Predicting H1N1 and Seasonal Flu Vaccines using Machine Learning Techniques

Problem Statement

In 2009, a pandemic that was caused by the H1N1 influenza virus, named "swine flu", spread like wildfire across the world. Researchers and Healthcare specialists estimated that in the initial year, it was responsible for between 150,000 to 600,000 deaths in the whole world. A vaccine against the H1N1 (swine flu) virus was made available in October 2009. We propose a data-driven machine learning model to predict the likelihood of a person being vaccinated against H1N1 and seasonal flu. We have implemented 9 deep learning models. The models include MlBox (model1), TPOT (model 2), Random forest (model 3), MLP (model 4), Linear regression, (model 5), Decision trees (model 6), polynomial feature (model 7), XgBoost (model 8) and CatBoost (model 9). CatBoost method outperformed as compared to rest of the methods in terms of prediction accuracy as 0.8617. The outcome of the research can better be utilized in healthcare sector to predict the vaccination of several other diseases in the situation of pandemic.

Background

H1N1 according to virology is a subtype of Influenza. It is a virus that is also written as (A/H1N1). The influenza ‘A’ virus was also a common reason for Influenza (flu) in 2009-2010 and also is associated with the great plague of the 20th century that devastated Spain (1918-1920). It is stated to be an Orthomyxovirus, it includes glycoprotein like, haemagglutinin and neuraminidase and that is also the reason it is known as H1N1 rely upon the type of H or N Antigens [1].The H1N1 virus was declared a pandemic in June of 2009 by W.H.O. (World Health Organization) which also confirmed that the strain comes from a swine origin. This virus was the cause of approximately 150,000 to 600,000 deaths worldwide recorded in tenure of 20 months [2]. Seasonal flu or Flu season, on the other hand, is quite common and not that deadly. It occurs in recurring time periods. It also mostly occurs in the cold weather. The three main virus families that can be pinpointed for the seasonal flu are Influenza virus A, B and C [3]. Researchers have tried to use various external environmental factors for improving the accuracy of influenza prediction models. Weather conditions like humidity, temperature, etc. affect the transmission of this virus. Multi-progression, artificial neural networks, long short term memory, convolutional neural networks are some of the machine learning algorithms which have been already used for influenza prediction. Both statistics and machine learning method for influenza prediction have been successfully implemented, but they have not been able to capture the effects of external environmental factors.

In this project, we have used various deep learning models for predicting the probability if an individual has taken the H1N1 vaccine or seasonal flu vaccine. We have compared the results of various machine learning models to improve the precision of prediction. Along with logistic regression, random forest, decision trees, etc. we have tried gradient boosting algorithms.

Not any particular research paper has been written in the same field as the topic of "Flu Shot Learning: Predict H1N1 and Seasonal Flu Vaccines” but some similar research paper topics have been published. Bish et al. [4] in their paper have described psychological and demographic factors influencing the spread of the H1N1 vaccine. The analysis showed that the perception of the vaccine as an effective strategy to prevent both levels of risk and vaccine follow-up posed by the 2009 pandemic flu outbreak has increased. In assessing the threat, factors influenced belief in the risk of developing H1N1 flu and anxiety. Past behaviour also had a strong effect. People who have had seasonal flu vaccines in the past are more likely to get the H1N1 vaccine. Population factors that increase vaccine use are: older people, sex, ethnic minorities, and for health professionals, a doctor. Some work based on deep learning models has also been done. Xue et al. [5] have worked on multiple regression models and artificial neural networks to monitor flu activity. This paper focuses on a wide variety of models that rely on data and values from the Google Flu Trends (GFT) and the Centres for Disease Control (CDC) to predict the characteristics of the flu. The results showed that the GFT + CDC regression model is better for monitoring influenza activity than the GFT and CDC regression models. After adding seasonal information (high flu vs. low flu), predictive results improved.A similar kind of model based on such data was proposed by Venna et al. [6] to predict flu in real time. This article proposes a new data-driven machine learning method using long-term multi-stage memory-based forecasting to predict flu. The results showed that the LSTM-based deep learning method showed better results than modern forecasting methods.

