Exploring Credit Reporting Errors: A Text Mining Analysis of Consumer Complaint Data

Final Project for Course PPOL 566: Data Science III – Advanced Modeling Techniques, Fall 2020

Andy Green

Georgetown University
December 16, 2020

Table of Contents

Executive Summary

This report describes the methodology, analysis process, and key findings from a project focused on analyzing the text data in consumer complaints submitted to the Consumer Financial Protection Bureau (CFPB). Specifically, the analysis focuses on complaints regarding errors that consumers have identified on one or more of their credit reports. The analysis is primarily exploratory in nature, with the goal of identifying major recurring themes across the pool of complaints. This may shed light on the types of errors that consumers find on their credit reports most frequently, as well as the underlying causes of those errors. As such, I focus primarily on unsupervised learning techniques, implementing K-means clustering and Latent Dirichlet Allocation (LDA) to accomplish these goals. I find that the following two issues emerge clearly from the clustering and topic modeling algorithms:

1. Accounts appear on a consumer's report but don't belong to them
    - Consumers often use words like "identity", "theft", "victim", and "fraudulent" when describing this issue.
2. Accounts that indicate the consumer is late on a payment or missed a payment, when they believe that they made the payment on time
    - Consumers often use words like "late", "balance", "payment", and "paid" when describing this issue.

Aside from these two issues that emerge fairly clearly, the results are relatively messy, and the clusters/topics don't always seem to have clearly identifiable meanings. To validate whether these results are due to shortcomings in the clustering process or just due to the messiness of the underlying data, I also implement a supervised learning method. In addition to the narrative text field where the user describes the issue in their own words, the dataset also contains a field where the consumer is prompted to select a broad issue category from a dropdown menu. I use a Naïve Bayes classifier to predict the complaint issue category that the consumer selected, using the text data as the feature set. I find that the model struggles to predict the correct value for all classes except for the "Information belongs to someone else" class.

Finally, I end the report with a discussion of key takeaways and limitations of the analysis. I discuss how the findings of the report underscore the importance of enacting legislation aimed at reducing the prevalence of errors on consumers' credit reports, and I raise the idea that there may be further opportunities for the CFPB to educate consumers on common drivers of credit reporting errors.

Introduction

A study from the Federal Trade Commission found that over 1 in 5 consumers may have a material error on at least one of their credit reports (FTC, 2012). For over 1 in 20 consumers, these errors may be impactful enough that they cause the consumer to fall into a higher credit risk tier (FTC, 2012). This can lead to a wide array of negative outcomes, from raising the cost of credit, to increasing the likelihood that a consumer is denied credit, to making it more difficult to obtain employment and housing (Wu, 2019).

When a consumer identifies a potential error on one of their credit reports, they can file a dispute with the relevant credit bureau and/or the financial institution that furnished their data to the credit bureau (CFPB, 2012). The Fair Credit Reporting Act (FCRA) mandates that the credit bureau and/or furnisher must then investigate the consumer's claim. If they find that the data in question is inaccurate, or if they are unable to verify its accuracy, the credit bureau is required to remove the data from the consumer's report (CFPB, 2012). However, consumer groups argue that the credit bureaus fail to live up to these statutory requirements, and that the system they rely upon to manage the process has major flaws and limitations (Wu, 2019).

An additional avenue that consumers may pursue when faced with an issue on one of their credit reports is filing a complaint with the CFPB. While doing so is not a substitute for initiating a dispute directly with the credit bureau or furnisher, the fact that the CFPB publishes these complaints, along with information on whether the company responded and how the complaint was resolved, may lead consumers to believe that doing so could improve their chances of having the error resolved.

This project focuses on analyzing the text data found in these complaints, with the goal of identifying major recurring themes across the pool of complaints. This may shed light on the types of errors that consumers find on their credit reports most frequently, as well as the underlying causes of those errors. Having a better understanding of the types of errors found on consumers' credit reports can help the CFPB and other policymakers determine whether regulatory action is warranted, and if so, what regulatory action would be the most effective in combating the issue of excessive credit reporting errors.

The rest of the report proceeds as follows. First, I provide more detail on the data used in the analysis. Next, I describe the unsupervised and supervised learning methods that I use to analyze the data. Then, I discuss my analysis process and findings in more detail. Finally, I reflect on the implications of the findings and limitations of the analysis. At the end of the report, I also provide an Implementation Appendix describing some of the preprocessing steps and replication documentation in more detail.

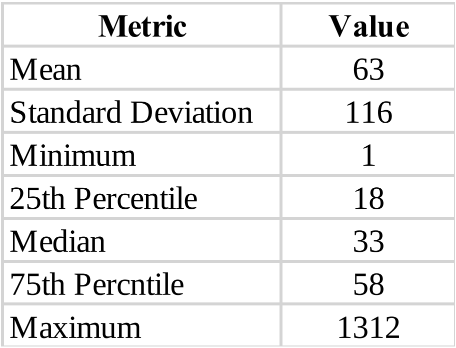
Data

The data used in this analysis is drawn from the CFPB's Consumer Complaint Database (CFPB, 2020). I begin by pulling out all of the complaints that were filed in a recent 12-month period (10/1/19 – 9/30/20), which span a wide range of different financial products and services. Of the 378,876 total complaints that were filed over this time period, 223,530 were about credit reports specifically (59%).[1] Of those 223,530 complaints, 153,934 were focused on the existence of incorrect information on the consumer's credit report in particular (69% of complaints about credit reports, or 41% of all complaints). Of these complaints, 50,366 consumers provided consent for the narrative text from their complaint to be made public in the database (with any personally identifiable information redacted). This set of complaints, and the narrative text that they contain, will serve as the focus of my analysis. However, given the computational expense associated with mining text data across this many documents, I will only be working with a random sample of 5,000 documents drawn from this donor pool.

The primary variable of interest in the dataset is the narrative text provided by the consumer. To prepare this text data for analysis, I employed a fairly standard set of text preprocessing techniques – tokenization, vectorization, lowercasing, removing stopwords, and more. The result of this process was a document term matrix (DTM) consisting of 5,000 documents and 3,394 terms. I then converted this to a weighted DTM by applying term-frequency-inverse document frequency weighting (TF-IDF), another common step in text analysis. I discuss TF-IDF weighting and the rest of the preprocessing techniques that I used in greater detail in the Implementation Appendix.

To help shed some light on the composition of the text data, Table 1 includes a series of summary statistics on the number of tokens per document after preprocessing. The table indicates that most of the documents are relatively short – 75% of documents have 58 tokens or fewer, with a median value of 33 – but a small number of relatively large documents drag the mean value to the right.

