Detecting Deceptive Language in the Enron Email Corpus

Colby Carter, Maura Cullen and Clay Miller
W266: Natural Language Processing with Deep Learning
UC Berkeley School of Information
colby.carter, mcullen, cmiller11 @berkeley.edu

Abstract

In this paper, we attempt to test several semi-supervised learning methods to evaluate the potential for deception-detection in the now public Enron emails. Our motivation comes from the not-so-infrequent attempts by individuals and company leaders to mislead and cover up harmful information and deceit, whether those be insureds trying to profit from fraudulent claims or executives misleading investors in order to line their pockets. We draw upon several studies on deceptive language and expand on the usual N-gram features with semi-supervised learning in order to generate clusters of suspicious looking emails and topics similar to those identified by targeted heuristics.

1 Introduction

Deception ranging from individual acts of fraud to wider corporate corruption can hurt companies, employees and customers alike. At the largest scale, the discovery of fraud committed by a successful and influential company can even affect the entire economy as Enron Corporation did in 2001. Enron was an American energy, commodities and services company which had to file for bankruptcy in 2001 after committing accounting fraud and revealing corruption by top executives.

Monitoring and combing through emails or other written statements to detect fraud or unlawful business practices within companies is expensive and tedious to do manually. Automating the fraud detection process is difficult because codewords, hints, and obfuscations in emails and other written statements are used with the intent of not being discovered. We attempt to mimic the human process of reviewing emails to be able to flag those suspicious communications, hoping to achieve modest precision. To learn from a real occurrence of fraudulent behaviour masked within company communications, we will be using the public emails gathered from the Enron case. Our goal is to be able to identify emails that appear to obscure potentially-damning information or actions, which could be at least loosely connected to larger fraudulent activity. Our goal is to create a process of identifying suspicious of deceptive behavior which can be used in a more generalized fraud detection within written statements in other contexts, such as insurance claims or credit fraud.

2 Background

In our research, we have found literature that has constructed similar goals of identifying suspicious or deceptive behavior in written language, emails or even verbal speech, though none yet specifically on the Enron corpus, to our knowledge.

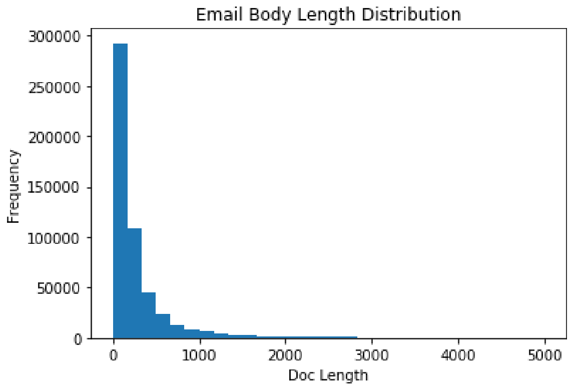
There have been several efforts to verify and label both written and verbal speech. Bachenko, Fitzpatrick and Schonwetter (2008) were able to tag 275 sentences in several criminal statements and depositions, plus the Enron congressional hearing, for truthfulness. They note “deception indicators” such as using choice verb tenses, “hedging” claims with qualifiers, and using extreme negative statements or exaggeration. In a similar and larger hand-tagging exercise by Feng et al. (2012), the authors find that using TFIDF-weighted unigram and bigram counts plus part-of-speech tags and features derived from sentence parse tree with a SVM classifier increases deception detection accuracy over the usual bag-of-words (BOW) approach. There have also been several similar studies to classify speech and topics specifically in emails: Cohen et al. (2004) sought to label email actions (e.g., “deliver” or “propose”), also using TFIDF-weighted features and part-of-speech tags.

Dredze et al. (2008) also pursued an email-focused effort using the Enron corpus, attempting to produce keywords in emails using unsupervised techniques. The authors’ approach included Latent Dirichlet Allocation (LDA), used here to create representations of topics in email mailboxes based on summary keywords. In our paper, we explore variants of LDA as originally developed by Blei, Ng and Jordan (2003). We also consider an approach of capturing topics with paragraph vectors, as developed more recently by Le & Mikolov (2014). These are trained very similarly to word vectors for language modeling, but with the benefit of producing a fixed-length vector of features unique to any given paragraph or document.