Many studies have been carried out to predict influenza at or beyond the city scale. However, it is difficult to predict it for a larger area. In addition, existing models often ignore spatial correlations of influenza activity between neighbouring regions, although they are very useful in predicting influenza. Xi et al. [7] used the CNN model for influenza predictions. The ReLu activation function is used for faster CNN training.The results of this deep learning model showed that the proposed deep residue model exceeded four basic models, namely: logistic regression, artificial neural network Long-term memory, and space-time LSTM. These four basic models, as well as CNN, have already been used to predict influenza, of which CNN performed better. Joshi et al. [8] did a very similar job using LSTM. They evaluated three text classification methods for identifying vaccination behaviour. A statistical approach, a rule-based approach, and a deep learning approach were implemented. After comparing the three methods it was found that a combination of statistical classifiers using task-specific classifiers and deep learning models that used a pre-trained language model and LSTM classifier give comparable results. Yang et al. [9] also used LSTM and RNN to predict the flu. It also points out the disadvantages of RNN gradient disappearance, so that LSTM can be used effectively to reduce current weight. Zhang and Nawata [10] used LSTM to predict the spread of influenza. They used four different LSTM multi-stage forecasting algorithms to predict the spread of influenza in several stages. The results showed that the highest accuracy was achieved using various single-channel forecasts in the six-layer LSTM structure.

Methodology

Data Pre-processing

The dataset used in this research work was provided by DrivenData which comes from the National 2009 H1N1 Flu Survey (NHFS). We have used simple imputer and hot encoder for data pre-processing [11]. Algorithms and packages have nominal categorical data but when we use regression models only numeric data is required. Hence we must assign a unique numerical value to each category in the objects column. But this gives rise to a new problem. When the model does not recognize data as categorical data and processes these functions on a scale, but does not accept their nominal value, because the category value affects the weights assigned to objects. To avoid this problem, we use one hot encoder. We encode a category in a 1-hot vector, where the position in the vector refers to each category and its size is equal to the number of categories. Simple imputer uses one of three ways to fill in the missing data, e.g. mean, median, mode. Pandas library in python always identify missing values as NaN.

