Topic Classification and Topic Modeling of Customer Complaints in the Financial Sector

Student Names and IDs: Niklas Steinfurth (149901), Jonathan Mathis Haug (149889), Konrad Heinrich Schulte (149872) & Thor Møldrup **(**127318)
Course: Natural Language Processing and Text Analytics
Programme: MSc Business Administration & Data Science
Page Count: 15
Character Count w. spaces: 33,836
28.05.2022

Abstract

This project explores different approaches to classify and create topics for financial consumer complaints by leveraging natural language processing models. First, five different topic classification models were used: Random Forest, Binomial Naive Bayes, Logistic Regression, Linear SVC, and a Convolutional Neural Network. We found that the Linear SVC model was the best at classifying ten issue topics with an accuracy of 76%. Additionally, the performance measures of precision, recall, and further results in the confusion matrix confirm that the model performs moderately well.

Moreover, we analyzed how clustering and unsupervised machine learning can be utilized to create topics with a predefined amount and to let the model decide the number of clusters. The two models that we used were Latent Dirichlet Allocation (LDA) and Top2Vec. In the end, we found that the LDA model works best at evaluating the current labels and improving upon them, whereas Top2Vec works best when the goal is finding new insights.

Finally, we concluded that a company needs to be aware of its needs before trying to use NLP for topic classification or modeling customer complaints. Topic classification only works when the company already has predefined topics and would want to classify incoming complaints within these topics. On the other hand, topic modeling is also useful when a company does not have predefined topics and still would like to cluster the complaints into different groups.

Table of Contents

1 Introduction

Customer support is an important function for every business and crucial for business success. Its primary goal is contented customers, achieved by helping them using a product or service, answering their questions, giving solutions, and troubleshooting with them. Customer support can also be defined even broader since every stakeholder in a firm can be considered a customer. This includes internal stakeholders, such as employees from different departments, but also external stakeholders, such as suppliers and customers.

Customer support can be very personnel intensive. In the earliest customer support configuration, all inquiries might have landed in a centralized inbox and were answered by an employee, which may not

be very scalable. When facing increasing complexity of support inquiries creating scalable processes that automate as much as possible becomes essential.
For this reason, Natural Language Processing (NLP) is a popular application in the customer support domain. A cutting-edge example of highly automated customer support comes from the Chinese company ANT Financial, whose chatbot system is beating human performance regarding customer satisfaction (Knight, 2017).
Part of efficient and automated customer support processes is an efficient content classification of support inquiries. Automatic recognition of the topic of a support request can help prioritize support tickets such as complaints and route them to the right employees.

In this paper we will apply topic classification and topic modeling on a dataset that contains customer complaints about financial products and services. We will answer the following research questions:
1a: How can topic classification be used to classify topics of customer complaints?
1b: How can topic modeling be leveraged to create clusters of customer complaints?
1c: How can companies use natural language processing to optimize their classification of customer complaints?

2 Related Work

Models such as random forest, multinomial naive bayes, and Linear SVC have all been used to do multi-label text classification. Omar, Mahmoud, Abd-El-Hafeez and Mahfouz (2021) studied a way to classify Arabic text documents, using a variety of methods such as the ones mentioned above, with different feature representations such as bag of words, TF-IDF, and N-grams(1,2). They found that the Linear SVC ended up performing the best for multi-label classification with TF-IDF feature representation. Further, they showed that tuning of the hyperparameters also increased the performance significantly (Omar et al., 2021).

Convolution neural networks have been frequently applied to text categorization tasks. One example is Johnson & Zhang (2017). The team created a convolutional neural network for sentiment classification and topic categorization with 15 weight layers. They were able to increase the accuracy of their model by increasing the depth of the neural network while at the same time maintaining low computational costs (Johnson & Zhang, 2017).

A recurring problem for many businesses is that they do not have their data labeled, which makes many machine learning models unfeasible. However, with the use of unsupervised machine learning models, this issue can be mitigated. Within topic modeling, there are many different state-of-the-art approaches

to generate topics for text analysis. A study from Egger & Yu (2022) explored the performances on some of these models such as Latent Dirichlet Allocation, Top2Vec and BERTopic for identifying topics from Twitter posts. In the end, they found that the Top2Vec has an advantage of discovering hidden patterns and topics within a corpus, which makes it perform a bit better than the other models. However, there are major differences between the models, and researchers should be aware of these when figuring out which models to apply to which situation and dataset (Egger & Yu, 2022).

3 Methodology

3.1 Dataset Description

The dataset used for our project originates from the Consumer Complaint Database (*Consumer Complaint Database*, 2022). This database is operated by the Consumer Financial Protection Bureau (CFPB, agency of the US government responsible for consumer protection in the financial sector). The CFPB offers US citizens the opportunity to file complaints about financial institutions. These complaints are then forwarded to the respective firms and their responses are returned to the customers. At the end of this process, the complaint is recorded in the publicly accessible Consumer Complaint Database. The fact that the database is maintained by only one agency ensures that the data is consistent in terms of categorization.

The database (or downloaded dataset) at the time of our project start (03.05.2022) contained 2,642,708 rows and 18 columns. It included complaints related to 6,364 different companies from March 2015 to April 2022. The complaints are in the customers' own words (with sensitive information removed) and were categorized into 165 different issues. Appendix 1 gives a short overview of the raw dataset with all its columns and their number of unique values.

3.2 Data Preprocessing and Exploratory Data Analysis

First, we downloaded the dataset from the CFPB database and got a rough overview.
For our project, two columns were of crucial importance, namely “Issue” and “Consumer complaint narrative”. A more detailed analysis of the "Consumer complaint narrative" revealed that this column contains 1,718,434 Null values. These observations are useless for our project, so we removed the respective rows from the dataset.
After removing all Null values, the dataset consisted of 924,274 rows, of which only 820,861 were unique complaints. A closer look at these duplicate narratives showed that the dataset contained the exact same narratives several times for different companies in different states at different times.

Furthermore, these identical complaints were partly assigned to different issues. Since this inconsistency in the data could reduce the performance of our models, we decided to remove these approximately 100,000 duplicates from the dataset.

The dataset cleaned of (complaint) Nulls and duplicates now contained 820,861 unique complaints, which were assigned to 160 different issues. However, 160 issues are too many for an accurate multiclass classification, so we investigated how we could minimize the number of issues and at the same time get the largest possible set of complaints. Appendix 2 shows how many complaints are covered by which topics (sorted by "size" of the issues in descending order). The plot shows that the issues were very unevenly distributed and that even a few issues covered a large part of the dataset. We therefore decided to further limit the dataset to the top 10 issues and thus the final dataset comprised 474,366 complaints distributed over 10 issues. These "Top 10 Issues" can be seen in Appendix 3.

After clearing our dataset, we started to analyze other columns to get a better understanding of the data. Appendix 4 shows the number of complaints over time. The number of complaints has increased significantly over the past years. This again emphasizes the increasing importance (of a good handling) of complaints. A more detailed analysis of the amounts of complaints per day (see Appendix 5) shows a strong outlier in 2017. We found out that on September 8 and 9, 2017, over 2,000 complaints were received concerning only the company "Equifax". A check revealed that Equifax had indeed experienced a major data breach in September 2017, which explains this outlier (*Equifax Data Breach Settlement*, 2019).

The distribution of character lengths across the different complaints in Appendix 6 shows that most narratives are between 1 and 1,000 characters long. After the analysis of our dataset showed no further anomalies, we finally reduced the dataset to the two necessary columns "Issue" and "Consumer complaint narrative". An illustration of the entire transformation from raw to final dataset can be seen in Appendix 7.

Finally, we preprocessed our complaints and corresponding issues for the later models. We tokenized the narratives first and then filtered out all stopwords. The most frequent words in the dataset can be seen in Appendix 8. Accordingly, the words "credit", "account" and "report" occur particularly frequently in complaints. In a second step we created a wordcloud of all words (see Appendix 9). This gives us further insights as the wordcloud also recognizes patterns such as frequent occurrence of two following words. According to this, words like "credit report", "identity theft" or "reporting act" occur a lot in complaints.

