CS229 Final Project - Medical Record Understanding

Justin Fu
justinfu@stanford.edu

Daniel Thirman
dthirman@stanford.edu

1 Introduction and Background

After completing an examination or treatment with a patient, doctors record detailed notes known as medical records. These records typically include a summary of past medical history, medications, a brief hospital course, discharge diagnoses, etc.

The ICD coding system is a relatively comprehensive system with support for most symptoms, operations, and diseases (the ICD-9 system has over 17,000 codes, and the ICD-10 system has over 100,000). ICD codes are important because they are a computer-readable summary that is invaluable for collecting statistics, and hospitals use these codes to predict risk factors, mortality rates, etc.

The problem of automatic ICD coding gained significant interest in the biomedical informatics research community following the release of the "2007 Computational Medicine Center International Challenge: Classifying Clinical Free Text Using Natural Language Processing". In particular, two styles of approaches were popular, rule-based algorithms (such as [5]), and machine learning algorithms. Rule-based methods were found to be surprisingly effective [1].

Previous work on using text-based machine learning algorithms have typically relied on using bag-of-words features and expert-crafted rules [4]. However, such methods have been shown to scale poorly with the large label space [3], and no results to date have achieved a practically usable accuracy on this problem. Recent work has applied intuitions about the structure of the problem, such as using hierarchical SVMs [2] to leverage the fact that the many labels are specific instantiations of others.

The goal of our project is to understand the errors made by previous work on this problem, discover additional structure in the data, and leverage that to build a better model and improve accuracy.

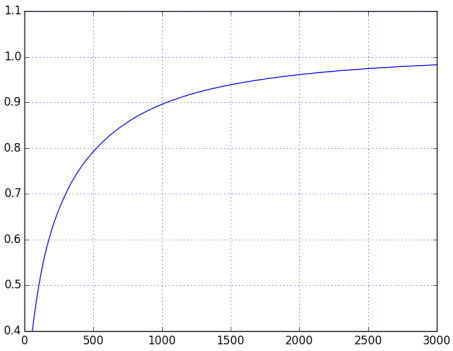

Figure 1: The distribution of codes on the MIMIC III dataset. The x-axis is the number of codes, and the y-axis is the percentage of coverage over the dataset.

2 Dataset

We used the MIMIC III dataset for our project, which contains roughly 50,000 emergency room discharge records from a single hospital. These notes were hand-labeled with ICD-9 codes by 3 experts (two hospitals and a company) and then aggregated together. A single note typically has 10-15 codes.

There are several well-known challenges with existing medical record datasets, some of which we do not attempt to tackle in this project:

1. **Label distribution**: The distribution of labels is incredibly lopsided, since only a small minority of symptoms and diseases are common. In our dataset, a total of 6985 labels were present, but the first 500 labels accounted for approximately 80% of all codes, and approximately 1500 codes had only 1 example. This distributino is shown in figure 1.

2. **Label noise**: ICD codes are known to be very noisy [6]. For example, in the cited report,

65% of Alzheimer's cases were present in the notes but not coded, and 5% of cases were coded without evidence in the note. Similar numbers hold for other diseases.

3 Baseline Model

3.1 Model

For our baseline model, we used a multi-label, multi-class logistic regression model with L1 regularization to enforce sparsity of features. We implemented our model using Theano and used Scipy for optimization.

3.2 Features

We tried several variations of bag-of-words features, and in the end, we settled on using the SPECIALIST lexicon as a dictionary containing relevant words and phrases. Phrases proved to be important, for example, the feature "urinary tract infection" is a great feature for the urinary tract infection, but the individual words are very generic and low-precision.

In total, there were 484,628 phrases we extracted from the lexicon, and after filtering by frequency in our dataset, we ended up with approximately 38,556 features.

3.3 Results

While our entire dataset had 6985 total labels, we focused on a small 9-label subset in order to do in-depth error analysis. For this subset, we achieved an macro-averaged F1 score of 60%. We scaled up to 30 labels, this score dropped to 45%. We report our results in table 1, and the top 3 features for each code in table 2. Unfortunately, we did not yet find other paper in the literature that uses the MIMIC III dataset, but F1 scores in previous work have typically ranged from the 80% range on 10 codes (with hand-engineered features), down to about 15% on 700 codes [2].

3.4 Results

Surprisingly, the features learned were very reasonable, except for "Pulmonary Hypertension", which had the lowest training support out of all 9 labels.

Features learned were typically drug names (such as "Albuterol", "Singulair" for Asthma), naming variations (such as HCAP, for hospital acquired pneumonia), or generally related procedures (such as intubation/extubation for respiratory failure).

3.5 Error Analysis

We observed several phenomena during our error analysis, and we categorized several commonly occurring ones. Typically, different codes present different types of errors, which means that this list is likely incomplete since we only analyzed 9 codes in-depth.