3 Methods

Our approach to identifying emails with suspicious phrases began with parallel efforts toward developing semi-supervised models: (1) with learnings from prior research on deceptive language, we explored the various types of emails in the Enron dataset (e.g. financial, personal, news reporting, hiring, etc.) and conducted targeted heuristics to identify and positively label emails with suspicious or information-masking language and phrases; and (2) guiding one of our algorithms toward email topics on specifically fraudulent programs within Enron, e.g. “condor”, “LJM” and “raptor.”

Our exploration began with a review of two-hundred randomly drawn email bodies from the first 50,000 examples in the dataset (which totals over half of a million, including a substantial number of redundancies with email threads and forwarded messages; a substantial effort was taken to preprocess the emails to handle digits, web and email addresses, punctuation, and other email-specific formatting, but much more could be done to improve feature generation). Email topics in the dataset include anything from legitimate discussions of buying and selling energy assets and other innocuous business dealings, to personal exchanges about birthday parties and fantasy sports leagues, all with varying lengths and content type, such as email addresses, links and image file names; a substantial majority of emails are under 3,000 tokens, while 20 emails are over 300,000.



Identifying attempts to obfuscate information requires a more targeted and nuanced approach, particularly when those attempting to hide information are not doing so obviously. From prior literature, targeted phrases can include a range of subtlety, from the obscure “*no recollection of*”/“*I don’t recall*” and odd uses of certain verb tenses (Bachenko & Fitzpatrick, 2008), to more egregious intentions in conjunction with the topic of, say, “*stock price*”. Similar leads were derived from Enron-specific keywords, or programs, such as “*LJM*”, “*condor*”, and “*raptor*”, and particularly the use of “*talking points*” and “*spin*”, often a tell that executives were seeking to mask information from or guide decisions to be made by the energy regulators (e.g., FERC). This targeted approach identified 30 emails (excluding redundant messages) that we felt met the threshold of suspicious in, say, a discovery process. An additional random sample of roughly 12,000 emails pertaining to the aforementioned fraudulent programs were also tagged for guiding LDA, as described in 3.3 below.

3.1 Simple K-Means with BOW

In parallel to the heuristic labeling approach, we developed a baseline clustering approach using simple BOW features, limiting the dataset to a random sample of 50,000 emails to make training more efficient. Feature generation was performed using NLTK's sentence tokenizer to split emails into sentences of individual tokens, plus preprocessing of email-specific formatting, and limiting email lengths to the first 3,000 words after identifying outliers beyond that threshold. The unigram counts were then vectorized into features, keeping only those N-grams that were present in at least five email bodies and no more frequently than 1% of the emails to limit dimensions, then transformed with TF-IDF to properly weigh both the more and less common words within and across documents. The resulting feature set was just over 37,000 frequency-weighted features, still a dimension too large for generalizable clustering but sufficient for a simple baseline. These were then used to train a K-means model with six clusters, a number approximating the types of high-level email topics discovered in our targeted review.

3.2 Paragraph Vectors with K-Means

Because of the dimensionality problem and the wide variety of email types in the corpus, we sought alternatives to the usual N-gram and part-of-speech or parse features used in the prior literature. Inspired by the dimensionality reduction but effectiveness of word vectors, we also trained paragraph vectors using the algorithm developed by Le and Mikolov (2014). The idea is to represent a paragraph, or entire document, as a fixed length vector, regardless of the document's word length. The algorithm works similarly to training word vectors by predicting the next word in a span and adjusting weights by stochastic gradient descent and backpropagation. The difference, however, is that an additional fixed-length vector representing the paragraph is concatenated to the word vectors, adding an additional set of features for prediction. The resulting paragraph vector is distinct to each document and can be used separately as features to models like logistic regression and K-means. Given both our large corpus and the desire to generalize, we chose a vector size of 2,000 features, trained using the longest typical N-gram of window-size equal to five. We found that even on a dataset of over half a million emails of varying lengths, the model was efficient to train and achieved a marked improvement--albeit still limited in cluster quality--over our simple baseline K-means with TF-IDF weighted BOW features.

3.3 Latent Dirichlet Allocation

A common technique in document clustering is Latent Dirichlet Allocation (LDA). The central idea behind LDA is that underlying the corpus of documents are *N* number of covered topics; for instance, example topics could be sports, politics, religion, etc. In this model, each document has some probability of being associated with each given topic. Furthermore, each word in the vocabulary has some probability of being associated with each of the topics. For the example topics listed earlier, the word "Obama" would have a much higher probability of being associated with a topic of politics than sports. This is a generative model and thus, once the distributions are calculated, we can generate documents by picking a topic from our distribution of topics, then pick words from our distribution of words.