The issues were extracted from the dataset and encoded for the supervised models (for the final encoding, see Appendix 10). The further steps to prepare the data for the respective models are described in the respective sections.

As the last part of our exploratory analysis, we originally wanted to investigate the "level of anger" in different complaints. Therefore, we applied and evaluated five models for sentiment analysis: Text2Emotion, TextBlob, NRCLex, Vader, and HappyTransformer. Unfortunately, we were not satisfied with any of the models' results and therefore decided to exclude them from both code and report.

3.3 Topic Classification and Modeling

3.3.1 Topic Classification

One goal of this project was to create a topic classification model that is able to assign different complaints of our dataset to the correct issue. In this paper, we are applying the classification into more than two labels, which is called multi-class classification. To start off with, we will present the different models that we ran on the dataset for multi-class classification. These are Random Forest, Multinomial Naive Bayes, Logistic Regression, Linear SVC, and Convolutional Neural Network.

Theory of Applied Algorithms

Random forest is a classification model that uses an ensemble of decision trees. The decision trees test each attribute, and then classify the data based on which has the highest probability. The Random Forest then combines many decision trees, to be able to classify the data (Donges, 2021)**.**

The Multinomial Naive Bayes Classifier is built upon the Bayes Theorem, which calculates conditional probabilities. These are calculated for each class, where the one with the highest probability is the finally assigned class (*Multinomial Naive Bayes Explained*, 2021)**.**

Multinomial Logistic Regression is a modification of the normal logistic regression model that tries to predict a binomial probability for each class based upon a numeric input. The multinomial logistic regression predicts a multinomial probability for each input example. The output changes from a singular probability value and into one for each of the class labels (Brownlee, 2020).

The Linear Support Vector Classifier (Linear SVC) is a method that applies a linear kernel to do the classification, which especially works well when performed on a large number of samples. One of the main differences between the Linear SVC and the normal SVC is that the kernel of the Linear SVC cannot be changed because it is always based on the linear kernel (DataTechNotes, 2020).

A Convolutional Neural Network is a neural network that is frequently applied on machine learning problems. As other neural networks, CNN consists of different layers. The amount and kind of layers used depends on the task that the neural network is supposed to solve. CNN performs particularly well at image classification tasks. Nevertheless, they can also be used for Natural Language Processing.

Methodology for Random Forest, Multinomial Naive Bayes, Logistic Regression and Linear SVC

To know which of the above models are most useful for making predictions on our dataset, we decided to first run Random Forest, Multinomial Naive Bayes, Logistic Regression, and Linear SVC against each other using cross-validation to determine which one has the highest accuracy. To improve the models further, we decided to try two different vectorizers for our input: the count vectorizer and the TF-IDF vectorizer. We then took the mean accuracy of the 5 cross-validations to determine the overall accuracy of each model. Due to the long runtime of the model, we decided to only use 10,000 rows for the cross-validation to determine the best model.

The Linear SVC model with the TF-IDF performed best out of all models as seen in Appendix 11. Therefore, we used Linear SVC for further analysis. To optimize the model, we performed a grid search to optimize the hyperparameter C, which tells the optimizer how small the margin-plane is for misclassifying the training points. Thus, the bigger the C value, the smaller the margin is. We chose to run the grid search on the 10,000 samples from our data, because of the run time of the search. When the hyperparameters had been tuned, we ended up with a model with a C value of 1, which we then fit on the entire dataset. The final classification metrics for this model will be presented in the results section of the paper.

Methodology for Convolutional Neural Networks

Another approach for solving the text classification problem was the application of two convolutional neural networks (CNN1 and CNN2) inspired by Janakiev (2018). CNN1 consists of six layers. The architecture of CNN1 is displayed in Appendix 12. CNN2 is an adjusted version of CNN1 and consists of eight layers. Additional batch normalization and dropout layers were added to avoid overfitting. The architecture of CNN2 is also displayed in Appendix 12. Both networks are equipped with an embedding layer for word vectors. However, the design of the embedding layer is different for CNN1 and CNN2. Whereas CNN1 uses custom created embeddings, CNN2 uses pre-trained word embeddings. For computational efficiency GloVe embeddings (Stanford University, 2022) were favored over Word2Vec for CNN2.

CNN1 and CNN2 were both trained on the full preprocessed dataset (train set) and evaluated with the test set. Hence, in contrast to the previous models, no cross validation was applied. Afterwards, hyperparameter tuning was applied on the best performing model. The results are shown in chapter 4.

3.3.2 Topic Modeling

After building algorithms to classify the labeled issues we used two different topic modeling techniques to categorize our complaints in an unsupervised way. The goal was to see if topic modeling enhances our understanding of the categories of consumer complaints.

We used Latent Dirichlet Allocation (LDA) and Top2Vec to split our complaints in different categories. LDA is like Principal Component Analysis (PCA) and focuses on maximizing separability among the different topics. LDA works with a bag-of-words corpus. To create this corpus and to summarize the same words in different versions to one element we first lemmatized and stemmed the words within each complaint and then created the bag-of-words. To then cluster the complaints LDA requires a predefined number of topics. It creates different topics by both maximizing the distance between the means of the topics and minimizing the variation within each topic. As LDA relies on a bag-of-words approach it ignores ordering and semantics of the words within a complaint.

Top2Vec is a more recent approach and consists of five different steps. First the documents are embedded with the help of Doc2Vec. After the embedding documents are closer to similar documents and thus the semantics are (partly) preserved. Then the document vectors are reduced to a smaller dimension with the help of UMAP. Afterwards the clustering algorithm HDBSCAN is used to find the centers of each cluster. After finding the centers, the center topic vectors are created and in the last step each document is assigned to the topic with the closest topic vector (Angelov, 2020).

4 Results

4.1 Topic Classification

4.1.1 Performance Measures for Supervised Models

To be able to compare and assess the performance between our models, we need to have some accuracy measures. A common one is accuracy which takes the number of correct guesses and divides it by the total number of guesses. However, for skewed datasets, this measure is not ideal, which makes it necessary to use other measures. Two of these are Precision and Recall.

Precision

Precision is a metric that calculates the True Positives that were predicted out of the total positive predictions. This would be the number of correct classifications out of the total amount of complaints classified by the model with this specific label.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Figure 1: Precision Formula

Recall

Recall calculates the true positives out of the total actual positives. This means that it measures how many predictions the model correctly classified as positives, out of the total amount of actual positives (Géron, 2019).

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Figure 2: Recall Formula

F1-Score

Usually, you might use either precision or recall depending on the problem that you are trying to solve. However, sometimes the problem at hand requires a compromise between the two. Here, the F1-Score can be used, which seeks to find a score that is between precision and recall.

$$F1\ Score = \frac{2}{Precision^{-1} + Recall^{-1}}$$

Figure 3: F1 Score Formula

These measures are usually calculated on binary targets. For our analysis we have 10 different labels, which means that we need to calculate the total precision and recall a bit differently. To find the total measures, we calculated the individual scores for each class and then averaged them together in two different ways. First, we calculated the weighted averages and secondly the macro averages. The weighted average computes the total accuracy with a specific weight for each label depending on the size of it. Because our dataset is skewed, we will not rely on this score.

The macro average score treats all classes equally even though they do not have the same size (Sokolova & Lapalme, 2009). This gives us a more reliable result (as it does not drag the accuracy up unnaturally).

Confusion matrix

A confusion matrix is a way to look more granularly at the performance of models. A confusion matrix tells you how many times the classifier has correctly identified each topic and how many times it misclassified with a specific wrong label. In the end, the confusion matrix gives a clear picture of the performance of the model, and it can also unveil patterns as to why the model might be misclassifying some topics.

