Cough-Based COVID-19 Detection

Dagger Chen (SUNet ID: dagger) Liyu Tan (SUNet ID: tlyhenry) Ryan Pan (SUNet ID: yp93)

Abstract

Covid-19 has impacted the lives of billions of people and a cough-based Covid-19 classifier can be extremely helpful to stop the spread of pandemic. In this project, we have investigated how well simple models such as Logistic Regression and SVM models can perform when contrasting to more complex models like LightGBM Classifier and CNN model. We have also looked into how the quality of cough recording can affect the prediction scores by applying a filter to those models. We found out complex models generally outperform simple models; however, the performance of simple models can be greatly enhanced with improvement of cough audio quality.

Introduction

2020 has been a challenging year for everyone as the world is shutting down because of Covid-19. Developing countries are facing diagnostic challenges as the tests are limited and pricey. Even in developed countries, people may avoid getting tested due to the unpleasant feeling of nasal swab testing [1]. The self-collected testing samples may lead to erroneous results, while testing done by medical staff brings additional risk of getting exposed to the virus for both patients and staff. It will be both safer and easier if the covid detection can be done based on cough patterns. People only need smartphones, which are universally accessible in developed countries and many developing countries [2], to record their coughs and get results in a few minutes.

As a screening technique, it can screen out people with low risk and identify people who may need complementary covid testings. By developing a classifier with high specificity and sensitivity, we can save people's money and time and stop the spread of Covid.

The team is especially interested in real-world problems as we have been working in the industry. This application project is meaningful in this particular period of time and challenge that motivates the team to resolve.

Related Work

Though coughs may sound the same for human ears, the prior study has already shown that latent features can be discovered in temporal and spectral space and it will change according to the airway flow mechanics [3]. From the cough classification point of view, Hanieh's study has shown that dry and wet coughs can be clearly separated through features related to the shape of sound waves [4]. There is more evidence in recent years' studies that show diagnosis and screening of pulmonary diseases, such as pneumonia, can be achieved through features of coughs [5-6].

Identification of Covid-19 infection brings new challenges, as Ali mentioned in his paper that there are more than thirty non-Covid-19 related medical conditions [7], it might be difficult to

differentiate Covid-19 related patterns from other latent features. Though the task is challenging, they have successfully developed an app that can distinguish Covid-19 coughs from other type of coughs with a multi-pronged mediator centered risk-averse AI architecture which contains Deep Transfer Learning-based Multi-Class classifier, Classical Machine Learning-based Multi-Class classifier, and Deep Transfer Learning-based Binary-Class classifier. The classifier they built performs well for cough detection and Covid-19 classification and has a high accuracy of prediction (~92.64%) [5].
Our project aims to use and compare the models we have learned in class such as Logistic Regression (LR) and Support Vector Machine (SVM), a model that is widely used in data science field (LightGBM Classifier), and Deep Learning (DL) models and tries to understand the difference between simple models and more advanced models.

Dataset and Features

The data we use come from an open sourced dataset called Coughvid. We make use of the following useful features:

1. Audio of cough
2. cough_detected: the probability that the audio contains one or more coughs
3. age: age of the patient
4. respiratory_condition: whether the patient is experiencing respiratory symptoms
5. fever_or_muscle_pain

As we will extract another 13 Mel-frequency cepstrum features from the audio of cough, we have in total 17 features (13 cough-related features + 4 supplementary information features). The dataset itself contains more columns; however, due to a large portion of the data in those columns being NaN or missing, we decided to move along with the features listed above.

Method

It is hard for a human to tell if a cough sound is related to COVID-19 and so it is for AI engines. In fact, a number of medical conditions can cause coughs, such as pertussis, pneumonia, and influenza. Furthermore, even healthy people may cough because of environmental reasons. On the other hand, we learned that AI engines learn features faster from visualized objects from previous studies. To implement further processing on the audio data, we decided to transform the dataset into spectrograms. In detail, we converted audios into Mel scale at first. Mel scale can evaluate changes of frequency in terms of changes in perceiving sounds. Mel scale can also preserve more information in the lower frequencies compared with other frequency spectrograms while cough sounds have more energies in lower frequencies. As a result, Mel scale is the optimized method to convert audios into materials that AI engines can use for distinguishing coughs.

