Submitted by: Sahar Ziv

Medical Symptoms Classification Interim Report

Introduction

This project focuses on using machine learning techniques in order to classify medical symptoms based on self-reported audio recordings, with the goal of improving conversational agents in the medical field. The dataset comprises thousands of audio snippets, totaling over 8 hours of recording time, these recordings were created through a multi-step process where contributors first provided textual descriptions of symptoms, followed by recording corresponding audio. However, challenges such as incorrect labels and poor audio quality necessitate thorough data cleaning before model training. By addressing these issues, the project aims to create robust models that can accurately classify medical symptoms, thus enhancing the efficacy of conversational agents and improving healthcare diagnosis.

The research questions are:

- Can we classify the symptom of the patient to cough or infected wound only by the audio file?
- Which model classify better, audio file classification model or NLP model of the audio files transcriptions?

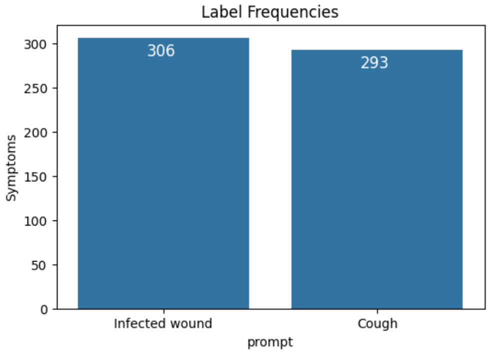
Data Description

The data found on Kaggle, the data was labeled using the Figure-Eight platform.

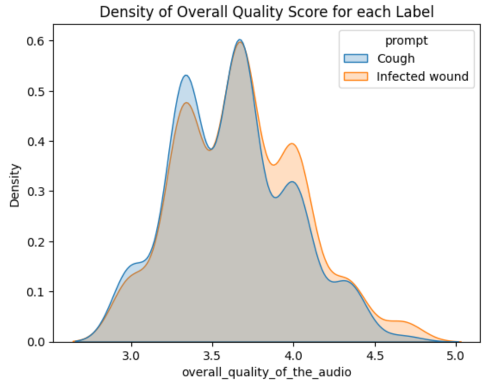
The dataset contains 6661 audio recordings and a corresponding CSV file, where each recording has 3 indicators of audio quality measures (audio clipping, background noise audible and quiet speaker) with confidence score for those indications (between 0 and 1), an overall numeric score between 3 and 5 of the quality of the audio which is uncorrelated with the other audio quality measures. Furthermore, for each audio recording the CSV file contains the transcription of the audio, the writer ID, the speaker ID, and the label which is one of 25 medical symptoms. In order to develop the NLP model, I used the English language dictionary.

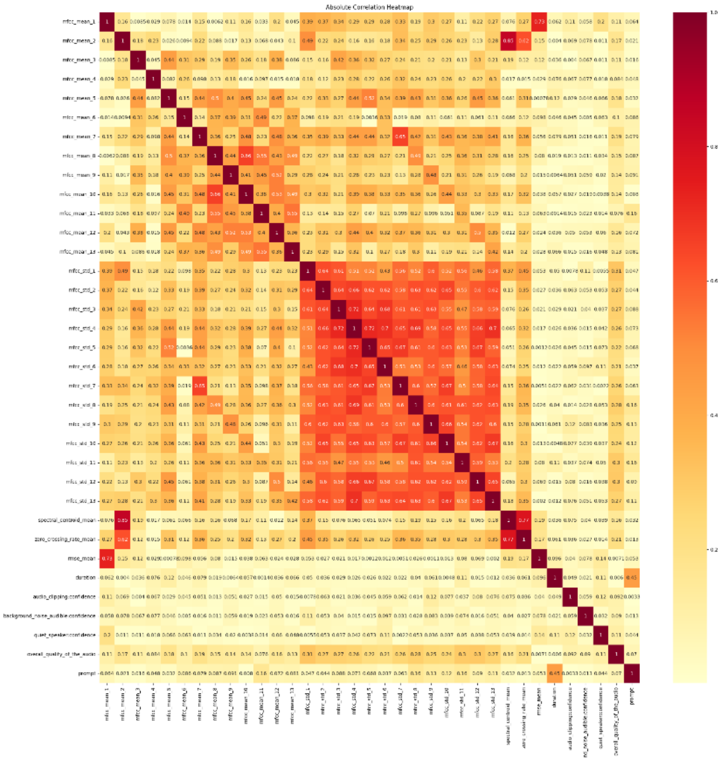
In order to answer the research questions, a sub selection of the data is needed, where the label is either "Cough" or "Infected wound".

In the plot on the right, we can see that the label frequency is pretty equal, so we do not have to account unbalanced observations. The subset of the data that we are interested in has 599 observations.