4.1.2 Results

Results from Linear SVC

If we look at the classification report, we can see that the Linear SVC model has an overall accuracy score of 76%. However, as we have a skewed dataset, this metric is not the best at measuring the performance of our model, which is why we will look at precision and recall with a focus on the macro

averages. We can see that the macro average for precision is 75% while it is 71% for recall and 72% for the F1-score. For the best performing issue, we have “Managing an account” with 92% in precision, 97% in recall and 94% in f1-score. One of the worst performers is the issue “False statements or representation”, which has a precision of 61%, a recall of 35% and an F1-score of 45%, which can be seen in Appendix 13.

In the confusion matrix we can see the performance of the Linear SVC for each topic (Figure 4). The diagonal line has the highest numbers, meaning that the model is generally good at classifying each issue correctly. However, there are some issues that are wrongly classified. The issue “Incorrect information on credit report” is misclassified as the issue “Incorrect information on your report " 36% of the time, where it is only correctly classified 40% of the time. This could be a case of the complaints being very similar in terms of words used, which then makes the classifier perceiving them as very similar. The misclassification of “Written notification about debt” as “Attempts to collect debt not owed” seems to stem from a similar problem.

The total runtime of the Linear SVC was 9 minutes and 20 seconds, which is a good running time, when there are many rows for the model to be fitted on. The runtime is a big advantage of using the Linear SVC model, especially in a business environment, where runtimes are critical.

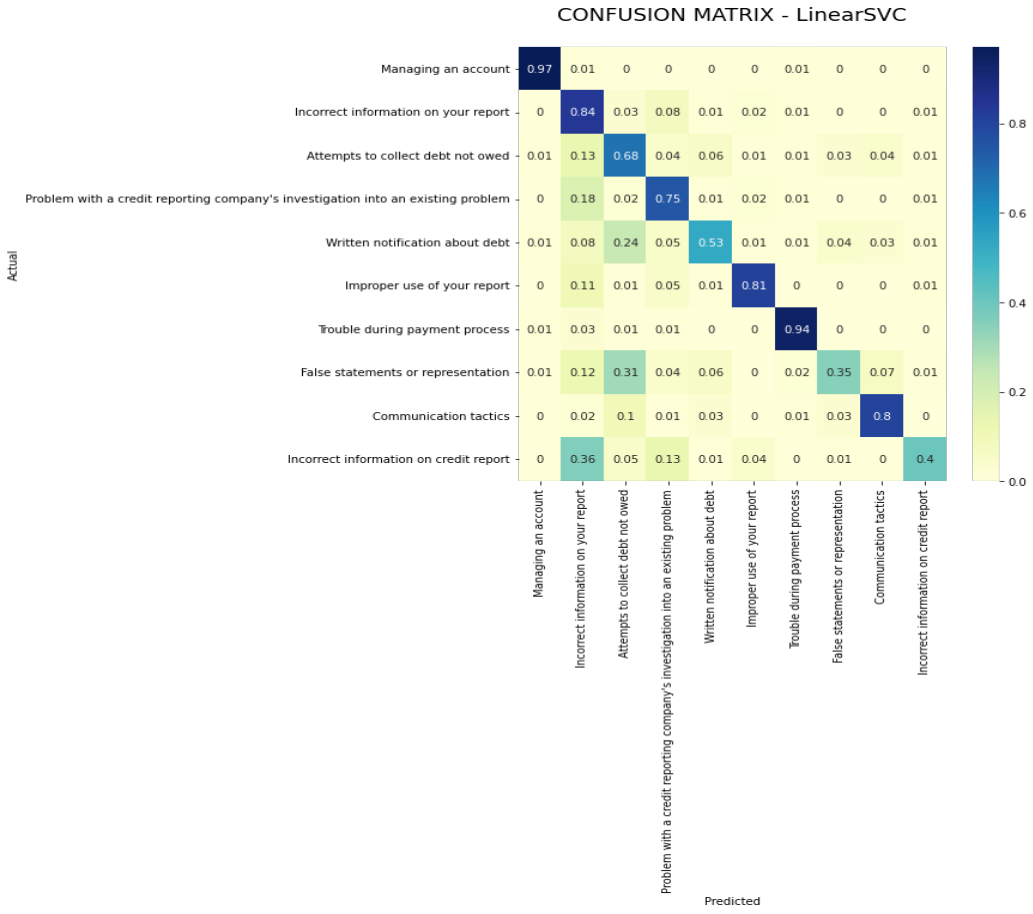


Figure 4: Confusion Matrix for Linear SVC

Results from Convolutional Neural Network

CNN1 had an overall accuracy of 66%. The macro average of the precision of CNN1 was 61%, the macro average of recall was 62%. All performance metrics of CNN1 are displayed in Appendix 14. The confusion matrix is displayed in Appendix 15.

Hyperparameter-Tuning for kernel size and filter was applied to CNN2. The tuned CNN2 was able to achieve a better performance than CNN1 in terms of all these performance measures. Accuracy was 69%, the macro average precision was 66%, macro average recall 66% and macro average F1-score 66%. The relative performance of the different complaint issues was very similar to the one of CNN1. Fortunately, we were able to improve the performance of smaller classes significantly. 'Trouble during payment process' had the highest precision of 88%, highest recall of 96% and F1 score of 92%. For the lowest F1 score we have 'Written notification about debt' of 38%. All performance metrics of CNN2 are displayed in Appendix 16. The respective confusion matrix is displayed in Appendix 17.

The runtimes of CNN1 and CNN2 were rather long, potentially due to the size of the dataset. CNN1 ran in total 5 hours and 32 minutes, CNN2 5 hours and 27 minutes. It is noteworthy that we were able to improve the performance from CNN1 to CNN2 while improving the runtime of the model.

4.2 Topic Modeling

Topic modeling is unsupervised by its nature. The evaluation of the resulting topics is thus subjective and dependent on the perceived coherence within the topic of the person evaluating it. While it is possible to calculate a coherence score for the LDA model, such a function is not implemented for the Top2Vec model yet and we therefore decided to focus on the analysis of the words within each topic.

Results from the LDA

We decided to set the number of topics in the LDA to the amount of Issue labels (10) in our dataset. With the number of issues and the number of topics being the same we can better analyze similarities between labels and the topics created by our unsupervised model.

We analyzed the words occurring in each topic of the LDA (see Appendix 18) and created short descriptions for each of the topics. The resulting descriptions are:

Topic 0: Inquiries about late payments and loans

Topic 1: [Broad topic - hard to find coherent description]

Topic 2: Inform and dispute about inaccurate balance asking for update or removal

Topic 3: Communication issues

Topic 4: Complaints about Equifax

Topic 5: Identity theft, fraud

Topic 6: Questions regarding loan and payment dates

Topic 7: Mortgage, payments, loans and checks

Topic 8: Validity of debt collection

Topic 9: Proof of debt origin

While some of the topics seem to be quite similar (esp. Topic 0,6 and 7), others can be clearly distinguished. To further analyze the topics, we looked at the distribution of the topics per issue label generated by the LDA (Appendix 19). Several topics show similarities with the labels and especially Topic 3 and "Communication Tactics" is a very good match.

The runtime of the LDA was 50 seconds and including the LDA specific preprocessing a total of 15 minutes runtime were necessary. While the time required to create the model is pretty low the topic modeling of all complaints took considerably longer and 1 hour and 24 minutes of runtime where necessary to find the LDA topic of each complaint.

Results of Top2Vec

In total, we created three different Top2Vec models. Our first model (Top2Vec_Full) created 2,283 different topics with some of the topics only consisting of 20 or less complaints. Such a high number of clusters might not be feasible for a company to analyze. To reduce the number of topics and to increase the number of complaints per topic in our second model we have set the minimum cluster size of HDBSCAN to 5,000 (Top2Vec_5000). This resulted in only two clusters and thus was a too high restriction on the number of clusters. Furthermore, the word clouds of these clusters do not seem to be not very coherent (Appendix 20). Our last and final model was limited to clusters with a size of at least 1,000 complaints (Top2Vec_1000). This resulted in 93 different topics. Looking at the word clouds of the few biggest clusters within each model we can see that both the Top2Vec_Full and Top2Vec_1000 model are clustering the complaints into coherent topics while our Top2Vec_5000 model is performing poorly.