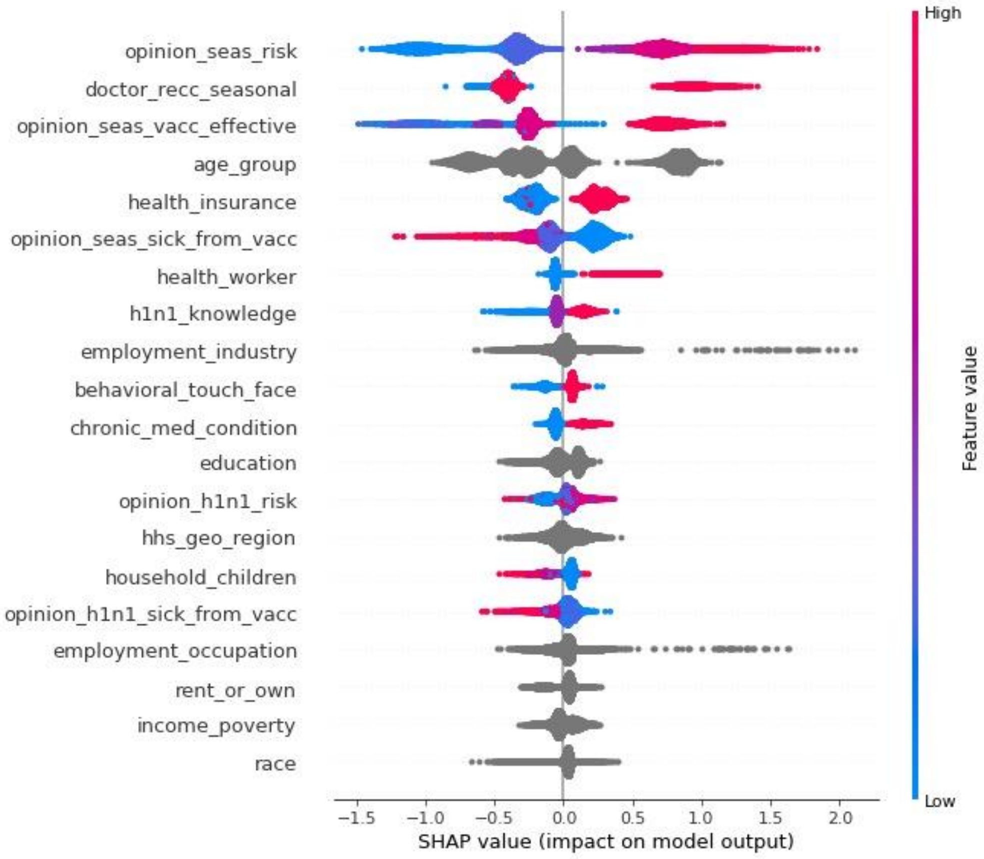

Fig. 1 Seasonal Flu feature importance for catboost

Missing values must be handled as they reduce the quality of any of our performance metrics. Unless the data is pre-processed to the extent that an analyst will encounter non-essential values such as NaN it can also lead to incorrect prediction or classification and can also cause a high bias for any given model. Missing values can be represented as a question mark (?) or a zero (0) or minus one (-1) or as a blank. Fillna() function is used to fill NaN values. We have also used Univariate selection. Univariate feature selection works by selecting the best features based on Univariate statistical tests. This can be seen as a prerequisite for the assessor. The scikit-learn feature displays the selection function as an object that implements the conversion method.

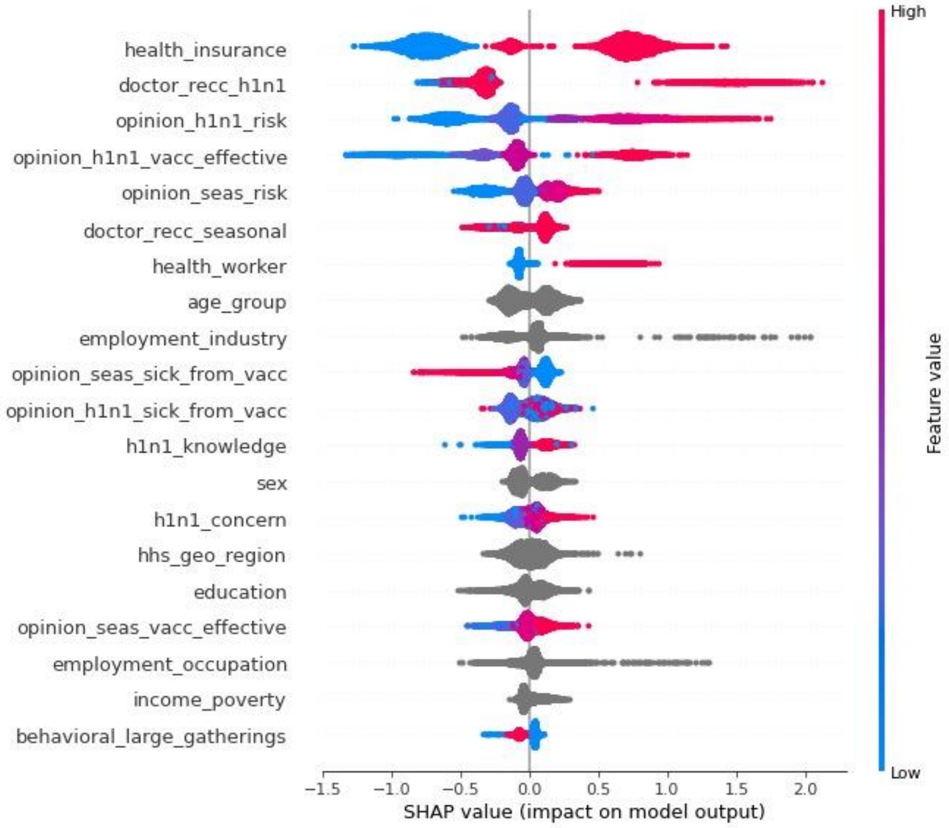

Fig. 2 HINI feature importance for catboost

CatBoost

CatBoost [12] is a new open library for increasing the gradient. It successfully handles categorical functions and provides better performance than other public gradient enhancement algorithms in terms of quality in public popular datasets. Mostly decision trees are used as base predictors for implementation of gradient boosting algorithm. But it's convenient only for numerical features. Many dataset contain categorical values which are a discrete set of values (like respondent id, or name of a city) which are not comparable with each other. Hence in most gradient boosting algorithms, they are converted to a numerical value before implementation. CatBoost is the algorithm that successfully handles categorical features and uses them while training as opposed to pre-processing time. Another advantage is that it uses a new scheme to calculate the price of the leaves when choosing the tree structure. It helps reduce over fitting. This algorithm is called CatBoost for categorical boosting. CatBoost has both CPU and GPU implementations. The formula (shown in equation (1)) is used to convert categorical features to numerical features.

$$\text{Target} = (\text{count} + \text{prior}/\ (\text{total Count} + 1)) \qquad (1)$$

count = How many times the label value was equal to "1" for the items with the current categorical value.
Priority (prior) = it is a temporary value of numerator
Total Count = total number of objects (up to current) that have a categorical value corresponding to the current one.

