BSChuang / ML-WEBH

Code Issues Pull requests Actions Projects Wiki Security Insights

gh-pages

ML-WEBH / index.md

ethan-w-roland Update index.md History

4 contributors

Raw Blame

99 lines (57 sloc) | 11.8 KB

Exposing Fake COVID-19 News

Introduction

Social media plays an important role in disseminating Covid-19 related information. However, despite its importance, there aren't strong controls on what information gets spread. The goal of our project is to create a machine learning model to detect Covid-19 related “fake news”. This includes information that is factually inaccurate or even dangerous.

Data scientists have created fake news classifiers in the past, but we are not aware at any efforts to classify information related specifically to Covid-19. Creating a classifier tailored to check social media posts about this topic will be extremely helpful in preventing the spread of healthcare misinformation online.

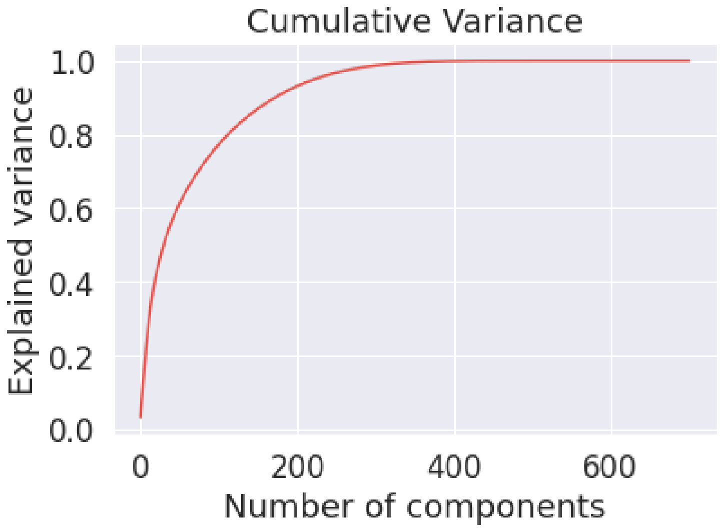
Methods

In the initial phase of the project, we focused on exploring the dataset using unsupervised techniques. This allows us to have a better understanding of our data, and to gain insights that may be useful when we proceed to the supervised learning step.

