Natural Language Processing for Symptom Detection in Unstructured Provider-Patient Conversation

Anisha Agarwal ANISHA24@MIT.EDU
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA, USA

Tiana Cui TIANACUI@MIT.EDU
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA, USA

Carlo Duffy CARLOD@MIT.EDU
Institute for Data, Systems & Society
Massachusetts Institute of Technology
Cambridge, MA, USA

Pranav Murugan PMURUGAN@MIT.EDU
Department of Electrical Engineering and Computer Science
Department of Physics
Massachusetts Institute of Technology
Cambridge, MA, USA

Abstract

The often-unrecognized high symptom burden placed on patients decreases care satisfaction and increases avoidable healthcare costs. As such, accurate identification of patient symptoms is crucial, especially for data whose downstream application is for work in machine learning in healthcare. Analyzing clinician-patient conversations and building models to automatically detect symptom-relevant components of the conversation can improve the current symptom detection pipeline. Using oncologist-patient conversations from Duke Medical Center and the University of Pittsburgh Medical Center, we build machine learning models that predict the general relevance of each line of dialogue to symptoms. Our best predictions arise from a BERT model, with an accuracy of 0.91 and an AUROC of 0.91, but the much more light-weight LSTM model was not far behind with an accuracy of 0.90. In general, we find that models designed for general conversational natural language processing (NLP) may outperform specialized medical NLP models due to poor transferability and the nature of physician-patient dialogue. This empirical analysis is a promising first step towards automating symptom detection from conversations.

1. Introduction

Accurate identification of patient symptoms is crucial for studies in machine learning and healthcare, especially for downstream quality of care. Electronic health records (EHRs) have often been the primary data source for capturing patient symptoms, but clinicians

. The repository for this paper is available at https://github.com/pmuruga/symptom-detection

may bias EHRs' accuracy (e.g., some clinicians may document only symptoms they remember or deem important). Audio recordings of conversations between patients and clinicians, on the other hand, provide the least biased sources for unearthing symptoms: hidden side effects, indications of a drug's effectiveness, or early symptoms of illnesses could exist that clinicians would have otherwise failed to record. The challenge then becomes the analysis of these unstructured conversations.

Using recorded conversations between patients and their oncologists, research assistants at the Dana-Farber Cancer Institute judged the extent to which each line of dialogue relates to symptoms in general. Human annotation, however, is incredibly time-consuming. The goal of our project is to aid the research assistants in their symptom-relevance tagging task. Future work on this project can we map the parts of the conversation that have been tagged as "symptom talk" to the relevant symptoms.

In this work we experiment with four machine learning models to predict relevance to symptoms from conversations between patients and oncologists and to map symptom-related conversations to specific symptoms.

Generalizable Insights about Machine Learning in the Context of Healthcare

Machine learning for healthcare is often perceived as designing and training frontier models that can diagnose or treat patients at or above the level of care provided by doctors. While research along that vein is important and has certainly led to breakthroughs in machine learning, such a perception ignores an equally important first step in the process: helping doctors and researchers build clean datasets that can be leveraged for future machine learning applications. With our work in this domain, we hope to analyze and compare the many different ways (ranging widely in level of technical complexity) text corpora can be analyzed, represented and classified with machine learning. We also hope to highlight the value of machine learning in the domain of creating clean, well labeled datasets. More generally, we hope readers recognize the importance of building analyzable models that provide practical and immediate use to medical professionals.

2. Related Work

Using NLP to process unstructured medical text data has been an active problem for the last 20 years. Before 2017, there were many different standard approaches; these include support vector machines, Bayesian and Markov chain classifiers, as well as relatively simple deep learning approaches (Li et al., 2021; Koleck et al., 2019). In 2017, Habibi et al. introduced bidirectional LSTMs with conditional random fields (CRFs), a modification that improves the LSTM's performance in sequential classification tasks (Habibi et al., 2017). These models have been shown to perform well in named entity recognition (NER) and multi-output tasks (Habibi et al., 2017; Wang et al., 2018). BiLSTMs with CRFs were the state-of-the-art until 2019, when BioBERT was trained from the general text transformer model, BERT. BioBERT is a powerful text transformer model trained on medical texts which is known to be good at NER classification tasks, as well as more sophisticated sequence-to-sequence tasks (Li et al., 2021; Lee et al., 2020). BioBERT, often with other augmenting features, is the current state-of-the-art in NER task performance, but comes with a much

greater computational cost compared to the BiLSTM method, as well as potential concerns about generalizability between specialties (Li et al., 2021; Lee et al., 2020; Tian et al., 2020). Although interpretability becomes more challenging as model complexity grows, techniques like saliency maps or perturbative methods are useful to identify causative features for model behavior (Wallace et al., 2020, 2019). Similarly, leave-one-out/leave-n-out techniques are common in NLP and involve iteratively masking one or n words at a time and comparing the change in model predictions (Dunn et al., 2021).

3. Methods

The problem can be framed as a binary classification problem, where the input data is the text of each turn and the target is whether this turn is *relevant* to symptoms. Per discussed with the Lindvall team, we define a turn to be *relevant* if it has an average symptom rating of at least 2 across all annotators and *irrelevant* otherwise. More details about the data and the rating system can be found in the Cohort section.

We experimented with four models: Bag of Words, LSTM, BERT, and BioBERT. We will discuss the methodology of each of these models in details in the subsections below. As suggested by Dr. Lindvall and her team, we tried fitting each model using both the original data and the preprocessed data where key phrases are mapped to their corresponding symptoms. We will compare the results in the Results section.