We quickly learned that traditional LDA was not adequate enough for our purposes because if we consider the set of documents that related to fraud or deception as one topic only, then that would still only make up a small percentage of the total corpus and would not likely get picked as one of the topics even with 10 topics using traditional LDA, so we incorporated a model called Guided LDA. GLDA allows us to pass in specific words that we want associated with certain topics. LDA will take its best guess for topic probability, but with so few keywords in a large corpus, it is unlikely to get the topics we are interested in correct. As a result, GLDA can help pick out the words we are interested in and make sure they all go into the same topic.

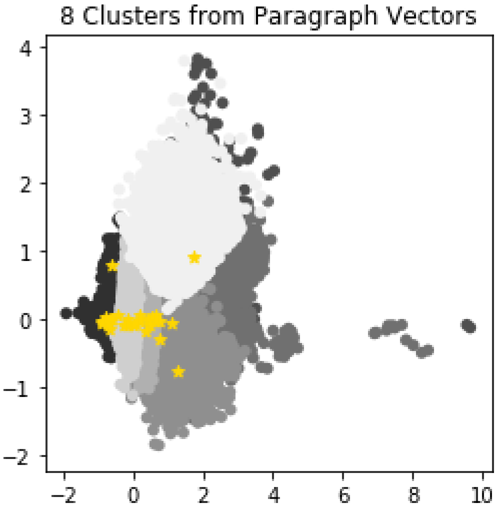
With documents clustered into topics, we can look at the documents clustered into the topics of interest, namely, the topics we put our keywords into, and see if it places fraudulent documents into that cluster.

4 Results and discussion

4.1 K-Means Clustering

Predictably, our baseline K-means model using the sparse BOW feature set did not perform well: with six clusters, chosen to represent the high-level topic variety of the company emails, there was an extreme imbalance in cluster sizes, presumably due to the wide variety of email lengths, content, word choices, spelling, and without additional preprocessing. As a result, the largest cluster contained roughly 96% of the emails, including our labeled examples that were captured in the random sample (N=50,000). This matched our intuition and left ample opportunity for improving cluster quality with a refined and reduced set of features.

With our set of paragraph vectors of length 2,000 on the entire dataset, we see marked improvement in cluster size balance and location of our 30 gold labels, but still with great limitations. We tested K-means with up to eight clusters, each improving only marginally in tightness with each additional cluster; our distance score marginally improved, roughly linearly from -40,000 with k=4 to -38,000 with 8 clusters. As seen below using principal components analysis, a small majority of our labeled examples are contained in or very near one central cluster; however, we do not see evidence that the remainder of the key cluster has likely and prevalent deception or suspicious features.



Evaluating our primary cluster and those examples nearest to our labeled data, we see mixed success identifying additional positive examples. Using a range of cosine similarities (roughly 0.5 to 0.8) to examine nearest neighbors for seven labeled examples in the primary cluster while also attempting to exclude likely redundancies picked up by the feature vectors, we find that five out of 42 neighbors (15%) are additional emails that would likely warrant review in a fraud context, while eight of the 42 exhibited some level of redundancy to the examples, either due to the example being quoted or contained in a later email chain; with more time (and reasonably high cost), it seems likely that our clustering would benefit greatly from additional preprocessing.