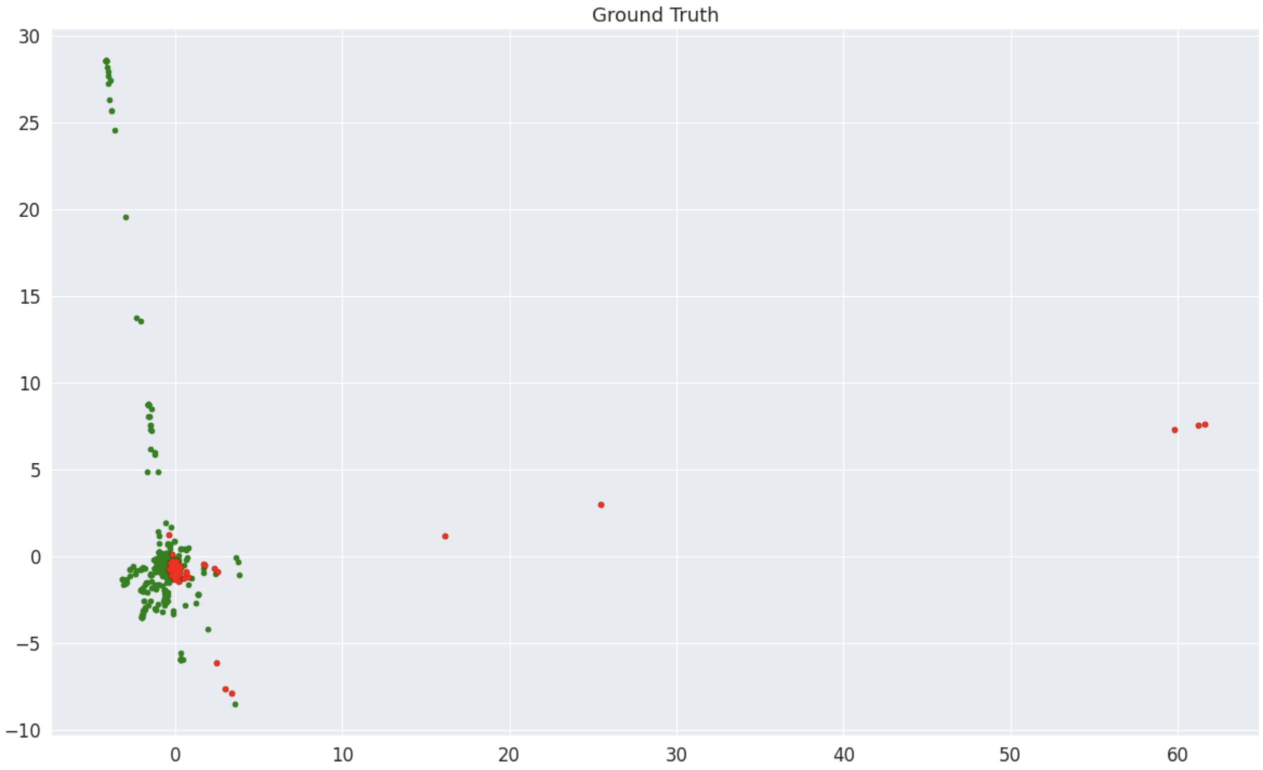
To obtain our initial dataset, we downloaded a set of Covid-19 related tweets which had already been labeled as true or false (decribed here). Using this dataset was more difficult than expected, since it only included tweet id's (reference numbers to an actual tweet). As a result, we had to create a web-scraper to search twitter for the tweet itself. Further, some tweets in the dataset had been deleted, requiring us to exclude those from the final dataset. After this initial data-cleaning stage, we were left with 1092 tweets to analyze. A simple analysis of the class imbalance showed that 12% were fake news, as compared to the 88% that were true.

We next performed a vectorization step to convert tweet text into numeric vectors. We used the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm as our vectorization method. Every unique unigram or bigram within the input text corpus became a distinct feature within out dataset. The value of that feature for a given tweet is dependent on the frequency of the word(s) corresponding to that feature within the tweet. This step also involved additional data cleaning. Here, we dropped all columns that were composed of English "stop words" - which are simple words like "the" or "is", which tend not to carry significant semantic value.

At the end of the vectorization step, our data contained 702 dimensions. We used PCA to create a representation of this data with a lower dimensionality. Specifically, we chose the minimum number of PCA components that would capture 99% of the variance within the dataset. We found that the original 702 dimensional representation could be reduced to only the first 314 PCA dimensions, while still explaining 99% of the variance. A visualization of the cumulative variance explained by each PCA dimension is shown below.