Table 1 – Summary Statistics on Number of Tokens per Document

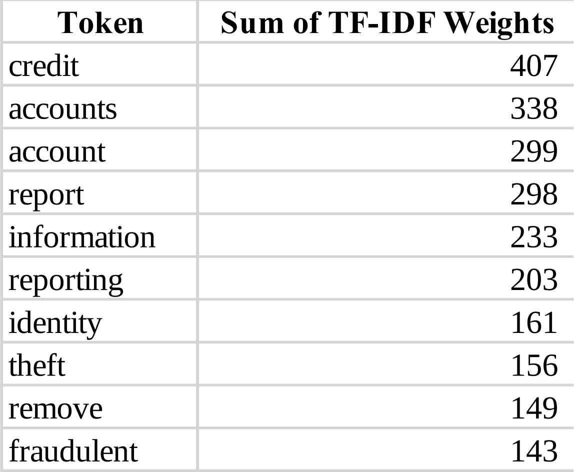


Metric	Value
Mean	63
Standard Deviation	116
Minimum	1
25th Percentile	18
Median	33
75th Percntile	58
Maximum	1312

[1] For reference, the next largest category, credit cards/charge cards, had 23,314 complaints (6% of the total).

It is also helpful to take a look at some of the most heavily weighted tokens in the corpus after preprocessing. Table 2 highlights the top 10 most heavily weighted tokens and the sum of their TF-IDF weights across all documents in the corpus. The table confirms that the top tokens all appear to be substantive and relevant to the topic at hand, an indication that the preprocessing strategies undertaken were relatively successful at preparing the corpus for analysis.

Table 2 – Tokens with Highest TF-IDF Weights

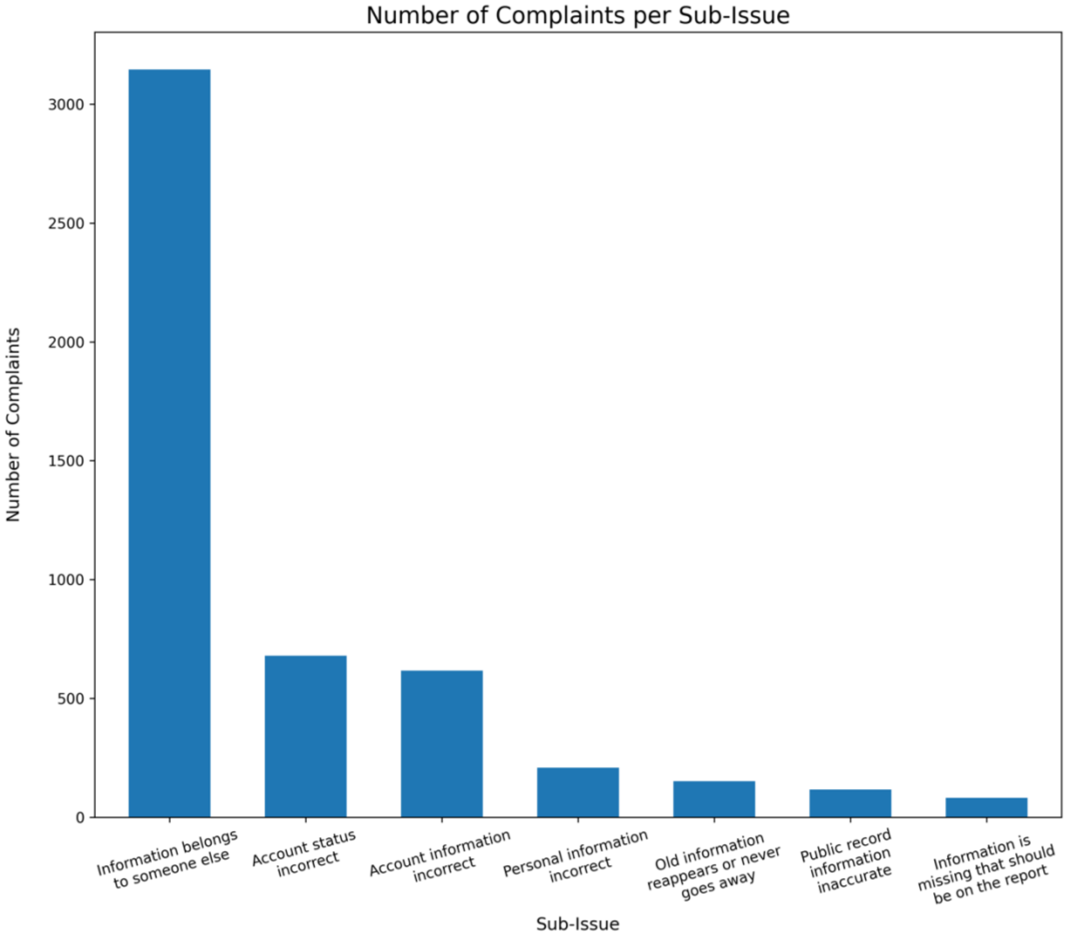


Token	Sum of TF-IDF Weights
credit	407
accounts	338
account	299
report	298
information	233
reporting	203
identity	161
theft	156
remove	149
fraudulent	143

In addition to the narrative text field, there is one more variable that I leverage in this analysis. This variable, called “Sub-issue” in the original dataset, stems from a question on the complaint form where the consumer is asked to select the value from a dropdown menu that best describes the reason for their complaint. This includes options like “Account information incorrect,” or “Information belongs to someone else.” This field is particularly valuable, because it gives us the ability to move beyond the unsupervised clustering techniques that make up the first part of the analysis, and begin to explore the data in a supervised learning context as well. In other words, we can treat this variable as the outcome variable in a machine learning model and see how well the narrative text performs as a way of predicting the issue at hand. This process is discussed further in the Methodology and Analysis & Findings sections of the report.

Figure 1 shows the distribution of issues cited by consumers across the body of complaints. The figure shows that a majority of consumers cite “Information belongs to someone else” as the primary issue driving their complaint (3,146 / 5,000, or 63%). “Account status incorrect” and “Account information incorrect” represent the next two largest issue categories.

Figure 1 – Number of Complaints per Sub-Issue



Before moving on to the Methodology section, it's worth briefly discussing some strengths and weaknesses of the dataset. One notable strength of the dataset is that it has both the "free text" style narrative text field, as well as the more structured "Sub-Issue" field where the user selects from a dropdown menu. If the dataset only had the former, it would be much more difficult to implement any supervised learning methods, and it would be more challenging to validate some of the results from the unsupervised learning analysis. On the other hand, a dataset without the narrative text field wouldn't allow us to pursue the clustering analysis and dig deeper into some of the key words that are associated with different types of issues. Thus, the combination of the two makes for an ideal pair for analysis purposes.

However, one downside of working with a free text field is that there is a wide range of content that consumers include. Complaints range from "Not my items reporting", which after removing stopwords is simply "items reporting," to submissions of lengthy blocks of legal language, and everything in between. This wide range of values makes it more challenging to derive meaningful insights from the total pool of complaints. While I think this is likely unavoidable to an extent given the nature of collecting information via free text fields, the quality of the inputs could perhaps be improved by asking more specific prompt questions, or by introducing a minimum word count threshold for submission.

