International Conference on Computer, Engineering, Law, Education and Management 2017 [ICCELEM 2017]

ISBN	978-81-933584-7-4		**VOL**	01
Website	www.iccelem.com		**eMail**	iccelem@asdf.res.in
Received	05-June-2017		**Accepted**	03-Aug-2017
Article ID	ICCELEM003		**eAID**	ICCELEM.2017.003

Automatic Classification of Consumer Queries using LSTM

Long Cai[1], Kokula Krishna Hari K[2], Rajkumar Sugumaran[3]
[1]Research Scholar, Association of Scientists, Developers and Faculties, Hong Kong
[2]Secretary General, ASDF International, London, United Kingdom
[3]Vice President, Human Resource Management, Techno Forum Group, Bangkok, Kingdom of Thailand

Abstract: *In this digital era, it is very important to understand the consumer needs while dealing with large volume of data. In this paper, we focus on consumer queries and complaints. It is a difficult task to manually sit and arrange the queries and complaints in forums or online discussion sites related to a specific topic. We propose a method which automatically classifies the queries posted by consumer to its correct class. The system does this classification by using a technique called Long Short-Term Memory(LSTM). The LSTM network has the capability of learning long-term dependency features directly from the dataset without any manual effort. The model showed considerable accuracy when tested with validation data.*

Keywords: Artificial Intelligence; Deep Learning; Feed Forward Network; Recurrent Neural Network; Long Short-Term Memory.

INTRODUCTION

Machine learning algorithms need pre-defined features to work. It is a very difficult task to identify salient features from data. Domain data knowledge is very essential for applied machine learning. The process of transforming data into features is called feature engineering. The process of feature engineering is a time-consuming method. In deep learning, the neural network will learn the features automatically from raw data.

Feed Forward Network (FNN) is a type of Artificial Neural Network (ANN) where the information goes in forward direction only. The simple FNN has no hidden layers. In case of FNN with one perceptron, the computed output will be the sum of the product of their weights. When we step back and look at the data, we will understand the pattern of that data. By storing these patterns, we can predict the next sequence by just seeing theprevious sequence.

Recurrent Neural Network (RNN) stores information in the memory over time. The vanishing gradient problem in RNN makes it difficult to store long term dependencies. The network is trained using backpropagation algorithm. It uses chain rule that gives derivatives or partial derivatives of a function.

RNN requires complex architecture than non-recurrent networks. The chain rule requires lots of computation. The output of RNN is not only used for computing recurrent value but also for computing next value for time periods. In deep neural networks, there are lot of hidden layers. The fundamental flaw of recurrent neural network is the number of multiplications required to compute the updated weights. The computed coefficients or weights in the past hidden layers are small numbers. So, it's hard for RNN to learn from the past. Consider this sample sequence A saw B, B saw C, C saw D. In this example, we need to predict the next sequence after 'A'. 'A' will strongly vote for 'saw' and 'B' will vote for comma. The word 'saw' has equal chances of predicting B, C or D. So, there are chances to make wrong prediction. To predict correctly, we need to see what happened in the previous steps.

LSTM is a type of RNN with a set of gates to control the flow of information. The gates will select and forget information when it enters the memory. The on/off gates will decide what to release as prediction and what to keep internal.

The dataset used for this work is US Consumer Finance Complaints. It is about issues people experienced in marketplace. The 'issue' and 'sub issue' columns in the data shows the problems faced by consumers. The product column shows products like mortgages, student loans, payday loans, debt collection, credit reports, and other financial products and services. Each record or sequence is a combination of 'issue' and 'sub issue' column. The product column corresponding to each record is taken as class label.

RELATED WORK

Oguzhan Gencoglu [1] proposed a method to categorize the messages from Finland's largest online health forum. It is to reduce the manual effort in managing messages in the forum. He used a Naïve Bayes classifier to classify messages into 16 categories.

The Search result diversification enables the modern-day search engines to construct a result list that consists of documents that are relevant to the user query and at the same time, diverse enough to meet the expectations of a diverse user population. However, all the queries received by a search engine may not benefit from diversification. Sumit bhatia, Cliff Brunk and Prasenjit Mitra [2] proposed an idea to analyze web search queries and classify those queries into one of the classes. They achieved Strong classification results for this classifier.

Lorenxo A Rossi and Omprakash Gnawali [3] analyzed the discussion threads from coursera forums. They investigated several language independent features to classify the discussion threads based on the types of the interactions among the users. The features related to structure, popularity, temporal dynamics of threads are extracted.

Bernard J. Jansen and Danielle Booth [4] proposed a methodology to classify automatically Web queries by topic and user intent. This technique can be used for real time query classification of web searches.

D. Irazú Hernández,Jansen Parth Gupta, Paolo Rosso and Martha Rochagy [5] proposed a method to automatically extract features from corpora and analyzed the distribution of features and used NaIve Bayes and SVM to classify them.

