International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-4, April 2019

Prediction of Telecom Churns and Consumer Behaviour using Recurrent Neural Networks

Lakshay Arora, Aayush Kapur, Lavanya K

Abstract - In today's world, telecommunication is a very common thing. People register on a company's platform. However, due to various issues a normal person might churns or unsubscribe to a particular telecommunication provider. So being able to predict churn is a pretty handy tool to a marketer or advertiser. They would be able to predict if a user will probably quit their telecommunication network or not. In this paper, we created a telecommunication churn prediction system by using recurrent neural networks. The basic idea of this is to create a system so that we can predict consumer behaviour and tell us whether a consumer will want to give a service based on his/her calling patterns, recharge frequency and amount and a host of other factors. This has been achieved by the help of recurrent neural networks. Recurrent neural networks are basically normal neural networks with a feedback loop. It takes its generic input and the output of the previous stage in each neuron. We initially, pre-processed the data, then creating the model of RNN by varying different parameters. The data was then passed to the RNN to train the model. Once that was done, the model was tested with testing data and results were produced. We were able to get pretty good accuracy with our model.

Keywords- Churn Prediction, Feedback loop, Neural Network, RNN

I. INTRODUCTION

In this age of internet-based businesses the ability to utilize data for optimizing business practices is sought after. Many businesses have a large number of users when they start off and slowly this number drops as the customers who earlier were part of the business lose their interest in it and thus the number of users slowly declines over time. This task of predicting the customer attrition over time is known as churn prediction. It occurs when the customers stop doing business with a company and the companies are very much interested in identifying the factors and reasons due to which their customers leave them so that they can change their business model accordingly thus losing less customers over time. A big help to solve this problem can be given by the usage of machine learning techniques which may help make the prediction based on the data collected in the past. We can utilize the deep learning techniques specifically the neural networks which can serve this purpose.

Manuscript published on 30 April 2019.
***** Correspondence Author (s)
Lakshay Arora, VIT University, Vellore, India
Aayush Kapur, VIT University, Vellore, India
Lavanya K, VIT University, Vellore, India

The recurrent neural networks are a specific type of neural networks which are suited for this since they have a component which takes into account the periodic happening and serves to make a better prediction.The ability to correctly predict the customer churn based on a host of factor is well sought after and different approaches have been tried to do it. We propose to use RNNs for telecom based customer churn where we predict whether a user will switch over to some other telecom service

II. II. RELATED WORKS

Mikolov et all worked to develop a new RNNs language model for speech recognition. They tried to reduce complexity in this process by using the RNN approach, they used a combination of RNN language models to reduce this complexity. They found that it led to reduction in the error late of speech recognition specially for words and thus led to better results.[3]Graves et all tried to increase the performance of RNNs for speech recognition, their usage is limited since the performance is not good for speech recognition however, they have tried to improve its performance for this use case.Wei, C.-P., & Chiu tried to implement a new technique for churn prediction using data mining, they proposed to utilize multi class classifiers combined together to predict the churn of telecom customers. Their work was targeted to the mobile phone subscribers whether he is going to switch over to a different provider based on a host factors, this can be pretty useful for providers as they can make individuals who are more likely to churn a focus of their selective customer retention strategy. [8]Hao et all used RNN for prediction of diseases, it was a multi modal disease prediction approach where they utilized RNN structure for finding out the probability of occurrence of a disease in an individual based on a host of factors, characterized by different attributes. [9]

III. III. BACKGROUND STUDY

A. Neural Network

An average neural network has anything from two or three dozen to hundreds, thousands, or even countless units in a movement of layers, all of which interfaces with the layers on either side. Some of them, known as information units, are expected to get distinctive sorts of information from the outside world that the network will attempt to get some answers concerning, see, or by and large procedure.

Distinctive units sit in actuality side of the network and banner how it responds to the information it's discovered; those are known as yield units. Amidst the data units and yield units are somewhere around one layers of hid units, which, together, structure the greater part of the fake personality. Most neural networks are totally related, which suggests each covered unit and each yield unit is related with every unit in the layers either side. The relationship between one unit and another are addressed by a number called a weight, which can be either positive (if one unit empowers another) or negative (if one unit covers or limits another). The higher the weight, the more effect one unit has on another. [4]Data courses through a neural network in two different ways. At the point when it's picking up (being prepared) or working typically (subsequent to being prepared), examples of data are sustained into the network by means of the info units, which trigger the layers of shrouded units, and these thus land at the yield units. This normal structure is known as a feedforward network.[5] Every unit gets contributions from the units to one side, and the sources of info are duplicated by the loads of the associations they travel along. Each unit includes every one of the sources of info it gets along these lines and (in the least difficult sort of network) if the entirety is in excess of a specific limit esteem, the unit "fires" and triggers the units it's associated with (those to its right side). Neural networks learn things in the very same way, regularly by a feedback procedure called backpropagation When the network has been prepared with enough learning precedents, it achieves a point where you can give it a completely new arrangement of sources of info it's never observed and perceive how it reacts.