3.5.1 Overfitting to Noise

This was one of the most common errors made by our model. For false negatives our model typically picked up a single strong signal that was correct, but it was drowned out by several hundred smaller features which summed up to cancel out the correct signal (each note has on the order of 300-700 features firing). A typical false positive looks like (in this case, we are trying to predict pneumonia):

Top Features	Weight	Bottom Features	Weight
pneumonia	2.108	ESLD	-0.233
PNA	1.130	neither	-0.228
levofloxacin	0.837	blastic	-0.210
LLL	0.448	mg oxide	-0.142
lower lobe	0.282	high normal	-0.130

A false positive typically has many somewhat related (such as "intubation"/"extubation" for acute respiratory failure) or completely unrelated words with small positive scores that sum up to a certain threshold. Although we added L1 regularization to enforce sparsity, it only zeroed out some of the weights and many were still left with small ones.

3.5.2 Label Structure

ICD-9 codes form a hierarchy, but if a patient has some specific disease such as "diabetes with renal complications", the doctor does not apply every code on the path from the root to the code of interest. Instead, there are specific coding guidelines, such as only the most specific code should be used, or that certain codes are mutually exclusive. If our model predicted diabetes and kidney failure, we would be wrong. In our 9 code analysis, we only used the most general forms of diseases, and thus, we had false positives whenever a more specific form of a disease was present.

3.5.3 Requires deeper inference/computation

A large class of these errors arose from test results in which the only evidence for a blood-related diagnoses is a low hematocrit (% of red blood cells

ICD Code	Precision	Recall	F1 Score	Support
Urinary Tract Infection	0.82	0.75	0.78	1057
Thrombocytopenia	0.60	0.39	0.47	493
Pneumonia	0.59	0.49	0.53	756
Acute Resp. Failure	0.70	0.71	0.71	1173
Anemia	0.49	0.42	0.45	884
Cardiac Arrest	0.63	0.37	0.47	209
Asthma	0.72	0.47	0.57	354
Rheumatoid Arthritis	0.78	0.28	0.41	112
Pulmonary Hypertension	0.06	0.02	0.03	56
Macro-averaged total	0.65	0.56	0.60	5094

Table 1: Test set F1 scores for baseline model.

ICD Code	First	Second	Third
Urinary Tract Infection	UTI	Urinary Tract Infection	Urinary Tract
Thrombocytopenia	Thrombocytopenia	HIT	Antibody
Pneumonia	Pneumonia	HCAP	Hospital Acquired Pneumonia
Acute Resp. Failure	Respiratory Failure	Intubation	Extubation
Anemia	Anemia	Normocytic	Dilution
Cardiac Arrest	Arrest	PEA	CPR
Asthma	Asthma	Albuterol	Singulair
Rheumatoid Arthritis	Rheumatoid	Rheumatoid Arthritis	Arthritis
Pulmonary Hypertension	Moderate	Contrast	Although

Table 2: Top indicators for baseline model.

by volume, for anemia) or low platelet count (for thrombocytopenia). For example:

Test Result	Diagnosis
PLT SMR-LOW PLT COUNT-84*	Thrombo.
Hematocrit is 28.8; platelets 357.	Anemia
Hematocrit of 36; platelet count 183.	Thrombo.

Our model can only pick up on single words, and cannot execute logic such as comparing test results against a threshold.

Some more difficult errors are ones involving judgement of the severity of an illness. Our model commonly predicted false positives for "acute respiratory failure" when the true label was "acute respiratory distress", since outside of explicitly mentioning "distress" and "failure", the two codes present similar features. The only difference we observed was that the "failure" cases were more severe than the "distress" cases, and indeed, the two codes are mutually exclusive by the ICD-9 coding guidelines.

3.5.4 Context

A nonzero amount of our false positives came from features firing from the "Medical History" section, which for some diseases should not be coded. However, for others, such as chronic diseases (such as rheumatoid arthritis, which is an autoimmune disease which attacks the joints), being present in the medical history is enough to justify a code.

3.5.5 Multi-word understanding

In many cases, the only good feature that fires is one related to a disease, but not necessarily enough to make a decision on its own. For example, the following is a sentence from a note coded with thrombocytopenia, or low platelet count.

```
She also had acute platelet drop
while on balloon pump and heparin.
```

"Platelet" is a feature that is obviously relevant to thrombocytopenia, but it is common to mention it with blood tests and other blood-related complications, so it is a very low-precision feature. However, the phrase "acute platelet drop" is very indicative of thrombocytopenia, especially when it is mentioned with heparin, which is an anticoagulant that commonly causes platelet levels to drop, resulting in heparin-induced thrombocytope-

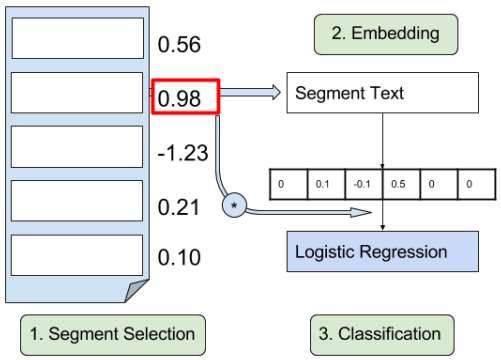

Figure 2: A diagram of our model

nia (abbreviated as HIT, which is one of the top features). Part of the issue here is that we don't have a good mechanism for automatically extracting good phrase features from the text.