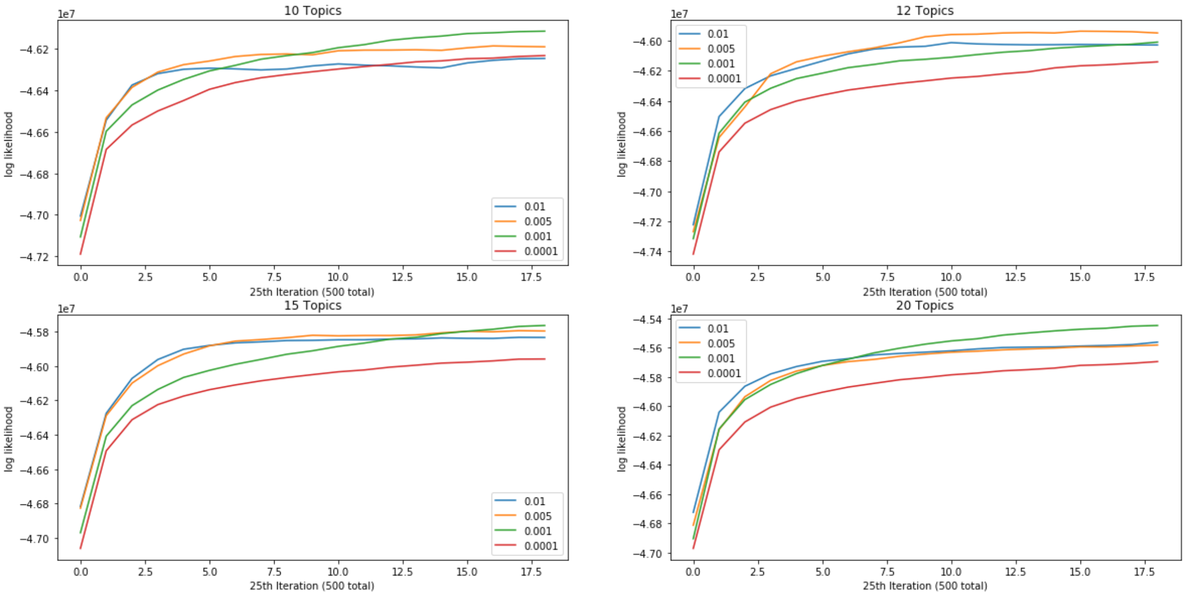
While the clustering with paragraph features does not perform well with the preprocessed dataset, it is successful in picking up interesting phrases in nearby emails, which gives us more confidence in this approach in terms of generalizing to more structured text datasets. For example, even though our trained paragraph vectors do not include N-gram features, neighbors to our evaluated labeled examples include the phrases:

- *manipulate*
- *good spin*
- *"books"* (sic, quotation marks used)
- *ammunition*
- *"steer" action* (sic)

We find the use of quotation marks particularly interesting, because within these contexts their usage can be intended to alter the meaning of otherwise innocuous words. While we suspect this might not be a normal language feature of more formal statements, we wonder if this similar paragraph vectoring can lead to interesting and useful tendencies in those other contexts.

4.2 Guided LDA

In order to determine the best parameters for the GLDA model, we ran several different models with different numbers of topics, and different values for Dirichlet parameters *alpha* and *eta*: *Alpha* represents the Dirichlet distribution over topics and *eta* represents the Dirichlet distribution over words.



We used the log-likelihoods to determine how well the model could fit the documents into the n topics. We found that as the number of topics increased the model performed better and better. We were worried about overfitting and most literature had topics in the range of 10-15, so we chose 15 as the optimal number of topics. Below is a figure showing how the different number of topics performs with different values of *alpha* with a fixed *eta* value of 0.1.

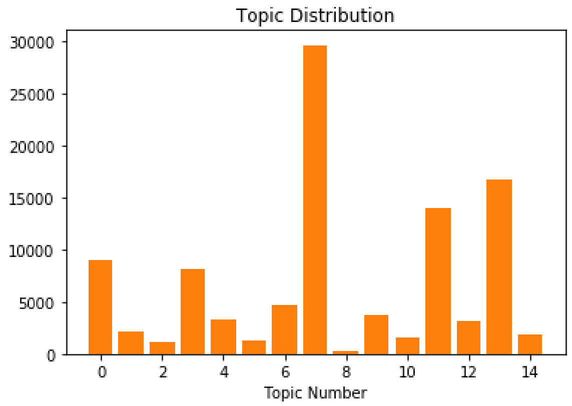
From these, we can see that when *alpha* gets too large, the models do not converge. We chose a final *alpha* of 0.01 and *eta* of 0.1 because these gave the greatest log-likelihoods that converged.

Once we settled on our model parameters we ran a sample of data, about 22,000 documents, through a count-vectorizer to get term counts, then we ran GLDA on that subset of documents to produce our probabilities for the vocabulary. We then used the resulting model to transform the rest of the emails in the corpus into the topic-space where we could compute topic probabilities.