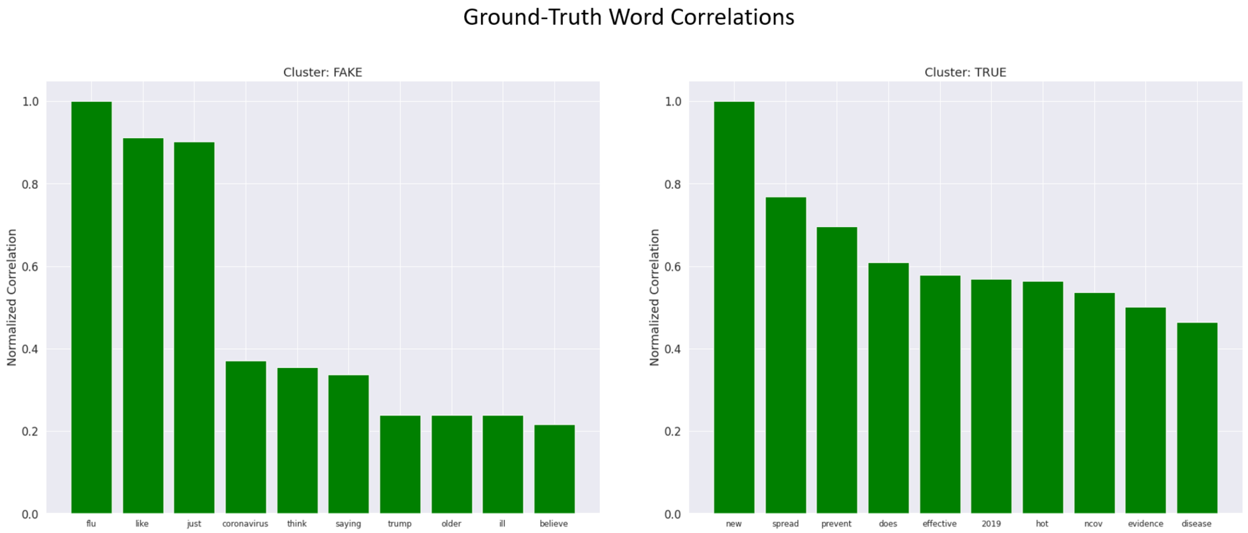


In order to gain an intuitive understanding of the dataset, we attempted to visualize the pre-existing label assignments. We employed two techniques. The first made use of a two-dimensional plot, where the first and second PCA dimensions are shown on the X and Y axis, respectively. Each tweet within the dataset is graphed on this plot, and colored according to it's truth value. This plot is shown below, with true tweets in green, and false tweets in red.



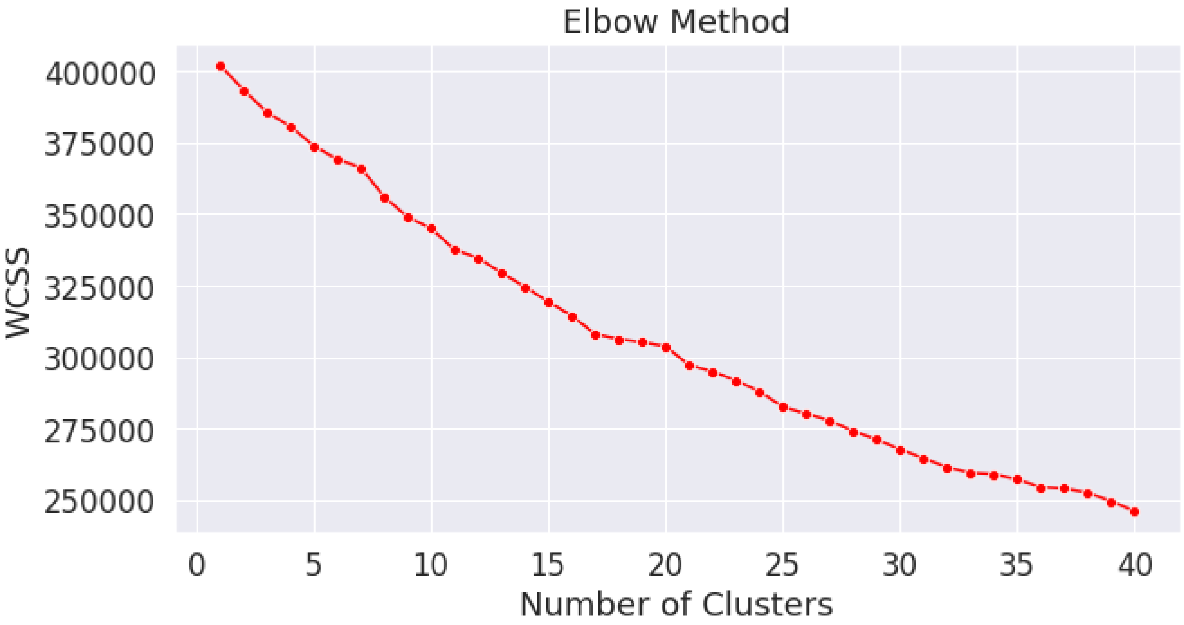
We were encouraged by this visualization, since it appears that each label forms fairly distinctive groups. This portends well for future supervised learning / classification efforts.