Appendix 21 shows a comparison of the top 7 clusters of the Top2Vec_Full and Top2Vec_1000. The topics are very similar, and 3 Topics are the same. The reduction of the cluster size thus seemed to conserve the main clusters while reducing the number of clusters. The reduction of the number of clusters to 2 clusters lead

The runtime of the models was around 1 hour and 30 minutes each (Top2Vec_Full: 1h 25min, Top2Vec_5000: 1h 21min and Top2Vec_1000: 1h 33min).

5 Discussion of Results

5.1 Topic Classification

Overall, the Linear SVC model outperformed the CNNs in terms of performance metrics and runtime. The overall accuracy was better with a value of 76% compared to the best CNN which ended up with an accuracy of 69%. Additionally, both the macro averages for precision and recall were superior in the Linear SVC model. Finally, the run time of the Linear SVC was 9 minutes and 20 seconds compared to the CNNs that ran for approximately 5 and a half hours.

The Linear SVC model does not natively support multi-label classification. However, by using the one vs rest approach, it is feasible to apply Linear SVC to our multi-label classification problem. Therefore, the Linear SVC acts as a binary classifier with the one vs rest approach to go through each class. This might explain the increase in performance because we are utilizing a power algorithm for binary classification and applying it to the multi-label classification.

The performance of the CNNs could potentially be improved with more extensive hyperparameter tuning that includes not only kernel size and number of filters of the convolutional layer, but also parameters such as batch size, number of epochs, learning rate etc. Additionally, the CNN can be further improved by adding more layers to it, thus making it deeper.

Overall, the best supervised model from the ones that we have used is performing moderately well on our dataset, with macro averages for recall, precision, and accuracy that are just around 75%. Additionally, we can see in the confusion matrix that it correctly classifies most of the classes correctly, however, there are still some that it is struggling to classify correctly. Given that the accuracy is around 75%, we cannot say that it can be used to reliably classify the issue types of the different complaints which leaves room for improvements. It especially lacks at figuring out the differences in issues, where they are closely connected, for example “Incorrect information on credit report” and “Incorrect information on your report". These results could imply the necessity for different issue categories from the companies.

5.2 Topic Modeling

Topic modeling can be used to get a deep understanding of the nature of customer complaints. We will answer our research question 1b by describing ways how LDA and Top2Vec can be used to leverage the creation of clusters and how these clusters can lead to enhanced understanding of the complaints in the following. LDA can be used if the number of clusters needed is already known and can enhance the understanding of the data. When labeled data is available, LDA can be used to understand dependencies between labels by analyzing how different topics of the LDA are distributed within the labels. This distribution can be monitored and (in case of too high dependencies between labels) the distributions can be used as a starting point of a discussion on how changes of the labels would lead to better treatment of complaints.

Top2Vec can be used to get granular clusters of complaints. With hyperparameter tuning it is possible to control the number of clusters Top2Vec produces. Analyzing the different clusters can lead to understanding of changes in consumer complaints and to the detection of newly important kinds of complaints. Using topic modeling next to topic classification can give companies a tool to sense changes in complaint clusters that would be hard to catch with supervised machine learning tools only. Our Top2Vec_1000 model, for example, detected a cluster of problems with ATM withdrawals (Appendix 21 - Topic 2). This is something we have not seen in this detail before and being able to detect this cluster of complaints enables companies to provide faster and better support whenever a new complaint is classified accordingly.

5.3 Topic Classification vs. Topic Modeling

The Linear SVC model and topic classification should be used by a company when they already have their data labeled, for example as issue topics or product topics. Topic classification will allow the company to automatically label incoming complaints and send them to the appropriate employees. If an issue is misclassified, the employee should notify the data department and specify which other category the complaint belongs to, which helps improve the model over time. This process can save the labor expenses for classifying complaints and increases precision.

If a company does not have labeled data, they should utilize the topic modeling approach. This allows them to classify and structure the complaints, so similar complaints will appear in the same place. However, if the company does not know the name of the clusters it would be difficult to know exactly which employees should handle it. On the other hand, the complaints were already not classified, and by doing this, an employee could look at the specific complaints and determine, based on the clusters, which employees would handle it the best.

To answer our research question 1c, the use of natural language processing in handling the classification of customer complaints, depends on what problem the company is trying to solve. Both approaches

have their merits, and it would be valuable for companies to explore both. However, it takes both resources and the correct skills to implement, which limits the feasibility of both models.

6 Conclusion

Throughout this paper, we analyzed how Natural Language Processing can be leveraged to assist financial companies in classifying and clustering customer complaints. For our topic classification, four different models were trained, where the LinearSVC was selected as the best. The hyperparameter of this model was then tuned, which resulted in an accuracy of 76%. Furthermore, two different Convolutional Neural Networks were trained and evaluated, one with custom created embeddings and another with pre-trained word embeddings. In the end, the second CNN was selected as the best out of those two, which ended up with an accuracy of 66%. We then ended up with the LinearSVC model as the best performing out of our topic classification models.
We also studied how clustering and unsupervised learning can help give a better understanding of how the issue topics can be structured in another (maybe better) way. Here we found that the LDA model works best when having to evaluate your current labels and improve them, whereas Top2Vec is best at discovering new insights and similarities in the complaints.

Finally, we can conclude that different approaches should be used based on the needs of a company. If they already have predefined issue topics, they can use the LinearSVC model, to automatically label new incoming customer complaints, which will save the company from having to sort through the complaints and give them a label. However, if a company does not have predefined labels on their complaints, they can use the one of either LDA or Top2Vec to divide the incoming complaints into topics based on the text within them. This creates groups that are more similar than what the company can do themselves.
In the end, we recommend that companies utilize the strengths of both approaches to improve their complaint management and increase overall customer satisfaction.

References

Angelov, D. (2020). *Top2Vec: Distributed Representations of Topics* (arXiv:2008.09470). arXiv. https://doi.org/10.48550/arXiv.2008.09470

Brownlee, J. (2020, December 31). Multinomial Logistic Regression With Python. *Machine Learning Mastery*. https://machinelearningmastery.com/multinomial-logistic-regression-with-python/

Consumer Complaint Database. (2022, January 1). Consumer Financial Protection Bureau. https://www.consumerfinance.gov/data-research/consumer-complaints/

DataTechNotes. (2020, January 7). *Classification Example with Linear SVC in Python*. https://www.datatechnotes.com/2020/07/classification-example-with-linearsvm-in-python.html

Donges, N. (2021, July 22). *Random Forest Algorithm: A Complete Guide | Built In*. https://builtin.com/data-science/random-forest-algorithm

Egger, R., & Yu, J. (2022). *A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts—PMC*. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9120935/

Equifax Data Breach Settlement. (2019, July 11). Federal Trade Commission. http://www.ftc.gov/enforcement/refunds/equifax-data-breach-settlement

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition [Book]*. https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/

Janakiev, N. (2018). *Practical Text Classification With Python and Keras – Real Python*. https://realpython.com/python-keras-text-classification/

Johnson, R., & Zhang, T. (2017). Deep Pyramid Convolutional Neural Networks for Text Categorization. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 562–570. https://doi.org/10.18653/v1/P17-1052

Knight, W. (2017). *Meet the Chinese Finance Giant That's Secretly an AI Company*. MIT Technology Review. https://www.technologyreview.com/2017/06/16/151178/ant-financial-chinas-giant-of-mobile-payments-is-rethinking-finance-with-ai/

Multinomial Naive Bayes Explained: Function, Advantages & Disadvantages, Applications in 2022. (2021, January 3). UpGrad Blog. https://www.upgrad.com/blog/multinomial-naive-bayes-explained/

Omar, A., Mahmoud, T. M., Abd-El-Hafeez, T., & Mahfouz, A. (2021). Multi-label Arabic text classification in Online Social Networks. *Information Systems*, *100*, 101785. https://doi.org/10.1016/j.is.2021.101785