B. Recurrent Neural Networks

A recurrent neural network (RNN) is a class of neural network where associations between hubs structure a coordinated graph along a transient grouping. This enables it to display worldly unique conduct. Not at all like feedforward neural networks, RNNs can utilize their inside state (memory) to process groupings of information sources. The expression "recurrent neural network" is utilized to allude to two expansive classes of networks with a comparable general structure, where one is limited impulse and the other is vast impulse. The two classes of networks display fleeting dynamic behaviour. A limited motivation behind recurrent network is a coordinated acyclic graph that can be unrolled and supplanted with a carefully feedforward neural network, while an unbounded drive recurrent network is a coordinated cyclic graph that cannot be unrolled.

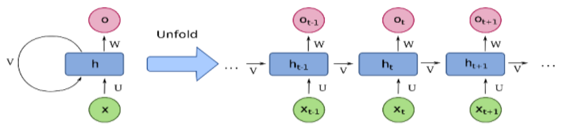
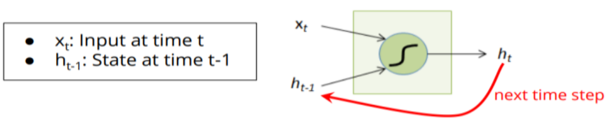


Fig 1 Unfolded basic recurrent neural network



$$h_t = f(W_h h_{t-1} + W_x x_t)$$

Fig 2 Recurrent Neuron

C. Customer Churn

So as to satisfy the need to hold the customer, different churning techniques have been created as an advertising procedure to continue making due in this unstable and fast developing condition. Churn is much of the time talked about in a correspondence setting, where it alludes to the inclination of cell phone users of switching suppliers. The most fundamental explanations behind churn are disappointment with a current supplier, the bait of a lower cost from an alternate supplier, an adjustment in the endorser's geographic area, the longing for expanded association speed, or a requirement for various or upgraded cell phone inclusion. A business needs to spend significantly more assets when endeavoring to win new customers than to hold existing ones. Subsequently, much research has been put into better approaches for distinguishing those customers who have a high danger of churning. Anyway, customer holding endeavors have additionally been costing associations a lot of asset. In light of these issues, the up and coming age of churn prediction should concentrate on exactness.

IV. IV. METHODOLOGY

RNN are pretty much a neural network but has a feedback loop. In this basically, at every stage, each neuron gets its regular input and the output of the previous stage. This is what differentiates RNNs from generic neural networks. [1] As they receive outputs from previous stages, they become suitable for time series predictions. Due to this feedback loop, the recurrent neurons basically gain some kind memory that helps it take into consideration previous states while making a decision. [5] We have used special type of neurons which have memory cells and are called LSTM (Long short-term memory). These are basically ordinary neurons with some extra properties that allow them to perform even better.[4] The first step we took was the collection of data and pre-processing that data. So, we initially collecting data which was easy, but formatting the data so that it could be put in the recurrent neural network was a pretty hard process. [2] First, we basically used a data integration tool to convert our telecom data into PostgreSQL and then to the csv format. [6] In this RNN, we predicted who was going to perform actions in the upcoming month so those who are not going to perform any action will be at high risk. So, then we decided to keep each column of the input as a representation of a recipient each row as a representation of a month. If a user calls or recharges at least once during that particular month then the value for that month is set to 1. [3] We also made the length of each timeseries equal by padding them with zeroes. Keras has been used to build the RNN as it is easy and pretty straightforward. Three layers instances have been used to build the linear stack which includes:

1) LSTM Layer: The main recurrent neural network layer of the constructed network.
2) Dense Layer: A highly densely-connected layer.
3) Activation Layer: This layer specifies the activation function that will be applied to the output. We chose the linear function as our activation function.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Pseudocode:

