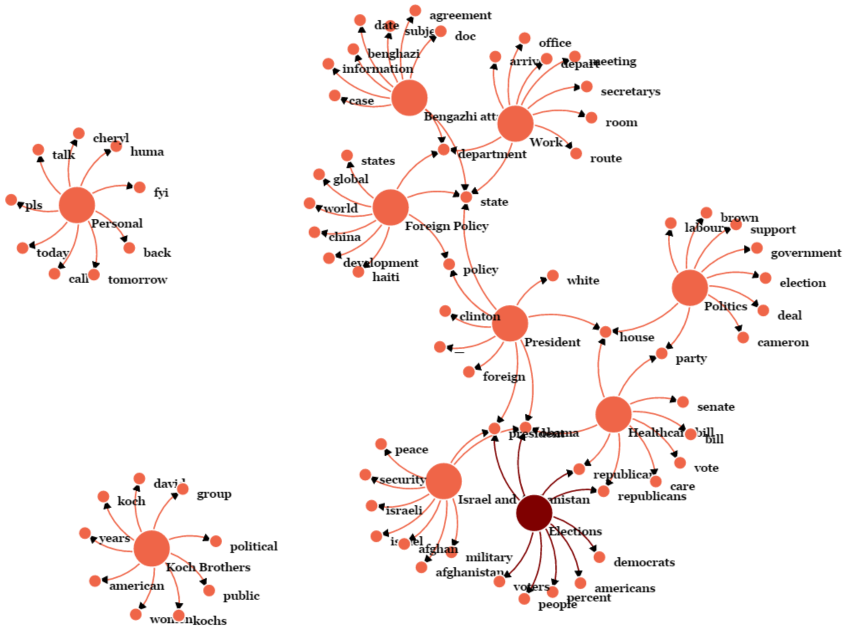
This was our initial implementation of the force directed graph. We enhanced this visualization further to foster the understanding. The enhancements that we made are:

- **Reduced the number of categories:**
  As the number of categories increases the graph gets cluttered and it becomes difficult to understand data from the visualization. Also, when the number of categories is more, the concentration of information in each topic reduces its significance diminishes. Hence, we have decided on modelling 10 categories for the emails.
- **Worked with the charge and link distance attributes of force element:**
  As we have multiple nodes being displayed and each of them have labels assigned, with less charge values, they node labels tend to get cluttered and are less readable. Hence, we had played with the charge and link distance attributes to have the nodes spread out. In addition,

with this done, the nodes tend to go beyond the assigned svg bounds due to high repulsive force. Hence, it was a trade-off between readability and the screen space that the visualization takes.

- **Handling selection:**
  We are using the color attribute to help the user understand the selection made. The initial selection defaults to 1st topic.

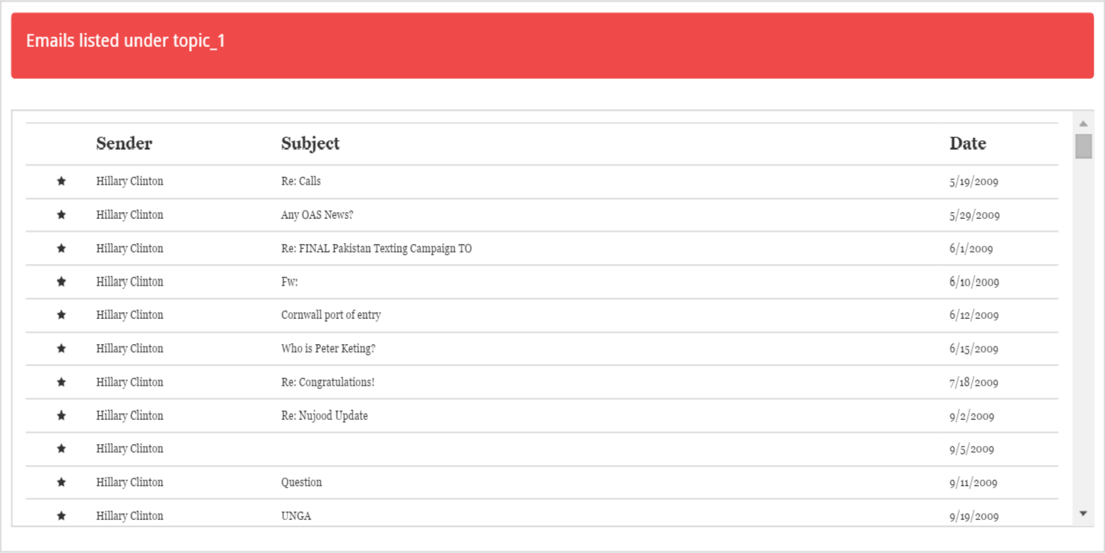
After working on the above mentioned tasks and few others, we have come up with the visualization below:



- **Email Data Display:**

Once the user choses a particular topic, the rest of the visualizations help in understanding the most frequently occurring terms in that topic and the people with whom she interacts the most in that respective topic. This data is derived from a huge list of emails and is filtered accordingly. This visualization gives an access to the emails which are aggregated to derive the needed data. On choosing any particular topic, the list of emails over the 4 years are filtered. This is done by taking the topic probabilities of the emails into consideration. For each email instance, there are probabilities assigned for each topic, which represents the probability with which the email belongs to that particular category. Each email is assigned to a topic that takes the highest probability. With this filtering done, whenever the user selects a particular topic, all the emails that have the highest probability in that topic are chosen and displayed.