Linear Regression:

Linear regression is a machine learning algorithm. It is used to predict a number of variables, such as salaries, sales, etc. Linear regression predicts the output of a class (y) dependent variable based on a given independent variable (x). It is called linear regression because it finds the linear relationship between input and output. Mathematically linear regression can be represented as shown in equation (2).

$$\mathbf{y = a_0 + a_1 x + \varepsilon} \qquad \mathbf{(2)}$$

Here,

Y= Dependent Variable (Target Variable)
X= Independent Variable (predictor Variable)
a0= intercept of the line (Gives an additional degree of freedom)
a1 = Linear regression coefficient (scale factor to each input value).
ε = random error

The values for the variables x and y are training data sets for the representation of the linear regression model.

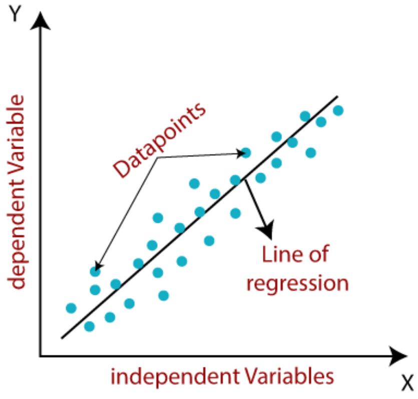

Fig. 3 example of linear regression [13]

Gradient boosting classifiers:

Gradient boosting algorithms just like random forests are supervised machine learning algorithms used for classification and regression. It is an ensemble learner. The implementation of this technique has various names; the most common name is gradient boosting machine (gbm) or XgBoost. XGboost is popular because it has been the winning algorithm in the Kaggle competitions. XGboost is eXtreme Gradient Boosting. GBM is also called MART (Multiple Additive regression trees).

Decision Tree:

A decision tree is a tool that makes use of a tree-like model of decisions and outcomes. Tree like algorithms are widely used. This method comes with high accuracy and accuracy. They can solve classification and regression problems likewise.

MLBOX

MlBox is an automated machine learning Python library. It supports distributed data processing, cleaning, formatting and complex algorithms such as Lite GBM and XgBoost. Also supports model stacking, which combines model information to create a new model to perform better than the individual model.