4 Full Model

4.1 Model

We wished to revise our model in order to fix some of the errors we observed - in particular, we focused on enforcing sparsity to reduce overfitting, high-level context understanding, and multi-word understanding. A pictoral representation of our model is located in figure 2.

Our model is roughly split into 3 parts.

1. We divide each document into multiple segments (we used a heuristic rule and divided be section, such as "medical history", "hospital course"), and score each segment with a coarse model. We simply used a linear function with weights initialized from our baseline model. This segment selection process helps reduce the number of features that fire, and allows the model to assign low scores to sections such as "Medical History".

2. We embed the best segment, either using an RNN (with the word embedding layer initialized with word2vec) or as a bag of words.

3. We score the embedding using logistic regression (linear layer + softmax). To alleviate non-differentiability problems, we multiply the segment score into the logits before passing it into the softmax.

Due to time limitations, we have currently only implemented this model to work for binary classification on one code.

Additionally, there are a few architectural problems which we are in the process of figuring out how to solve. One is that this model trains extremely slowly due to the non-differentiability of using a max for segment selection, and we are currently multiplying in the score to the logits as a workaround instead of resorting to Monte-Carlo methods such as REINFORCE. The second is that noisy features still end up affecting the segment selection, which further slow down training since only weights from the best segment get adjusted due to the aforementioned differentiability problems.

4.2 Results

Again, due to how slowly the model trains, we have run very few experiments for this model. So far, we have done experiments on anemia with a balanced dataset consisting of equal numbers of positive and negative labels, and have achieved a test F1 score of 42% with the bag-of-words embedding, and 35% with the RNN embedding. These results are only slightly worse than our baseline, but the problem is much easier due to the balanced dataset.

5 Conclusions

We presented a baseline linear model, which to the best of our knowledge, has achieved comparable results to several other pure classification-based methods that have previously used.

We have not seen good results from our full model yet, but we are in the process of running more training iterations and possibly rethinking our architecture to make it easier to train.

6 Future Work

There are several potential problems of interest that we have not addressed in this project. One is exploiting the structure of the codes/labels, since we know that some labels are mutually exclusive, some are a conjunction of two other labels, and some are subclasses of other labels. The subclass observation has been explored in previous work, but not the others.

Another potential problem to tackle is addressing the codes which have little training support be exploiting outside knowledge. Each code has a short description, and using knowledge bases or the Internet can provide the additional information

necessary for a model to justify a label. However, since these are rare codes, it's not clear if there is a strong practical need to get these correct, and much of previous work (especially ones with hand-engineered features) have only focused on subsets of codes, as we have done in this project.

Finally, an important problem to tackle is the issue of noisy labels in the dataset. We wish to apply ideas such as semi-supervised learning and bootstrapping to improve our model's ability to handle label noise.

References

[1] Ira Goldstein, Anna Arzumtsyan, and Ozlem Uzuner. Three approaches to automatic assignment of icd-9-cm codes to radiology reports. In *AMIA Annual Symposium*, 2007.

[2] Adler Perotte, Rimma Pivovarov, Karthik Natarajan, Nicole Weiskopf, Frank Wood, and Nomie Elhadad. Diagnosis code assignment: models and evaluation metrics. In *Journal of the American Medical Informatics Association (JAMIA)*, 2014.

[3] Stefano G. Rizzo, Danilo Montesi, Andrea Fabbri, and Giulio Marchesini. Icd code retrieval: Novel approach for assisted disease classification. In *Data Integration in the Life Sciences (DILS)*, 2015.

[4] Suchi Saria, Gayle McElvain, Anand K. Rajani, Anna A. Penn, and Daphne L. Koller. Combining structured and free-text data for automatic coding of patient outcomes. In *AMIA Annual Symposium*, 2010.

[5] Illes Solt, Domonkos Tikk, Viktor Gal, and Zsolt T. Kardkovacs. Semantic classification of diseases in discharge summaries using a context-aware rule-based classifier. In *Journal of the American Medical Informatics Association (JAMIA)*, 2009.

[6] Wei-Qi Wei, Pedro L. Teixeira, Huan Mo, Robert M. Cronin, Jeremy L. Warner, and Joshua C. Denny. Combining billing codes, clinical notes, and medications from electronic health records provides superior phenotyping performance. In *Journal of the American Medical Informatics Association (JAMIA)*, 2015.