In addition to these drawbacks with the narrative text field, there are also some drawbacks with the sub-issue category field. First, as indicated by Figure 1, the classes are relatively imbalanced, which will present some challenges later on in the analysis, as discussed further in the Analysis & Findings section of the report. Additionally, some of the categories seem to have considerable overlap, and consumers may be unsure which to select in certain situations, which may make classification and validation more challenging.

Methodology

I implement two unsupervised learning techniques and one supervised learning technique as part of this analysis. For unsupervised learning, I use the K-means clustering technique and the Latent Dirichlet Allocation (LDA) topic modeling technique. For supervised learning, I use a Naïve Bayes classifier. Each of these methods is discussed further in the sections below.

Unsupervised Learning

The first technique that I use is K-means clustering. K-means is a partitional clustering algorithm that makes use of “prototypes” or “centroids” to group each observation with whichever centroid is closest or most similar to that observation (Brodnax, 2020b). The algorithm iterates many times, and in each round, every document will be associated with the centroid that it is most similar to; then, the centroids will be re-calculated such that they’re at the center of all points now associated with that cluster (Brodnax, 2020b). This process repeats until the changes between iterations level off, or until the algorithm reaches some maximum number of iterations.

As with all methods, K-means comes with some advantages and disadvantages. On the plus side, K-means is likely the most “tried and true” clustering method, with widespread use across many applications (Brodnax, 2020b). Additionally, K-means results in a discrete cluster assignment for each observation, which may be well-suited for this particular application if we believe that most of the complaints are about a discrete issue. However, one drawback of K-means is that it requires you to select a specific number of clusters upfront, and there isn’t necessarily a certain value that this situation calls for. There are some techniques that can be used to help optimize the value of K, which I do implement and discuss further in the Analysis & Findings section below, but they don’t end up being particularly helpful in this specific situation. Another drawback of K-means is that the centroids are typically assigned randomly at the beginning, and this random assignment can have an effect on the ultimate outcome of the algorithm (Brodnax, 2020b). I aim to address this issue by using a slight modification of K-means called K-means++, which picks the initial centroids in a non-random way such that they are sufficiently spread out, thus helping the algorithm to reach convergence faster and more effectively (Brodnax, 2020b).

One interesting example of K-means clustering comes from Lucas de Sá (2019), who used this technique to cluster the national anthems of different countries across the world. The author leverages many of the same preprocessing steps as I use in this analysis and implements K-means clustering with five clusters. From inspecting the top words associated with each cluster, the author determines that the clusters correlate with concepts like liberty, religion, and war. In other words, the national anthems of the countries in one cluster focus more heavily on liberty, while the anthems of the countries in another cluster are more religious in nature. This analysis serves as an insightful example of how K-means clustering can be leveraged to analyze and explore text data further.

In addition to K-means clustering, I also implement Latent Dirichlet Allocation (LDA). LDA is a soft clustering technique, which means that, unlike K-means, each document is not given a discrete label/topic assignment. LDA makes use of two parameters as part of its allocation process, which tell us a) how strongly each document is associated with each topic, and b) how strongly each word is associated with each topic (Brodnax, 2020e). These two parameters then feed into a latent allocation process, which identifies key underlying topics across the corpus.

The fact that LDA does not assign a discrete label to each document may be a strength if we think that many of the complaints span multiple topics. Additionally, if we believe that the complaint data is noisy and unlikely to cluster cleanly in a hard clustering algorithm like K-means, LDA may be better suited to identify distinct underlying topics. However, one important downside of LDA is that, similar to K-means, it requires selecting a certain number of topics at the beginning. This is even more challenging for LDA, given that there aren’t some of the same validation techniques available that exist for K-means. In a situation like this where there isn’t necessarily an obvious choice for the number of topics, it usually requires trying out different values and manually inspecting the results to see which value makes the most sense.

An interesting example of LDA comes from Edison and Carcel (2020), who used this technique to identify key topics discussed by the U.S Federal Open Market Committee over a 10-year period. The authors find that the committee talked extensively about economic modeling during the years of the financial crisis, and then shifted their focus to the banking system in the years following. LDA was likely a useful technique in this case, as it’s very likely that an individual document (i.e., a transcript of a given meeting) would span a number of different topics. By using LDA, the authors avoid being forced to categorize each document into a discrete category. This analysis serves as a useful example of how LDA can be used to identify key topics that emerge in text data.

Supervised Learning

While unsupervised learning is the primary focus of this analysis given the exploratory nature of my objectives, I wanted to supplement the clustering work with a supervised learning method. As a result, I decided to implement a Naïve Bayes classifier. Naïve Bayes is a probabilistic classifier, which means that it assigns a probability that each document belongs to each class; the model then selects whichever class has the highest probability as the predicted value for that document. Behind the scenes, the algorithm makes use of: a) the base rate of each class, b) the likelihood of seeing the set of feature values amongst each class, and c) the base rate of that set of feature values amongst the whole dataset (Brodnax, 2020a). The algorithm takes these inputs and uses them to assign probabilities that each document belongs to each class.

Naïve Bayes has a few characteristics that make it particularly well-suited for this application. First, it works well in situations where the dataset has many features (Brodnax, 2020a), a characterization that certainly applies to this dataset with 3,394 features. Additionally, Naïve Bayes is effective at dealing with noise and irrelevant features (Brodnax, 2020a). As discussed in the Data section, the free text nature of the complaint field means that much of the text data is rather noisy, making Naïve Bayes a wise choice for this situation. Further, Naïve Bayes appears to be very efficient computationally, producing predictions almost instantly in my analysis, as compared with other methods that took longer periods of time to run. However, one notable disadvantage of Naïve Bayes is that it doesn't handle correlated features very well (Brodnax, 2020a). This may present a problem if we think that certain words are likely to appear together. Additionally, Naïve Bayes' conditional independence assumption is often violated in practice (Brodnax, 2020a).

One interesting application of Naïve Bayes comes from Vik Paruchuri (2015). The author uses Naïve Bayes to predict the sentiment of movie reviews, classifying them as either positive or negative. Behind the scenes, the algorithm is determining the base rate of positive/negative reviews, the likelihood of seeing the set of words contained in the review given that it's either positive or negative, and the probability of seeing that set of words in general in the dataset. These factors combine to help predict the sentiment of each review. This process mirrors how I'll use Naïve Bayes in this analysis to predict the category/topic of each complaint, and is thus a useful example for understanding my application of the method.

Analysis & Findings

Unsupervised Learning

I begin my analysis by using K-means clustering to cluster the complaints. In an effort to determine the optimal value of K to use, I leverage two common evaluation methods – the sum of squared errors (SSE) and the silhouette score. The SSE is a measure of the cohesion of the clusters, while the silhouette score reflects both the cohesion and separation of the clusters

(Brodnax, 2020c). Figures 2 and 3 demonstrate how the value of K used in the model affects the SSE and silhouette scores, respectively. Unfortunately, neither of these methods provide much assistance in determining the optimal value of K in this instance. Given that the SSE will generally decrease as the number of clusters increases, what we are looking for is an "elbow" in the plot, where the decrease in the SSE starts to level off as we move from one value of K to the next (Brodnax, 2020b). This is because the insightfulness of clustering may decrease as we allow the number of clusters to grow. For example, we could technically minimize the SSE by giving each observation its own cluster, but that wouldn't give us any further insight into the data. As a result, the idea is to strike the right balance between having a lower SSE, while still having a manageable and interpretable number of clusters. Unfortunately, the validation curve is almost exactly linear in this instance, which doesn't provide much help in selecting a value of K.