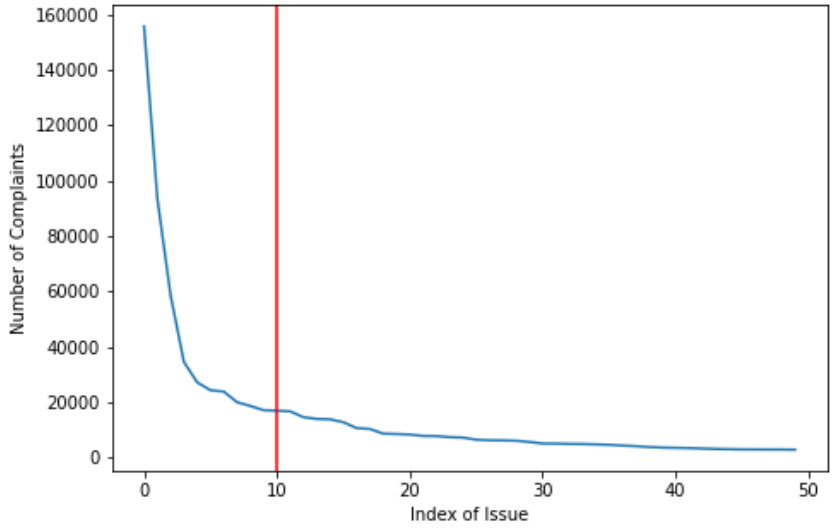
Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Stanford University. (2022). *GloVe: Global Vectors for Word Representation*. https://nlp.stanford.edu/projects/glove/

Appendix

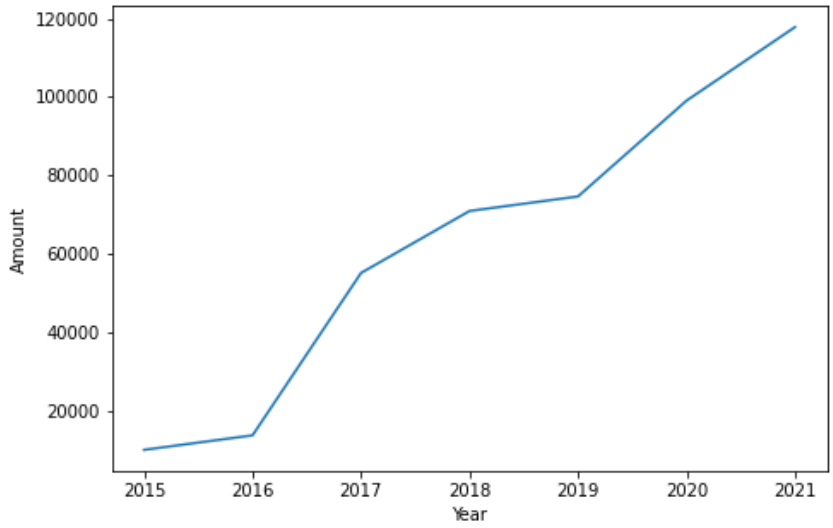
Column	Unique Values	Null Values
Date received	3,805	0
Product	18	0
Sub-product	76	235,163
Issue	165	0
Sub-issue	221	657,945
Consumer complaint narrative	820,861	1,718,434
Company public response	11	1,544,156
Company	6,364	0
State	63	39,583
ZIP code	58,977	39,839
Tags	3	2,332,526
Consumer consent provided?	4	775,150
Submitted via	7	0
Date sent to company	3,754	0
Company response to consumer	8	3
Timely response?	2	0
Consumer disputed?	2	1,874,250
Complaint ID	2,642,708	0

Appendix 1: Overview of Raw Dataset

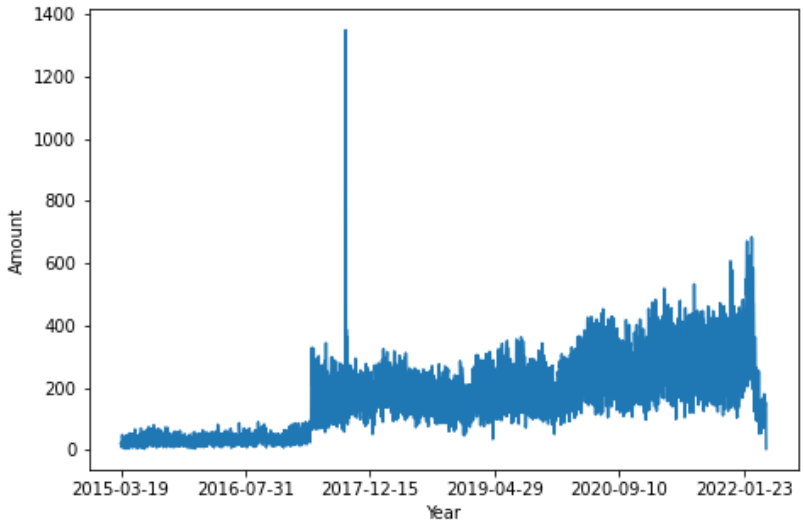


Appendix 2: Number of Complaints per Issue

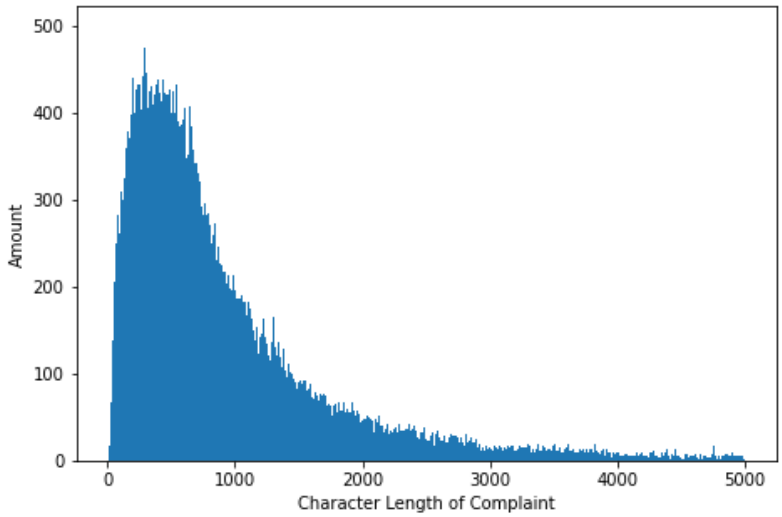
Appendix 3: Ten Most Frequent Issues



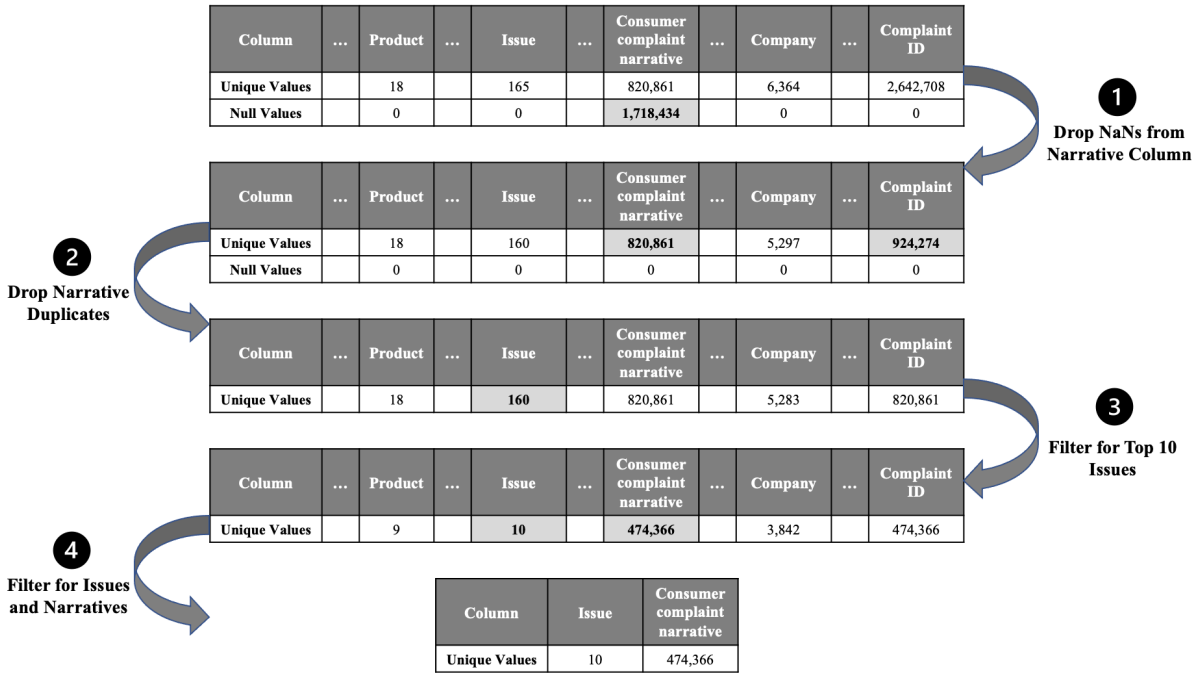
Appendix 4: Number of Complaints over Time