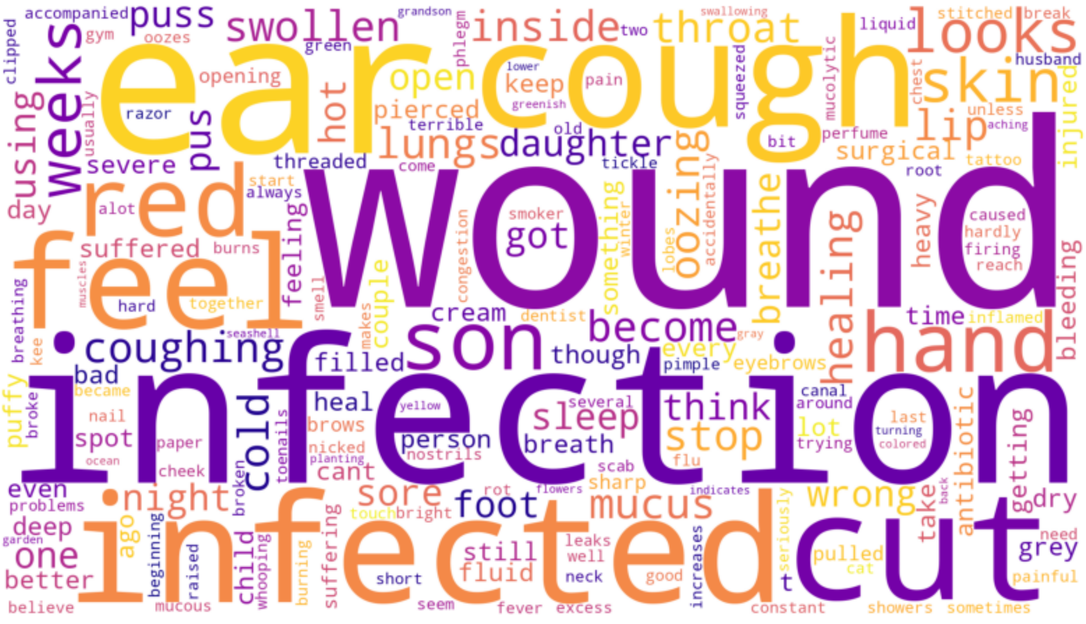
In the plot on the right, we can see the density of the overall quality score of the audio files by each label. We can see that there are more audio files with better overall quality for the "Infected wound" label than the "Cough" label.





In the plot on the left, we can see the absolute correlation between all the features I modified in the *Model Engineering* section. We can see really high correlation between some features (in bright red), which could result in bad predictions of our label. I decided to deal with it using LASSO and will be explained in the *Model Engineering* section.

In the plot under, we can see a word cloud of the non-stop words in the transcription for the audio files. The size of the word is proportional to its appearance in all the transcriptions.



It is easy to see that in order to classify between "Infected wound" and "Cough", removing those words from the phrases is crucial, as they have high correlation to themselves and would affect the prediction accuracy.

Modeling

In order to develop a model for classification **from the audio files** I converted each audio file to the following features:

- The duration of the audio file in seconds.
- The mean value of the MFCC coefficient values across all time series frames.
- The standard deviation value of the MFCC coefficient values across all time series frames.
- The mean spectral centroid across all time series frames, calculated by
$$Centroid(t) = \sum_k \frac{S[k,t] \cdot freq[k]}{\sum_j S[j,t]}$$
Where, t is the time frame, s is a magnitude spectrogram and $freq$ is the array of frequencies of the rows of s.
- The mean zero crossing rate across all time series frames, which means the mean rate at which the signal changes its sign (the number of times the audio waveform crosses the horizontal axis).
- The mean rms (root mean square) across all time series frames.

To that I added the quality measures confidence score and the overall quality score, which resulted in 34 features.

In order to develop a model for classification **from the transcriptions** I used some conventional method in language process, I deconstructed words (I'm=I am, I've=I have…), lower cased and strip out punctuations. Then I removed the stop-words, lemmatized and removed the words "cough", "infected", "infection", "wound" and "cut" and did one-hot encoding, to that again I added the quality measures confidence score and the overall quality score, which resulted in 174 features.