For the silhouette score plot, we are looking for a value of K where the plot comes to a distinct peak (Tan et al., 2020). Unlike with the SSE, it's common to see silhouette scores rise and fall across various values of K. However, in this case, the silhouette score line rises fairly consistently as the value of K increases. There is a bit of a local peak at a value of K = 4, but then it rises again at higher values of K.

Figure 2 – SSE by Number of Clusters

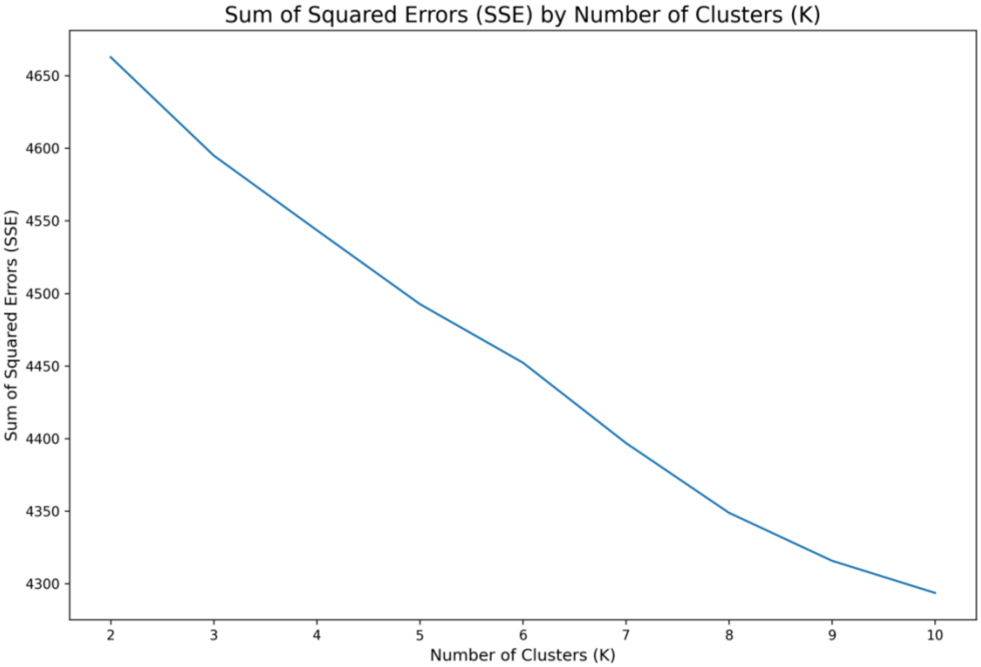
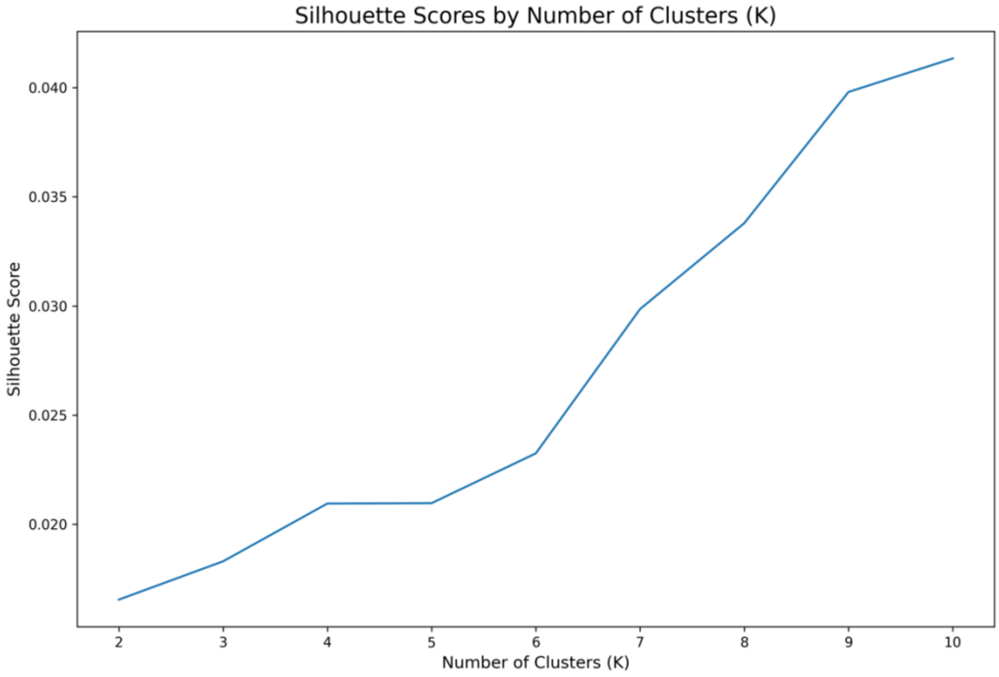


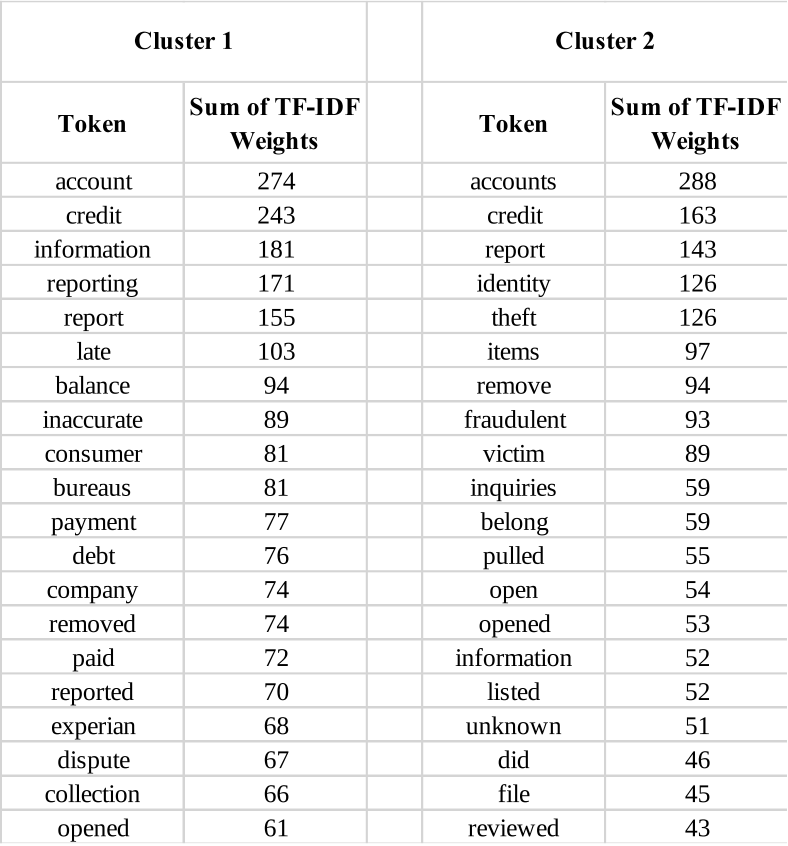
Figure 3 – Silhouette Score by Number of Clusters



