DSCI 550 Project 3 - Report

Team Name: MIMECRAFT

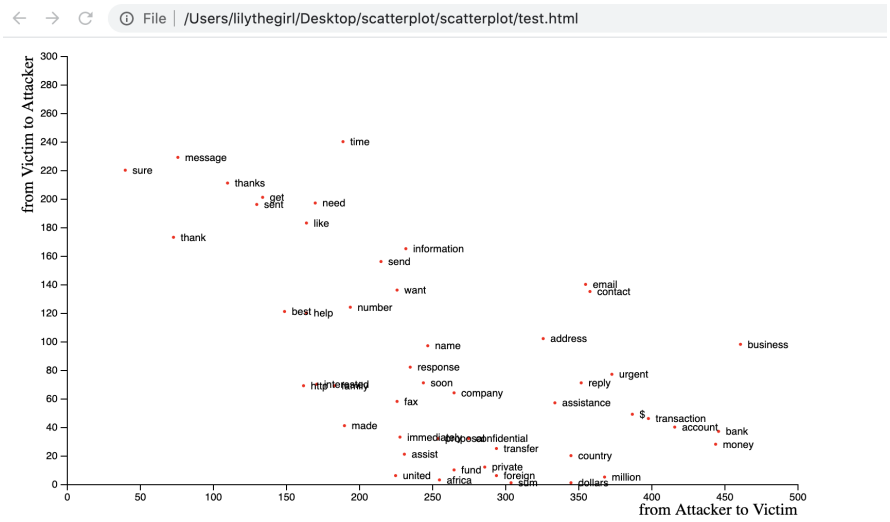
Team Member: Joshua Huang, Saumya Shah, Sungho Lee, Xinran Liang, Yue Liu

Five D3 Visualization Types

The core standard to choose a visualization mechanism is suitability and clarity. We believe that the visualization mechanism plays an important role in representing the data structure and data relationships. Taking the output from assignment 1 and 2, we extracted tsv, csv and json files out, building the data script for d3 visualization. We pick 5 types of visualizations: scatterplot, pie chart, heatmap, zoomable treemap and bar chart.

Scatterplot

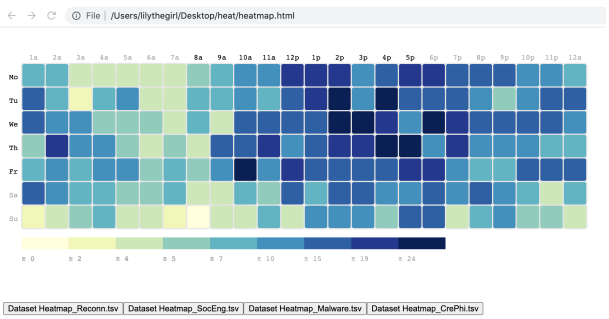
From all high-frequency words, which emerged in the email bodies, we reduced some meaningless words, like is/are or you, and picked 50 ones to count their appearance in both attacker-to-victim and victim-to-attacker emails.
From the visualization, words like “send”, “help” and “best” have similar frequency in both victim and attacker emails. In the attacker emails, words related to money, e.g. business, bank, account, transaction, have the heaviest attributions. In the victim emails, highly frequent words would be “message”, “sure”, “thanks”. Among all these words, attacker-to-victim ones take the dominance, whose amount is almost 1.5 times of the victim-to-attacker emails.
From this, we can further tell which words are more likely from emails in one direction. This would be meaningful for the reverse email type detection in the future.



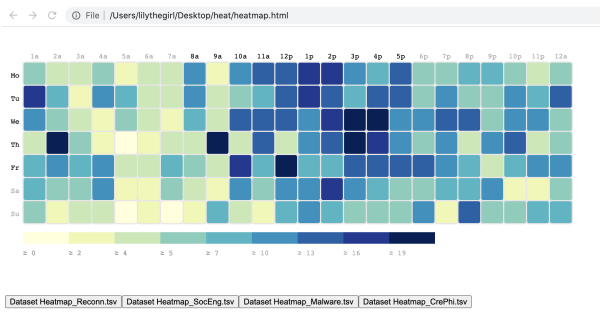
Heatmap

We chose the heatmap to see if there is any pattern on sending time for each attacker type. Heatmap shows the frequency of the fraudulent email over time and day of week. Row dimension represents a day of week and the column dimension represents an hour. We splitted the visualization into four based on the attacker type. As shown below, attackers related to reconnaissance mostly sent an email during weekdays and during day time (around 10 a.m. to 7 p.m.). Similar pattern was found from the Attackers related to Social Engineering but with less density. Attackers related to Malware and credential phishing had less frequency during day time and weekdays than reconnaissance.