We also wanted to gain an understanding of the semantic differences between true and false tweets. To do this, we calculated the correlation between each class label, and all the single-word tf-idf features. We then sorted these tf-idf features based off of the correlation. In our visualization, we chose the top 10 mostly strongly correlated words for each class, and plotted their correlation strengths along the y-axis of a bar graph. We chose to normalize these correlations in order to better show the relative strengths of the relationships among all words shown. This plot is can be seen below.



This visualization allows us to get an inuitive understanding of the topics associated with each label. For instance, it appears that true tweets tend to focus on topics related to disease prevention and "evidence", while false tweets tend to contain comparisons to the common flu, and may contain referenes to political leaders.

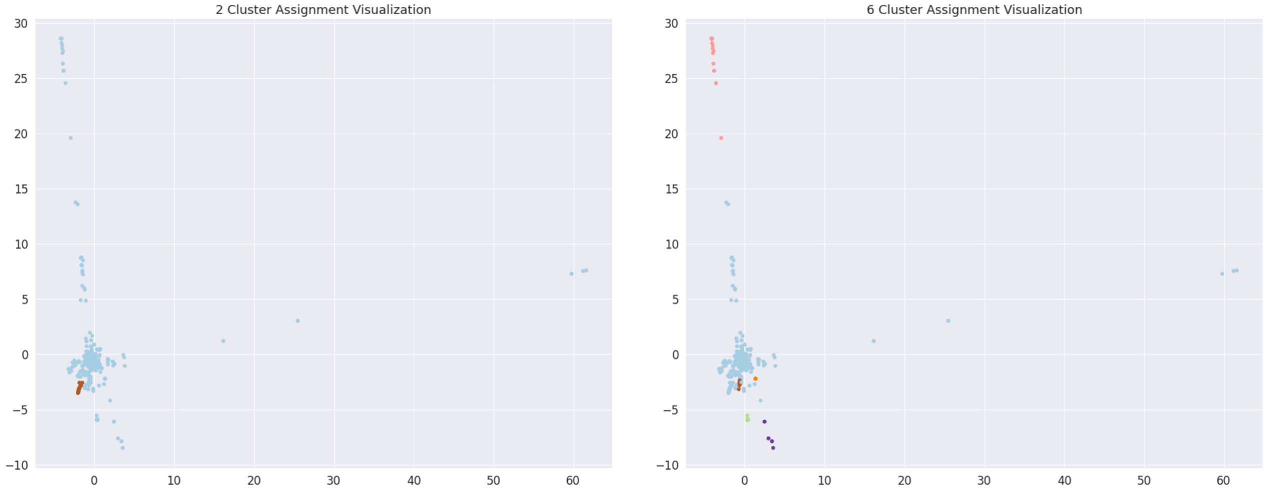
After the earlier dimensionality reduction step, we also wanted to apply unsupervised clustering techniques to our data. We first attempted to employ K-Means clustering. In order to determine the proper number of clusters, we attempted to apply the elbow method of optimization. A plot comparing Within-Cluster-Sum-Of-Squares (WCSS) to the number of clusters is shown below.



As can be seen, the decrease is fairly linear with respect to increasing cluster number. This may indicate that K-Means is not the best approach for this dataset. Likely, this stems from the shape of the data, which appears to have one core circular region, and two oblong outer regions. Since K-Means is best suited to circularly clustered data, it likely cannot properly cluster the two oblong regions. Nonetheless, we were curious about further exploring the K-Means results. We performed K-Means with both 2 and 6 clusters. We explored these results through similar visualizations from the exploration of the ground truth values.