This display of emails provides a quick access to the sender, subject and dates of the emails. This is a Gmail like layout, wherein we can click on any email to see the content of the email along with the sender, subject and date information.

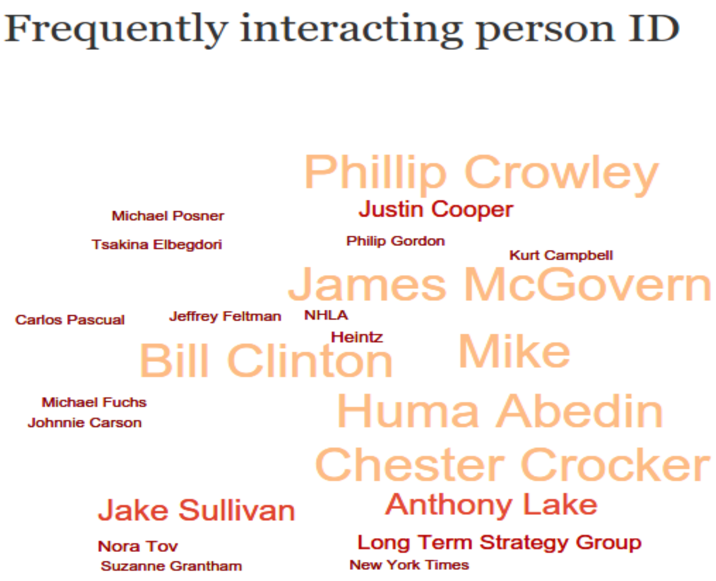


Emails listed under topic_1

	Sender	Subject	Date
★	Hillary Clinton	Re: Calls	5/19/2009
★	Hillary Clinton	Any OAS News?	5/29/2009
★	Hillary Clinton	Re: FINAL Pakistan Texting Campaign TO	6/1/2009
★	Hillary Clinton	Fw:	6/10/2009
★	Hillary Clinton	Cornwall port of entry	6/12/2009
★	Hillary Clinton	Who is Peter Keting?	6/15/2009
★	Hillary Clinton	Re: Congratulations!	7/18/2009
★	Hillary Clinton	Re: Nujood Update	9/2/2009
★	Hillary Clinton		9/5/2009
★	Hillary Clinton	Question	9/11/2009
★	Hillary Clinton	UNGA	9/19/2009

This is the initial implementation of the email data display.

- **Word cloud:**
  Another important piece of information that we wanted to display was the people most contacted regarding particular topics. We chose a word cloud to represent this information where the color and size channels describe the frequency of interactions by an individual regarding a topic. The main challenge in creating this visualization was figuring out how to avoid overlaps between different names and it required a lot of parameters tweaked.
  Some notable features of the visualization are Bill Clinton being present in almost all topics and former National Security Advisor to Bill Clinton, Anthony Lake showing up in topics of Benghazi Attacks and Foreign Policy. Below is a screenshot of the word cloud visualization:



IMPLEMENTATION:

The Statistical language R is used for preprocessing, data cleaning and finally Topic modeling using the Latent Dirchlet Allocation algorithm.

The visualizations have been implemented using D3, JavaScript, HTML and CSS. We have modularized the code for the different visualizations, and hence each visualization has its own JavaScript file that has methods for every task that needs to be done for implementation of any visualization. We have implemented the below mentioned files:

- **Script.js** : This file is used to perform the tasks involved in initializing the content such as loading data and defining instances for all the JS files by making calls to constructors. The methods in this js file are:
    - startHere() : This is the method that reads all the csv files. We use a d3 queue here considering the huge amount of data to read. Only after the completion of reading all the files the dataLoaded() method is invoked.
    - dataLoaded() : In this method, the data read from the CSVs is copied to appropriate JS variables.
    - Init() : In this method, all the javascript files are instantiated by making calls to their constructers. Also the event handling method is implemented here which is invoked when there are any selections made in the force directed graph. The selection, which is the topic name is then sent to the other JS files to change their data to be displayed based on the selection.
- **WordCloudVis.js** : This file handles the word cloud implementation. The instance to this file is created in the Script.js and the appropriate data is passed. Within this file, the data received from the Script.js is filtered to have data in the form of the count of the number of times a sender interacts with Hillary Clinton in any specific topic.

This data is then displayed appropriately in the form of word cloud using appropriate methods.

