Checkpoint 4: Machine Learning Findings

The Enchanted Badgers
Alexander Einarsson, Sergio Servantez, Marko Sterbentz
November 12, 2020

The theme of our project is to identify meaningful categories for uncategorized complaints by analyzing the relationship that exists between the complaint category and the complaint report narrative, and to use this new information to explore various aspects of complaint investigations and outcomes. To this end, we sought to answer the following set of questions by applying machine learning techniques:

1. We would like to model new complaint categorizations based on the raw complaint text. The relevant fields for this would be the "summary" column in the "data_allegation" table. We will use latent Dirichlet allocation (LDA) to build these new categories.
2. We would like to determine whether a complaint's summary and category type are predictive of its final outcome. To do this, we will build a transformer language model and evaluate its accuracy in predicting the outcome of a complaint given the summary and category.

The answers and analysis are provided below.

Question 1: We would like to model new complaint categorizations based on the raw complaint text. The relevant fields for this would be the "summary" column in the "data_allegation" table. We will use latent Dirichlet allocation (LDA) to build these new categories.

Latent Dirichlet Allocation is an unsupervised machine learning technique for topic modeling. The algorithm takes in documents of words, together with a pre-set number of topics *k* and words per topic, and finds which words belong to which topic. The algorithm treats each document as a bag of words, meaning the order doesn't matter.

To form the topics, the algorithm initializes each word in each document randomly to one of the *k* topics, then goes through each word *w* in each document *d* and calculates the proportion of words in that document belonging to topic *t*, excluding word *w*. If many words in *d* belongs to *t*, it's likely that *w* also actually belongs to *t*. It then calculates what proportion of words in *t* is exactly the word *w* over all documents, as if *w* has a high probability of being in *t* then any document containing word *w* will be more associated with topic *t*. The algorithm then multiplies these two probabilities to get the probability that word *w* has topic *t*.

We believe topic modeling, and in particular the commonly used LDA, is a good fit for our project because we are interested in different and new ways of categorizing complaints, and we want to explore doing so both in a supervised manner (which we explore in checkpoint 5), and in an unsupervised way by only taking into account what the officers wrote as a summary. By

letting the LDA create the topics for us in an unbiased manner, looking only at the narrative summaries, we want to examine both whether the created topics are interpretable for humans, and if they provide another venue for categorization for people who want to do research on the data.

Because the number of complaints with narrative summarizations was relatively small, we immediately decided that we need to keep the number of topics low. We decided on 20 topics to match the number of supercategories (categories in the original dataset). This assumes that none of the original categories need to be merged or split into more distinct subcategories. But without the domain knowledge to determine the appropriate number of categories a priori, we decided this was the best route. The choice for the number of words per topic was largely arbitrary. We found that 12 words seemed to give us interpretable and relatively concise topics. Because the LDA modeling is non-deterministic, we decided that it would be worth our time to train several different models and pick the best one among them.

Before building the models, we did some rudimentary text pre-processing, and after looking through what words occurred the most frequently in the summaries, we opted to add some domain specific words to the list of stopwords. Words such as “allege”, “accuse”, and “officer”, while not present in every summary, occurred often enough to appear meaningful to the algorithm. However, we knew that most of these words would occur often across categories without having any deeper meaning in this domain. For this reason, we examined both topic words and graphs visualizing word frequency in order to identify candidate words to be added to the stopword list.

We trained 10 different models on the dataset using the hyperparameters outlined above, and decided on the final model through qualitative analysis. We ranked the models based on how human interpretable their topics were according to our group members, and then created bar charts of the word frequencies in each topic for each model. Thus, our final selection of a model (model 7) was based on interpretability and distribution across topics.

While the model topics are largely interpretable, a small number of them prove difficult to map from the words to a distinct topic, instead containing mainly words that are either departmental or filled with domain-specific words. Because of our relative lack of domain expertise, we opted to accept those topics as possibly worthwhile in the model. We examined them via visualizations earlier in the project, and determined that they didn’t appear to be “catch-all” topics, but future research by people with more domain knowledge is required to make a final determination as to whether the model should contain more topics to allow for catch-all topics.

The algorithm proved largely successful in finding interpretable topics purely from the narrative summaries. Several of the topics map directly onto the categories from our original dataset, such as “grab”, “push”, “punch”, “head” clearly relating to the Use of Force category, and “search”, “warrant”, “apartment”, “enter” relating to the Illegal Search category.

We also used the final topics to examine the uncategorized category from the original dataset, and found that the 101 complaints largely mapped onto two topics. Both of these topics contained features prominent in the original Supervisory Responsibilities category. We may theorize that complaints inside the department were more likely to be uncategorized because the complaints reveal more serious issues within the department, but that is pure speculation. More research is needed, preferably from people with more domain expertise than the members of this group.