3.1. Bag of Words

Bag of words is one of the oldest and simplest NLP techniques available. It is called a "bag" because it does not account for the position of which a word in a sentence, only whether a word exists and how many times it exists (Harris, 1954). For each sentence, it counts the number of times that each word appears in this particular sentence – a process known as count vectorization. These counts then become the features that we feed into some machine learning classifiers.

In terms of the choice of machine learning classifiers, We first used a logistic regression classifier in which, all else equal, each predictor is linearly related to the log-odds of a turn being relevant. Logistic regression allows for straightforward interpretations of associations between predictors and the target, but at the possible expense of predictive performance.

Second, we used an XGBoost model. XGBoost is an optimized distributed gradient boosting model designed to be highly efficient, flexible and portable (Chen and Guestrin, 2016). It often outperforms other methods, sometimes including deep learning methods, when the input data is structured (Bag of Words makes the text data structured).

3.2. LSTM

A Long Short Term Memory (LSTM) network is a type of recurrent neural network designed to learn and do inference on sequential data. These types of models extend the capabilities of traditional recurrent neural networks by including persistent and updateable hidden states

that significantly increase the effective memory of LSTMS (Hochreiter and Schmidhuber, 1997). These models have been widely used in the field of natural language processing, both in general and for medical data (Van Houdt et al., 2020; Jagannatha and Yu, 2016; Mascio et al., 2020). In particular, they provide a good balance between being sufficiently complex to model natural language data and still simple enough to train quickly using modern computational resources (in comparison to, say, a transformer-based model) (Mascio et al., 2020).

In this project, we use an LSTM, without the conditional random field that it is often accompanied by. Because we do not need word-level context in this task (because it is classification at the sentence level), we simply implement a bidirectional LSTM with a fully-connected layer to perform the classification task based on the latent features the LSTM learns. To prevent overfitting, we also used a dropout layer with $p = 0.2$ between the LSTM and fully-connected layer. Because of the class imbalance of the data, which was almost 90% negative samples, we used a weighted binary cross-entropy loss to weight the positive samples higher during training and improve the model's sensitivity.

3.3. BERT

BERT, which stands for Bidirectional Encoder Representations from Transformers, is a state-of-the-art model for Natural Language Processing tasks. It was developed and published in 2018 by several researchers at Google in the paper "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" (Devlin et al., 2018). It is trained on a massive corpus with 2.5 billion words from Wikipedia and 0.8 billion words from BooksCorpus. Unlike its precedents, BERT produces contextualized representations by jointly conditioning on both left and right context in all layers.

The original BERT models use 12 layers and 768 hidden nodes or 24 layers and 1024 hidden nodes (Devlin et al., 2018). While we don't have to train these models from scratch, fine-tuning these models is computationally expensive. Considering the limited capacity of our laptops and the possibility of future tuning on Dr. Lindvall's side, we decided to use BERT Small, which is a smaller and lighter version made available on Tensorflor Hub (Turc et al., 2019). BERT Small has 4 layers and 512 hidden nodes, which has shown to be sufficient for our task. Please check the appendix for more information about the structure we used for fine tuning.

3.4. BioBERT

BioBERT is a pre-trained biomedical language representation model. When we leverage generic pre-trained models for a healthcare use-case, we run the risk of our models performing poorly due to the domain shift: the words commonly used in biomedical text are very different from the words used in regular language, on which BERT was trained. BioBERT, on the other hand, was trained on a large-scale biomedical text corpora, which allows it to handle the domain shift from regular text to biomedical text. Since our data are conversations between doctors and patients, we have a mix of regular speech and biomedical speech in our corpora. This motivated us to try both BERT and BioBERT in our experiments.

Our method for BioBERT is as follows: First, we pre-process each "turn" (pre-defined segments of conversation, about 1-2 sentences long) to remove any stop words and punctuation. Then, we pass the pre-processed turns as input into a pre-trained BioBERT embedding model by Lee et al. (2020) to get embeddings for each turn. The embeddings returned are vectors of length 768 for each turn.

Since our final goal is to classify each turn, we use the embedding vectors to train a Linear Classifier. Our linear classifier consists of 3 fully connected layers with ReLU layer, 2 batch normalization layers, and a drop-out layer. Similarly to our procedue for the LSTM, we use a Binary Cross Entropy Loss for training the linear classifier, and weight the positive samples 20 times higher than the negative samples to make up for class imbalance. Additional details regarding our model architecture can be found in our git repository, as well as Appendix A.

4. Cohort

Our main dataset takes conversations and represents each line, or turn, of dialogue (between a patient and oncologist) as a row. There are roughly 79,000 rows after removing duplicate rows (each turn can sometimes be annotated by multiple annotators). Besides the text from the dialogue, we can also see each turn's speaker role (facilitator, clinician, patient, family or other), patient ID, patient sex and human annotations for function content, illness understanding content and symptoms content. Each human annotation is a discrete score from 0 to 3, with 0 being not relevant to the topic and 3 being certainly relevant.

Another source of information is a symptom keyword library. Using a subset of conversations, the research collaborators at Dana-Farber focused on turns with a human-annotated symptom score of at least 2, extracted their words, and added symptom-relevant words to the library. These words were then grouped into one or multiple symptoms.

4.1. Cohort Selection

Conversations between patients and oncologists were originally recorded for the Communication in Oncologist-Patient Encounters (COPE) study, a randomized clinical trial by Porter et al. that tested whether COPE–an online tool "that teaches patients how to communicate their emotions to their oncology providers"– increases patient propensity to express emotional concerns to oncologists (Porter et al., 2015).