We then obtained Mel Frequency Cepstral Coefficients (MFCC) by implementing Cepstral analysis on the Mel spectrum. Every sample produced a MxN MFCC feature matrix where M is the number of MFCC features and N is the number of frames. For each cough sample, we took the mean of each MFCC feature for all the frames to get a Mx1 vector. We also took the top P Mx1 Principle Component Analysis (PCA) projections of each MFCC feature for all the frames to get another Mx1 vector. Finally we concatenate the two Mx1 vectors into a 2Mx1 vector. As we repeated this approach to all the samples, we acquired the MFCC features of the audio dataset.

As we had acquired the MFCC features, we by far had two sets of datasets to work on: a feature table and a series of MFCC spectrum. We considered utilizing Logistic Regression, SVM, DL, and ANN but given the materials learned in class as well as the time frame of this quarter we ended up with LR and DL. The LR is basic but still effective to learn the feature matrix. Besides, it was widely known that a DL model would be effective to train MFCC spectrms. As so far, we have developed a Logistic Regression model training on the feature matrix.

The team will develop a ML model based on material learned in the class. Potential options of algorithms include Naive LR, SVM, and ANN. The team will make a decision depending on future research as the course is in progress. Besides setting up a model from scratch, the team plans to implement a deep learning model on the data for comparison purposes. The team picked a deep learning model, probably a CNN model, because this kind of model is highly efficient in learning computer vision problems.

Experiments/Results/Discussion

i. Logistic regression

Data Preprocessing

After MFCC feature extraction, we got 5386 patient records, in which around 90.9% of the test records have negative labels and the rest are positive. We also calculate the mean value for each feature column with missing data, and then use the mean to replace NaN in these columns. Furthermore, due to the imbalancement of positive and negative labels, we decided to balance the data by upsampling the positive class. To further improve the f1 score and sensitivity, we standardized the features and applied L1 and L2 regularization to the model. Lastly, we added a cough_detected filter to improve the quality of training data and simulate a better audio sample to predict.

Feature Selection and Model Fitting

The results turned out to show the standardization and resampling do have a positive effect on the f1 score. By tuning the penalty terms and balancing the data, the f1 score has increased from around 0.210 to 0.252. We also tried to drop some features and check the resulting score. We found that dropping the “respiratory_condition” actually increases the f1 score by a little bit. At the same time, by applying filter cough_detected > 0.8 to both training and testing data, the f1 score increases dramatically to 0.556 with an accuracy of 0.851 which indicates the LR model is heavily dependent on cough audio quality.

Metric & Difficulties

At first we tried to use accuracy as our metric and didn’t balance the data before fitting the LR model. However, when we made predictions on the test data, the accuracy was surprisingly high (~0.910) and we later realized that the classifier always identifies the case as negative and the portion of the negative class is just 0.91. It motivated us to do the data balancing and choose a better metric (f1 score) that weighs more on false negatives.

ii. Support-Vector Machine

The data preprocessing for SVM is exactly the same as the one for LR. The standardization is very useful for SVM as scaling is not an invariant for SVM models. However, with all the data preprocessing and enhancement, the best f1 score for SVM is still very low (~0.207) with an accuracy of 0.623. With a filter cough_detected > 0.8, we observed the same leap of f1 score and accuracy score, which means it is also heavily dependent on the quality of cough audio. As the performance of SVM is poorer compared with LR model, we didn't spend more time to tune parameters and improve its performance.

iii. LightGBM Classifier

Intuition

LightGBM Classifier is a widely used and successful classifier which is based on decision tree algorithms [8]. With its unique techniques: Gradient-based One-Side Sampling and Exclusive Feature Bundling, the classifier becomes both powerful and efficient [8]. We chose this model because it fits into various scenarios and can serve as a great reference for the other models we learnt in class.

Data Preprocessing

We added a filter for the cough_detected column to screen out some training rows, so we can investigate the influence of probability of cough on accuracy and f1 score.