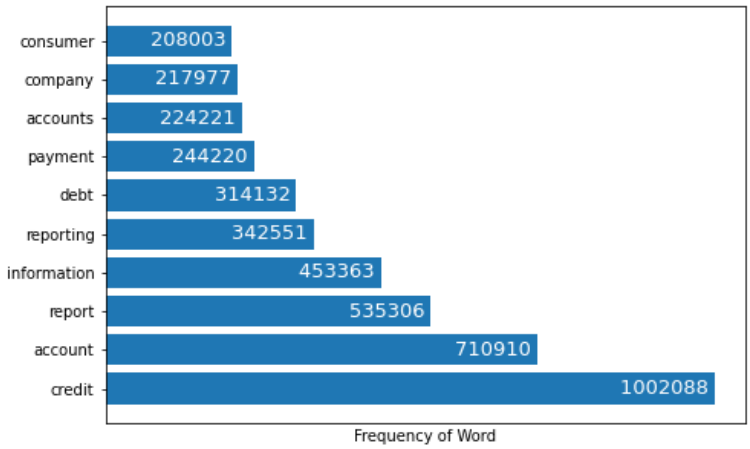
Appendix 5: Number of Complaints per Day over Time



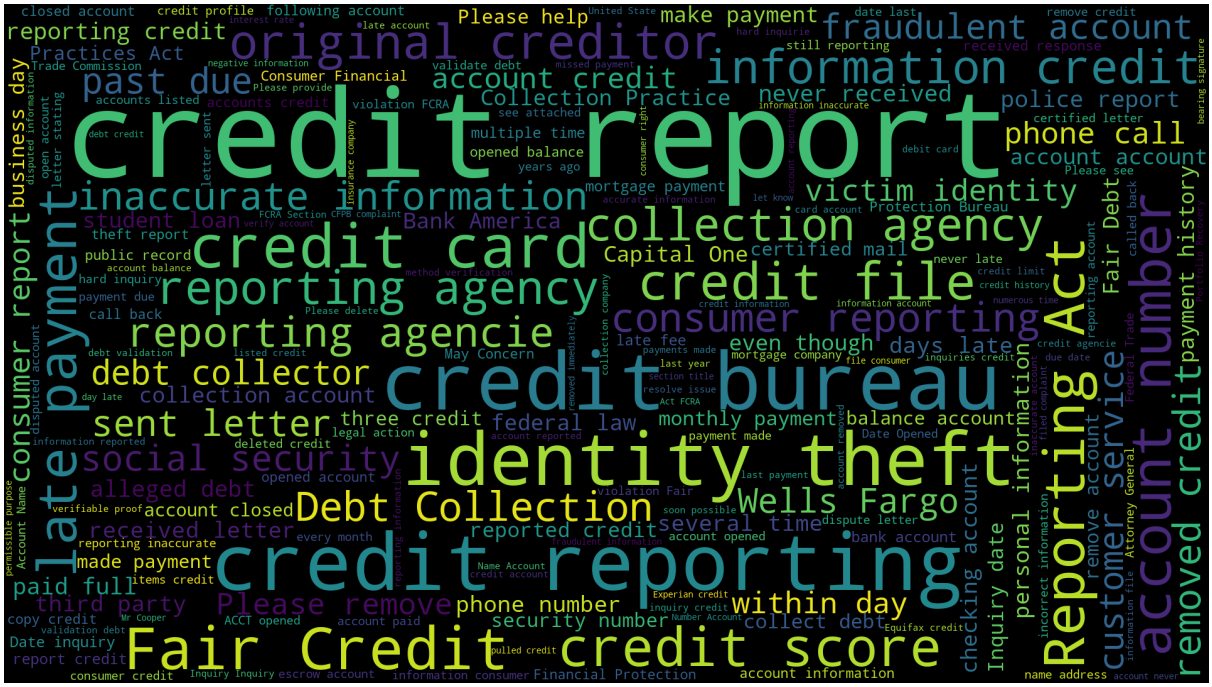
Appendix 6: Character Length of Complaints



Appendix 7: Dataset Preprocessing Overview



Appendix 8: Ten Most Frequent Words



Appendix 9: Wordcloud of Whole Dataset

0 = Managing an account
1 = Incorrect information on your report
2 = Attempts to collect debt not owed
3 = Problem with a credit reporting company's investigation into an existing problem
4 = Written notification about debt
5 = Improper use of your report
6 = Trouble during payment process
7 = False statement or representation
8 = Communication tacticts
9 = Incorrect information on credit report

Appendix 10: Label Encoding

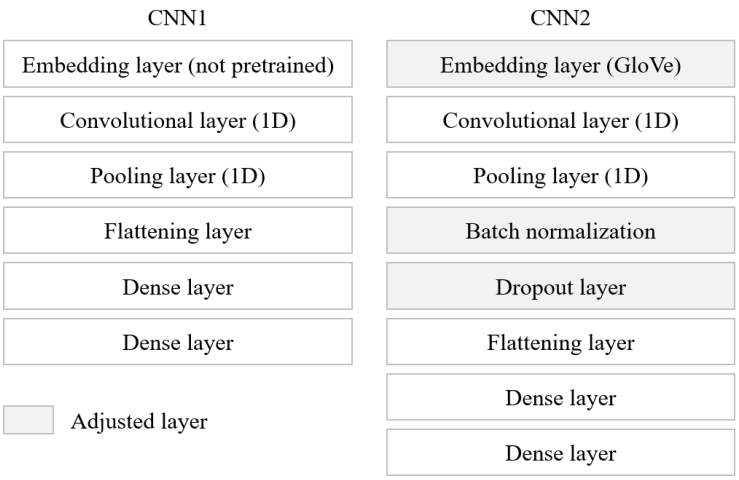
Count Vectorizer

name_of_model	Mean Accuracy
LinearSVC	0.6108
LogisticRegression	0.6353
MultinomialNB	0.6047
RandomForestClassifier	0.6473

TF-IDF Vectorizer

name_of_model	Mean Accuracy
LinearSVC	0.6818
LogisticRegression	0.6549
MultinomialNB	0.4970
RandomForestClassifier	0.6421