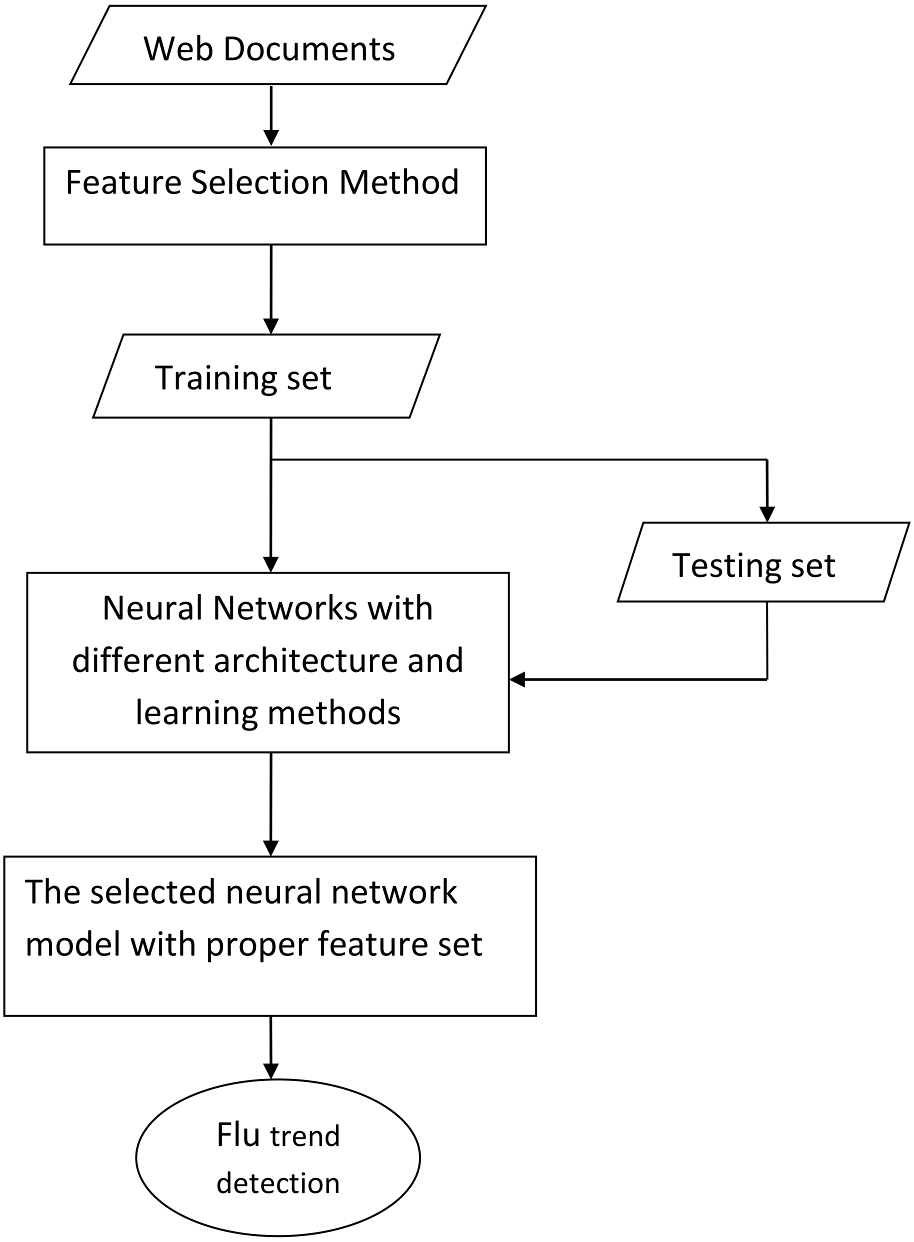
Web Documents
Feature Selection Method
Training set
Testing set
Neural Networks with different architecture and learning methods
The selected neural network model with proper feature set
Flu trend detection

Fig. 4 flow chart for influenza prediction

Hyper-Parameters

Machine learning algorithms like gradient boosting algorithms, neural networks for regression, random forest involve many hyper-parameters that need to be set before running the model. Hyper-parameters are the variables that determine the structure of the network (number of levels) and determine the learning rate. Accuracy can be increased by many units hidden within a level with regularization techniques. Fewer units can cause insufficient adaptation. The learning speed is the speed with which it determines the speed with which the weights in the neural networks or the coefficient in the regression change vessel. If the learning speed is low, it slows down the learning process but can be easily adapted. But if the speed of learning is high, it increases the speed of learning, but it cannot converge.

Experimental Design

A total number of nine algorithms were used to get the results. Out of which CATBoost gave the best result. All the algorithms used as the basis of machine learning models are summarised in Table 1.

Table 1. Details of performance measure

S. No.	Algorithm	Accuracy
1.	Polynomial Feature	0.72
2.	Random Forest	0.793
3.	Decision Trees	0.80
4.	MLP	0.82
5.	TPOT	0.828
6.	Linear Regression	0.83
7.	MlBox	0.85
8.	XgBoost	0.8564
9.	CatBoost	0.8620

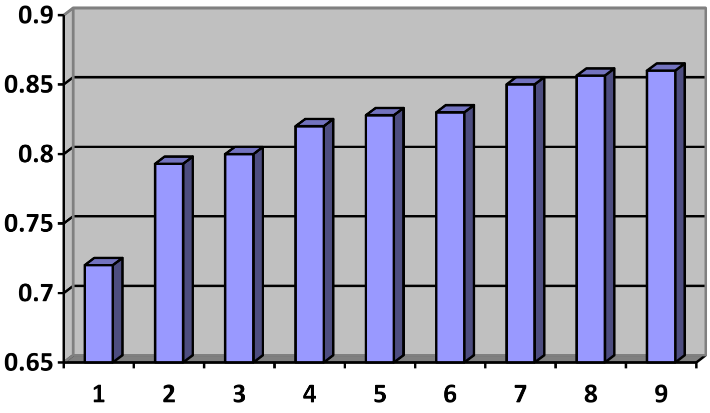

Fig. 5 diagrammatic representation of result

Evaluation Measures

As this is a competition arranged by DrivenData it uses ROC AUC (Area under Acquiring Operating Attributes) as a competitive evaluation metric. The result is probability, not binary labels. ROC is a probability curve and AUC is a measure of degree or isolation. The higher the AUC, the better model is on differentiating between patients with the disease and someone with the disease. The AUC value lies between 0.5 to 1, 1 being the best. TPR=True positive rate, FPR=False positive rate, TN=True negative, FN=False negative

