INTELLIGENT SYSTEM FOR ANALYZING SENTIMENTS OF FEEDBACK

Shailendra Singh Kathait

Co-founder & Head
Artificial Intelligence & Machine Learning Lab
Valiance Solutions, Noida, India, 201301
shailendra.kathait@valiancesolutions.com

Shubhrita Tiwari

Data Scientist
Artificial Intelligence & Machine Learning Lab
Valiance Solutions, Noida, India, 201301
shubhrita.tiwari@valiancesolutions.com

Anvesha Bagaria

LNMIIT, Jaipur, Rajasthan, India, 302031
anvesha.bagaria@valiancesolutions.com

Vinod Kumar Singh

Senior Research Scientist
Artificial Intelligence & Machine Learning Lab
Valiance Solutions, Noida, India, 201301

Abstract:

Today almost in every industry, people are making use of data to increase their revenue. The data of feedbacks by the customers is used by the companies to analyze their reputation and performance. Manually reading each and every feedback and analyzing their sentiments is herculean. An automated process is required that predicts the sentiments from the feedbacks when given as input. Much research has been done in this field making use of different algorithms. This study designed an Intelligent System to predict the sentiments of the reviews. We have trained and tested several well-known classification algorithms on the manually labeled customer reviews (banking and insurance sector) scraped from different online sites. The study compared the performance of four well-known classification algorithms i.e., Convolutional networks, SVM, Bernoulli Naive Bayes and Bidirectional recurrent neural network. Results show that Bidirectional Recurrent Neural Networks with multiple LSTM (Long Short-Term Memory) layers attained the Maximum accuracy of 88%. This automated approach can be an ideal choice for an organization to find opinion about their product after launch.

Keywords: Sentiments analysis, reviews, Intelligent System, Convolutional neural networks, Bidirectional Recurrent Neural Networks B-RNN, LSTM.

1 Introduction:

In recent years the amount of data generated on internet had increased rapidly and continue to grow exponentially in near future. Every day, large amount of data is generated by social media, financial transactions, behaviour of the internet user, consumer's browsing and purchasing history. This data is being continuously explored by industry and academia for useful insights that can enhance revenue of the industry and user experience on internet[1].The data also includes huge chunk of raw text data in the form of product reviews, news or research articles, blogs, song lyrics, poems, etc.[2]. Labeling or categorization of this text data helps in efficiently searching relevant information about the product or query, from the huge data on internet.

The financial organizations are more concerned about their products and their reputation. Hence, they rely on customer reviews for improving their services and product. Recently, various text mining and machine learning techniques have been explored to draw insight about the sentiment polarity of the reviews[3].

The proposed work is the comparative study of performance of deep learning techniques and traditional classification techniques to find polarity of customer reviews of Banking and Insurance domain. The aim is to simplify the task of manually rating each and every feedback and automating them. This approach will give good estimate about the company's reputation in the market in very less time, so, that optimal decisions can be made in real time. The methodology employed deep learning techniques like convolution neural network (CNN), bidirectional recurrent neural network (RNN), bidirectional long short-term memory (LSTM) and two traditional text classification algorithms i.e., support vector machine (SVM) and Naive Bayes (NB).

2 Methodology:

The methodology consists of two main steps, the first step consists of data crawling from web resources followed by its manual rating for classifier training. The Second step involves training of classification algorithms.

2.1 Data construction:

Different online sites like www.mouthshut.com, www.wallethub.com etc. that consists of huge number of customer feedbacks on different banks and insurance agencies are fetched using different Python libraries such as Scrapy, Beautiful Soup etc. The data is then manually rated in three categories i.e., negative, neutral and positive.