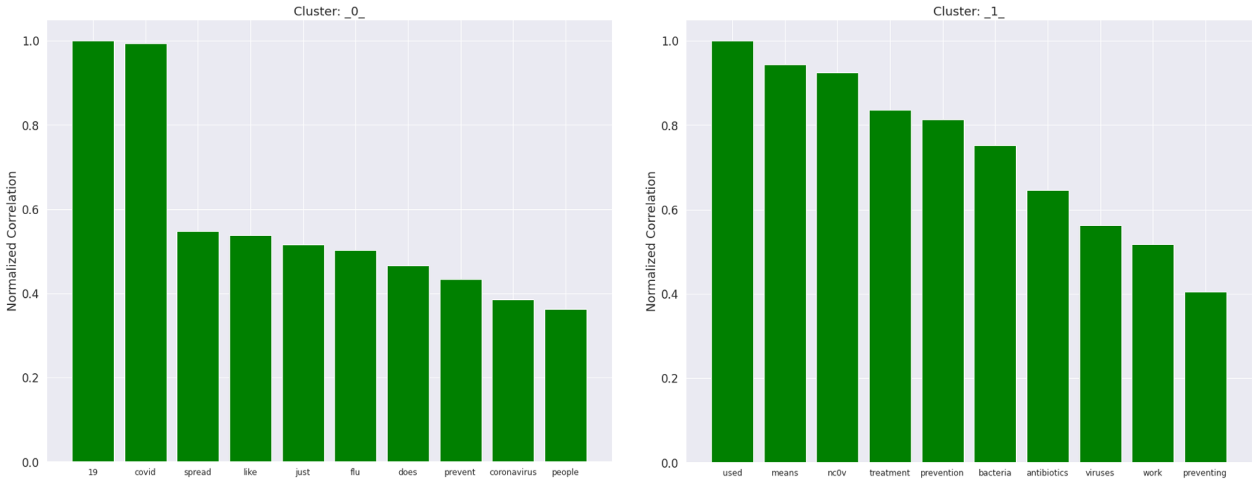
The two-dimensional PCA plots for both 2-cluster and 6-cluster assignments can be seen below.