Porter et al. detail the data's population. The conversations occurred between 2010 and 2014, and were recorded at Duke Medical Center (DUHS) and University of Pittsburgh Medical Center (UPCI). At both locations, clinicians and patients were recruited from medical, surgical and radiation oncology clinics. DUHS has 30 medical oncologists, 16 radiation oncologists, 8 hematologists treating malignancy and 23 fellows. At DUHS, during the COPE study, 73 percent of patients are white, 19 percent African American, 1 percent Asian and 0.6 percent Hispanic. UPCI has 28 medical oncologists, 6 radiation oncologists, 16 hematologists treating malignancy and 25 fellows. At UPCI, during the study, about 82 percent of patients are white, 9 percent African American, and 1 percent other races

(Asian, Hispanic, etc.). At both locations, there are approximately equal shares of male and female patients.

Given this population, Porter et al. selected patients for the study based the following criteria: the patients had to (i) speak English, (ii) be treated for advanced malignancies, (iii) have access to the Internet and email, (iv) not suffer from physical ailments that would diminish mental capacities or preclude internet use, and (v) experience noticeable "emotional difficulty with their cancer" (Porter et al., 2015).

4.2. Data Extraction

Originally, the data presented turns classified on a scale from 0 to 3, where 0 meant the turn was irrelevant to symptoms, and 3 meant the turn was extremely relevant. As recommended by the researchers at the Lindvall Lab, we turned this multi-class classification problem into a binary classification problem by computing the average symptom score for each turn and labeling turns with score greater than or equal to 2 as `1` and less than 2 as `0`.

For the BioBERT and LSTM models, we avoid the problem of data leakage by using a customized train-val-test split function to ensure that data from the same patient all get assigned to the same split. Unfortunately, we were not able to re-train BERT with this split. We did, however, ensure that turns from the same conversations end up in the same split; this mitigates, but does not resolve, the data leakage issue for BERT. This is discussed more in the Limitations section.

It is also to the interest of the Lindvall team to examine the effect of preprocessing the data using the Symptom-Keyword Dictionary on the model performance. Due to the page limit, we include the results table in the appendix.

4.2.1. Bag of Words

When performing the count vectorization, there are two things that we tweaked. First, we removed the stop words, which are words that don't carry any meaningful information, e.g. "the", "a", "are". Second, we set the hyperparameter max_features (the maximum number of words that will be incorporated into the feature space) to be 1,000. We intentionally set it to be a relatively small number because most sentences are very short and thus a large max_features can result in a sparse feature space.

4.2.2. BioBERT

After pre-processing the turns to remove the stop words (words that don't carry weight with respect to the meaning of a sentence, such as "a", "the", or "it"), we linked the embedding vectors to their ground truth labels via the text of the respective turn. This is relevant because it caused our dataset to decrease from about 79k turns to 55k turns (many turns in the dataset, such as "No" and "Yes" were repeated many times across different conversations). Also, some turns with the same text may have been classified differently at different occurrences, and this implementation removes that granularity. However, because our model is designed as context-independent across different turns, these discrepancies in labeling of the same turn texts are considered noise rather than meaningful additional information.

Method	Accuracy	AUC
BoW + LR	0.88±0.01	0.86±0.02
BoW + XGBoost	0.87±0.01	0.81±0.02
LSTM	0.93±0.01	0.89±0.01
BERT	0.91±0.01	0.91±0.01
BioBERT	0.76±0.02	0.85±0.03

Table 1: Quantitative metrics for each model on the non-preprocessed data.

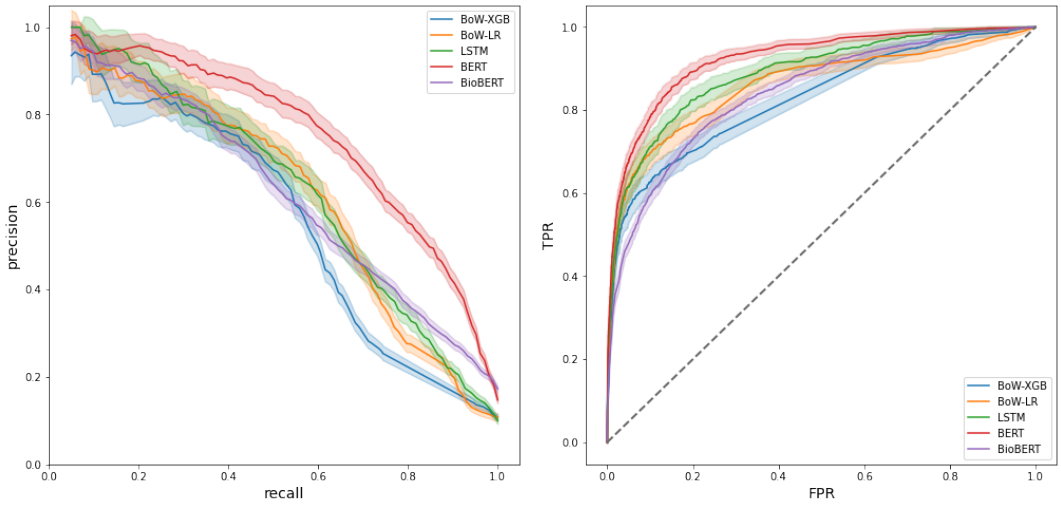

Figure 1: Left: Precision-recall curve for each method tested. Right: Receiver-Operator Characteristic (ROC) curves for each method. The dashed gray line indicates the curve for a perfectly random classifier. The associated AUROC for each of these curves can be found in Table 1. Error bars show the 95% confidence interval and were estimated with a bootstrap method using 250 resamples per threshold value.

4.3. Feature Choices

We choose to focus on the dialogue of each turn for our analysis, rather than any of the other features available in the data. Future work could analyze how including other features from the data might change the classification, but we decided it was best to focus on the text itself.

5. Results on Real Data

5.1. Evaluation Approach

We evaluated each model using four quantitative metrics: accuracy, AUC, precision, and recall. Please see Table 1 and Figure 1 for details.

