Assigning medical codes to each hospital patient admission is an expensive and tedious process, particularly because as of now, it is performed manually by doctors and nurses. These codes are used across different healthcare agencies, meaning that this deceptively small task actually spans a multi-billion dollar industry. As a result, it is important to minimize the possibility of errors and biases in code labeling. This project aims to remove human error from this process by using natural language processing to predict proper codes for a given admission. Furthermore, it attempts to utilize natural language processing to provide explanations for why a specific code was chosen for a given case.

To do this, we attempt to interpret the clinical notes that accompany medical code assignments and identify key phrases that indicate which label is most fitting. The dataset used is MIMIC-III, which contains notes detailing patient admissions and their medical codes at the Beth Israel Deaconess Medical Center. The text is passed through a hierarchical attention network, made up of a word GRU encoder, a word attention layer, a sentence GRU encoder, and a sentence attention layer. By using network architecture that mirrors the hierarchical structure by which a document is built on, the design offers a more detailed processing of the word and sentence levels that make up a document. The attention mechanism allows the machine to not only predict the most accurate code assignment, but also offer clarification into why a code was chosen for a specific case. Such a improved system for medical coding will eliminate costs and improve efficiency in healthcare.
1 Introduction

The International Classification of Diseases (ICD) provides alphanumeric codes for doctors, health insurance companies, and public health agencies to specify various diseases, disorders, injuries, infections, and symptoms. One or more of these medical codes are assigned to every hospital admission, which is then used across different healthcare parties. These codes are arranged hierarchically, and some examples can be seen in Figure 1. While the codes provide a standard identification protocol across different healthcare parties, they are often poorly annotated and do an inadequate job explaining the reasoning behind a diagnosis[4]. Additionally, manually labeling the patients can be prone to error[2] or bias to receive greater reimbursements[3]. According to the Centers for Medicare & Medicaid Services (CMS), in the fiscal year 2017, these errors caused an estimated $36.21 billion in improper payments[1].

In an effort to minimize these costs, there have been some attempts to use machine learning to standardize code assignment across hospitals. This is done by using natural language processing to recognize, extract, and interpret the other documents that accompany medical codes to predict proper coding and billing. Additionally, this data can be used for predictive purposes to decrease the likelihood of error or bias. Creating a system that predicts the codes based off clinical notes holds great potential for the healthcare industry.

1.1 Problem Description

This project aims to predict ICD-9 medical codes based off healthcare professionals’ clinical notes. As mentioned before, this is in an effort to eliminate the error and bias of manual code assignment, as well as to use NLP why a specific code is chosen for a patient.

Part of the problem with clinical notes is that there is often inconsistencies across different hospitals and different doctors. This includes the length of documents, style of notes, and how the data is stored. As a result, this project only uses notes from a single hospital. Another challenge is that the label space is relatively large. ICD-9 has approximately 13,000 codes, where each code is 3-5 characters in length. Furthermore, the transition from ICD-9 to ICD-10 will prove an additional challenge as the codes become more specific (3-7 characters...
in length) and increases the space to approximately 68,000 codes in total. This project specifically uses only data with ICD-9 codes.

1.2 Existing systems

Although medical codes are currently manually labeled, there have been some preliminary studies looking into the possibility of using neural networks for predictive coding. Most pertinent is the study performed by Mullenbach et al. in which they designed a model called Convolutional Attention for Multi-Label classification. This model passes text through a convolutional layer to compute a base representation of the text, then combines this with attention weights that select the most relevant parts of the document for each possible code. Their use of an attention mechanism allows them to distinguish which text snippets were the most significant in predicting code labels for clinical texts[4]. This provides clarification about the rationale behind each code assignment on top of the predictive code labeling, an important feature for this project.

Another model combines the bridges the gap between context-based word embeddings and topic modeling, which represents ignores order and semantic relationships among words by representing documents by word distribution[5].

These current studies do not use state-of-the-art word embedding methods, usually just word2vec. Additionally, there is room to explore different levels of granularity for attention mechanisms. Newer methods can provide more in-depth information about the context of each word. However, the approach of Mullenbach et al. provides a clear example of how NLP can be used to explain the logic behind a code assignment.