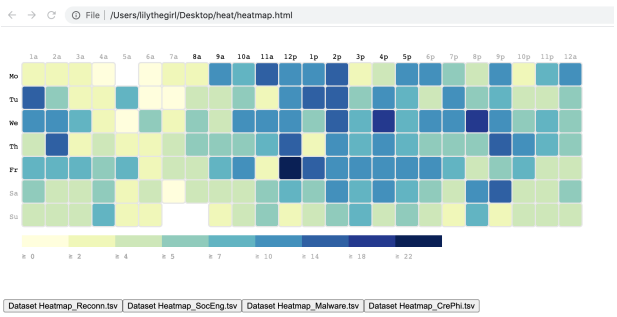
reconnaissance



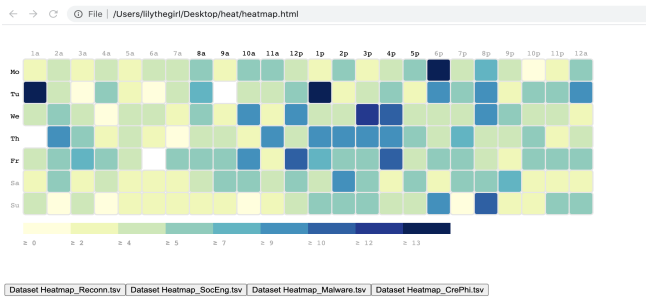
Social Engineering



Malware

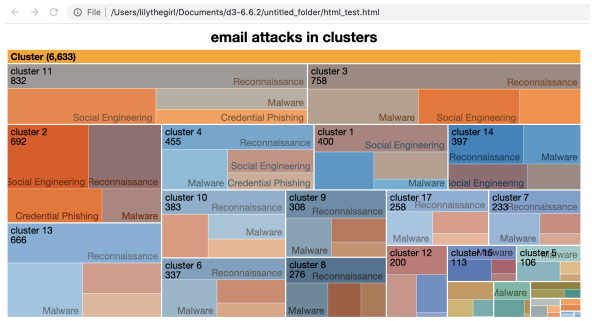


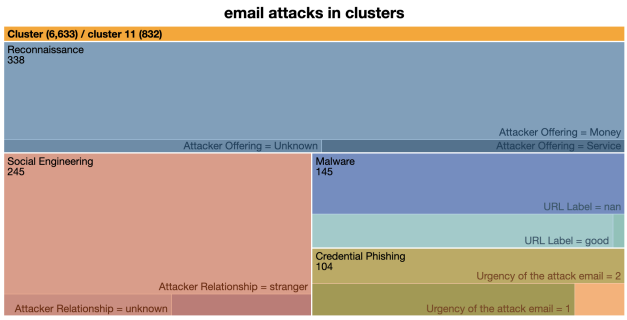
Credential Phishing

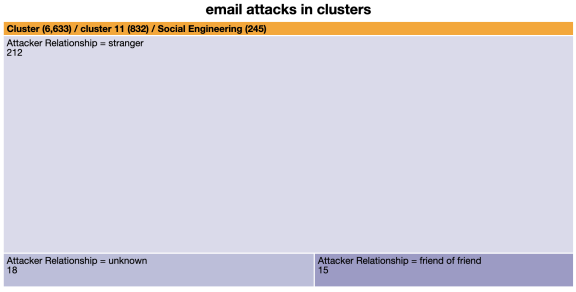


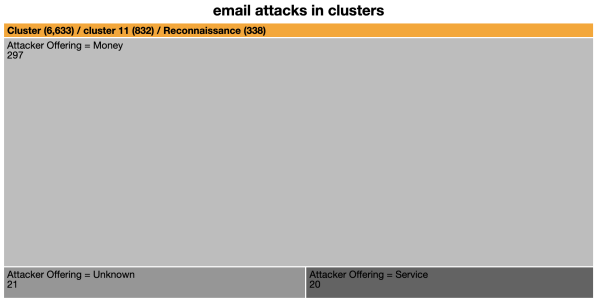
Zoomable TreeMap

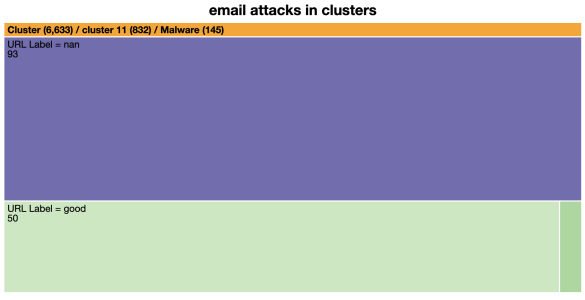
Inside the zoomable treemap, the initial state shows 17 different clusters calculated from Jaccard Similarity. If users click in a certain cluster, then the graph will show 4 attack types' count inside the current cluster. By clicking on a certain attacker type, users will know a more detailed summary based on features in the attack type. The reason that we choose zoomable treemap is this technique can show the detailed distributions of emails under different attacker types. What's more, the colors and shapes can offer the audience more straightforward understanding about the data structure.

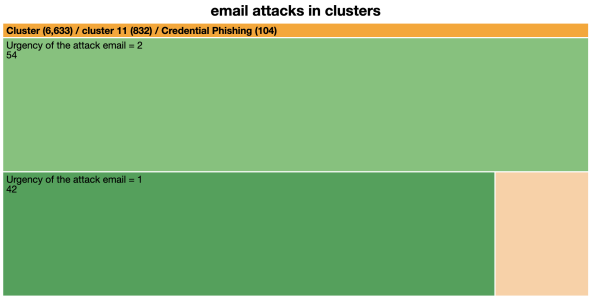






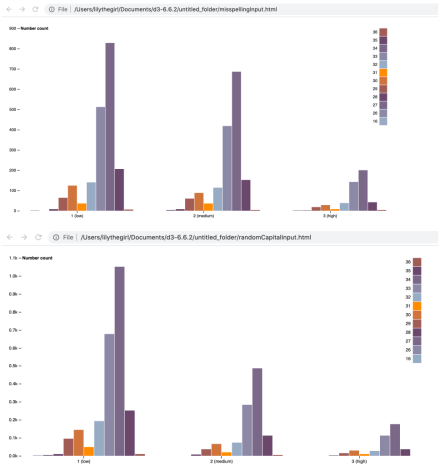






Grouped bar chart

We used grouped bar charts to show the distribution of the email attacks. It's pretty surprising that both attacker for the misspelling and random capitalizations showed similar, almost exactly the same, distribution patterns. As a result, it's pretty precious that different features of email attack do not have a significant relationship with the age group. These bar charts are displaying the level of Misspelling and Random Capitalization prevailing across different age groups. The y axis represents the count of emails and the x-axis represents the level of misspelling or random capitalization present (3 being the highest). The coloured panels represent the age group distribution as shown in the label on the right.



Pie Chart using ElasticSearch

This pie chart shows the URL Classification based on URL labels : 'good' or 'bad'. We chose Pie Chart to prove there is a significant URL classification difference between the good URL and the bad URL. Using an additional data set we have evaluated if a url is classified under a 'bad' category (meaning it's a known spam link) while the others are categorized 'good'. We also have a category of url types and we analyze their distribution with url labels. This visualization can clearly indicate dominance of some url types under different labels. The legends combined with the count also tell people exactly how many urls consist of a type.

According to this chart, we can see that the Arts is the major part of the good URL while arts is one of the lowest parts of the bad URL. To build this we have used the aggregations method from Elasticsearch. We have leveraged Logstash to ingest our data into elasticsearch index and have written an aggregation query to request the data from index and show the pie charts.