Next, we examined the top *N* words for each of the computed topics (shown below). While we see that topic #0--the topic we put all of our keywords into--does have words that likely represent categories that would be of interest to us, we also notice that other topics do as well, such as topic #3 and #13. However, also of interest is that there appear to be some topics that are related to non-business topics, which is good to see as a check that the topics provide some meaning. For instance, topic #14 appears to be travel ad-related emails and topic #7 appears to be a lot of coordination related emails, such as people scheduling meetings or phone calls.

Topic #0: *ferc et order rto transmission meeting commission comments pm issues filing information sent know message original attached draft need request*
Topic #1: *09 dgdgdgdgdg gas dgdgdgdgdgdg intercontinentalexchange data error 000 database pool dec date 01 com power index 0f hpl natural click*
Topic #2: *company said million stock financial new billion dow jones investors business shares energy credit trading news mr dynegy debt year*
Topic #3: *com agreement message sara swap sent pm doc corp original thanks attached mail shackleton isda intended credit know fax ena*
...
Topic #5: *3d font td br nbsp size scientech com tr align width images gif face href arial updated right new span*
Topic #6: *market california iso ferc price power na energy order time electricity markets jeff cap prices ees dasovich generation costs wholesale*
Topic #7: *com message sent know time just like original mail pm good day going week think free ll new did let*
...
Topic #14: *image pm sat day tx ct new rates com scheduled houston outages london travel pt specials hilton hotel airport ca*

Next, we want to ensure that our topics are not just being assigned randomly and have some inherent meaning. We can look at the distribution of the most likely topic for each document to ensure it is not uniform in the below figure.



From this figure, we can see that topic 7 is by far the most common, followed by topics 13 and 11. Importantly, we do not see a uniform distribution and thus the GLDA is not just randomly assigning docs to topics at a constant probability.

Lastly, we can look at some example emails in our topics to see if these topics prove fruitful. Below are excerpts from emails from topic #0, the topic with which we seeded our keywords.

...stan pieringer of locke liddell & sapp helped put together two of the three transactions that need to be unwound and is otherwise familiar with structures of this nature…

...western gas resources just an fyi about the sell of assets to western gas resources <s> with the exception of clearing up a few loose ends involving some accounting issues the sell of the gomez plant and related gathering system the mitchell co2 plant…

One other topic that seemed like it was clustering docs that had relevance to Enron's illegal procedures was topic #3. An example of an email found in this topic was something that would certainly point to fraud:

...we have a second tranche of $ 608m (gross) or $ 396m (net ena 's share) to be placed into raptor DG on december…

5 Conclusion

We conclude that while this exercise on the Enron email corpus had limited success, we do believe that similar approaches could produce higher accuracy identifying suspicious phrasing given a more structured language source and verifiable history. For example, in the insurance industry, roughly one percent of claims have fraud successfully detected and payments denied using mostly manual detection and investigation techniques, which include sources such as witness statements, police reports and background checks, as well as details collected and written by claims adjusters. We suspect that similar document feature generation and keyword detection, when combined with other fraud factors, could improve automated detection and, at a minimum, contribute to identifying fraud earlier in the process to minimize improper losses.

References

Bachenko, Joan, Eileen Fitzpatrick, and Michael Schonwetter, "Verification and Implementation of Language-Based Deception: Indicators in Civil and Criminal Narratives," *ACL*, 2008, http://www.aclweb.org/anthology/C08-1006

Blei, David M., Andrew Y. Ng, Michael I. Jordan, "Latent Dirichlet Allocation," Journal of Machine Learning Research 3 (2003) 993-1022, http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf

Cohen, William W., Vitor R. Carvalho, and Tom M. Mitchell, "Learning to Classify Email into 'Speech Acts'," *EMNLP*, 2004, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.329.4178&rep=rep1&type=pdf

Dredze, Mark, Hanna M. Wallach, Danny Puller, Fernando Pereira, "Generating Summary Keywords for Emails Using Topics," *IUI*, 2008, https://research.google.com/pubs/archive/34948.pdf

Feng, Song, Ritwik Banerjee, Yejin Choi, "Syntactic Stylometry for Deception Detection," *ACL*, 2012, http://www.aclweb.org/anthology/P12-2034

Le, Quoc and Tomas Mikolov, “Distributed Representations of Sentences and Documents,” *Proceedings of the 31st International Conference on Machine Learning*, Beijing, China, 2014. JMLR: W&CP volume 32 https://cs.stanford.edu/~quocle/paragraph_vector.pdf