K-Means Clustering

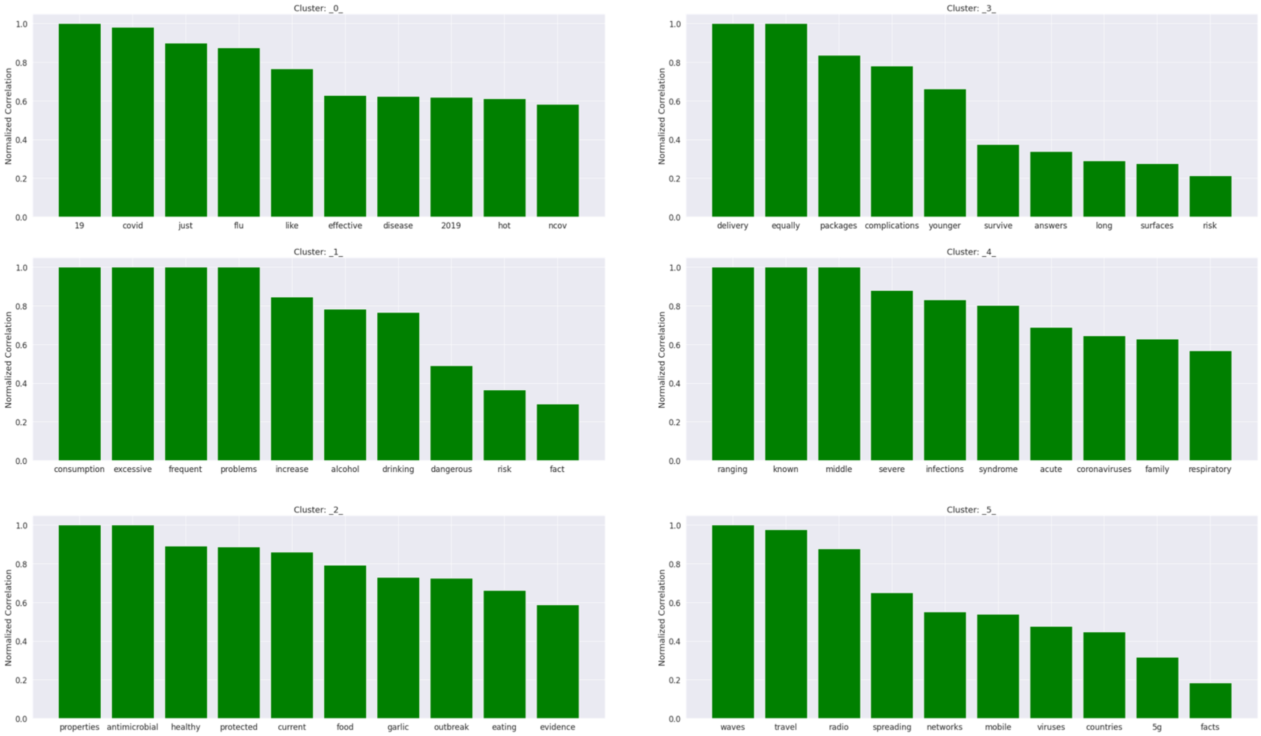


It's notable that these cluster assignments are visually quite different than the ground truth labels, likely indicating that the clusters produced by K-Means have created divisions of the original dataset that are semantically different than the ground-truth true/false division. We can visualize the semantic content of each cluster using the word correlation technique from earlier. The relevant bar charts are shown below.