- **BubbleChartVis.js** : This file handles the implementation of Force Directed Graph. The instance to this file is created in the Script.js and the appropriate data is passed. Within this file, the data received from the Script.js is filtered to have data in the form of names of topics and the 9 most frequent terms in each topic. This data is then displayed appropriately in the form of a labelled force directed graph using appropriate methods.
- **EmailTextVis.js :** This file handles the implementation of Email Text Display. The instance to this file is created in the Script.js and the appropriate data is passed. Within this file, the data received from the Script.js is filtered to have data in the form of emails assigned to different topics by taking the maximum probability with which the data occurs within any topic. This data is then displayed appropriately in the form of a Gmail like layout using appropriate methods.

All the above mentioned files for the visualizations have common methods that are stated below:

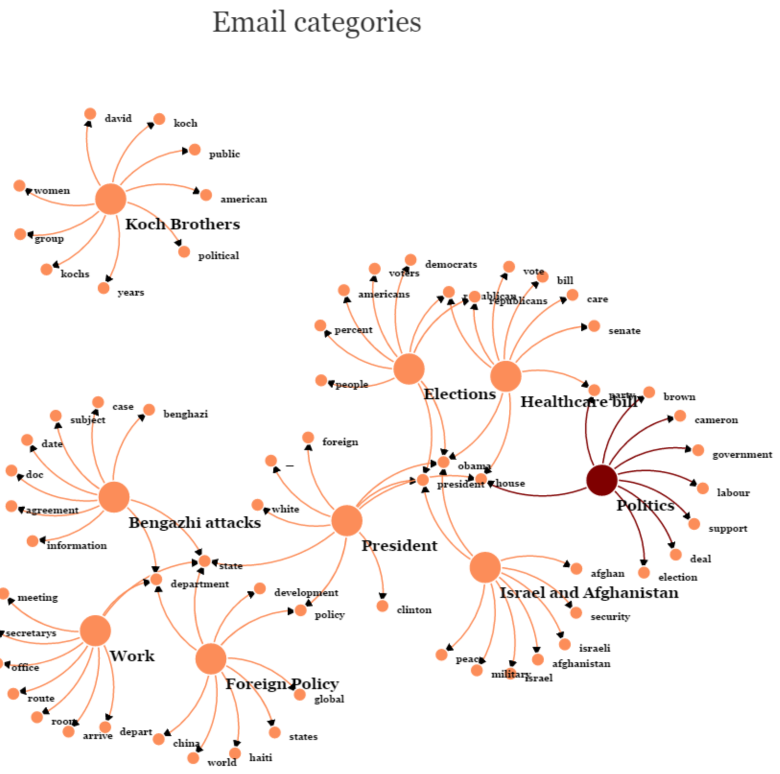
- Constructor : All the JS files have a constructor that takes in the data from script.js and assigns it to the variables specific to that visualization which have scope throughout the file.
- wrangleData() : This method is used to take the data that is inputted to the file, filter and convert it to an appropriate format that is needed for the visualization.
- initVis() : This method is used to initialize all the elements of the svg once the data wrangling is completed.
- updateVis() : This is central to the every visualization. It is in this method that the structure of the entire visualization is defined and the data binding to d3 is done. With there is a change in data to be displayed, this method is responsible for updating the visualization.
- onSelectionChange() : This method is triggered whenever there is any selection change done in the force directed graph. This modifies the 'topic_selected' variable that is central to data wrangling and then makes a call to the dataWrangle() function that first updates the data and then makes a call to updateVis() method.

INTERACTIONS WITHIN THE VISUALIZATIONS:

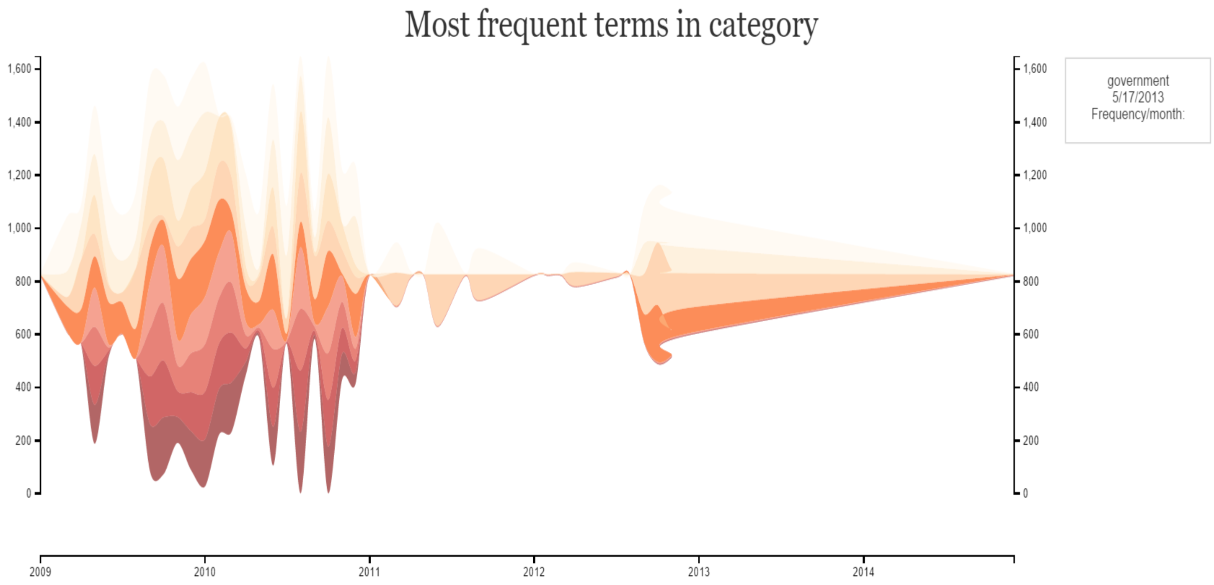
The main visualization element is the force directed graph at the top which contains the most frequently occurring words for each topic. When a particular topic is chosen from this graph, the stream graph gives the temporal relationship with the most frequently occurring words within the topic selected. Each of the individual streams represents a word in the topic and the size of the stream depicts the frequency of that word used per month. In the third visualization which is a word cloud, users can also see the most frequently mailed people in that topic. To the end, we have an email text box which updates to show the list of emails concerning the topic. The user can also view the full content of the email by clicking on them. Below are the screenshots showing interactions between the visualizations stated above:

Scenario 1:

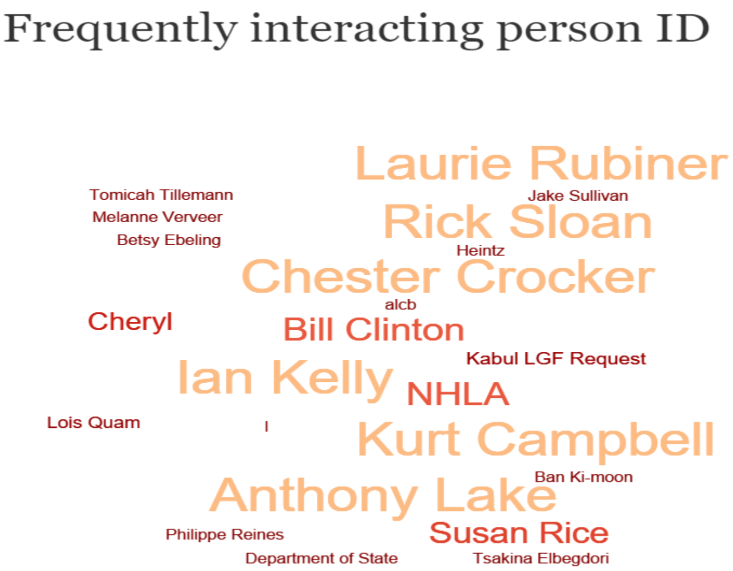
1. Selecting the topic Politics in the first visualization:



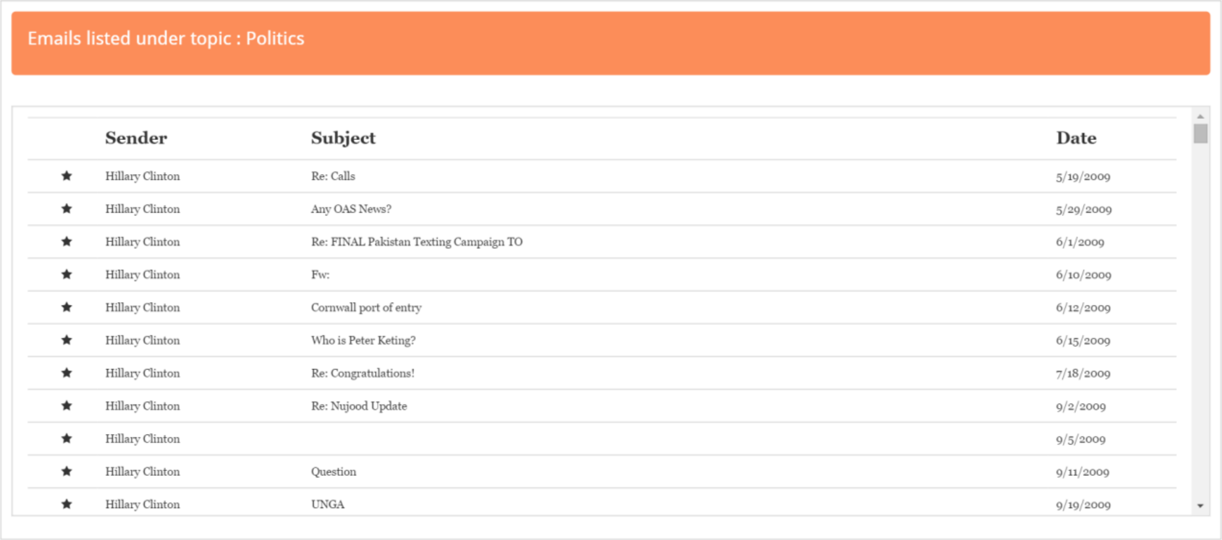
2. Stream graph for ‘Politics’:



3. Word cloud for ‘Politics’:



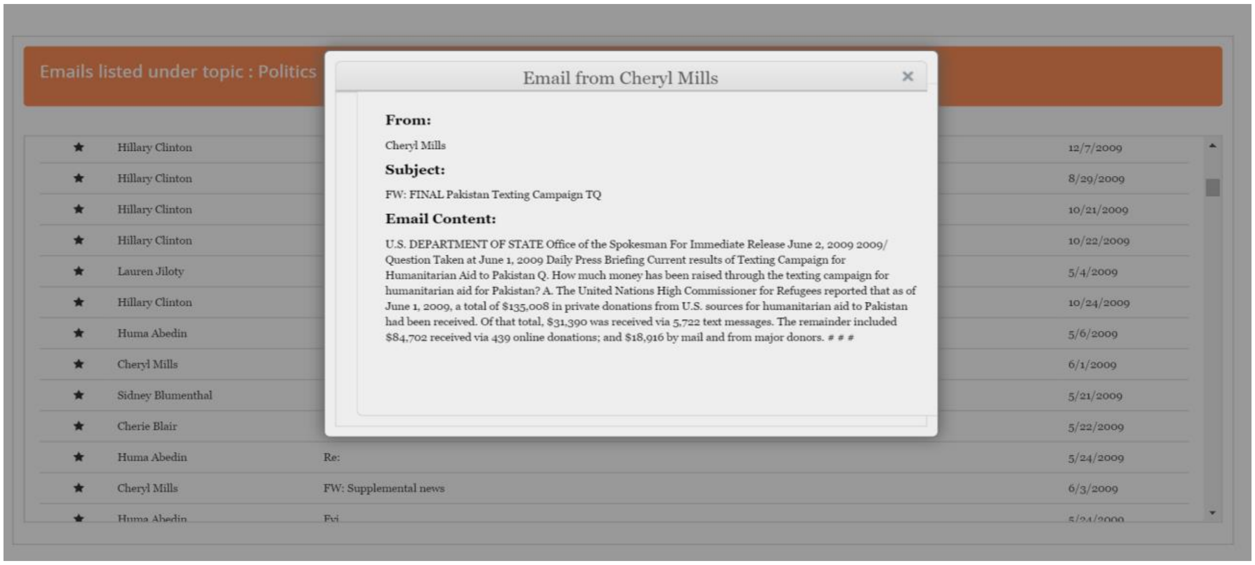
4. Email data for the selection:



Emails listed under topic : Politics

	Sender	Subject	Date
★	Hillary Clinton	Re: Calls	5/19/2009
★	Hillary Clinton	Any OAS News?	5/29/2009
★	Hillary Clinton	Re: FINAL Pakistan Texting Campaign TO	6/1/2009
★	Hillary Clinton	Fw:	6/10/2009
★	Hillary Clinton	Cornwall port of entry	6/12/2009
★	Hillary Clinton	Who is Peter Keting?	6/15/2009
★	Hillary Clinton	Re: Congratulations!	7/18/2009
★	Hillary Clinton	Re: Nujood Update	9/2/2009
★	Hillary Clinton		9/5/2009
★	Hillary Clinton	Question	9/11/2009
★	Hillary Clinton	UNGA	9/19/2009

5. Email pop up for the display of email content on click:



Emails listed under topic : Politics

Email from Cheryl Mills ×

From:

Cheryl Mills

Subject:

FW: FINAL Pakistan Texting Campaign TQ

Email Content:

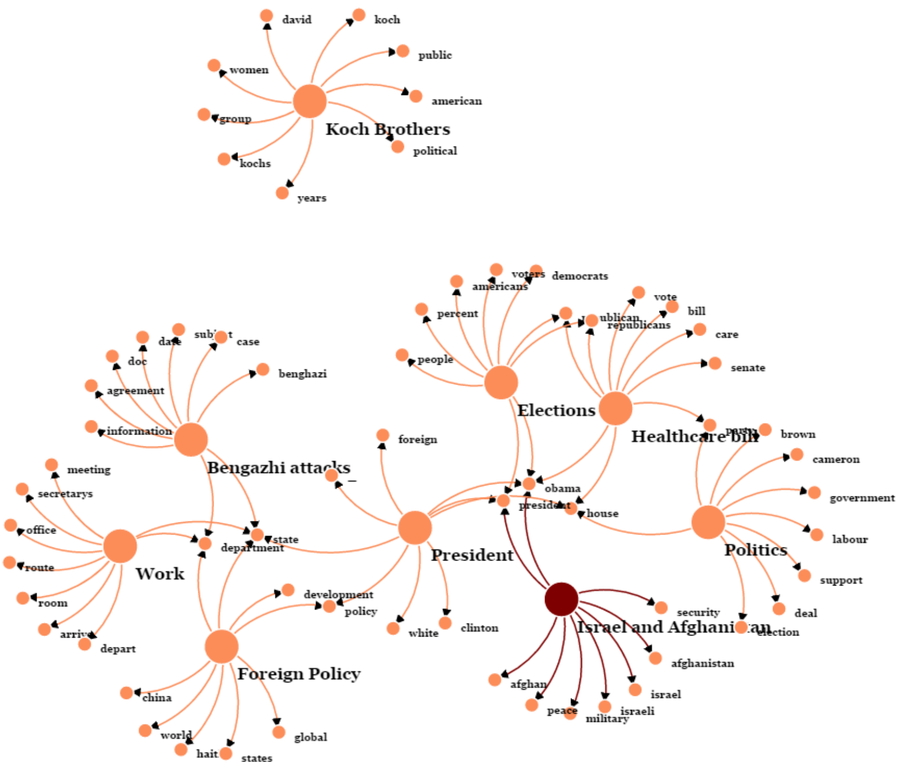
U.S. DEPARTMENT OF STATE Office of the Spokesman For Immediate Release June 2, 2009 2009/ Question Taken at June 1, 2009 Daily Press Briefing Current results of Texting Campaign for Humanitarian Aid to Pakistan Q. How much money has been raised through the texting campaign for humanitarian aid for Pakistan? A. The United Nations High Commissioner for Refugees reported that as of June 1, 2009, a total of $135,008 in private donations from U.S. sources for humanitarian aid to Pakistan had been received. Of that total, $31,390 was received via 5,722 text messages. The remainder included $84,702 received via 439 online donations; and $18,916 by mail and from major donors. # # #