Target: churn prediction of telco consumer dataset

Inputs: telco dataset from kaggle

1. Import the dataset
2. Split the into test train parts (70% train dataset and 30% test dataset)
3. Create two models
   3.1 for model 1
      3.1.1 create LSTM based layer
      3.1.2 highly dense layer was added
      3.1.3 activation was chosen as linear function.
   3.2 for model 2
      3.2.1 create three LSTM based layers.
      3.2.2 highly dense layer was added
      3.2.3 activation was chosen as sigmoid function.
4. create a modified model
   4.1 create five LSTM based layers.
   4.2 there should be a dropout of 0.2 after every LSTM layer
   4.3 create the hidden layers
   4.4 activation function was chosen as sigmoid function.

The batch size of the LSTM layer was set to 300 and the batch size of the dense and activation layer was set to 1349.The number of the neurons in the first layer was set to 1349, the dataset size was set to 300 and the number of neurons per layer was set to 300.

The data was then split into training data and testing data. We chose mean square error as the loss function for splitting the dataset into training and testing data as this function minimized the average of the squares of the errors produced during propagation and hence was very useful. We trained the model for 50 epochs, 80 epochs and 200 epochs which generated various different charts. It was noticed that the loss in the training dataset decreased as we increased the epochs value. However, we also noticed that on the testing dataset the loss did not linearly decrease but increased and decreased randomly. However, when we used a larger dataset it gave us better results. From that we realized that this learning works only for large datasets and not for smaller datasets. In small datasets, it memorizes the training points and cannot generalize on unseen data.

V. RESULTS

We implemented the RNN with various timesteps, the dataset used was the Kaggle telecom dataset which had been modified for our purpose we removed the categorical variable fields relating to the plans of the customers, the yes and no were encoded with 1's and 0's.

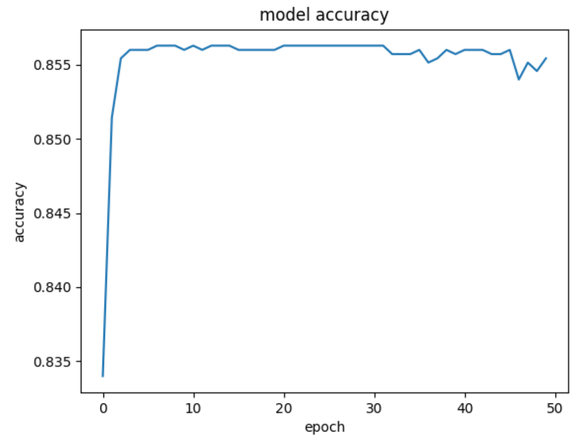
We utilized a standard model approach where we had two models, in the first model we had one LSTM layer and then the hidden layers completed by the linear activation function, since our output is categorical that is whether the customer will churn or not thus the performance was not very good.

In the second model we had three LSTM layers followed by hidden layers and sigmoid activation function, it showed better performance compared to the previous model. For the standard model we had the following performance graphs

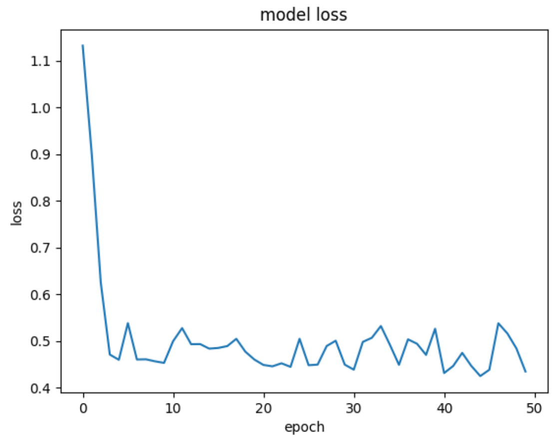
After trying out on with timesteps 1,2 and 5 over both the models, the performance was the best at timestep 5 with 80 epochs though the performance at timestep 5 was almost as good.

Then we implemented a new modified model based on this information with five LSTM layers and dropout of 0.2 after every layer, followed by hidden layers and the activation function was chosen as sigmoid, it gave comparable results to the standard model 2.