In addition to some of the topics directly relating to the original category, the algorithm also proved successful in breaking categories into multiple meaningful topics. For example, the model created multiple topics including the word 'search', potentially related to the Illegal Search category. Looking primarily at topics #11 and #14, we see that topic #11 appears to relate to Illegal Search of vehicle, including words such as 'justification', 'stop', 'vehicle', 'impound', and 'traffic'., while topic #14 seems to indicate a relation to Illegal Search of the complainants home, containing words such as 'apartment', 'enter', 'residence', and 'door'. It's promising that the algorithm managed to break down the topics into more granular categories, as this may let researchers build categories not currently present in the larger dataset.

While a single unsupervised algorithm never would completely change the landscape of the handcrafted categories, we have shown that by merely using the narrative summaries of the complaints we can craft meaningful and interpretable topics for classification, and that these topics are robust enough to create consistent mapping from the unlabeled/uncategorized category onto the algorithm-generated topics. While it doesn't replace the handcrafted and very granular categorization in the original dataset, it may serve to augment people who are interested in critically reviewing the current categories. Because of the relative lack of complaints with an accompanying narrative summary, there is also a chance that the algorithm can create more meaningful topics given more data, and that more data would also allow for a greater number of topics, which would increase the granularity.

We recommend that more research be done in this field, and note that by following the steps laid out in the Appendix of checkpoint 3, a person with domain expertise in policing, but perhaps lacking some expertise in data science, can easily run the LDA and generate their own topics for a qualitative and critical review of the current categories in the CPD complaints data.

Question 2: We would like to determine whether a complaint's summary and category type are predictive of its final outcome. To do this, we will build a transformer language model and evaluate its accuracy in predicting the outcome of a complaint given the summary and category. *(Originally: "We would like to model how consistent the police categorizations are for a complaint (e.g. would two identical or similar complaints be labeled the same way?). The relevant fields would be the "summary" and the "most_common_category_id_integer" columns in the "data_allegation" table.)*

Please note that this question is different than the one we had originally planned to answer. As our project developed and we began answering questions as part of past checkpoints, this new

question seemed much more pertinent to answer and better fits our overall project theme. Additionally, as we began thinking more deeply about ways to measure complaint similarity automatically, we found that we ultimately were going to be looking at proxies for complaint similarity like complaint length, word similarity, etc, rather than the semantics of the complaints. As a result of changing this question, we no longer need to rely on similarity metrics that are mere proxies to what we are really looking for and unlikely to provide any satisfactory answer to the original question.

In order to begin answering this new question, we first needed to gather and clean the dataset with which we were going to train the model. For Checkpoint 2, we had cleaned and integrated narratives from the database with a set of narratives that the Invisible Institute had acquired more recently. For more information about that process, please see the Appendix: Cleaning and Integrating Additional Narratives from the Checkpoint 2 writeup (up to the section entitled “Using Language Models to Classify Uncategorized Complaints”).

The resulting narratives dataset consisted of 16010 complaints with narrative summaries, their associated IDs, the complaint category, and the final outcome. However, not all of these complaints had known or non-null final outcomes. There were many rows that had values of NULL or “Unknown” for the final_outcome column. As a result, the complaints with no known final outcome were removed from the dataset since we need labeled data for training the transformer. This resulted in 10029 total samples with the following final outcome class distribution:

Final Outcome Class	Number of Complaints
1 Day Suspension	219
10 Day Suspension	100
12 Day Suspension	7
120 Day Suspension	5
15 Day Suspension	39
2 Day Suspension	75
20 Day Suspension	30
25 Day Suspension	16
28 Day Suspension	6
3 Day Suspension	76

30 Day Suspension	48
365 Day Suspension	8
4 Day Suspension	10
45 Day Suspension	12
5 Day Suspension	96
6 Day Suspension	8
60 Day Suspension	9
7 Day Suspension	19
90 Day Suspension	5
No Action Taken	8544
Penalty Not Served	45
Reinstated By Court Action	7
Reprimand	305
Resigned	72
Separation	18
Suspended Over 30 Days	9
Violation Noted	215
other_outcome	26

Note that any final outcome class that had less than 5 samples associated with it was lumped into a category called “other_outcome”. We do this since any category with less than 5 training samples likely does not have enough training data to properly infer outcomes belonging to this type.

Additionally, by looking at the class distributions above, we can see that they are heavily skewed towards “No Action Taken” which has 8544 complaints associated with it, and represents roughly 85% of the samples. Training a model on this data would likely result in a model that learns to predict “No Action Taken” most of the time, and would have quite high accuracy on the test set as well. To alleviate this issue and better balance the class distribution, we sought to reduce the number of samples for “No Action Taken” and thus took a random

sample of 10% of these complaints and used that as part of our final dataset. The resulting dataset has the same distribution as above, but with just 854 complaints with the final outcome of “No Action Taken,” resulting in 2339 total samples in the dataset.