$$\text{TPR (/Recall/Sensitivity=} \quad \text{TP/ TP+FN} \tag{3}$$

$$\text{FPR = 1-Specificity =} \quad \text{FP/ TN+FP} \tag{4}$$

$$\text{Specificity =} \quad \text{TN/ TN+FP} \tag{5}$$

In this research we have compared the results given by the various machine learning algorithms. The evaluation metric used is ROC AUC for each of the two target variables. The average of these two scores is considered as final score. These curves give probability as the output. The higher the probability better is the performance of the algorithm. For example, the probability given by CatBoost algorithm is 0.86 and by XgBoost is 0.8564. Hence CatBoost algorithm is the better performer among the two.

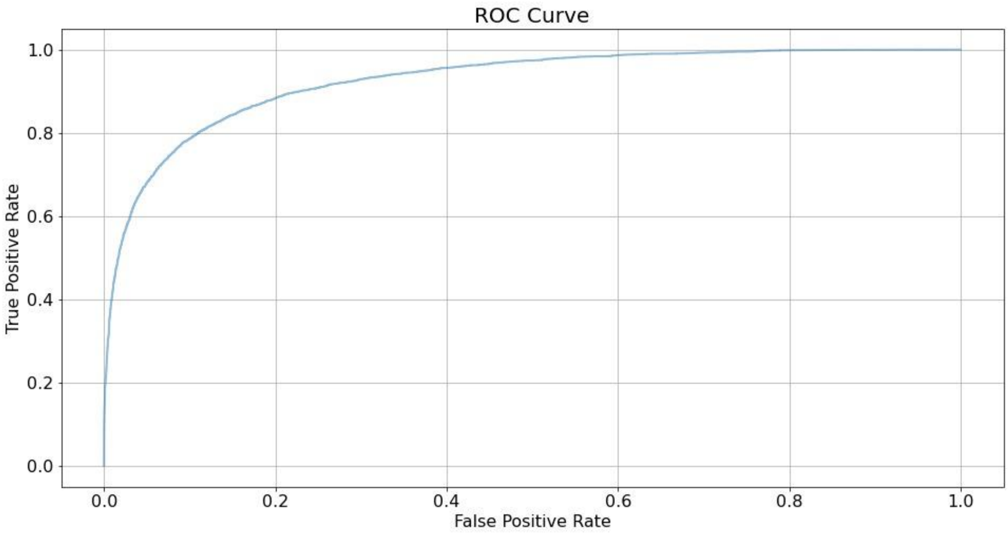

Fig. 6 ROC AUC for H1N1 flu (catboost)

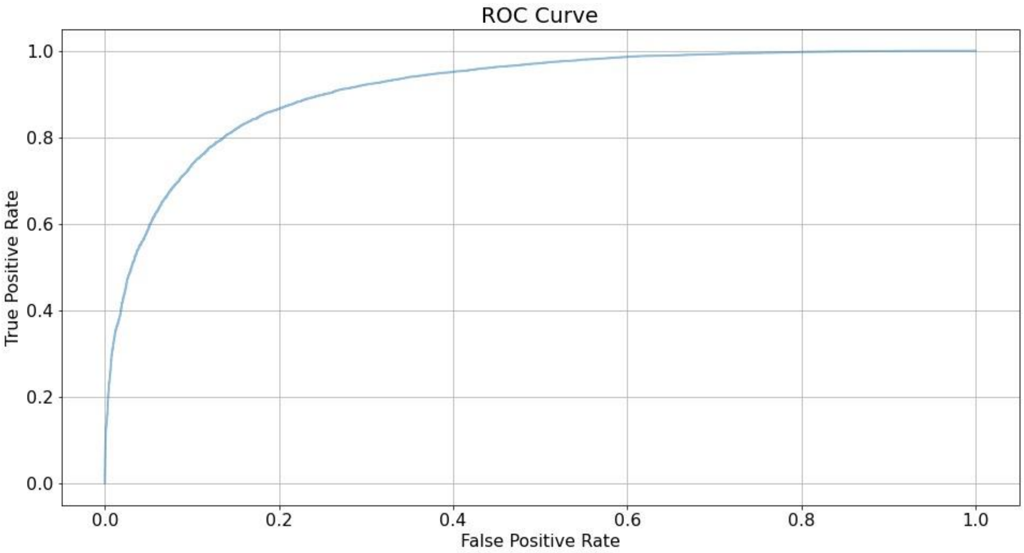

Fig. 7 AUC - ROC Curve for seasonal flu (catboost)