For both classification methods I split the data to train and test 70%-30%, trained a scaler using only the train set and transforming both train and test set. I chose to fit 3 models Logistic Regression, KNN and SVM. For each model I have the following tuning parameters:

- Logistic Regression – 'penalty': LASSO or Ridge, 'C': in $[1, e^3]$ – penalty inverse parameter, as it is smaller the stronger the regularization.
- KNN – 'K': number of nearest neighbors between $[3{,}241]$ in odds numbers.
- SVM – 'kernel': linear, polynomial, or radial, 'degree': between $[2{,}6]$ for the polynomial kernel, 'gamma': between $[0.00005{,}1]$ for the radial kernel.
(Will be added in the final report)

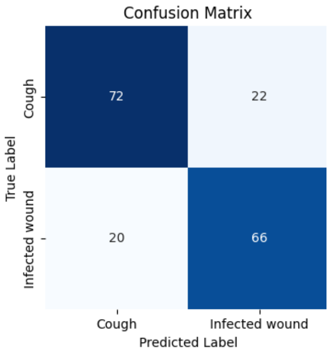
I will find the best tuning parameters by using 10-fold cross validation and measuring the accuracy of the validation sets for each combination of tuning parameters. The best combination model will be compared with the other models to determine the best model out of those three.

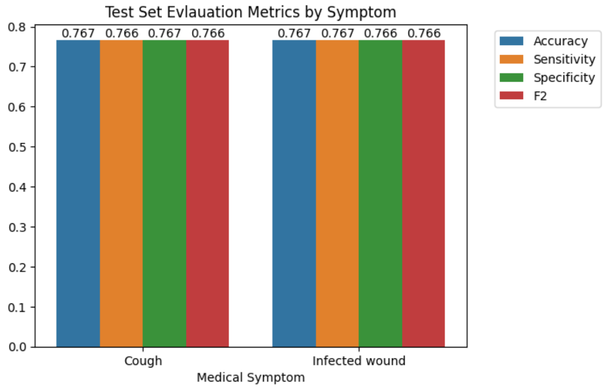
Results – Audio files data

The logistic regression best parameters (by the best train set accuracy) are:

- $C = 6.05,\ Penalty = Ridge$

The logistic regression model is easy to understand, the mode estimates the probability that a give input belongs to particular class, it works pretty well when the data is not completely separated between the classes. The running time of all 1200 cross-validation combinations is 48_{sec}, the train set accuracy is 76.1%, which means the model is not overfitting and we can see that the test set accuracy is 76.7% which is fairly good!

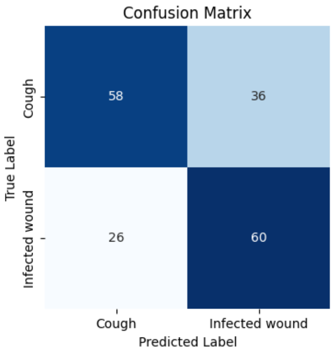


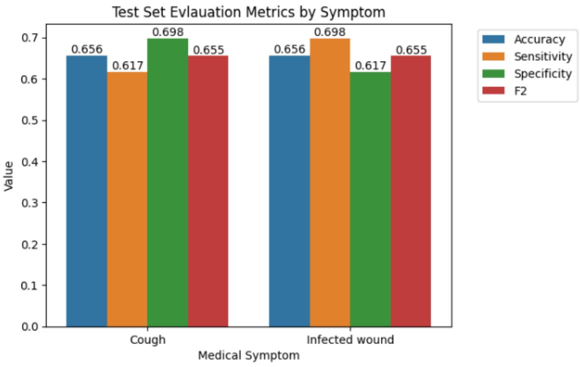


Another property the logistic regression holds is that the regularization can result coefficient value 0 by the importance of the features in the model. (Will use it in the final report)

The KNN best parameter is: $K = 37$

The running time of all 1200 cross-validation combinations is 15_{sec}, the train set accuracy is 68.7%, which means the model is not overfitting and we can see that the test set accuracy is 65.6% which is pretty good too, but worse than the logistic regression model.



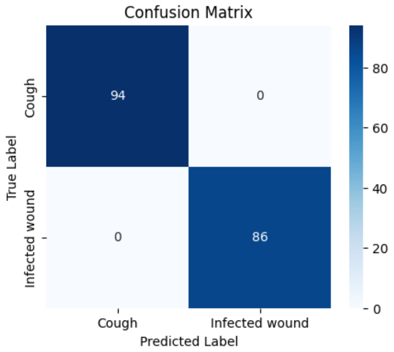


Results – Transcriptions data

The logistic regression best parameters (by the best train set accuracy) are:

- $C = 1\ ,\ Penalty = LASSO$

In this case the prediction is perfect, and it might be because of the small sample size. Also, NLP models are extraordinarily strong in accuracy if preprocessed correctly but since the model have feature for each non-stop word it takes a long time to train the model, the running time is 22_{min}.



Conclusion and Future work

In conclusion, we can say that the best way to predict a symptom using self-reported audio recordings is by transcript it to phrase using speech-to-text software and use the NLP logistic regression model for binary classification, but if we can't do it for some reason, the classification using logistic regression on the audio files themselves could give good results too. For the final report, I intend to do a multiclass classification and use more models.