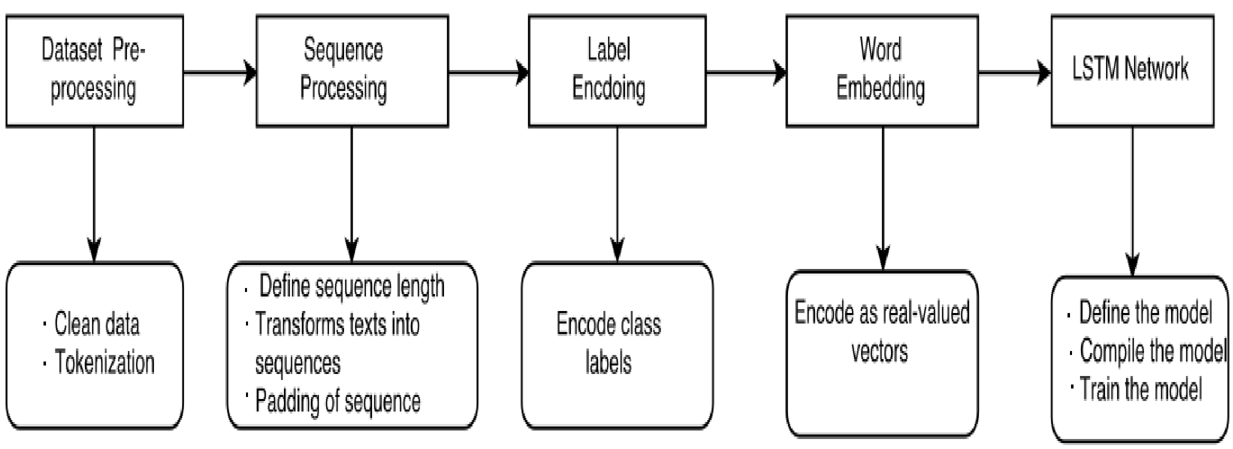

Fig. 1. Workflow of Model

Kristof Coussement, Dirk Van den Poel [6] introduced a technique to improve complaint-handling strategies through an automatic email-classification system that distinguishes complaints from non-complaints. This methodology reveals linguistic style differences between complaint emails and others.

SOLUTION APPROACH

The workflow of the model is shown in figure 1. The approach of sequence classification model is explained below.

Dataset Pre-Processing: Tokenization and stop word removal are the common pre-processing steps. Each record is converted into unit-gram tokens. Keras Tokenizer API is used for tokenization and other basic filtering of text. The most common words are removed from the raw text. Other unwanted symbols and numbers are also removed from the data using regex operations.

Sequence Processing: The first step is to transform each record to sequences. A vocabulary is created based on tokens. Each word in the dictionary is represented with a unique number. The next step is to pad the sequence length to the defined size. If the sequence length is smaller than the defined size, zeros will be added to pad the sequence. We can discard it if the size is higher than maximum sequence length.

Label Encoding: The algorithm will not be able to read class labels. The class labels are transformed into an array of numbers.

Word Embedding: The process of representing words in a continuous vector space based on position of words. This representation gives semantic similarity between words. The distributed representation of words are given as an input to the embedding layer.

LSTM Network: The model is defined by giving the number of memory neurons, activation function etc. We used SoftMax as the activation function. The total dimension represents the features. These features are converted into memory units. It is a fully connected network. LSTM network learns what to select and forget from the features. The model is then complied by defining the optimization algorithm and loss function. Adam optimizer algorithm is used in the network. It is then fitted to the model. The model is evaluated with the validation data.

Overfitting is the main problem in LSTM networks. The network will not be able to predict for unseen data. Our dataset may have thousands of parameters or dimensions. In this case, the parameters will try to adjust with the noise in the data. Then, the training accuracy will be high and out of sample data gives low accuracy. Adding dropout to the data will assign zeros to a percentage of data. This will happen for each epoch. Adding drop out layers can reduce the overfitting in LSTM networks. The loss functions are used by the optimization algorithm in every epoch to update the weights in every epoch. To predict categories, we have specific loss functions in keras library. The Hyper parameter tuning includes tuning of batch size, epochs, learning rate, activation functions, dropout layers, number of neurons etc.

RESULTS

This is an on-going work. The number of total records is 555957. A sample of 500 instances is taken from each class and a test set is generated. The validation score was 62.2 when tested with validation data. The accuracy of the model can be increased. The hyper-parameter tuning is going on to increase model's prediction accuracy. we have to evaluate the model with different parameters using Grid search process.

CONCLUSION

In this work, the main focus is automatic classification of complaints and queries in internet forums or sites. The usual machine learning classification problems needs pre-defined features or manual intervention is needed to create features from dataset. The future plan is to optimize the model and maximize the accuracy. Many predictive modeling problems of sequence classification can be solved using this method.

REFERENCES

1. Oguzhan Gencoglu, "Automatic Classification of Forum Posts: A Finnish Online Health Discussion Forum Case", EMBEC 2017, NBC 2017: EMBEC & NBC 2017pp 169-172J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
2. Sumit bhatia, Cliff Brunk and Prasenjit Mitra, "Analysis and automatic classification of web search queries for diversification requirements", Proceedings of the American Society for Information Science and Technology Volume 49, Issue 1, Version of Record online: 24 JAN 2013.

3. Lorenzo A. Rossi, and Omprakash Gnawali, "Language Independent Analysis and Classification of Discussion Threads in Coursera MOOC Forums", 15th IEEE International Conference on Information Reuse and Integration (IRI 2014), At San Francisco, CA.
4. Bernard J. Jansen, and Danielle Booth, "Classifying Web Queries by Topic and User Intent", April 14–15, 2010, Atlanta, GA, USA
5. D. Irazú Hernández, Parth Gupta, Paolo Rosso, and Martha Rocha "A Simple Model for Classifying Web Queries by User Intent", January 2012.
6. Kristof Coussement, and Dirk Van den Poel "Improving customer complaint management by automatic email classification using linguistic style features as predictors", Decision Support Systems 44 (2008) 870– 882.

.