Appendix 11: Mean Accuracies of Count Vectorizer and TF-IDF Vectorizer



Appendix 12: Architecture of CNN1 and CNN2

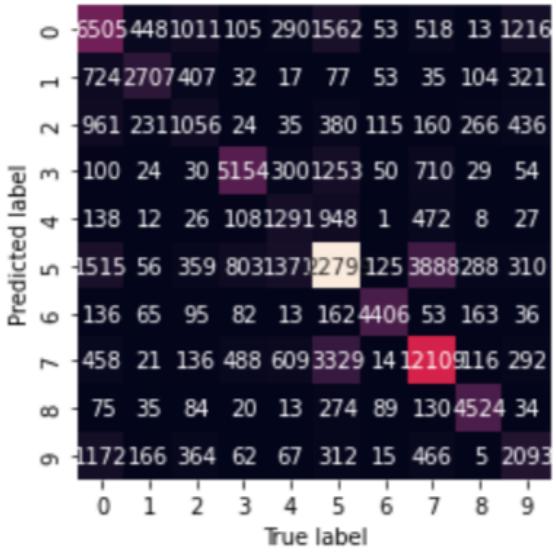
CLASSIFICATIION METRICS REPORT

	precision	recall	f1-score	support
Managing an account	0.92	0.97	0.94	4921
Incorrect information on your report	0.76	0.84	0.80	31096
Attempts to collect debt not owed	0.64	0.68	0.66	11784
Problem with a credit reporting company's investigation into an existing problem	0.77	0.75	0.76	18541
Written notification about debt	0.65	0.53	0.58	4819
Improper use of your report	0.80	0.81	0.80	6878
Trouble during payment process	0.89	0.94	0.91	5516
False statements or representation	0.61	0.35	0.45	3568
Communication tactics	0.76	0.80	0.78	3765
Incorrect information on credit report	0.72	0.40	0.52	4006
accuracy			0.76	94894
macro avg	0.75	0.71	0.72	94894
weighted avg	0.75	0.76	0.75	94894

Appendix 13: Performance Metrics for Linear SVC

		precision	recall	f1-score	support
Managing an account	0	0.55	0.55	0.55	11784
Incorrect information on your report	1	0.60	0.72	0.66	3765
Attempts to collect debt not owed	2	0.29	0.30	0.29	3568
Problem with a credit reporting investigation	3	0.67	0.75	0.71	6878
Written notification about debt	4	0.43	0.32	0.37	4006
Improper use of your report	5	0.72	0.73	0.73	31096
Trouble during payment process	6	0.85	0.90	0.87	4921
False statements or representation	7	0.69	0.65	0.67	18541
Communication tactics	8	0.86	0.82	0.84	5516
Incorrect information on credit report	9	0.44	0.43	0.44	4819
	accuracy			0.66	94894
	macro avg	0.61	0.62	0.61	94894
	weighted avg	0.66	0.66	0.66	94894

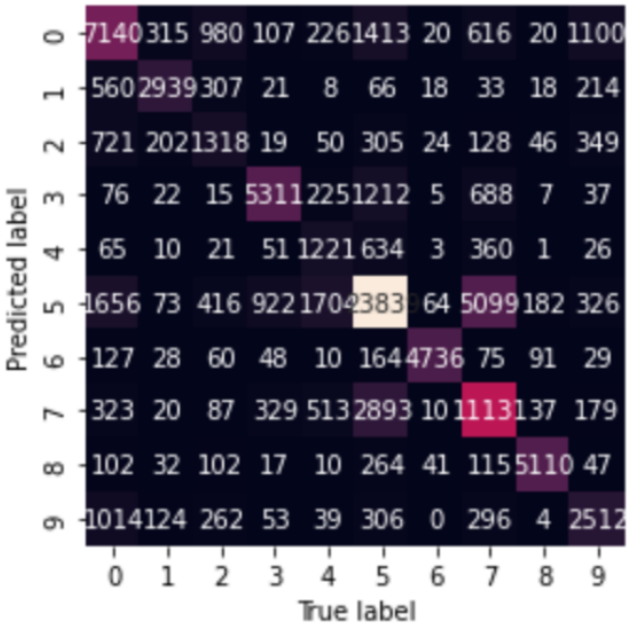
Appendix 14: Performance Metrics for CNN1



Appendix 15: Confusion Matrix for CNN1

		precision	recall	f1-score	support
Managing an account	0	0.60	0.61	0.60	11784
Incorrect information on your report	1	0.70	0.78	0.74	3765
Attempts to collect debt not owed	2	0.42	0.37	0.39	3568
Problem with a credit reporting investigation	3	0.70	0.77	0.73	6878
Written notification about debt	4	0.51	0.30	0.38	4006
Improper use of your report	5	0.70	0.77	0.73	31096
Trouble during payment process	6	0.88	0.96	0.92	4921
False statements or representation	7	0.72	0.60	0.65	18541
Communication tactics	8	0.88	0.93	0.90	5516
Incorrect information on credit report	9	0.54	0.52	0.53	4819
	accuracy			0.69	94894
	macro avg	0.66	0.66	0.66	94894
	weighted avg	0.68	0.69	0.68	94894

Appendix 16: Performance Metrics for CNN2



Appendix 17: Confusion Matrix for CNN2

Topic: 0

Words: 0.048*"payment" + 0.028*"inquiri" + 0.018*"bank" + 0.017*"make" + 0.015*"late" + 0.011*"would" + 0.010*"call" + 0.010*"month" + 0.009*"receiv" + 0.009*"loan"

Topic: 1

Words: 0.046*"consum" + 0.033*"inform" + 0.016*"agenc" + 0.013*"section" + 0.012*"disput" + 0.011*"creditor" + 0.009*"violat" + 0.009*"day" + 0.009*"delet" + 0.008*"furnish"

Topic: 2

Words: 0.028*"date" + 0.023*"balanc" + 0.022*"inform" + 0.018*"payment" + 0.017*"remov" + 0.014*"disput" + 0.012*"late" + 0.012*"inaccur" + 0.010*"open" + 0.010*"updat"

Topic: 3

Words: 0.038*"call" + 0.017*"tell" + 0.013*"say" + 0.013*"would" + 0.012*"receiv" + 0.011*"inform" + 0.011*"time" + 0.011*"compani" + 0.011*"phone" + 0.010*"back"

Topic: 4

Words: 0.022*"equifax" + 0.012*"number" + 0.012*"bank" + 0.010*"disput" + 0.009*"inform" + 0.009*"file" + 0.008*"state" + 0.008*"remov" + 0.008*"address" + 0.008*"capit"

Topic: 5

Words: 0.041*"inform" + 0.023*"ident" + 0.021*"theft" + 0.019*"file" + 0.018*"remov" + 0.016*"fraudul" + 0.015*"disput" + 0.013*"item" + 0.011*"verifi" + 0.011*"delet"

Topic: 6

Words: 0.019*"loan" + 0.018*"payment" + 0.016*"time" + 0.012*"make" + 0.011*"year" + 0.011*"pay" + 0.010*"compani" + 0.009*"late" + 0.009*"remov" + 0.009*"month"

Topic: 7

Words: 0.021*"mortgag" + 0.018*"check" + 0.015*"payment" + 0.014*"loan" + 0.011*"pay" + 0.011*"insur" + 0.009*"escrow" + 0.009*"receiv" + 0.009*"would" + 0.008*"month"