Without a conclusive result for the optimal value of K from the validation methods, the best option is to try out different values of K and subjectively judge the results to see which value makes the most sense. To that end, I begin by running the K-means algorithm with a value of K = 2. The top 20 words for each of the two clusters are shown in Table 3. While there is definitely some overlap with common words between the two clusters (e.g., account(s), credit, report(ing), etc.), a few things stand out to me in interpreting the meaning of the clusters. In Cluster 1, I focus on the following words: "late", "balance", "payment", "debt", "paid", and "collection". I interpret these words to indicate that consumers are raising the issue that one of their accounts is showing as having a late or missed payment/debt, when in reality they believe that they made their payment on time and in full. This would align most closely with either the "Account status incorrect" or "Account information incorrect" sub-issues, which were the second and third largest complaint categories, respectively. In Cluster 2, I focus on the following words: "identity", "theft", "fraudulent", "victim", and "belong". I interpret these words to mean that consumers are seeing accounts on their report that do not belong to them, and believe it is due to identity theft. This would align most closely with the "Information belongs to someone else" sub-issue, which was the largest complaint category by far.

It is worth briefly noting that it may not be the case that identity theft is truly to blame in all situations where consumers find accounts on their credit reports that don't belong to them. A report from the CFPB on credit reporting cites data errors and issues with the "matching" process, where information from one consumer is inadvertently matched to another consumer's credit report, as two of the major issues that lead to credit reporting errors (CFPB, 2012). The CFPB talks about how matching errors can stem from information being entered incorrectly (e.g., consumers switching two digits on a social security number), the furnisher omitting information like date of birth or social security number on their transmission, or situations where family members have common names (e.g., Jr. and Sr.). Of course, there is no way for consumers to know about such issues, leaving them to reasonably conclude identity theft is to blame for an unknown account appearing on their credit report. The report from the CFPB doesn't indicate the relative frequencies of data entry and matching errors as compared with identity theft, but the report's focus on the former suggests that such issues are likely to blame for at least some portion of the complaints in this category.

Table 3 – Top 20 Words per Cluster with K = 2



Cluster 1			**Cluster 2**	
Token	**Sum of TF-IDF Weights**		**Token**	**Sum of TF-IDF Weights**
account	274		accounts	288
credit	243		credit	163
information	181		report	143
reporting	171		identity	126
report	155		theft	126
late	103		items	97
balance	94		remove	94
inaccurate	89		fraudulent	93
consumer	81		victim	89
bureaus	81		inquiries	59
payment	77		belong	59
debt	76		pulled	55
company	74		open	54
removed	74		opened	53
paid	72		information	52
reported	70		listed	52
experian	68		unknown	51
dispute	67		did	46
collection	66		file	45
opened	61		reviewed	43

After reviewing the clustering results at a value of K = 2, I proceeded to try running the algorithm with progressively larger values of K, reviewing the top words for each cluster after each round. However, I struggled to identify a coherent takeaway from many of the clusters in the specifications with larger values of K. While the two major themes from clustering with K = 2 appear to hold constant, and some clusters appear to show new themes/underlying issues emerging, you also see many of the same key words appearing across multiple clusters, suggesting that the different clusters are not truly separating out the complaints effectively by different categories. To illustrate this point, the top 20 words for each cluster with a value of K = 7 are shown in Table 4. Given that there are seven distinct categories in the sub-issue field, the hope would be that K-means would be able to loosely group the complaints into corresponding clusters.

Cluster 2 in this specification appears to be very similar to Cluster 1 in the earlier model, with words like “late”, “payment”, “balance”, “paid”, etc. indicating that their account status/information is incorrect. Cluster 3 in this specification appears to be similar to Cluster 2 in the earlier model, with words like “theft”, “victim”, and “identity” moving even higher up the rank order, and new words like “police” strengthening the focus on identity theft even further. Cluster 1 appears to be capturing many complaints with legal language included, as indicated by words like “shall”, “subsection”, “promptly”, and “notice”. However, I’m not able to pick out a cohesive issue category from that cluster. Cluster 4 may be picking up errors with consumers’ personal information, as indicated by words like “information”, “personal”, and “outdated”, though it’s hard to say definitively. I struggle to identify cohesive categories for Clusters 5-7, with words that I associated with earlier clusters like “fraudulent”, “identity”, “debt”, and “balance” reappearing.

Table 4 – Top 20 Words per Cluster with K = 7

Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7
consumer	account	theft	inaccurate	accounts	credit	acct
agency	late	victim	credit	report	information	account
information	credit	identity	accounts	remove	report	charge
section	payment	accounts	information	credit	reporting	opened
block	balance	credit	reviewed	items	accounts	claim
reporting	reporting	report	reporting	pulled	debt	balance
shall	report	relate	report	fraudulent	experian	compliance
theft	paid	information	file	unknown	account	response
identity	payments	result	outdated	open	oh	dispute
furnisher	reported	affidavit	incomplete	identity	items	collection
file	closed	does	personal	opened	company	writing
subsection	removed	notarized	inquiries	listed	removed	related
reseller	company	fraud	profile	did	remove	receive
blocked	date	id	bureaus	time	inquiry	statement
report	information	police	inaccuracies	belong	address	requesting
services	collection	items	knowledge	couple	bureaus	claims
identified	opened	blocked	accusations	social	sent	truth
cra	bureaus	submitted	caused	things	file	fraudulent
promptly	card	creditors	noticed	derogatory	equifax	request
notice	loan	transactions	recently	inquiries	reported	acts

While all of the above clustering and interpretation were done without explicitly using the sub-issue categories selected by the user (hence why it is referred to as unsupervised learning), the fact that we do have that information gives us the additional capacity to analyze the results through that lens. This concept will be explored in more depth in the supervised learning section that follows, but it's worthwhile to take a brief detour here to compare our interpretation of the K-means clustering results with the sub-issue categories selected by consumers. To that end, Table 5 shows the breakdown of how the cluster assignments align with the sub-issue complaint categories.