In addition to the quantitative metrics, we also performed some qualitative analysis on the models. For the traditional Machine Learning classifiers that we used for Bag of Words, we are able to visualize the feature importance, but we are not able to do so for the neural network models. For these models, we focus on analyzing where the model did well (True Positive and True Negative) and where it does not do well (False Positive and False Negative).

5.2. Quantitative Metrics

As can be seen in Figure 1, the BERT model performed the best at classifying the turns. We can also see that its AUC is the best at 0.91 ± 0.01. This aligns with our expectations, as BERT is a Transformer, which is the among the current state of the art models for natural language processing. Although the LSTM has a higher accuracy than BERT, the precision-recall and ROC curves are much more indicative of model performance on this data: since about 80% of turns are negative, a model can easily have an inflated accuracy simply by outputting negative predictions with higher frequency, regardless of the quality of its learned representation of the turn. Interestingly, the most basic model, the bag of words and logistic regression approach, had comparable AUC and accuracy scores to the other models that are more complex and computationally intensive.

With the context of BERT's performance, it may seem surprising, then, that BioBERT performed the worst according to the Precision/Recall curve 1, and the accuracy metric (its accuracy is 0.76 ± 0.02 at a threshold of 0.7), and second worst according to the ROC curve, with an AUC of 0.85 ± 0.03. Upon further investigation, we see the reason for BioBERT's poor performance: BioBERT is trained with medical data from doctor's notes, not conversational text. At first, we hypothesized that this might improve performance due to the presence of medical jargon in both medical notes and doctor/patient conversations. However, doctors do not speak to their patients in the same way they write their notes. Rather, they avoid medical jargon and speak rather conversationally in order to communicate more effectively with their patients. So, the domain on which BioBERT was trained is not as applicable to the domain of doctor/patient conversations as we had hoped. Furthermore, BioBERT has been know to struggle with domain shift between medical texts from different disciplines of medicine. This weakness makes the domain shift from medical text to doctor/patient conversations even worse for the BioBERT model.

Besides the two BERT models are the Bag of Words models, the XGBoost with an accuracy and AUC of 0.87 ± 0.01 and 0.81 ± 0.02 respectively, and the LSTM with an accuracy and AUC of 0.93 ± 0.01 and 0.89 ± 0.01 respectively. These two show performances as expected. It is of note, however, that the Bag of Words model with Logistic Regression outperforms both BioBERT and Bag of Words with XGBoost. However, it still misses out on vital information due to the lack of sequential context for each word in the feature vector. In contrast, the LSTM is able to encode both spatial context and represent the turns with non-linearity, which allows it to perform well. We can see that the LSTM outperforms both Bag of Words and BioBERT. It still, however, does not perform as well as the BERT model.

Further discussion of these models using the original data versus the data pre-processed with the Keyword Library can be found in Appendix A. See table for specific metrics.

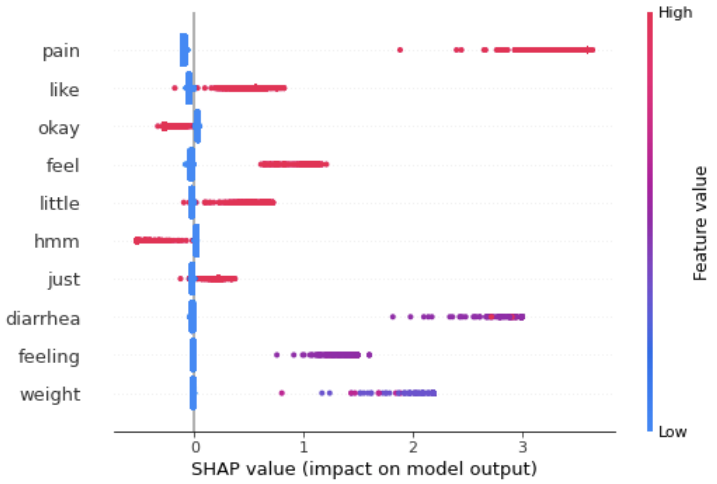

Figure 2: SHAP Plot using XGBoost

5.3. Qualitative Analysis

5.3.1. Bag of Words + Logistic Regression

We first present results using the bag of words and logistic regression approach. Using both the original and processed data. Figure 4 present the key terms whose coefficients are largest in magnitude. While results using original data produce terms that are irrelevant, results using the processed data produce many more symptom-relevant terms. Section 5.3.2 reiterates and elaborates on this finding.

5.3.2. Bag of Words + XGBoost

The SHAP plot for the original data is shown in Figure 2. Since SHAP plots can be hard to interpret for first-time users, we include a short manual here. The y-axis contains the words arranged in the order of importance from top to bottom. The x-axis has the SHAP values, which indicate the impact of each word on the probability that a sentence is relevant to symptoms; the color indicates the direction of such impact. For instance, more appearances of the word *pain* increases the probability of that a sentence is symptom-relevant, whereas more appearances of the word *hmm* decreases that probability.

We observe that the model targets words that patients use while describing their symptoms, such as *like*, *feel* and *feeling*, instead of words directly reflecting symptoms. In fact, out of the top 10 most important words, only 2 are directly reflective of symptoms, e.g. *pain* and *diarrhea*. This is likely due to the fact that patients have lots of ways to describe the same symptom. These different descriptions often have low stand-alone frequency, hence making it difficult for the model to catch the signals. This problem is mitigated when we used pre-processed data where keywords are already mapped to symptoms (see appendix).

5.3.3. LSTM

From the quantitative section above, and from what we expect from prior literature, the LSTM performs better than a simple bag-of-words model but worse than a more complex model like BERT (Mascio et al., 2020). The ability of the LSTM to incorporate the context of the words around it improves its prediction ability, and there were not many sentences that were false negatives because a known clinically significant phrase was missed.