Table 1: Timestep 2 standard model 1



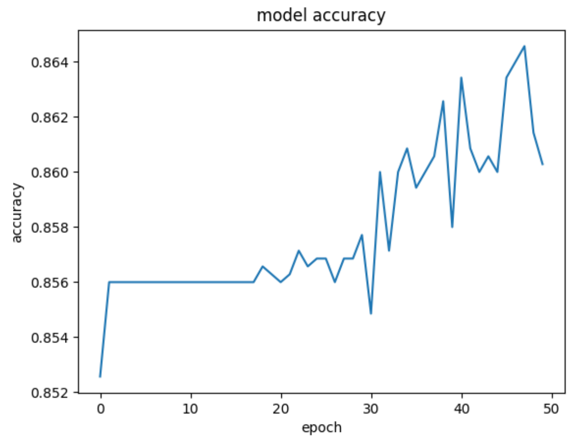
50 epochs accuracy



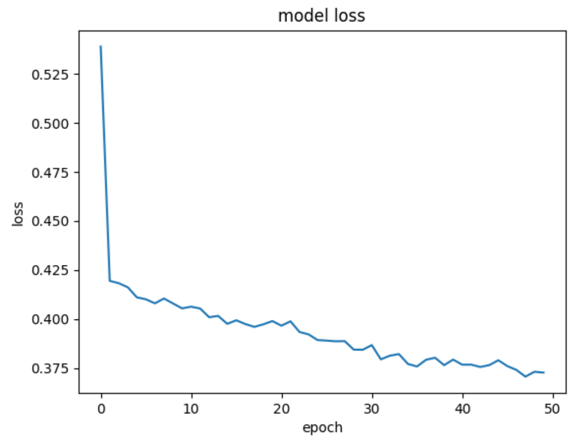
50 epochs loss

Table 2: Timestep 2 standard model 2

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

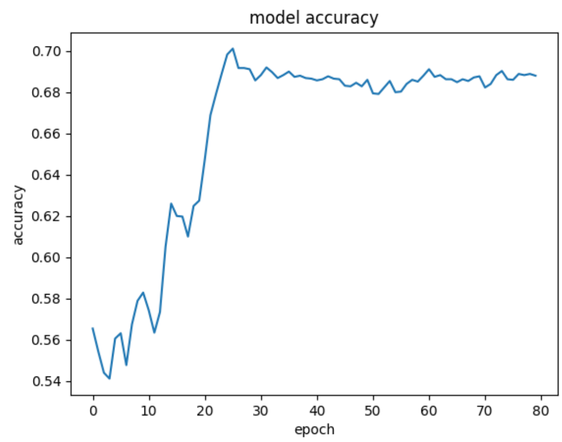


50 epochs accuracy

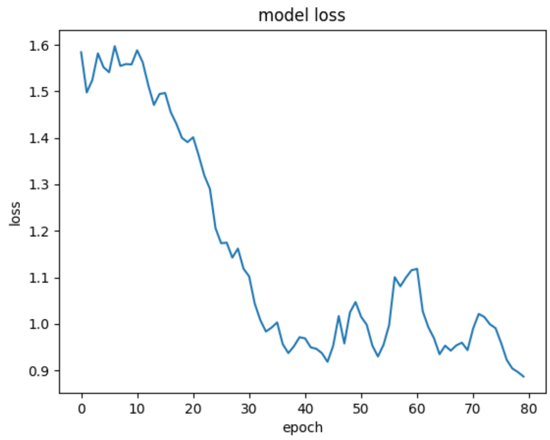


50 epochs loss

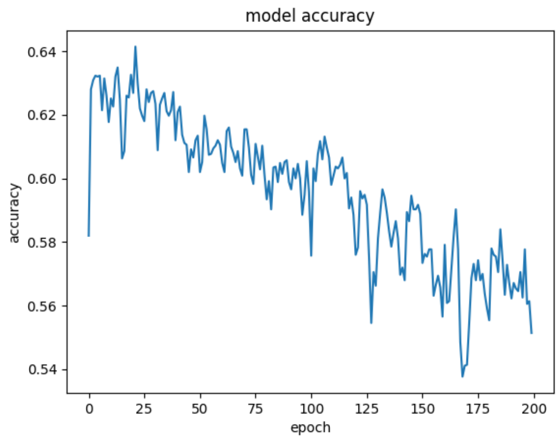
Table 3: Timestep 5 standard model 1



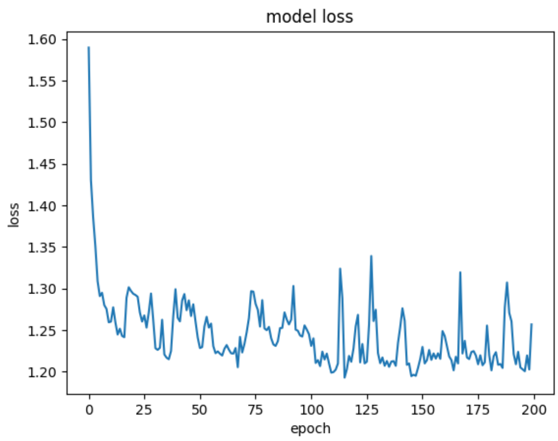
80 epochs accuracy



80 epochs loss

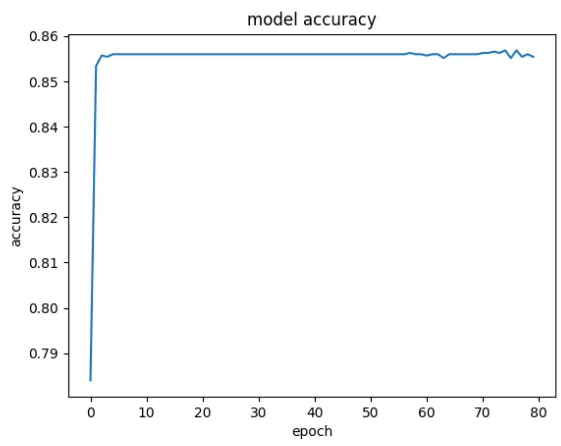


200 epochs accuracy

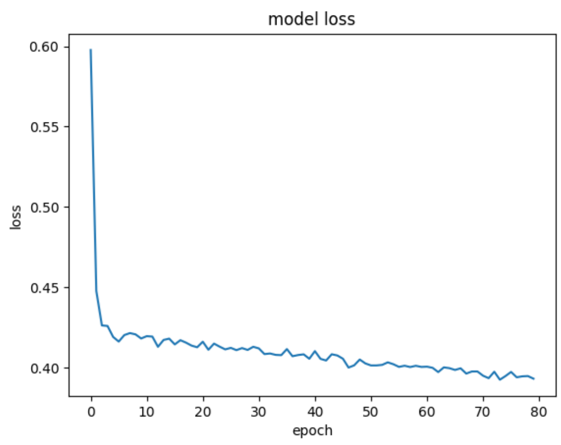


200 epochs loss

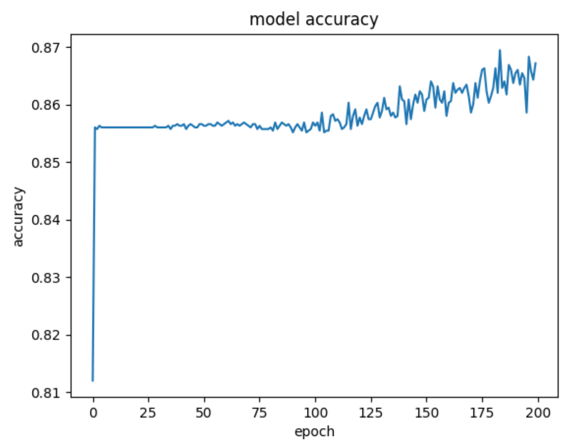
Table 4: Timestep 5 standard model 2



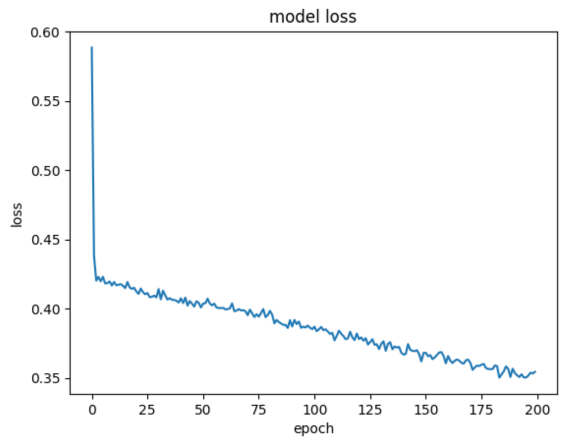
80 epochs accuracy



80 epochs loss

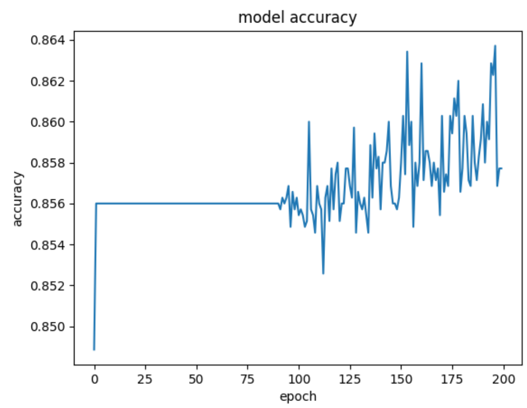


200 epochs accuracy

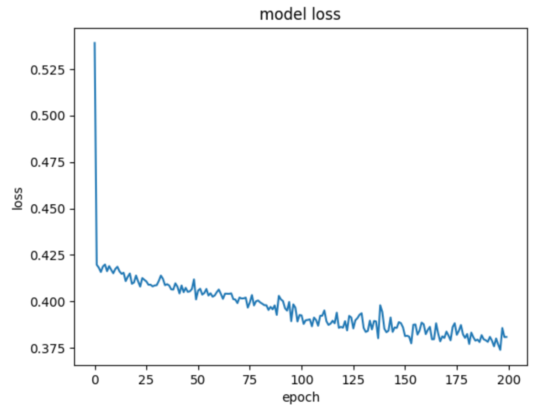


200 epochs loss

Table 5: Epochs 200 modified model



accuracy



loss

VI. CONCLUSION AND FUTURE WORK

Predicting whether a user will churn out your advertisements is a very important tool to a marketing company. [2] There are various ways this can be achieved by using data science. There are some things to keep in mind if you want to use churn prediction.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