2-Cluser K-Means Word Correlations



6-Cluser K-Means Word Correlations

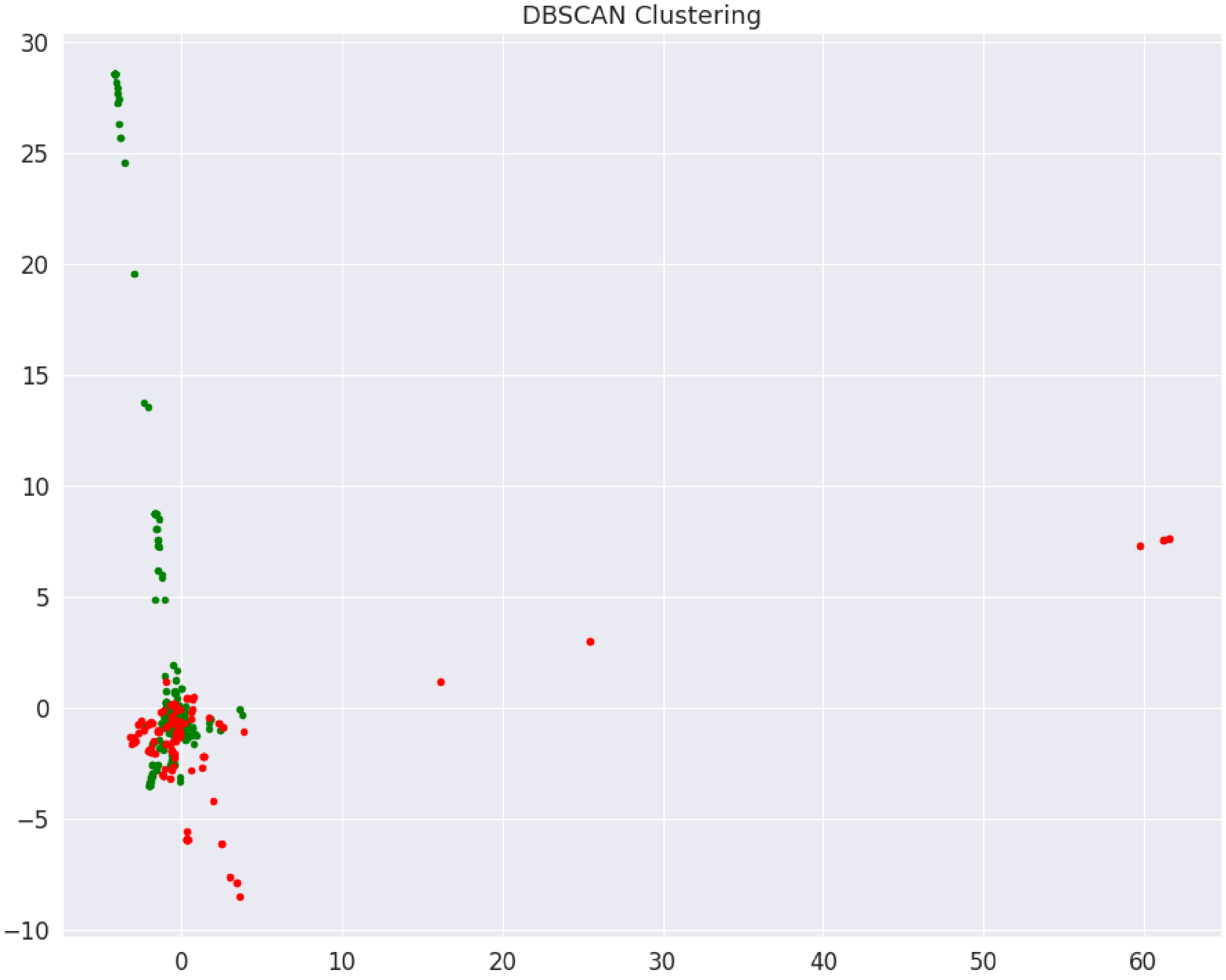


There are some notable takeaways from these graphs. Although the 2D plot of the 2-cluster K-Means result is quite different from the 2D plot of the ground truth, the semantic visualization shown above seems to indicate that the topics contained in the 2-cluster K-Means class assignments seem to closely mirror the topics contained by the ground truth labels. We can see that cluster **0** contains similar words to those contained by the "Fake" label, as demonstrated by words like "just" and "flu". Similarly, we can see that cluster **1** contains similar words to those contained by the "True" label, as demonstrated by words like "spread" and "prevent".

Additionally, the 6-cluster K-Means results are interesting because each cluster seems to represent a semantically distinct topic. For instance, it appears that cluster **5** is distinctly related to 5G mobile technology, likely reflecting tweets related to conspiracies that propose a link between 5G and covid-19.

In addition to K-Means clustering, we also we felt that DBSCAN clustering may produce interesting results. DBSCAN has two main hyperparameters: epsilon and min_samples. Epsilon controls how "far out" DBSCAN can look from an existing cluster assignment in order to claim an additional point. Min_samples controls the minimum number of samples required to for a cluster to count as one. We tuned this hyperparameters using a grid search to maximize classification precision relative to the ground truth, only considering hyperparameter values that resulted in a total of two DBSCAN clusters. This resulted in a precision of 86.9% when using epsilon = 32 and min_samples = 16.