However, the LSTM struggled to achieve comparable precision levels at the same recall compared to BERT, especially in the high-recall regime we are most concerned with; there were many instances of false positives from sentences like "pulling them" which was not medically related in this dataset, but one can imagine contexts in which pulling (e.g. a muscle) could be symptom talk, leading to misclassification by the model. In addition, at high recall thresholds, there are many false positives that are just sentence fragments like "Yeah" and "Good", but this may be solved by preserving context between turns.

5.3.4. BERT

As shown in the Table 1 and Figure 1, BERT outperforms all other models in all metrics except accuracy (which, in this domain, is not as meaningful). It has a low false positive and a relatively high false negative rate (confusion matrix shown in Figure 6 in the Appendix). While there are certainly cases where the model simply did not pick up the signal, a lot of these false positives and false negatives are either difficult even for humans to identify or likely to be mislabeled. For instance, here are two examples where the model predicts relevant but the label says irrelevant: "And prednisone's for your bones." and "Yeah. I'm getting pretty good letting it go completely limp and then just kind of letting it come out. That seems to help some." Similarly, here are two examples where the model predicts irrelevant but the label says relevant: "Oh my, yeah. Oh my, yeah" and "It will". This suggests that there's some degree of noise in our training data.

5.3.5. BioBERT

Much of the BioBERT's weakness comes from its high false positive rate, as can be seen in Table 7: it labels almost half the testing data as positive, although only about a fifth of the ground truth labels are positive. The true negatives that the model is able to correctly classify are the short, generic turns such as "– how nice I am. Um." or "goodness gracious."

As mentioned previously, there is some noise in the dataset as a result of the ground truth labels being aware of the context of the conversation as a whole. Since our models only analyze one turn at a time as input, we do not encode the context of the conversation as a whole. This is apparent in the relatively few turns that the model falsely labels negative. Turns such as "occasionally" and "just barely" are only symptom talk in the context of the larger conversation, which the BioBERT model does not encode. Such examples make up the majority of the false negatives. However, some false negatives are examples of the model's failure, such as "Let me ask you this, have you had any problems with cough?"

Longer turns with more medical jargon, on the other hand, the BioBERT model seems to classify as positive even when they are labeled as negative. Since the researchers at the Lindvall Lab are looking for a high sensitivity, the threshold for positive prediction was set relatively high at 0.7, meaning we err on the side of labeling a turn as positive. So, longer

turns with a higher density of medical jargon are more likely to end up being marked as a positive example, although its ground truth classification is negative.

6. Discussion

In summary, we explored four models: Bag of Words, LSTM, BERT, and BioBERT. The BERT-based classifier performed the best, with an AUROC of 0.91 and a precision-recall curve significantly higher than any other model. The advantage over the second-best model, the LSTM, was most pronounced in the high-recall regime that is most clinically applicable; when used in tandem with human labelers, having a low false negative rate is important to minimize work necessary to manually check labels for likely negative samples.

This performance advantage came at a cost, however, as the BERT encodings took longer and required more computational resources than the second-best LSTM-based models. In addition, the equally-bulky BioBERT encodings performed quite poorly in the sentence classification task. As noted in Lee et al. (2020), BioBERT struggles with cross-domain transfer of its encodings; it is even necessary to fine-tune it on data from a new medical specialty as the difference in syntax and jargon can be significant. The poor performance of BioBERT suggests that this symptom-detection task utilizes more of BERT's general informal conversation training.

The optimal model choice in a clinical setting is a slightly more complex question. As alluded to earlier, generating BERT encodings is not a real-time process; if deployed on a central server with sufficient resources, a transformer-based model may be appropriate for pre-labeling symptom-talk data. However, given the performance constraints of a typical portable computer that a physician would carry, the LSTM model performs just slightly worse on a much faster timescale (order of seconds instead of hours). Thus, if deployed today, the LSTM may be a better choice. Our results, however, are still promising; as advances in transformer-based NLP techniques on general conversational text continue, we are more likely to find a lightweight way to integrate these powerful models into a text classification pipeline.

We have thus shown that it is possible to apply natural language processing in detecting symptom-related talks in conversational data, and identified the benefits and drawbacks of the different approaches. These models make it possible to automate the labeling process, which is currently done manually by human annotators in the Lindvall lab, among others. In addition, these models pave the way for a fully automated pipeline, where conversations between patients and clinicians can be first converted to transcripts via speech recognition and then passed into NLP models for symptom detection. This allows doctors to be able to more aware of their patients' needs, hence improving the patients' experience.

6.1. Limitations

6.1.1. Noise in the Data

Results of our models rely heavily on accurately labeled data. Since there are relatively few positively-labeled conversation turns, noisy labels will have a larger effect on this models compared to a more balanced dataset. In addition, as noted in the qualitative results, a potentially significant amount of context is lost when dividing the conversation into turns.

These models would likely have performed better if the previous turns had been available as context. In the future, it may be possible even to train the model to learn the amount of context it needs to correctly classify a turn, enabling it to more seamlessly integrate into an automated pipeline for symptom detection.

6.1.2. Possibility of Data Leakage

Another problem is the possibility of data leakage. For the BioBERT and LSTM and XGBoost + BoW models, we split the data into train-test-val by patients. By doing so, we ensure there is no data leakage for these models. For BERT, unfortunately, we did not have enough time to re-run BERT with the data split on patients rather than conversations. So, there is some data leakage for the BERT model. We did, however, ensure that turns from the same conversation end up in the same split. Therefore, our method for BERT only mitigates but does not completely prevent this problem.