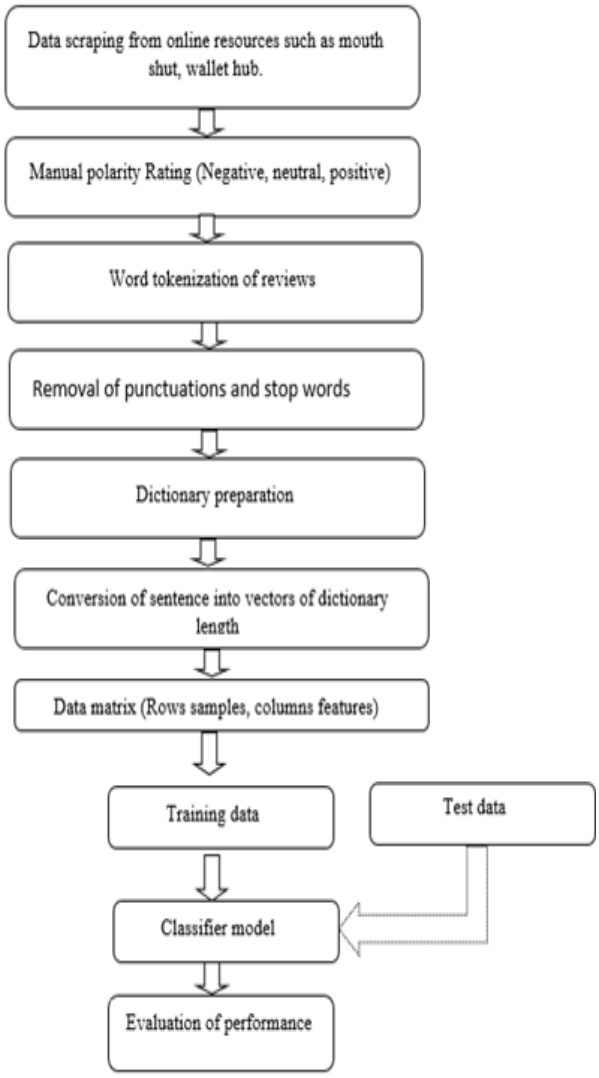
The constructed dataset consisted of labeled 5000 reviews in the document. The punctuations are irrelevant, therefore, removed from the reviews. The unique words of the dataset were ordered according to their frequency. The stop words such as the, is, an, about, etc., were also removed from the dictionary because they do not affect the sentiment polarity and are present in high frequency in all documents. The final dictionary of size D was prepared which have unique words of the dataset. Each review is represented as a binary vector of size D having 1 at index of dictionary location if that word is present in

the reviews. Hence, whole document can be represented as matrix:

$$D = [X^1, X^2, \ldots, X^{Tr+Ts}]^T = \begin{pmatrix} x_1^1 & \cdots & x_D^1 \\ \vdots & \ddots & \vdots \\ x_1^{Tr+Ts} & \cdots & x_D^{Tr+Ts} \end{pmatrix}$$

Where,Tr and Ts are the number of training and test review samples respectively. Each row of the matrix represents a review binary vector in $\Re^D$.

2.2 Methodology flow diagram:



The training samples were used for learning following supervised learning methods for comparative study.

3 Support vector machine (SVM)

Support vector machine is an efficient discriminative supervised classification model. It has been widely used in different classification problems of the industry due to its high prediction accuracy and ability to handle high-dimensional data[4]. These models separate two classes on the basis of two key concepts: In the first step, the kernel function is transformed from non-linearly separable input data to linearly separable high dimensional feature space. In the second step, the margin that separates optimal hyperplane is maximised that act as decision boundary for the classification[5].

4 Naive Bayes classifier

Naive Bayes (NB) is a generative (probabilistic) model for classification based on the assumption of independent features. It is applied to solve business intelligence problems like text mining, computer visionwhen training examples are less but features are independent of each other[6,7]. As this is a generative classifier, it learns a model of the joint distribution $P(X, y_j)$ of input and output, where input data is X and the output (class label) isy_j. The posterior from joint distribution is obtained using Bayes rule, i.e., the probability of class y_j for the input data X [6].

$$P(y_j|X) = \frac{P(X, y_j)}{P(X)}$$

$$= \frac{P(X|\, y_j)\ P(y_j)}{\sum_{y_k \in L} P(X|\, y_k)\ P(y_k)} \qquad (eq.1)$$

The parameters of distribution $P(X|\, y_j)$and $P(y_j)$were estimated by various parameter estimation methods like Maximum likelihood, Expected maximization, [6]. Finally, NB assigns the label of most probable target class Y to any given data instance x_i , i.e.,

$$L(x_i) = \arg\max_{y_j} P(y_j|\, x_i)$$

(eq. 2)

Where, $L(x_i)$is the label assigned to given data instancex_i.