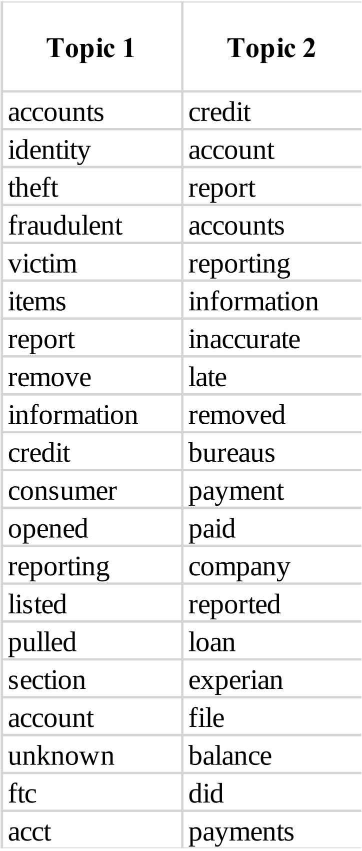
Generally speaking, the results shown in the table confirm my subjective interpretations of the K-means clustering results. Cluster 2 does seem to be picking up many of the account information/status incorrect complaints, and Cluster 3 is almost entirely comprised of complaints about information on the report belonging to someone else. Additionally, it's clear that the "Information belongs to someone else" category is widely spread across many clusters (i.e., Cluster 3 is almost entirely made up of this sub-issue, but it still only contains a small portion of all complaints with that sub-issue), which explains why we see words like "fraudulent" and "identity" showing up across different clusters. Similar phenomena seem to exist on a smaller scale for other sub-issue categories as well. Overall, the results show that the clustering analysis picks up some of the main complaint themes, but that the clusters don't track particularly closely with the sub-issue categories.

Table 5 – Comparison of K-means Clustering Results to Complaint Sub-Issues

	Account information incorrect	Account status incorrect	Information belongs to someone else	Information is missing that should be on the report	Old information reappears or never goes away	Personal information incorrect	Public record information inaccurate	Total
Cluster 1			88		1			89
Cluster 2	311	351	316	17	57	1	12	1,065
Cluster 3	1	1	344			4		350
Cluster 4	12	45	238			47	3	345
Cluster 5	8	8	888	3	4	1	5	917
Cluster 6	285	275	1,106	61	89	156	95	2,067
Cluster 7			166				1	167
Total	617	680	3,146	81	151	209	116	5,000

While LDA is a fundamentally different algorithm than K-means, as discussed in more detail in the Methodology section, it turns out that they produce fairly similar substantive results in this case. I begin the LDA analysis process by using a value of two for the number of topics produced by the algorithm. The top 20 tokens for each topic are included in Table 6. Topic 1 strongly resembles the topics from K-means focused on someone else's information showing up on a consumer's credit report, with words like "identity", "theft", "fraudulent", and "victim" appearing at the top of the list. Topic 2 strongly resembles the topics from K-means focused on consumers' account information/status being incorrect, with words like "late", "payment", "paid", and "balance".

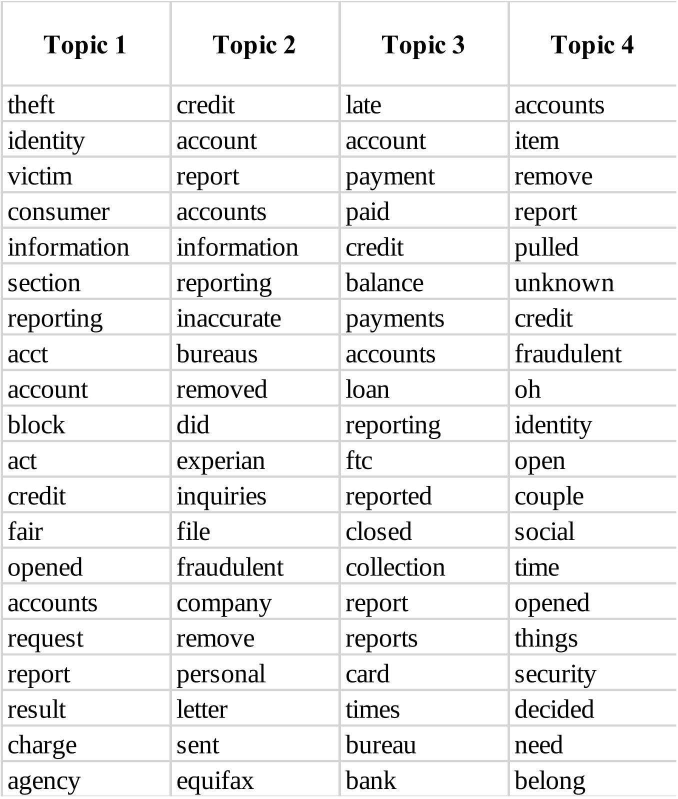
Table 6 – Top 20 Words per Topic with Two Topics



Topic 1	Topic 2
accounts	credit
identity	account
theft	report
fraudulent	accounts
victim	reporting
items	information
report	inaccurate
remove	late
information	removed
credit	bureaus
consumer	payment
opened	paid
reporting	company
listed	reported
pulled	loan
section	experian
account	file
unknown	balance
ftc	did
acct	payments

Similar to with K-means, I tried increasing the number of topics and monitoring how the top words associated with each topic changed accordingly. Once again, the two primary topic categories held mostly constant, and it was difficult to parse out the meaning of many of the remaining topics. To illustrate this point, the top 20 words associated with each topic, when using a value of four topics, are shown in Table 7. Topics 1 and 3 in this specification remain fairly consistent with the two topics from the earlier model and with those from K-means. However, it's hard to discern a cohesive category from Topics 2 and 4. The results do not seem to improve or change significantly when using more than four topics.

Table 7 – Top 20 Words per Topic with Four Topics



Topic 1	Topic 2	Topic 3	Topic 4
theft	credit	late	accounts
identity	account	account	item
victim	report	payment	remove
consumer	accounts	paid	report
information	information	credit	pulled
section	reporting	balance	unknown
reporting	inaccurate	payments	credit
acct	bureaus	accounts	fraudulent
account	removed	loan	oh
block	did	reporting	identity
act	experian	ftc	open
credit	inquiries	reported	couple
fair	file	closed	social
opened	fraudulent	collection	time
accounts	company	report	opened
request	remove	reports	things
report	personal	card	security
result	letter	times	decided
charge	sent	bureau	need
agency	equifax	bank	belong

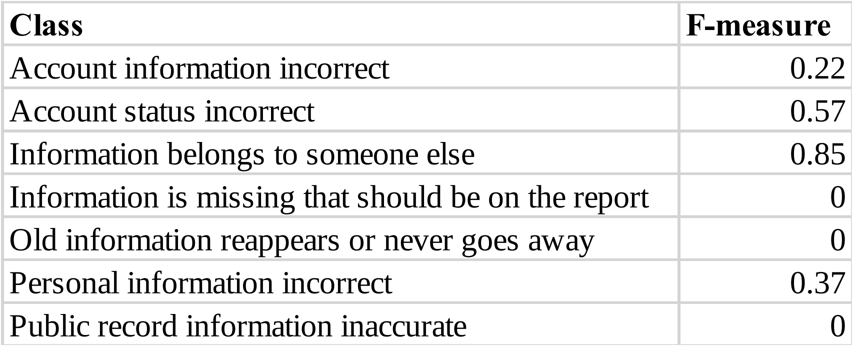
Before moving on to supervised learning, it's worth briefly summarizing the main takeaways and insights gathered from K-means and LDA. First, it does seem as though these methods succeed at highlighting some of the largest issues that consumers find on their credit reports (i.e., accounts showing up that are not theirs and accounts showing up incorrectly as being behind on payments). Additionally, they do provide insight into how consumers think about these issues and the words they use to describe them. However, the results are messy and the clusters/topics don't always seem to have clearly identifiable meanings. The hope would be that clusters/topics either line up well with the sub-issue categories that consumers selected, or that they form new cohesive categories that may even improve upon the preset categories, but neither objective seems to be met here. Instead, the results seem to highlight the few largest issues pretty clearly, but struggle to produce interpretable results beyond that.