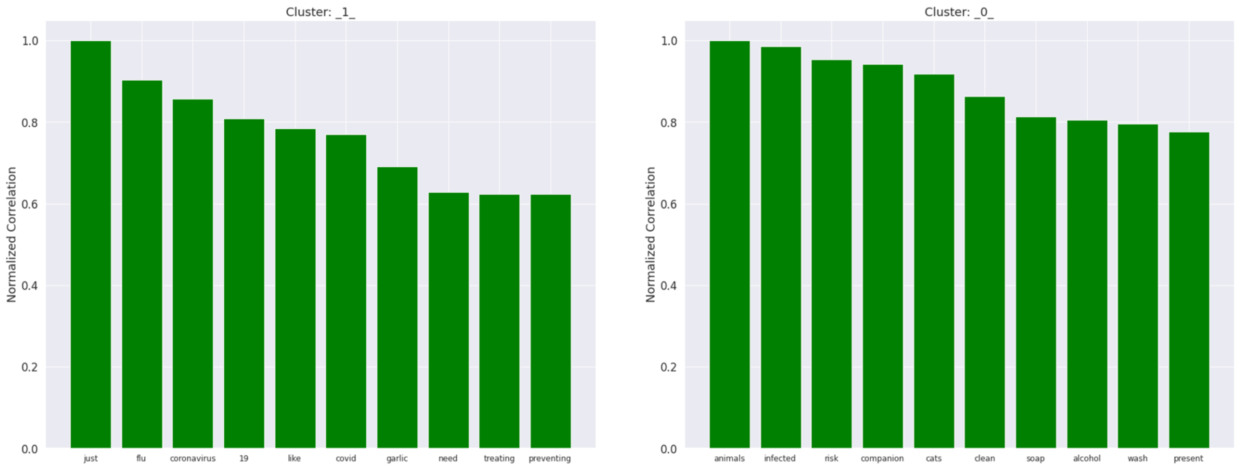
As with K-Means, we wanted to visualize our clustering results using the same methods as earlier. The 2D PCA plot of the DBSCAN cluster assignments is shown below.



Note that this plot has far more visual similarity to the plot of the ground truth labels, relative to the clustering produced by K-Means.

We also wanted to explore the semantic relationships of each cluster, which is shown in the word correlation graphs below.

DBSCAN Word Correlations



As expected the topics covered by the each DBSCAN cluster seem to closely mirror the topics covered by the ground truth groupings.

Results Summary

In the unsupervised portion of this project, we were able to complete all of our original goals and gain deep insights about the nature of our dataset. The three main tasks completed include:

- successfully production of numeric vectors from text using TF-IDF
- dimensionality reduction using PCA
- data clustering and visualization using DBSCAN and K-Means

Crucially, we met our goal for the unsupervised learning phase by successfully producing clustered data and comparing those clusters to ground truth values.

Looking forward, we can see that, while the supervised learning effort will likely produce useful results, the visual ambiguity in parts of the 2D PCA plot indicates that the system may have difficulty differentiating between true and false in some cases. This may help shape usage policy for our final classifier. We advocate it's use solely to flag potentially misleading tweets, which are then sent to a human for manual review. This approach can help avoid unnecesarry censorship.

As a check-for-success in the supervised learning portion of this assignment, we intend on producing a classifier that accurately differentiates between fake and true news, and producing metrics for its performance.

Discussion

Ideally, our project will successfully categorize posts as ‘fake news’ or ‘not fake news’. If we are successful, our project could be used to flag fake news posts on social media for review. This would involve some room for error, as fake/real news can use similar phrases. Human review would allow for false positives without immediately taking down real new posts. Finally, our model would help control the spread of fake news and misinformation on social media sites, improving trust by the user.

Our main outcome this phase was the production of vectors from our dataset, a process which included some data cleaning and dimensionality reduction using PCA. We found success with the K-Means and DBSCAN algorithm, which may also be useful for exploring our results during supervised learning portion of this project. In this unsupervised learning portion, the bar charts we generated helped us understand what features are most significant in our dataset. In the next phase of our project, we will similar bar graphs to visualize the output of our classifier. We also plan on training our classifiers using the output of our PCA implementation. We will also investigate the SVM and random forest algorithms for classification.

References

How Facebook Is Using AI to Fight COVID-19 Misinformation, Tekla S. Perry, https://spectrum.ieee.org/view-from-the-valley/artificial-intelligence/machine-learning/how-facebook-is-using-ai-to-fight-covid19-misinformation

Fake News Detection: A Deep Learning Approach, Aswini Thota, Priyanka Tilak, Simrat Ahluwalia, Nibrat Lohia, https://scholar.smu.edu/cgi/viewcontent.cgi?article=1036&context=datasciencereview

CoAID: COVID-19 Healthcare Misinformation Dataset, Limeng Cui, Dongwon Lee, https://arxiv.org/pdf/2006.00885.pdf

Links

Google Colab

Midterm Video