2 Approach

2.1 Network Architecture

This project uses a hierarchical attention network to process documents, which is shown in Figure 2. This structure is based off the model created by Yang et al[6]. The network contains several parts, namely a word encoder, a word attention layer, a sentence encoder, and a sentence attention layer.

2.1.1 Sequence encoder

Each encoder is a GRU based sequence encoder, composed of a reset

![Figure 2: Architecture of the Hierarchical Attention Network](image)
gate and an update gate. At each step, the new state of the GRU is:

\[ h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \] (1)

where \( \tilde{h}_t \) is equivalent to:

\[ \tilde{h}_t = \tanh (W \cdot [r_t \odot h_{t-1}, x_t]) \] (2)

The reset gate decides how much past information to forget, and is defined by the following equation:

\[ r_t = \sigma (W_r : [h_{t-1}, x_t]) \] (3)

The update gate learns how much information to pass on and defined by the following equation:

\[ z_t = \sigma (W_z : [h_{t-1}, x_t]) \] (4)

### 2.1.2 Loss Function

The loss function used was binary cross-entropy loss:

\[ \mathcal{L}_{BCE}(X, y) = - \sum_{\ell=1}^{L} y_{\ell} \log(\hat{y}_\ell) + (1 - y_{\ell}) \log(1 - \hat{y}_\ell) \] (5)

### 2.2 Datasets

The dataset that is used is MIMIC-III, a public relational database that can be accessed online. The database contains information relating to patients from the intensive care units at Beth Israel Deaconess Medical Center. More specifically, the table that is used is the ‘noteevents’ table, which contains all notes for patients. Each admission is labeled with one or more ICD-9 codes, indicating the diagnoses and procedures used during the patient’s stay.

Figure 3 shows the size of the MIMIC-III dataset. Because both the input data and the number of labels are so large, an abbreviated dataset of the top 50 most frequent labels was first used for training. This included the instances that had at least one of the top 50 codes.
2.3 Evaluation Methods

For each epoch, a number of metrics were collected. The metrics include accuracy, precision, recall, F1, and area under the ROC curve (AUC). For each metric, both the micro-averaged values and macro-averaged values were calculated. Micro-averaged refers to treating each (text, code) pair as a separate prediction, while macro-averaged refers to averaged metrics per-label, placing more emphasis on rare label predictions.

3 Results

The results of the HAN model on the 50 most frequent labels dataset can be seen in Figure 4, while the baseline comparisons (from the Mullenbach et al. paper) are seen in Figure 5. Compared to the baseline results, neither the AUC nor the F1 scores were particularly high when run on the test data - not even compared to logistic regression. However, after observing the much higher scores on the training data, it seems like the model was likely overfitting to the data. The training data was remarkably high, with a micro AUC of 0.947 and a micro F1 of 0.693. This overfitting is something that would be fixed if there were more time for the project, in future work.

![Figure 4: Hierarchical attention model results for 50 label](image1)

<table>
<thead>
<tr>
<th>Epochs</th>
<th>AUC</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Macro</td>
<td>Micro</td>
</tr>
<tr>
<td>50 epochs on test</td>
<td>0.7993</td>
<td>0.8422</td>
</tr>
<tr>
<td>50 epochs on train</td>
<td>0.9344</td>
<td>0.9467</td>
</tr>
</tbody>
</table>

![Figure 5: Baseline results based off Mullenbach et al. paper.](image2)
4 Conclusion

There is great potential for natural language processing to improve the efficiency and accuracy of medical code assignment. As ICD-9 expands into the more detailed, more expansive ICD-10 coding, these improvements will become increasingly relevant to having productive healthcare.

Future work includes integrating this model with CAML to see how applying different levels of granularity for the attention mechanism affects the predictions. Additionally, testing improved embedding methods such as BioBERT, rather than just using word2vec, could boost the performance of the model.

5 Acknowledgements

I would like to thank my advisor, Professor Dragomir Radev, for providing me with the opportunity to work on this project. I would also like to express a special thanks of gratitude to Yavuz Nuzumlali and Irene Li for their patience and instruction in narrowing down the design of this project, helping with troubleshooting, and offering invaluable advice.

6 Appendix

Code for this project can be found at:

References