With this dataset in hand, the last major step was to put it in the proper format so that we’re using both the complaint category and summary to predict the final outcome class. To do this, we add special tokens around the category name and prepend that to the beginning of the complaint summary. We used the special tokens “@@”, and the training samples thus looked liked the following example taken from the dataset:

> @@Operation/Personnel Violations@@ An off-duty CPD Officer was alleged to have failed to report that he discharged his firearm, failed to immediately identify himself as a police officer, failed to submit a report regarding the discharge of his firearm, impeded the investigation when he falsely reported that he did not discharge his firearm, and failed to properly secure his firearm after it discharged and malfunctioned. It is also alleged that the Officer provided several counts of false statements, including telling the first responding officers that he had discharged his firearm, telling the sergeant that he did not discharge his firearm, stating he never spoke to an uniformed sergeant, informing every officer that he spoke with that he discharged his firearm, and stating that he made a timely notification of the discharge of his firearm to the Department. Finally, it is alleged that the Officer brought discredit upon the Department regarding the circumstances of the discharge of his weapon.

This example had the label “Resigned”.

We then make use of Scikit-Learn’s train_test_split() function to divide the resulting set of samples into training, validation, and testing data sets with a train/validation/test split of 60/20/20. We then create three TSV files containing the set of samples and their class (i.e. final outcome) for each of the training, validation, and testing data sets.

With the final training and testing datasets put together, we can finally train the language model classifiers. For this task, we trained two models: a bag of embeddings model and a BERT transformer model. The bag of embeddings model is similar to a bag of words model except we use word embeddings rather than raw words or n-grams.

The training process makes use of the AllenNLP framework, which provides a wrapper around a variety of transformer implementations, including the BERT model. The BERT model has already been pretrained on a massive corpus of text, and all we need to do now is add a classification output layer and fine-tune it on the set of complaint narratives with categories and their final outcome classes. The code for this is included in the `src` directory of our submission, as are the instructions for training and testing the model. The final script used for producing the training, validation, and test data is `process_raw_data.py` in the `src/q2` directory of Checkpoint 4.

The result is a bag of embeddings classifier which achieves a classification accuracy of 53.6% on the test datasets, and a trained BERT classifier that achieves an accuracy of 53.1%. The accuracy of both of these models is quite low, and indicates that perhaps the complaint summary and category are not predictive of the final outcome. However, in a more ideal world, it would seem that these complaints would actually be predictive of their final outcome (assuming the complaint is an accurate accounting of the situation).

Yet, we're not seeing this predictive power in our models. There are a number of possible reasons for this. The first is that a large portion of the complaints do not capture all of the details of the situation. This would result in the model making incorrect predictions because there is some confounding factor that leads to the outcome being something other than what might be expected. However, there is nothing we can presently do to alleviate this situation if it is the case. Similarly, some of the complaint summaries are quite noisy, with some containing incomplete sentences and words joined together when they should not be. Again, this is something that would be difficult for us to control for, and would require much more thorough and manual data cleaning.

Another possible reason is that the model is unable to adequately distinguish between the multitude of suspension outcome classes. In the original distribution of final outcome classes there are 20 different Suspension classes that differ only in the length of the suspension (e.g. "1 Day Suspension" vs. "3 Day Suspension). It is quite possible that the model can in fact recognize a suspension would be the ultimate outcome, but is unable to determine the length of the suspension. To test this, we performed another round of processing on the dataset and bucketed the suspensions into just two suspension outcome classes: Short Term Suspension and Long Term Suspension. Short Term Suspensions are defined as any suspension that has a length of 30 days or fewer, and Long Term Suspensions are those that last more than 30 days. The final class distribution after this bucketing is as follows:

Final Outcome Class	Number of Complaints
Long Term Suspension	48
No Action Taken	854
Penalty Not Served	45
Reinstated By Court Action	7
Reprimand	305
Resigned	72
Separation	18

Short Term Suspension	749
Violation Noted	215
other_outcome	26

The total number of samples is the same as before (2339 samples) and is processed and split into training/validation/testing sets as before (a 60/20/20 train/validation/test split). We again train the bag of embeddings and BERT models on these samples. The bag of embeddings model attains an accuracy of 67.7% on the test dataset, and the BERT classifier achieves an accuracy of 69.8%. These are 14-16% higher than previously, yet the accuracy remains quite low. This would suggest that the complaint summary and category has limited capabilities when it comes to predicting the final outcome of the complaint. It is possible that with a better categorization of the complaints (perhaps using the categories produced by the topic modeling performed in Question 1) we would be able to better predict the final outcome of a complaint filed with the Chicago Police Department.