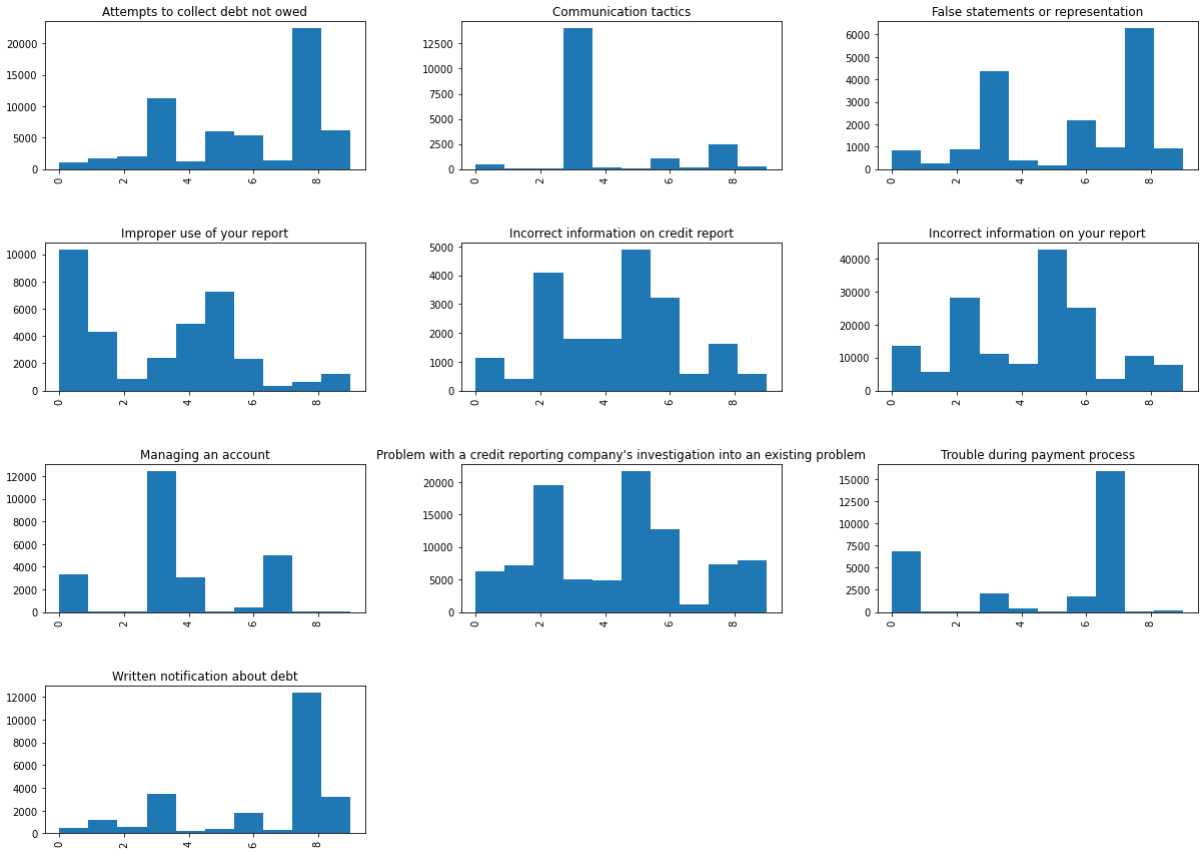
Topic: 8

Words: 0.058*"debt" + 0.041*"collect" + 0.019*"letter" + 0.018*"disput" + 0.018*"receiv" + 0.017*"send" + 0.015*"compani" + 0.012*"valid" + 0.010*"amount" + 0.010*"agenc"

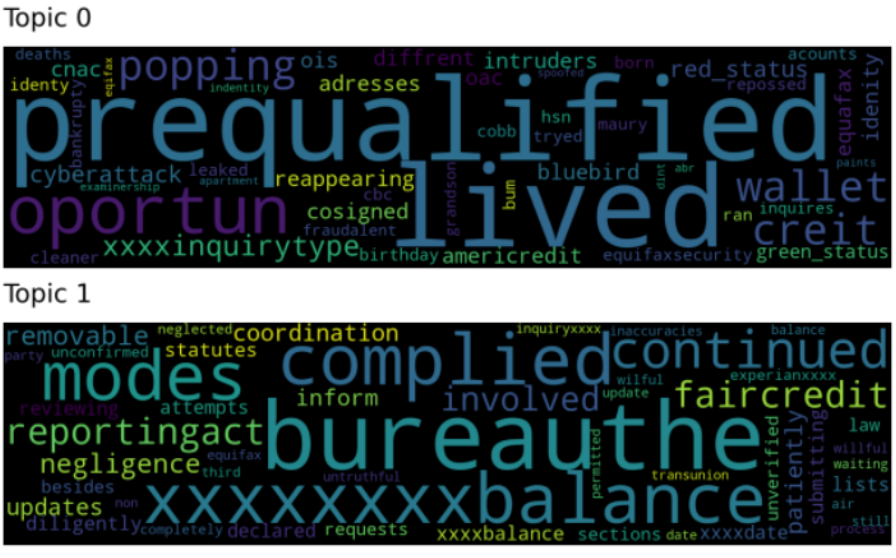
Topic: 9

Words: 0.022*"debt" + 0.018*"request" + 0.018*"provid" + 0.015*"valid" + 0.012*"document" + 0.012*"collect" + 0.012*"alleg" + 0.012*"claim" + 0.011*"proof" + 0.011*"origin"

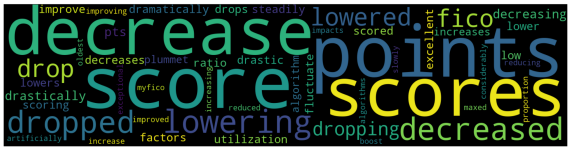
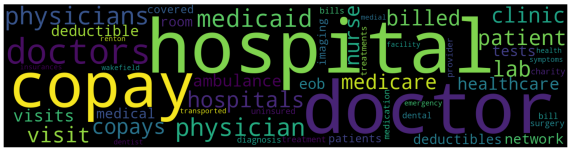
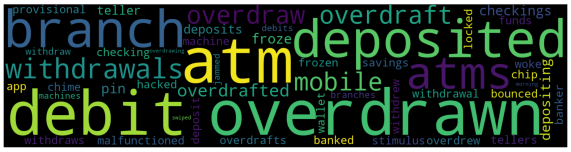
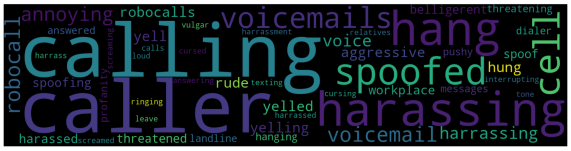
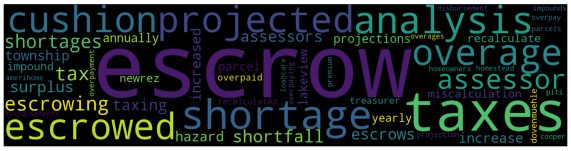
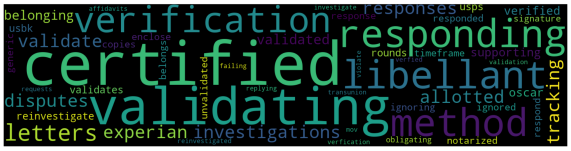
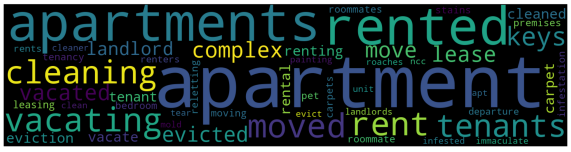
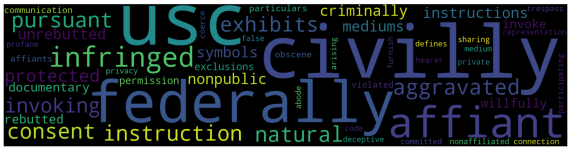
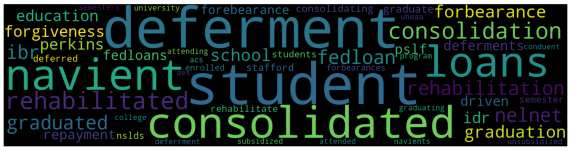
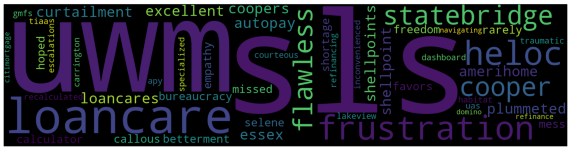
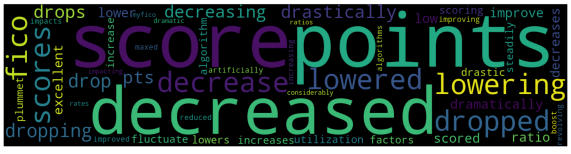
Appendix 18: Words Occurring in Each Topic Created by the LDA



Appendix 19: Distribution of LDA Topics per Issue



Appendix 20: Wordclouds of the Topics of Top2Vec with min. Cluster Size 5000

Top2Vec_Full	Top2Vec_1000
Topic 0 	Topic 0 
Topic 1 	Topic 1 
Topic 2 	Topic 2 
Topic 3 	Topic 3 
Topic 4 	Topic 4 
Topic 5 	Topic 5 
Topic 6 	Topic 6 

Appendix 21: Comparison of Top 7 Topics from Top2Vec_Full and Top2Vec_1000