6.1.3. Interpretability of Deep Learning Models

Beyond the scope of this project, there are many avenues of investigation that could yield better and more interpretable results. Our results are limited in the breadth of the architecture search we were able to perform in the scope of this project. In particular, while we were able to tune hyperparameters, a systematic architecture search for the design of additional layers may improve these results even if the same core model is used. The inclusion of e.g. attention layers and other novel architecture designs for NLP show promising performance improvements and may yield better results (Liu and Guo, 2019). Finally, the interpretability of more complex models is critical to furthering our understanding of these models in both an NLP and clinical application context. As we observed a modest improvement in model performance after pre-processing with known symptom phrases, using perturbative and gradient-based methods to identify causal inputs for these larger LSTM and BERT models may improve accuracy and enable clinicians to better evaluate and understand the model predictions (Wallace et al., 2020).

7. Acknowledgments & Member Contributions

- Anisha Agarwal implemented the BioBERT model. She also performed and wrote all BioBERT-related quantitative and qualitative analysis. She wrote the Quantitative Metrics section (5.2), the Data Extraction general information section (4.2), and the Abstract. She also worked with Carlo to write the Introduction, and she worked with Pranav and Tiana to write the Limitations section (6.1.1 and 6.1.2).

- Tiana Cui implemented the BERT model and XGBoost Model and performed all quantitative and qualitative analyses regarding BERT and XGBoost. She worked with Carlo in obtaining the bag of words features, which served as inputs for the XGBoost model. She worked with Anisha and Pranav in 1) completing the poster and 2) writing the Discussion and Limitations sections (specifically she worked on 6.1.2 and the last paragraph of the Discussion).

- Carlo Duffy wrote the Introduction and Cohort sections, and implemented the the logistic regression model using bag of words to derive features (see section 5.3.1). He also took charge in understanding the data source and assisted in performing exploratory data analysis.
- Pranav Murugan implemented the LSTM model and performed all associated qualitative and quantitative analysis of this model. He also calculated the comparative analyses between the models (Figure 1), wrote the Related Work section, and worked with Anisha and Tiana to write the Abstract, Discussion (paragraphs 1-3), and Limitations (6.1.1 and 6.1.3) sections.

Many thanks to all members of the Lindvall team at the Dana-Farber Cancer Institute!

References

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. 2016. doi: 10.48550/arXiv.1603.02754. URL https://arxiv.org/abs/1603.02754.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. 2018. doi: 10.48550/arXiv.1810.04805. URL https://arxiv.org/abs/1810.04805.

Andrew Dunn, Diana Inkpen, and Răzvan Andonie. Context-sensitive visualization of deep learning natural language processing models. 2021. doi: 10.48550/ARXIV.2105.12202. URL https://arxiv.org/abs/2105.12202.

Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48, 07 2017. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx228. URL https://doi.org/10.1093/bioinformatics/btx228.

Zellig S. Harris. Distributional structure. *¡i¿WORD¡/i¿*, 10(2-3):146–162, 1954. doi: 10.1080/00437956.1954.11659520. URL https://doi.org/10.1080/00437956.1954.11659520.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997. doi: 10.1162/neco.1997.9.8.1735.

Abhyuday Jagannatha and Hong Yu. Bidirectional recurrent neural networks for medical event detection in electronic health records. 2016. doi: 10.48550/ARXIV.1606.07953. URL https://arxiv.org/abs/1606.07953.

Theresa Koleck, Caitlin Dreisbach, Philip Bourne, and Suzanne Bakken. Natural language processing of symptoms documented in free-text narratives of electronic health records: a systematic review. *Journal of the American Medical Informatics Association : JAMIA*, 26, 02 2019. doi: 10.1093/jamia/ocy173.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, R. Andrew Taylor, Harlan M. Krumholz, and Dragomir Radev. Neural natural language processing for unstructured data in electronic health records: a review. 2021. doi: 10.48550/ARXIV.2107.02975. URL https://arxiv.org/abs/2107.02975.

Gang Liu and Jiabao Guo. Bidirectional lstm with attention mechanism and convolutional layer for text classification. *Neurocomputing*, 337:325–338, 2019. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2019.01.078. URL https://www.sciencedirect.com/science/article/pii/S0925231219301067.

Aurelie Mascio, Zeljko Kraljevic, Daniel Bean, Richard Dobson, Robert Stewart, Rebecca Bendayan, and Angus Roberts. Comparative analysis of text classification approaches in electronic health records. 2020. doi: 10.48550/ARXIV.2005.06624. URL https://arxiv.org/abs/2005.06624.

Laura Porter, Kathryn Pollak, David Farrell, Meredith Cooper, Robert Arnold, Amy Jeffreys, and James Tulsky. Development and implementation of an online program to improve how patients communicate emotional concerns to their oncology providers. *Supportive care in cancer : official journal of the Multinational Association of Supportive Care in Cancer*, 23, 02 2015. doi: 10.1007/s00520-015-2656-2.

Yuanhe Tian, Wang Shen, Yan Song, Fei Xia, Min He, and Kenli Li. Improving biomedical named entity recognition with syntactic information. 04 2020. doi: 10.21203/rs.3.rs-21994/v1.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962v2*, 2019.

Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. A review on the long short-term memory model. *Artificial Intelligence Review*, 53, 12 2020. doi: 10.1007/s10462-020-09838-1.

Eric Wallace, Jens Tuyls, Junlin Wang, Sanjay Subramanian, Matt Gardner, and Sameer Singh. Allennlp interpret: A framework for explaining predictions of nlp models. 2019. doi: 10.48550/ARXIV.1909.09251. URL https://arxiv.org/abs/1909.09251.