Supervised Learning

Part of the reason why I decided to include a supervised learning method in addition to the unsupervised learning analysis was to validate some of the messiness from the clustering and topic modeling. In other words, I wanted to see if we could have more success doing supervised classification than in trying to cluster the complaints without an outcome variable. My thinking was that if we aren’t even able to classify the complaints effectively in a supervised context, then it might explain some of the messiness that we saw with clustering. Of course, these two concepts aren’t perfectly related, but it may provide some insight into the matter.

To explore this idea, I implement a Naïve Bayes classifier, using the text data of the complaint to predict the sub-issue category selected by the consumer. At first glance, the model seems to perform relatively well, with a mean test accuracy of 0.72 using 5-fold cross-validation. However, digging a little deeper into the class predictions presents a more complicated picture. Perhaps due to the class imbalance in the underlying data, where 3,146 / 5,000 documents were classified as “Information belongs to someone else,” the model is overwhelmingly picking this class in its predictions. The model selected this class for 4,131 predictions, while three of the seven classes did not receive a single prediction. Given these results, using a metric like the F-measure may give a better idea of the true model performance. The F-measure is a score that combines a model’s precision and recall values for a given class into a single score that runs on a 0-1 scale, with 1 being the best score (Scikit-learn, 2020). Table 8 shows the F-measure values for each of the complaint classes. The results in the table reflect how the model works fairly well for the “Information belongs to someone else” class, but it struggles with most of the other classes. The “Account status incorrect” class is the next highest performing, which tracks with what we saw popping out in the clustering analysis as well.

Table 8 – F-measure Values by Complaint Class



Class	**F-measure**
Account information incorrect	0.22
Account status incorrect	0.57
Information belongs to someone else	0.85
Information is missing that should be on the report	0
Old information reappears or never goes away	0
Personal information incorrect	0.37
Public record information inaccurate	0

Overall, it’s hard to determine the extent to which the supervised learning model is harmed by imbalanced classes as opposed to just the messiness/noisiness of the data. However, regardless of which is more to blame, I think the results of this portion of the analysis help explain and underscore some of the challenges with the clustering analysis. It’s not necessarily surprising that we saw terms like “fraudulent” and “identity” showing up in various different clusters given the prevalence of the class those terms are associated with, and the noisy nature of much of the data can only make the process more challenging.

Conclusion

There are a few ways in which the insights from this analysis can be useful to policymakers seeking to improve the credit reporting process. First, the analysis underscores the extent to which consumers find accounts on their credit reports that do not belong to them. Given how harmful this can be to consumers' credit scores, it is imperative that the credit reporting agencies take every available step to reduce the prevalence of such errors. When these errors do manage to slip through, the dispute resolution process must allow consumers to receive a fair and prompt resolution to the problem. Fortunately, the House of Representatives recently passed legislation that would help address these challenges. Amongst many other provisions, the Comprehensive CREDIT Act of 2020 would improve the accuracy of credit reports by requiring that the credit bureaus use stricter matching standards when adding furnished information to a consumer's report, and strengthen consumer rights and protections in the dispute resolution process, including a newly added right of appeal. (Americans for Financial Reform, 2020). These reforms would go a long way toward reducing the issues that we see emerging most prominently from this analysis. I strongly recommend that both the House and the Senate take swift action to pass this legislation at the start of the 117th Congress.

Additionally, the results of this analysis suggest a potential opportunity for educating consumers about the reasons why consumers may find an account on their credit report that doesn't belong to them. As discussed earlier, consumers seem to come to the reasonable conclusion that such accounts are the result of identity theft, when it's likely that at least some portion of these instances is actually the result of data entry or matching errors. This may cause consumers unnecessary stress or lead them to spend time and money seeking to protect themselves from identity theft when their identity hasn't actually been stolen. The CFPB could feasibly provide resources to consumers explaining the various potential causes of this issue after the consumer has submitted a complaint that falls into this particular sub-issue category.

Finally, it's worth discussing a few limitations of the analysis and considerations for the future. In the Data section of the report, I discussed how the messiness of the text data, and the class imbalance and overlapping nature of the sub-issue category field make the analysis more challenging. In addition to these important limitations, working with text data can also be challenging from a computational expense standpoint, and I was forced to work with only a sample of complaints as a result. While I believe that the sample size of 5,000 randomly selected complaints should be sufficiently large to preserve most of the meaningful variation in the data, it would nonetheless be interesting to analyze the full set of complaints using additional computing power. Lastly, it's good practice to consider the role of ethical limitations in any machine learning analysis. To the extent that ethical considerations would play a role in this analysis, they would likely be most present upstream in terms of what data the CFPB is able to publish and what they choose to redact. However, I have no reason to believe that not having access to any of this redacted information negatively impacted the results of my analysis, and I am not aware of any ethical considerations that would restrict the use of the data this is publicly available.

References

Americans for Financial Reform. (2020, January 27). *Letter to Congress: Letter in Support of the Comprehensive CREDIT Act of 2020.* Americans for Financial Reform. Retrieved From: https://ourfinancialsecurity.org/2020/01/letter-congress-letter-support-comprehensive-credit-act-2020/

Brodnax, N. M. (2020a, February 10). PPOL 565: WK 5 – Classification.

Brodnax, N. M. (2020b, September 23). PPOL 566: Module 4 – K-means Clustering.

Brodnax, N. M. (2020c, October 5). PPOL 566: Module 5 – Clustering Evaluation.

Brodnax, N. M. (2020d, October 14). PPOL 566: Module 6 – TF-IDF Weighting.

Brodnax, N. M. (2020e, November 18). PPOL 566: Module 8 – Topic Modeling.

Consumer Financial Protection Bureau. (2012, December). *Key Dimensions and Processes in the U.S. Credit Reporting System*. Consumer Financial Protection Bureau.

Consumer Financial Protection Bureau. (2020). *Consumer Complaint Database*. Consumer Financial Protection Bureau. Retrieved From: https://www.consumerfinance.gov/data-research/consumer-complaints/

de Sá, L. (2019, December 17). *Text Clustering with K-Means: Clustering national anthems with unsupervised learning.* Medium. Retrieved From: https://medium.com/@lucasdesa/text-clustering-with-k-means-a039d84a941b

Edison, H., & Carcel, H. (2020). Text data analysis using Latent Dirichlet Allocation: an application to FOMC transcripts. *Applied Economics Letters*, *28*(1), 38–42. https://doi.org/10.1080/13504851.2020.1730748

Federal Trade Commission. (2012, December). *Report to Congress Under Section 319 of the Fair and Accurate Credit Transactions Act of 2003*. Federal Trade Commission.