5 Artificial Neural Networks (ANN)

Recently artificial neural network and its variants have been widely exploited for classification tasks to make intelligent systems for business decision making like predicting financial frauds, hand-writing recognition, computer vision, text mining, self-driving cars, etc. These models mimic the behaviour of brain neurons to learn from the given situations. The simplest form of ANN consists of only two layers of neurons, i.e., input layer and output layer, and can be applied for linear regression and linear classification purpose. The non-linear classification problems such as XOR, and needs to addressed by the introduction of hidden layers to introduce complexity to the model[8]. The size of the hidden layer (number of neurons/ layer)is also reduced significantly by adding more hidden layers. Additionally, the increment in hidden layers may cause overfitting of the model. Therefore, thetradeoff between complexity and overfitting should be considered while building a model[9]. Various architectures of ANN have been proposed for different problems.

5.1 Feed forward neural network

In feed forward neural network, each neuron or node in one layer is connected to every neuronin the next layer. Hence information is constantly "fed forward" from one layer to the next. The pairs of input and output values are fed into the network for many cycles to minimise errors using back propagation algorithm to update weights, so that the network can learn the relationship between the input and output[8]. The networks that have many hidden layers are deep neural networks (DNN), and each of the successive hidden layers learns more complex patterns than previous one. However, the introduction of successive hidden layers may make the model more specific to training examples which cause bad performance on the test or unseen instances. Another problem is faced in deep neural networks is “vanishing gradient problem”. The different layers in DNN are learning vastly at different speeds eg. the later layers in the network are learning well, on the other hand, the earlylayers may get stuck during training, learning almost nothing[10].

5.1.1 Convolution neural network:

Convolution is a particular case of DNN which overcomes the "vanishing gradient problem" by using weight initialization, feature preparation (through batch normalisation—centering all input feature values to zero), and rectified linear units (ReLU). This approach has been successfully used to extract deep features for classification tasks[11] and has been widely used in computer vision. Convolution network combines three architectural ideas to ensure some degree shift, scale, and distortion invariance: local respective field, shared weights (weight replication), and spatial or temporal subsampling[12].

Basically, a CNNconsists of two primary layers. In the case of computer vision, First, convolution layers that convolve local image regions independently with multiple filters, and the responses are combined according to the coordinates of the image regions. Second, the pooling layers summarise the feature responses, and pooling is processed with a fixed stride and a pooling kernel size[13]. The convolution neural networks (CNNs) do not consider contextual dependencies between different image regions because both convolution and pooling operations are locally applied on image areas separately. The contextual information is crucial to obtain real meaning from the raw sequential text data. Hence, other architectures of DNN have been developed to capture contextual information like recurrent neural networks (RNN) and its variant long short-term memory (LSTM) [13].

5.1.2 Recurrent neural network (RNN):

Various learning tasks require information from sequential data. The processes such as time series prediction, speech recognition, language modelling, translation, musical information retrieval, textmining, and video analysis, a model must learn from the sequential input. The current neural network (RNN) is a class of DNN designed for learning contextual dependencies among sequential data by using the recurrent (feedback) connections[13].

These are connectionist models that capture the dynamics of sequences via interconnected networks of simple units. In simple words, the architectures RNN can be considered as multiple copies of the same network, each passing a message to a successor. Unlike standard feedforward neural networks, this architecture enables RNNs information from an arbitrarily long context window. Although in past recurrent neural networks were difficult to train due to millions of parameters. However, recent advances in optimisation techniques, network architectures, and parallel computation have enabled successful large-scale learning with them[14].

The learning with RNNs is challenging due to difficulty in learning long-range dependencies. The problems of vanishing and exploding gradients occur when backpropagating errors across many successive time steps[10,15].The long short-term memory (LSTM) architecture of RNN described in next subsection uses precisely designed nodes with recurrent edges with fixed unit weight as a solution to the vanishing gradient problem.

5.1.2.1 Long short-term memory (LSTM)

LSTM is an RNN architecture designed to handle with long time-dependencies in sequential data such as sentences, speech etc. It was motivated by an analysis of error flow in existing RNNs ,where long time lags were inaccessible to existing architectures because the backpropagated error either blows up or decays exponentially[14]. Truncating the gradient where this does not do harm, LSTM can learn to bridge minimal time lags in excess of1000 discrete time steps even in the case of noisy, incompressible input sequences, without the loss of short time lag capability. This is done by enforcing constant error flow through "constant error carrousels (CEC)" within special self-connected units i.e., multiplicative gate units. These unitsact as memory cells and learn to open and close access to the constant error flow[15,16]. Hence, LSTM is designed to get rid of the vanishing error problem.