Eric Wallace, Matt Gardner, and Sameer Singh. Interpreting predictions of NLP models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, pages 20–23, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-tutorials.3. URL https://aclanthology.org/2020.emnlp-tutorials.3.

Xuan Wang, Yu Zhang, Xiang Ren, Yuhao Zhang, Marinka Zitnik, Jingbo Shang, Curtis Langlotz, and Jiawei Han. Cross-type biomedical named entity recognition with deep multi-task learning. 2018. doi: 10.48550/ARXIV.1801.09851. URL https://arxiv.org/abs/1801.09851.

Method	Accuracy	AUC	Precision	Recall
BoW + LR	0.88±0.01	0.68±0.02	0.77±0.02	0.38±0.02
BoW + XGBoost	0.91±0.01	0.83±0.02	0.58±0.03	0.57±0.03
LSTM	0.91±0.01	0.89±0.01	0.71±0.03	0.60±0.01
BERT	0.92±0.01	0.92±0.01	0.79±0.03	0.59±0.03

Table 2: Results on best models using preprocessed data

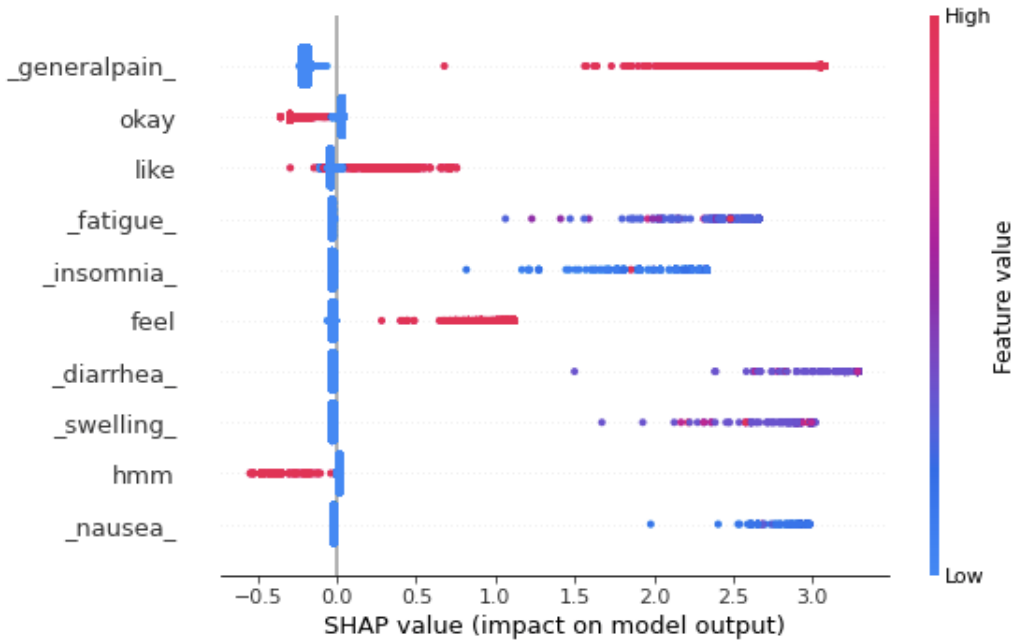

Figure 3: SHAP Plot using preprocessed data. Words that are enclosed in underscores are mapped from the Symptom Keyword Library

Appendix A. Effect of Leveraging Keyword Symptom Library

As discussed before, it is to the interest of the Lindvall team to examine the effect of preprocessing the data using the Symptom-Keyword Dictionary on the model performance. In this section, we include the performance of all models using preprocessed data.

It is important to note, however, that utilizing prior knowledge that isn't known in the original dataset can lead to additional risk of data leakage.

XGBoost - SHAP plot on preprocessed data

As shown in the summary Table 2 and the SHAP plot Figure 3, XGBoost is able to pick up the signals a lot better when we mapped the keywords to their corresponding symptoms. 6 out of the top 10 most important words are directly indicative of symptoms (compared to 2 when we train the exact same model on the original data).

We observe some discontinuity in the SHAP values for the mapped symptom keywords, which is a sign of noise (Figure 3). After careful investigation, we find two potential sources of noise. First, noise may come from the original data due to mislabeling, which is discussed previously in the Qualitative Analysis section. Second, noise may come from the mapping process. Since the machine does not understand context, it maps a keyword to its corresponding symptom whenever it sees one. For instance, *breath* is one of the key phrases in the dictionary, corresponding to the symptom *shortness of breath*. Therefore, sentences like "Take a deep breath." is mapped even though it is not really related to shortness of breath.

Deep Learning Models

In contrast to XGBoost, the Keyword Symptom Library does not meaningfully improve the performance for the three deep learning models (Table 2). This is possibly due to the fact that all three deep learning models we used are pre-trained on an extremely large corpus, and hence this extra piece of information does not add too much to the models.

Appendix B. Supplementary Results

Bag-of-Words + Logistic Regression

Feature Importance

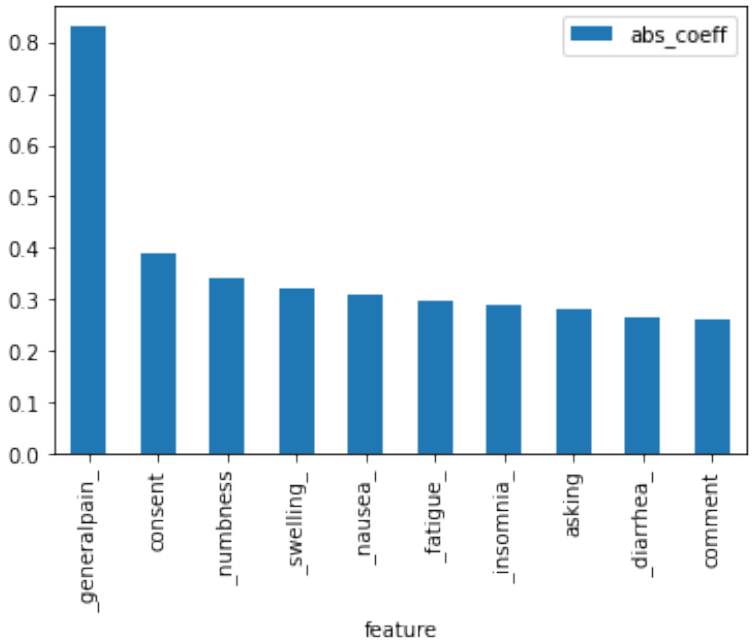

Figure 4: Most important features from the logistic regression using the processed data.