Model Fitting

For the LGBM Classifier, we used class_weight to compensate for the imbalance of the positive and negative class. We changed the ratio of weight of positive class to negative class from 1 to 100 and selected the best one. We found that the weight ratios that maximize accuracy and f1 score oftenly appear to be the same, and the resulting f1 score and accuracy are significantly higher than their counterparts in the LR and SVM models.
We have also seen the effects of cough_detected threshold on final scores. When we have no filters, the best f1 score is 0.551 with an accuracy of 0.909. Surprisingly, the scores of LightGBM Classifier is good even without a cough_detected filter, which can be understood as the model works well even for low-quality data.

iv. CNN

Intution

In the data pre-processing, we are getting the mel-spec of the audio that detects the cough from the patient. We are using the Mel Spectrogram image to determine if a patient is likely to get covid or not.

Model

Each image had 480 * 640 * 3 pixels, we build the cnn model multiple layers, the first layer we are using a 2 * 2 maxpool layer, the second layer is a 2 * 3 convolution2D layer, the third layer is 2 * 2 maxpool layer, the fourth layer is a 0.15 dropout layer, the sixth layer is another 2 * 3 convolution2D layer, the seventh layer is a 4*4 max pool layer, then is a 0.15 dropout layer. The last one is a flatten model with a fully connected layer.

Conclusion/Future works

In the experiments, we have found how f1 score can be a better metric than accuracy score when working with unbalanced data. We have also realized the importance of resampling, regularization and feature standardization when working with simple models such as LR and SVM. Even simple models can work well when we pre-process the data properly.

For comparison among different models, we can observe that the LightGBM model and CNN model give good scores when filter is not applied, so they can work well even with low-quality data. In the aspect of cough_detected filter, we can see the improvement of audio quality can greatly enhance the scores for LR and SVM models. However, it doesn't affect much for the more complex and powerful LightGBM model, which performs well at all scenarios. We have also noticed that the CNN model has much worse performance compared to the literature, it might be due to our implementation or lack of features and data as we only extract 13 features for training with 4000 rows of data in total. In the future, we will try to use a larger dataset and extract more features from audio, common information (age, date_detected, sex, etc..), and symptoms to further explore the CNN model.

Table 1: F1 score and Accuracy Comparison between Four Models with and without Filter

	LR	SVM	LightGBM	CNN
Best f1	0.252	0.207	0.551	0.62
Accuracy	0.655	0.623	0.704	0.61
Best f1 with filter	0.556	0.482	0.569	N/A
Accuracy with filter	0.851	0.856	0.910	N/A

Reference

[1] Beeching, Nick J., Tom E. Fletcher, and Mike BJ Beadsworth. "Covid-19: testing times." (2020).

[2] Poushter, Jacob. "Smartphone ownership and internet usage continues to climb in emerging economies." *Pew research center* 22.1 (2016): 1-44.

[3] W. Thorpe, M. Kurver, G. King and C. Salome, "Acoustic analysis of cough," *The Seventh Australian and New Zealand Intelligent Information Systems Conference, 2001*, 2001, pp. 391-394, doi: 10.1109/ANZIIS.2001.974110.

[4] H. Chatrzarrin, A. Arcelus, R. Goubran and F. Knoefel, "Feature extraction for the differentiation of dry and wet cough sounds," *2011 IEEE International Symposium on Medical Measurements and Applications*, 2011, pp. 162-166, doi: 10.1109/MeMeA.2011.5966670.

[5] I. Song, "Diagnosis of pneumonia from sounds collected using low cost cell phones," *2015 International Joint Conference on Neural Networks (IJCNN)*, 2015, pp. 1-8, doi: 10.1109/IJCNN.2015.7280317.

[6] C. Infante, D. Chamberlain, R. Fletcher, Y. Thorat and R. Kodgule, "Use of cough sounds for diagnosis and screening of pulmonary disease," *2017 IEEE Global Humanitarian Technology Conference (GHTC)*, 2017, pp. 1-10, doi: 10.1109/GHTC.2017.8239338.

[7] Imran, Ali et al. “AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app.” *Informatics in medicine unlocked* vol. 20 (2020): 100378. doi:10.1016/j.imu.2020.100378

[8] Ke, Guolin, et al. "Lightgbm: A highly efficient gradient boosting decision tree." *Advances in neural information processing systems* 30 (2017): 3146-3154.