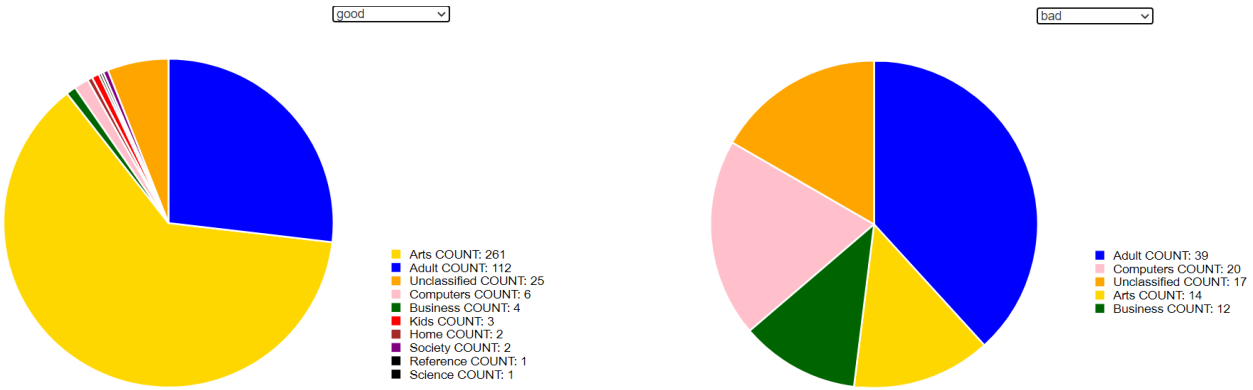
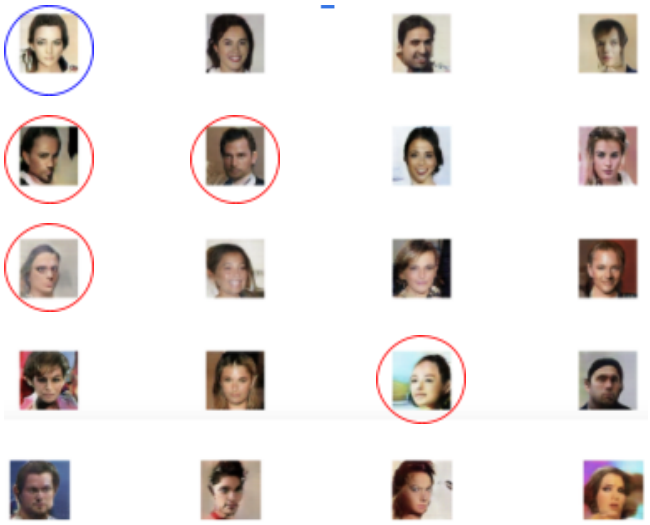
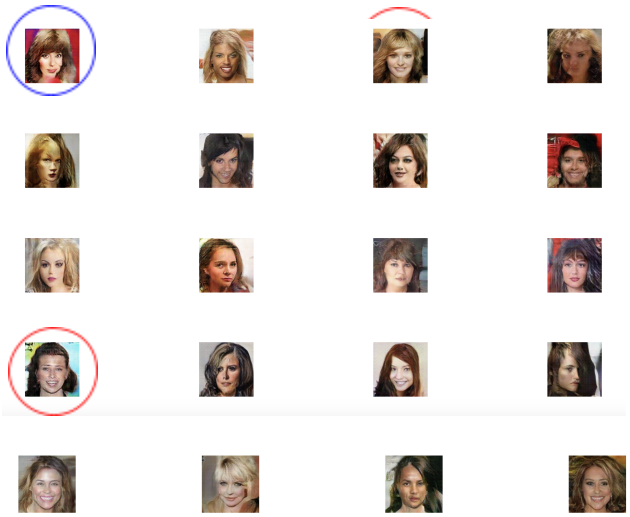
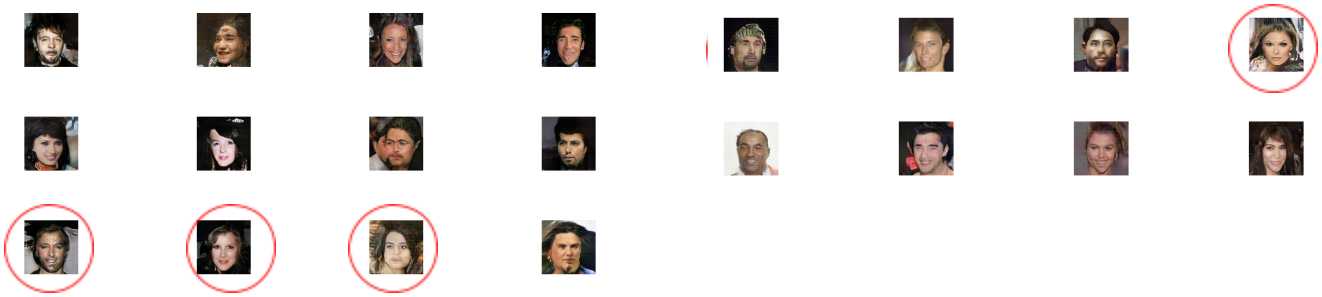


Image Space

Taking the fake attacker images we generated in assignment 2, we used the docker version Image Space to build a ImageSpace search index. The core function is to search the image forensics and similarity(SMQTK).
By using docker, users can easily build containers for Image Space, ImageCat and SMQTK. The procedure may bring some troubles to Windows system users but work well for Mac system users. Solving docker memory problems for uploading folders with large amounts of pictures takes some time, and the solr port can't be accessed directly causing a little difficulty. Overall, the final ImageSpace search page has very friendly UI designs and is easy to use.