LSTM

The LSTM was trained on padded, tokenized sequences (total turn length varied depending on the preprocessing pipeline, but ranged from 300 - 700 tokens). Each token was mapped to a trainable embedding space with 64 dimensions before training with the LSTM. The LSTM layer itself contained single a hidden layer with 16 dimensions with bidirectional inputs. These LSTM encodings were then passed through a dropout regularization layer with $p = 0.2$ before a linear layer computed a logit output for the given layer.

BERT

Fine Tuning Structure

The model structure that we used is shown in Figure 5. As shown, the input texts are passed into a preprocessing layer, which is a companion of BERT models to preprocess plain text inputs into the input format expected by BERT. This processed input is passed in to the BERT encoder, which produces contextualized embeddings for each sentence. Next, the embeddings go through a dropout layer, which randomly masks 5% of the neurons during training. This is a common technique used to prevent overfitting. Finally, the data goes through a dense layer with a sigmoid activation function, which returns a decimal between 0 and 1 that represents the predicted probability of this particular sentence being relevant to symptoms. In addition, we added an early stopping condition where the model will stop training if the validation AUC does not improve for 3 epochs.

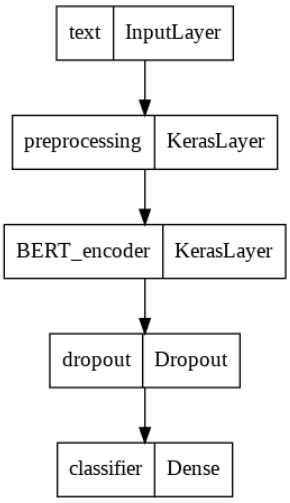

Figure 5: Structure for Fine-Tuning BERT

BioBERT

Linear Classifier

Our linear classifier consists of 3 fully connected layers of size 768, 64 and 1 respectively, with a ReLU layer and a batch normalization layer between the first 2 fully connected layers, followed by another batch normalization layer and a dropout layer between the second and third fully connected layers. The classifier was trained for 10 epochs.

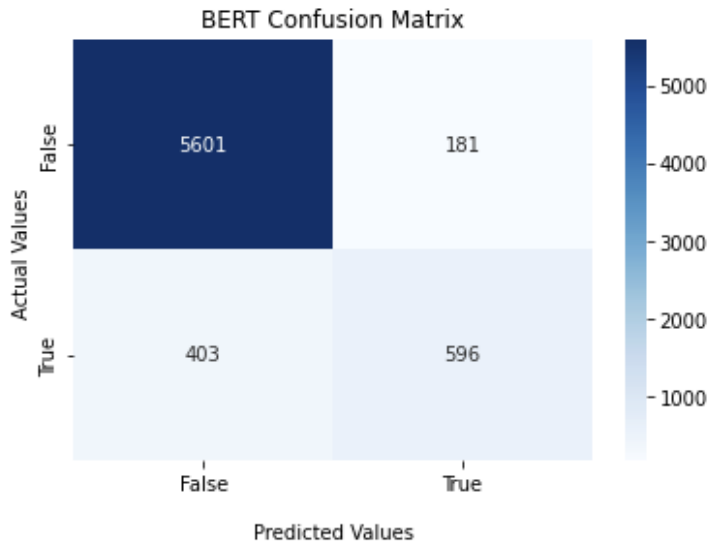

Figure 6: BERT confusion Matrix

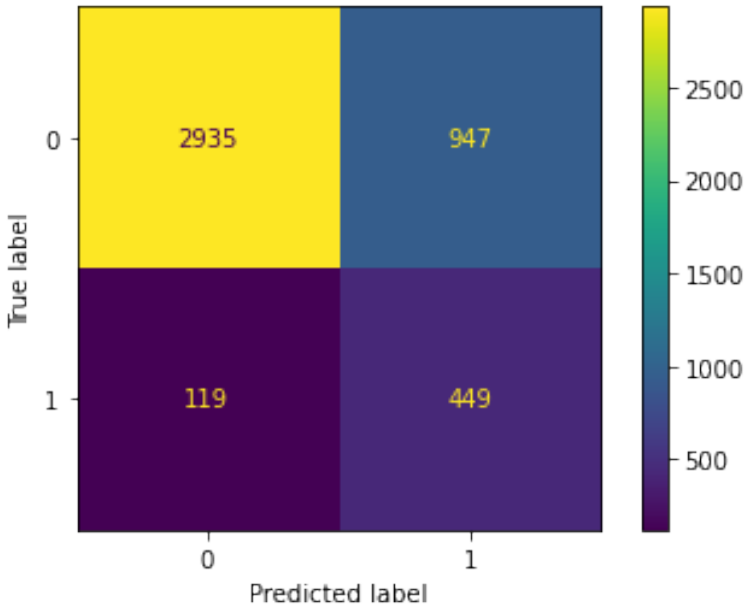

Figure 7: BioBERT confusion Matrix