5.1.2.2 Bidirectional Recurrent Neural Network with multiple LSTM layers

The main idea of bidirectional LSTM (BLSTM) recurrent Neural Network is to capture context of both sides of the current word at $s(t)$ i.e., $s(t-n)\ to\ s(t)$ & $s(t)\ to\ s(t+n)$, to encode the text and make decision. A BLSTM processes input sequences in both directions with two sub-layers. Due to context capturing behavior these models have many applications in the field of image captioning, speech recognition and language modeling, and text mining [17].

6 Experimental setup and Results:

The performance of the above classification models on the review data compared. Although the Bernoulli Naïve Bayes had been widely used for text classification when data was less. However, in present scenario, the data is available in sufficient amount which is ideal for deep learning tasks[18].Our study also proves that deep learning techniques (BLSTM and CNN) do better sentiment classification compared to another conventional method due to the ability to capture more complex features and context on a large dataset (Table 1)[17].

Table 1: Summary of parameters setup used to train various models and their performance

Algorithm	Experimental setup	Batch size	Accuracy
BLSTM	Number of hidden layers= 2, number of neurons in hidden layer=500, number of LSTM layers=10	128	**88 %**
CNN	Pooling filter size= 2×2, Number of convolution layers=2	128	**84 %**
Bernoulli NB	Maximum likelihood parameters estimation form training data. i.e. probability of positive class.		**74 %**
SVM	Gaussian kernel, parameters were optimized through grid search algorithm. cost		**65 %**

7 Conclusion

This study showed bidirectional long short-term memory RNN is the ideal choice of the classifier to find polarity of review sentiments. This study can prove useful for the organizations to quantify their reputation or their product quality in real time so that necessary steps can be taken. Other potential applications of this work can be social media monitoring such as public opinion on certain topics, tracking sentiment towards products, movies, politicians, etc., improving customer relation models, detecting happiness and well-being, improving automatic dialogue systems, etc.

References

:

1. Lakhno VA, Nikolaievskyi OY, Skliarenko EV, Lytvynenko LO. Models and Tools for Automatization of the Linguistic Research. Journal of Theoretical and Applied Information Technology. 2017;1595.
2. Kumar V, Minz S. Poem Classification using Machine Learning Approach. In: Kumar Kundu M, Mohapatra DP, Konar A, Chakraborty A, editors. Advanced Computing, Networking and Informatics-Volume 1. Cham: Springer International Publishing; 2014. pp. 57–66.
3. Fang X, Zhan J. Sentiment analysis using product review data. Journal of Big Data. Springer International Publishing; 2015;2: 5.
4. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning [Internet]. Elements. 2009.
5. Vapnik V. The Nature of Statistical Learning Theory [Internet]. New York, NY: Springer New York; 2000.
6. Duda RO, Hart PE (Peter E, Stork DG. Pattern classification. Wiley; 2001.
7. Kohavi R, John GH. Wrappers for feature subset selection. Artificial Intelligence. 1997;97: 273–324.
8. Mitchell TM. Machine Learning. Burr Ridge: McGraw Hill; 1997.
9. Stathakis D. How many hidden layers and nodes? International Journal of Remote Sensing. 2009;30: 2133–2147.
10. Hochreiter S. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. World Scientific Publishing Company ; 1998;6: 107–116.
11. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. Aistats. 2010;9: 249–256.
12. Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998;86: 2278–2324.
13. Zuo Z, Shuai B, Wang G, Liu X, Wang X, Wang B, et al. Convolutional Recurrent Neural Networks: Learning Spatial Dependencies for Image Representation. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. 2015; 18–26.
14. Lipton ZC, Berkowitz J, Elkan C. A Critical Review of Recurrent Neural Networks for Sequence Learning. 2015; 1–38.
15. Bengio Y, Hochreiter S, Frasconi P. Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies. A Field Guide to Dynamical Recurrent Networks. 2001.
16. Hochreiter S, Schmidhuber JJ. Long short-term memory. Neural Computation. 1997;9: 1–32.
17. Mohamed AR, Seide F, Yu D, Droppo J, Stoicke A, Zweig G, et al. Deep bi-directional recurrent networks over spectral windows. 2015 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2015 - Proceedings. 2016; 78–83.
18. Kim S-B, Rim H-C, Yook D, Lim H-S. Effective Methods for Improving Naive Bayes Text Classifiers. Springer Berlin Heidelberg; 2002. pp. 414–423.