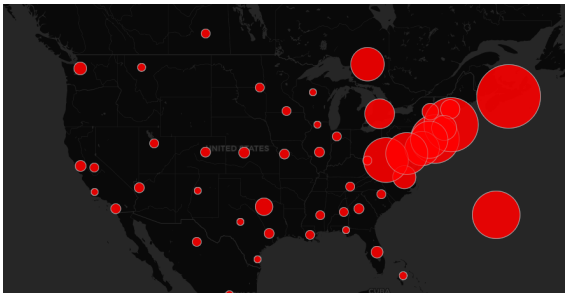
For the search result, in many cases, Image Space aggregates similar pictures by the shape of eyes, nose, mouth or face inside images. These partial features are actually easy to be ignored by humans. For example, before using Image Space, we won't think pictures marked in Ex1 are similar. The first reason would be figures' different genders; besides, their different expressions/moods also distract humans. However, after the results shown on the webpage, we could figure out that these figures have a common feature: the V-shape face. Ex2 and 3 then show how shapes of nose, mouth and eyes in the input picture bring the output pictures together. All these findings happened after Image Space returned some results.

Ex1:(the searched picture's surrounded by a blue circle)	Ex2:(the searched picture's surrounded by a blue circle)
	

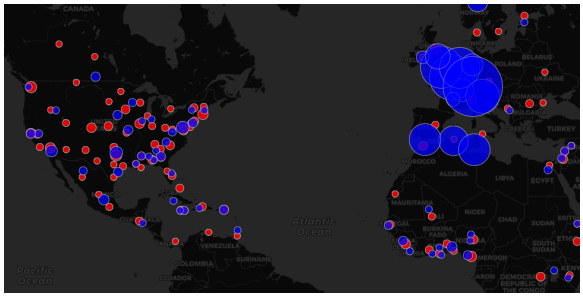
Ex3:(the searched picture's surrounded by a blue circle)


GeoParser

First of all, to execute the geo-parser, we have used the "EmailText" from featureTable_v2.tsv from HW2. Geo-parser automatically extracted location information such as country, county, or city name from the content of the email. One different aspect from HW1 is that one email content can have multiple locations on this map. So for example, if one email contains the location name "Chicago" and "Iowa", this email will be located in both Chicago and Iowa in geo-parser.



Fraudulent email distribution in the United States



Distribution of fraudulent email containing word "money"

We have looked through our geo-parser result carefully. What we could easily see was that most of the fraudulent emails are related to the United States and the most emails were distributed in the eastern part of the United States. And also when we searched for the word “money”, geo-parser showed a lot of emails containing the word “money” from Europe.

References

(didn’t use)ETLlib: https://github.com/chrismattmann/etllib
Contributors: Chris A. Mattmann

D3 - Zoomable Treemap Template from ganeshv’s Block 6a8e9ada3ab7f2d88022
http://bl.ocks.org/ganeshv/6a8e9ada3ab7f2d88022

D3 - Scatterplot with Multiple Series from Mike Bostock’s Block 3183403
https://bl.ocks.org/mbostock/3183403

D3 - Grouped Bar Chart from Mike Bostock’s Block 3887051
https://bl.ocks.org/mbostock/3887051

D3 - Day / Hour Heatmap from Tom May’s Block 5558084
http://bl.ocks.org/tjdecke/5558084

D3 - Pie Chart
https://www.d3-graph-gallery.com/graph/pie_changeData.html

ImageSpace: https://github.com/nasa-jpl-memex/image_space/wiki/Quick-Start-Guide-with-ImageCat
Contributors: Chris A. Mattmann

Geoparser: GitHub - nasa-jpl-memex/GeoParser: Extract and Visualize location from any file
Contributors: Chris A. Mattmann

Web Page design template inspired from: https://github.com/USCDataScience/ufo.usc.edu

Logstash: https://www.elastic.co/logstash

Contribution:

- **Joshua Huang:** Gathering information for five D3 visualization, research, set up and debug Image Space, designing web pages
- **Saumya Shah:** Using Elasticsearch for ingesting data using logstash and data visualization, Web page design to show the 5 charts and elasticsearch implementation using client call, gzipping and tarring up of elasticsearch index
- **Sungho Lee:** Generating input files for all five D3 visualization, Executing geoparser for visualizing location, web page design
- **Xinran Liang:** Generating D3 visualizations, Building Elasticsearch chart with logstash, Ingesting and implementing Image Space with docker, tarring up index files from docker CLI, designing web page
- **Yue Liu:** Generating D3 visualizations, Ingesting and implementing Image Space, tarring up index files, designing web page