Paruchuri, V. (2015, March 17). *Tutorial: Predicting Movie Review Sentiment with Naïve Bayes.* Dataquest. Retrieved From: https://www.dataquest.io/blog/naive-bayes-tutorial/

Scikit-learn. (2020). *Documentation: sklearn.metrics.f1_score.* Scikit-learn. Retrieved From: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

Tan, P.-N., Steinbach, M., Karpatne, A., & Kumar, V. (2020). Chapter 7: Cluster Analysis: Basic Concepts and Algorithms. In *Introduction to data mining* (pp. 525–612). Pearson.

Wu, C. C. (2019, February). *Testimony before the U.S. House of Representatives Committee on Financial Services Regarding "Who's Keeping Score? Holding Credit Bureaus Accountable and Repairing a Broken System"*. National Consumer Law Center.

Implementation Appendix

Preprocessing

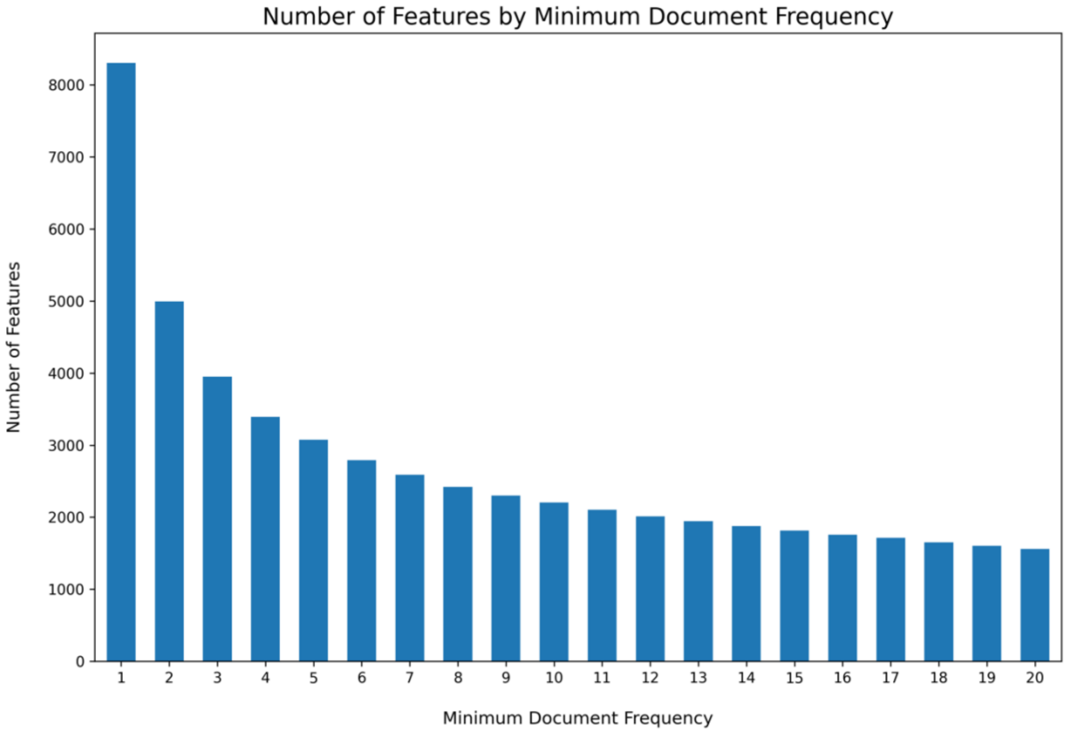
This section describes the preprocessing steps applied to the text data in greater detail. The first step in the process was tokenizing the data, which means separating out each complaint into a list of tokens. In this case, I chose to only use unigrams, as opposed to bigrams or trigrams, which means that each token is a single word. After tokenization, the next step is vectorization, which means formatting the corpus such that each document is a row and each unique token is a column. The values of the cells correspond with the number of times a given token appears in a given document.

There are also a number of other preprocessing steps baked into this process. All of the following steps were included as part of tokenizing and vectorizing:

- Lowercasing: all words are converted to only lowercase characters
- Removing all punctuation and symbols
- Removing stopwords: common words such as “the”, “and”, “or”, etc. are removed
- Removing redaction placeholder text (e.g., “xx” and “xxxx”)
- Removing words that appear in < 4 documents

While it is a common step to remove words that appear in only a few documents, the exact number to use as the threshold is subject to the researcher’s discretion. In an attempt to make an informed decision, I decided to utilize a validation chart showing the number of features remaining in the corpus for various values of the minimum document frequency threshold. Figure 4 contains the graph I used in making this decision. The graph shows that there is a major drop-off in the number of features going from a minimum document frequency of one to two, but then it quickly drops off from there. I selected a value of four, as this is around the point at which you begin to see diminishing returns from raising the threshold further, but you could reasonably select a number of other values and the results would likely be rather similar.

Figure 4 – Number of Features by Minimum Document Frequency



Finally, the last important step in preparing the text data for analysis was applying term frequency-inverse document frequency (TF-IDF) weighting to the document term matrix. TF-IDF allows us to use both the term frequency (how many times a term appears in a given document) and the document frequency (how many documents a term appears in) to give a weight to each term in the corpus. Higher term frequencies are associated with higher importance for a given term, while a higher document frequency is associated with lower importance for a given term (Brodnax, 2020d). Combining these two measures into a single value is a convenient way of weighting our document term matrix for analysis purposes.

Data and Code Files

This section provides a brief overview of the code and data files used in this analysis for replication purposes:

- File: 'text_analysis_consumer_complaints_1.ipynb'
    - Purpose: this file is used to read in the full batch of 378,876 complaints that were filed in the Consumer Complaint database over a 12-month period, and narrow it down to the 5,000 complaints that serve as the focus of my analysis.
    - Inputs: 'complaints.csv'

        - Note: this file is not included in the repository as it is over 280 MB in size. As a result, it's best not to run the code in this file, as it will error out without the proper data inputs.
    - Outputs: 'complaints_sample.csv'
        - Note: this file is what will be used in the next code file, and it already exists in the repository so that this file doesn't need to be re-run.
- File: 'text_analysis_consumer_complaints_2.ipynb'
    - Purpose: this file contains the code for all of the analysis steps described throughout this report.
    - Inputs: 'complaints_sample.csv'
    - Outputs: this file produces a wide array of .csv and .png outputs containing the tables and graphics used in this report. These files can all be found in the "Tables_Graphics" sub-folder in the repository.
        - Note: the code chunks that produce these outputs are currently commented out to avoid creating unexpected files on the user's machine.