Conclusion

We were provided with a database of H1N1 and flu survey that was conducted in 2009. We were supposed to predict if a person will take vaccination in upcoming season or not based on their previous behaviour. We used 9 algorithms on the dataset provided and were able to get a good accuracy rate with a lot of them. The best accuracy was given by “CatBoost”. The accuracy was 0.86 or 86%. We conclude this paper by stating that “CatBoost” gave the best results of all the algorithms used and could be used for further prediction models in this category.

Software and Hardware Requirements

Python based Machine Learning libraries have been exploited for the development and experimentation of the project. Tools such as Anaconda Python, and libraries such as skit-learn, pandas have been utilized for this process. Training was conducted on Google colabs online platform. NVIDIA GPUs for extensive training of the machine learning models for can also be utilised.

References

[1] “Influenza A virus subtype H1N1”. Retrieved from https://en.wikipedia.org/wiki/Influenza_A_virus_subtype_H1N1, Accessed 19 May 2020.

[2]”What is the pandemic (H1N1) 2009 virus?”(24th February 2010). Retrieved from https://www.who.int/csr/disease/swineflu/frequently_asked_questions/about_disease/en/, Accessed 18 May 2020

[3] Charles Patrick Davis**. "**Flu (influenza, conventional, H1N1, H3N2, and bird flu [H5N1]) facts**".** Retrieved from https://www.medicinenet.com/influenza/article.htm, Accessed 19 May 2020.

[4]A. Bish, L. Yardley, A. Nicoll, and S. Michie, "Factors associated with uptake of vaccination against pandemic influenza: A systematic review, *Vaccine*. 2011, doi: 10.1016/j.vaccine.2011.06.107.

[5] Xue, Hongxin & Bai, Yanping & Hu, Hongping & Ldfs, Hdsajkkd. (2017). Influenza Activity Surveillance Based on Multiple Regression Model and Artificial Neural Network. IEEE Access. PP. 1-1. 10.1109/ACCESS.2017.2771798.

[6] S. R. Venna, A. Tavanaei, R. N. Gottumukkala, V. V. Raghavan, A. S. Maida, and S. Nichols, "A Novel Data-Driven Model for Real-Time Influenza Forecasting," *IEEE Access*, 2019, doi: 10.1109/ACCESS.2018.2888585.

[7] [1] G. Xi, Y. Li, L. Yin, and S. Mei, "A deep residual network integrating spatial-temporal properties to predict influenza trends at an intra-urban scale," in *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery, GeoAI 2018*, 2018, doi: 10.1145/3281548.3281558.

[8] A. Joshi, X. Dai, S. Karimi, R. Sparks, C. Paris, and C. R. MacIntyre, "Shot Or Not: Comparison of NLP Approaches for Vaccination Behaviour Detection," 2019, doi: 10.18653/v1/w18-5911.

[9] C. T. Yang *et al.*, "Influenza-like illness prediction using a long short-term memory deep learning model with multiple open data sources," *J. Supercomput.*, 2020, doi: 10.1007/s11227-020-03182-5.

[10] J. Zhang and K. Nawata, "Multi-step prediction for influenza outbreak by an adjusted long short-term memory," *Epidemiol. Infect.*, 2018, doi: 10.1017/S0950268818000705.

[11] Pathak, Manish (January 6th 2020)."Handling Categorical Data in Python"Retrieved from https://www.datacamp.com/community/tutorials/categorical-data,Accessed 12 June 20

[12] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: Unbiased boosting with categorical features," in *Advances in Neural Information Processing Systems*, 2018., Accessed 12 June 2020

[13] Granville, Vincent (May 20 2018). "Why logistic regression should be the last thing you learn when becoming a data scientist". Retrieved.from https://st4.ning.com/topology/rest/1.0/file/get/2808358994?profile=original, Accessed 9 June 2020.