★	Hillary Clinton		12/7/2009
★	Hillary Clinton		8/29/2009
★	Hillary Clinton		10/21/2009
★	Hillary Clinton		10/22/2009
★	Lauren Jiloty		5/4/2009
★	Hillary Clinton		10/24/2009
★	Huma Abedin		5/6/2009
★	Cheryl Mills		6/1/2009
★	Sidney Blumenthal		5/21/2009
★	Cherie Blair		5/22/2009
★	Huma Abedin	Re:	5/24/2009
★	Cheryl Mills	FW: Supplemental news	6/3/2009
★	Huma Abedin	Fvi	5/24/2009

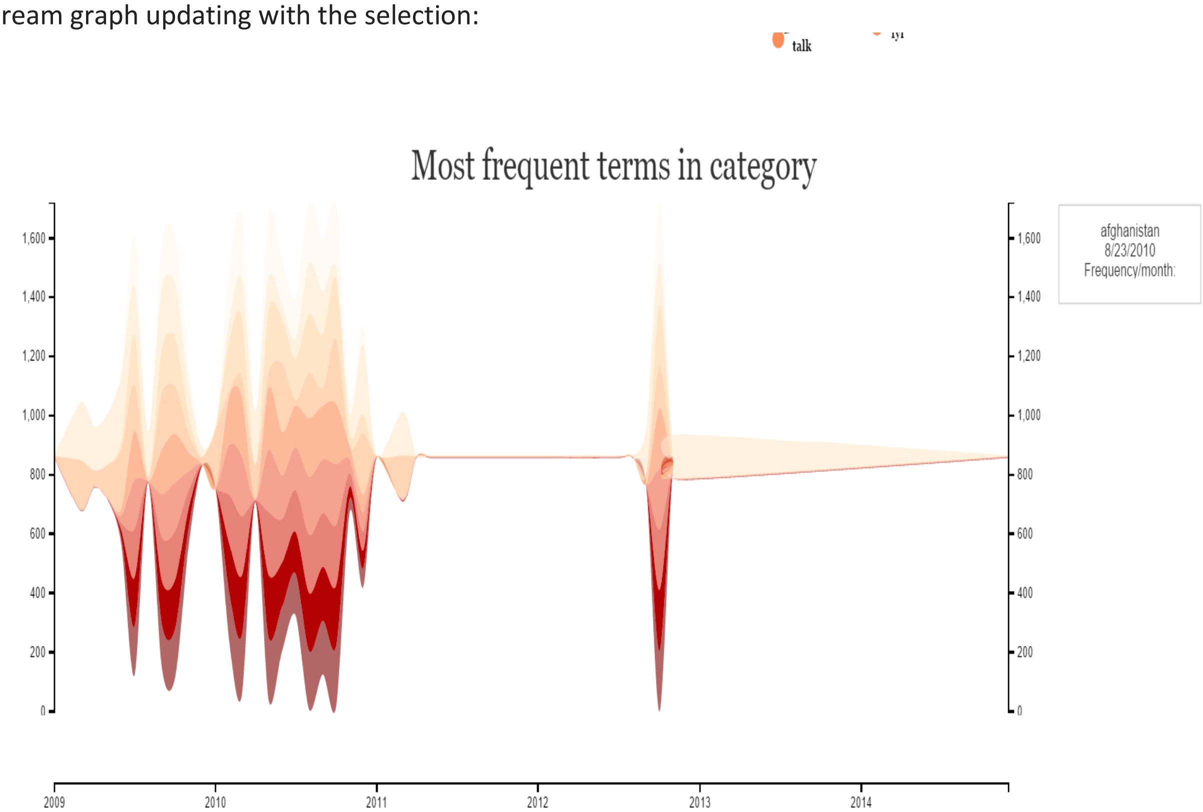
Scenario 2:

1. Selecting topic 'Israel and Afghanistan':

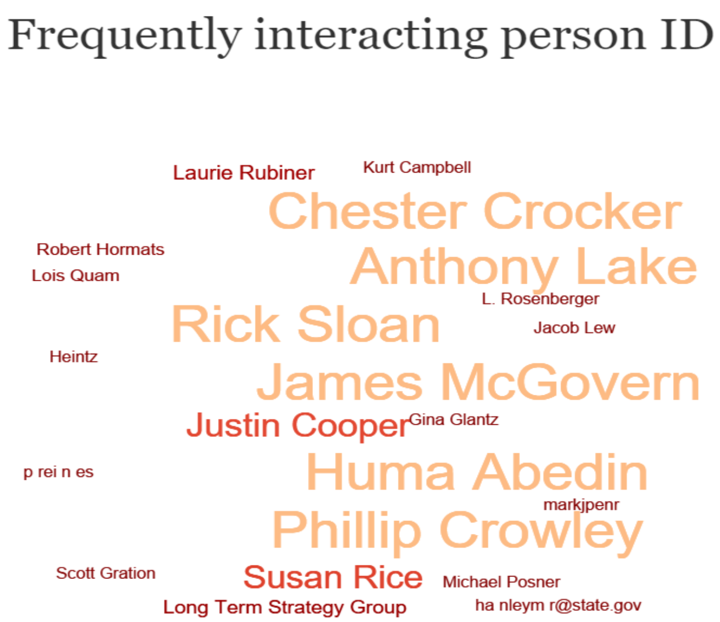
Email categories



2. Stream graph updating with the selection:

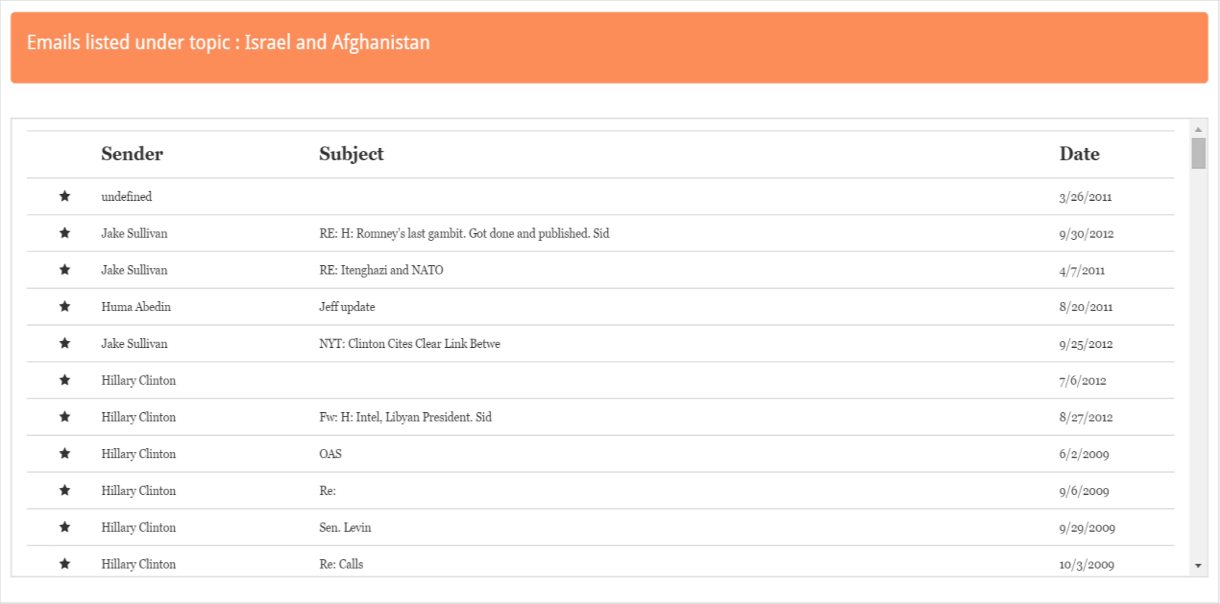


3. Word cloud for the selection:



Frequently interacting person ID

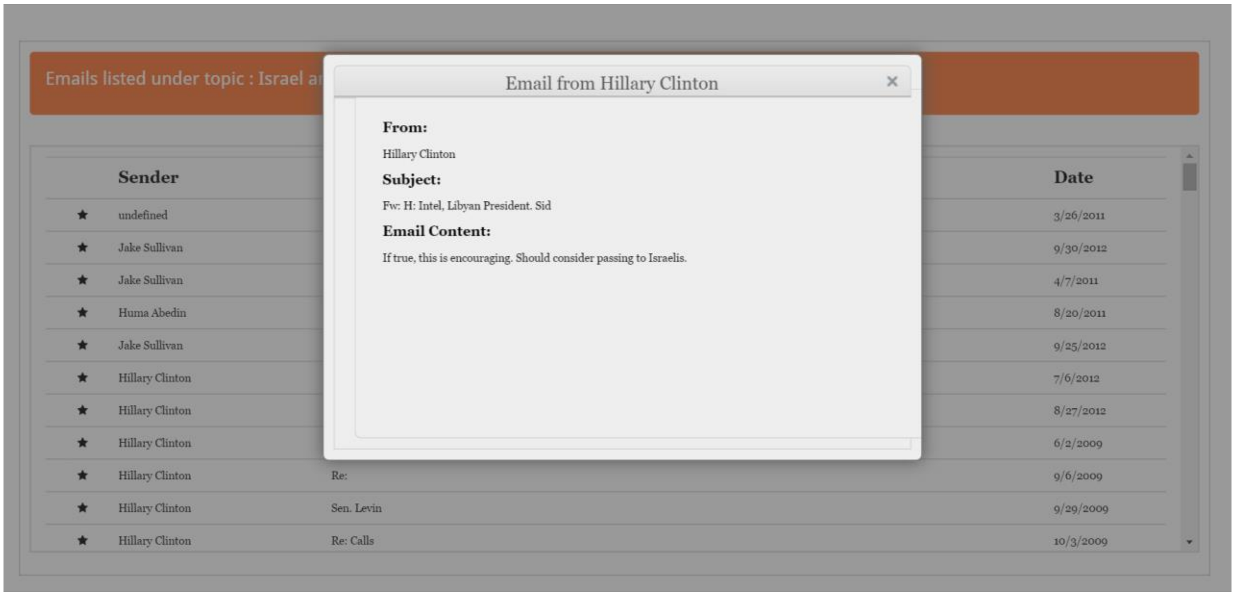
4. Emails display for the selection:



Emails listed under topic : Israel and Afghanistan

	Sender	Subject	Date
★	undefined		3/26/2011
★	Jake Sullivan	RE: H: Romney's last gambit. Got done and published. Sid	9/30/2012
★	Jake Sullivan	RE: Itenghazi and NATO	4/7/2011
★	Huma Abedin	Jeff update	8/20/2011
★	Jake Sullivan	NYT: Clinton Cites Clear Link Betwe	9/25/2012
★	Hillary Clinton		7/6/2012
★	Hillary Clinton	Fw: H: Intel, Libyan President. Sid	8/27/2012
★	Hillary Clinton	OAS	6/2/2009
★	Hillary Clinton	Re:	9/6/2009
★	Hillary Clinton	Sen. Levin	9/29/2009
★	Hillary Clinton	Re: Calls	10/3/2009

5. Pop up to display email content on selection of any email for the category Israel and Afghanistan:



Emails listed under topic : Israel a

Email from Hillary Clinton

From:

Hillary Clinton

Subject:

Fw: H: Intel, Libyan President. Sid

Email Content:

If true, this is encouraging. Should consider passing to Israelis.

	Sender		Date
★	undefined		3/26/2011
★	Jake Sullivan		9/30/2012
★	Jake Sullivan		4/7/2011
★	Huma Abedin		8/20/2011
★	Jake Sullivan		9/25/2012
★	Hillary Clinton		7/6/2012
★	Hillary Clinton		8/27/2012
★	Hillary Clinton		6/2/2009
★	Hillary Clinton	Re:	9/6/2009
★	Hillary Clinton	Sen. Levin	9/29/2009
★	Hillary Clinton	Re: Calls	10/3/2009

EVALUATION:

- **Insights from the visualization:**
    - The visualizations can answer a wide variety of questions such as “What are the most important topics discussed by Hillary Clinton in general?”, “What word in a particular topic she seems to talk more in a particular month of interest?”, “What is her emails content in topic named “Koch brothers?”.