First, most of these predictions are an iterative process and need to be updated as new data is generated. [10] Secondly, one dataset might fail totally whereas another might work great. This is because there might be specific information you to divide by which makes the difference. Finally, any difference hidden in the data will affect the work you do in trying to predict churns. As seen in this paper, recurrent neural networks provided us with a great way to create a model for churn prediction. We pre-processed the data, then created the model of RNN by varying different parameters. The data was then passed to the RNN to train the model. Once that was done, the model was tested with testing data and results were produced. We were able to get pretty good accuracy with our model. This tool can be used by various marketers and adverting companies to help creating a better advertising and marketing plan for themselves.

ACKNOWLEDGMENTS

We are very grateful to Prof. Lavanya K, our project guide, without whose guidance we would not be able implement and modify the proposed model. We would also like to thank VIT University management for giving us this opportunity to undergo this subject and providing us facilities to do this project.

REFERENCES

1. Markou, Eleni. “Recurrent Neural Networks for Email List Churn Prediction - DZone AI.” Dzone.com, 12 Sept. 2017,dzone.com/articles/recurrent-neural-networks-for-email-list-churn-pre-1.
2. Donges, Niklas. “Recurrent Neural Networks and LSTM.” Towards Data Science, Towards Data Science, 25 Feb.2018,towardsdatascience.com/recurrent-neural-networks-and-lstm-4b601dd822a5.
3. Mikolov, Tomáš / Karafiát, Martin / Burget, Lukáš / Černocký, Jan / Khudanpur, Sanjeev (2010): "Recurrent neural network-based language model", In INTERSPEECH-2010, 1045-1048.
4. Jain, A. K., Jianchang Mao, & Mohiuddin, K. M. (1996). Artificial neural networks: a tutorial. Computer, 29(3), 31–44. doi:10.1109/2.485891
5. Krizhevsky, Alex, et al. “ImageNet Classification with Deep Convolutional Neural Networks.” Communications of the ACM, vol. 60, no. 6, 2017, pp. 84–90., doi:10.1145/3065386.
6. Graves, A., Mohamed, A., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. doi:10.1109/icassp.2013.6638947
7. Burez, J., & Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. Expert Systems with Applications, 36(3), 4626–4636. doi: 10.1016/j.eswa.2008.05.027
8. Wei, C.-P., & Chiu, I.-T. (2002). Turning telecommunications call details to churn prediction: a data mining approach. Expert Systems with Applications, 23(2), 103–112. doi:10.1016/s0957-4174(02)00030-1
9. Hao, Yixue, et al. “Recurrent Convolutional Neural Network Based Multimodal Disease Risk Prediction.” Future Generation Computer Systems, vol. 92, 2019, pp. 76–83., doi:10.1016/j.future.2018.09.031.
10. Orac, Roman. “Churn Prediction.” Towards Data Science, Towards Data Science, 26 Jan. 2019, towardsdatascience.com/churn-prediction-770d6cb582a5.