    - The visualizations also give us some interesting insights such as that she considers some people namely Bill Clinton, her personal staffer Huma abedin extremely important because they appear in the word cloud for multiple topics and with high frequency which is depicted by the large size of their names in the word cloud.
    - The presence of the email textbox actually serves us two ways. It not only gives us a way to directly validate our results by clicking on the emails and seeing the content but also it will be interesting for users to check on the emails if they find the subject of the email interesting.

- **Effectiveness and Ways to improve:**
    - The visualizations are clean and neat and they don’t require any domain knowledge from the part of audience. With the help of efficient thresholding we made sure that the visualizations don’t overwhelm the audience by displaying too much data.
    - They are also effective in what they do and with the preprocessing part done priorly the visualizations are as slick as possible with minimal computation required from the user side.
    - If we had more time, we would like to experiment with spiral graph in the place of word cloud. The reason we chose not to for our project is because of it’s more complex

implementation and also we thought that it will not be as effective as the word cloud if the number of words are more and the each phrase may have more than two words which would be hard to see in the spiral graph.

- Also, we would have further extended the project by also considering other important data present in our dataset and would have done a visualization which might describe her sentiment towards different nations changing over time. We would have implemented a stream graph for this purpose and chose to display her sentiment towards the top-10 economically and militarily powerful nations.