

BAAT: Banking Chatbot In English Language For Customers Using Machine Learning

Vishu Madaan¹, Prateek Agrawal^{1,2}, Rahul Dandotiya¹

¹Lovely Professional University, Punjab, India

²University of Klagenfurt, Klagenfurt, Austria

Abstract

Customer helpdesk is the prime focused area of any organization. This paper proposed a system BAAT for listening customer complaints and providing solution to those problems. In this paper, various machine learning algorithms have been discussed. BAAT is implemented using classifiers as Adaboost classifier and Decision Tree. System is tested by self and from bank experts. Satisfactory results are obtained from system. Proposed system can be used as a good example based on NLP and machine Learning (ML) to learn. This type of system can be designed for various applications using different datasets.

Keywords: Chatbot, Machine Learning Algorithms, Adaboost, Classifiers, Decision Tree

1. Introduction

Everybody is seeking for quick solutions. No longer waiting exists now a day. A chatbot on web application is one of the solutions for the problems faced by customers. It is based on the technology which is capable of talking with humans by some form language or natural language like English. People will not be denying talk to someone who is supportive, amusing and fascinating speakers even though they are robots? These are the machines which provide solutions to human problems. Chatbots are the one which allows human to talk with them and they will try the answer that anybody going to ask them if they are trained for it or if the answer in his scope of learning. Chatbots obtained high accuracy to provide accurate answer of even complex questions with the help of machine learning or deep learning. Chatbots are computer programs or softwares which can talk with the humans and can listen to their problems [1]. Chatbots are of two types: Chatbots for general conversation e.g. Microsoft Tay bot and chatbot which are specifically designed like Goal Oriented e.g. Cortana, Alexa and Siri. The specifically designed chatbots are designed with a purpose to fulfil. These are developed in order to solve the problems of the people by using natural language while general chatbots are designed for conversation with people on several topics. The objective of this proposed work is to develop a chatbot which can be useful for government in reduction of money and effort which government invest for solving small queries of people of its country by integrating chatbots to its online platform like banking website, railways site, healthcare website etc. The paper explains the development of a chatbot which is able to talk with people and able to understand what they want to say to it. Another objective of this proposed work is developing a chatbot which can be used in government sector and that will help government in governance because better communication with the people lead towards the better governance. Developing a web application which is used for conversation with the user. The web application should we

interactive and the same time it should not be too complex for user to understand how to chat. Developing a chatbot which is able to answer a question or able to solve a user query which is related to banking sector. Integrating both frontend and backend on a local host in order to get the answer of user queries from the backend and pass it to frontend so that user is able to see answer of its query the reasons behind making this chatbot is to reduce the effort and money spent by government. The fundament goal of our chatbot is to provide the answer of user queries related to banking sector. We have trained our model on the FAQ's of the different-different banks. Our system deals with the answering different- different question of user through web application. In the backend of our system we have used different- different method of nlp to deal with the all the problem related to processing of natural language and different- different machine learning model are used to predict the answer of user queries.

1.1 Natural Language Processing(NLP)

NLP is the technique through which computer are able to understand human's language and through which people are able to talk with computer or that computers are called chatbots which can interact with humans. NLP is the technique through which developers are trying to perform task like sentiment analysis, topic segmentation from document, speech recognition and relationship extraction. There are lot more Natural language processing tools or techniques available to train a computer to chat with humans or we can use these techniques in order to perform some task.

1.1.1 POS Tagging

Part-of-speech (POS) tagging is a process where you read some text and assign parts of each word or token, such as noun, verb, adjective, etc. POS tagging becomes extremely important when you want to identify some entity in a given sentence. The first step is to do POS tagging and see what our text contains. Classifying words of document according to some form of English or labeling them is known as POS tagging. This is explained using an example of a string: *"I have done this impossible project"*.

Table. 1. POS example of a string

Word1	Word2	Word3	Word4
I	Done	Impossible	This
Noun	Verb	Adjective	Pronoun

1.2 Machine Learning Model

Machine learning, is a field of computer science which is evolved from learning pattern and its recognition. It is related to computational learning theory in artificial intelligence. Machine learning is the training and designing of algorithms which are capable of further learning and predictions on the data sets [2]. All the prediction which is done by the system is purely based on the machine learning model [4]. Whenever user want to chat with chatbot or want to find to answer of its query then user question directly passed to trained machine learning model and then this model predict the answer of that question. The primary machine learning model that we have used in our system is Adaboost classifier.

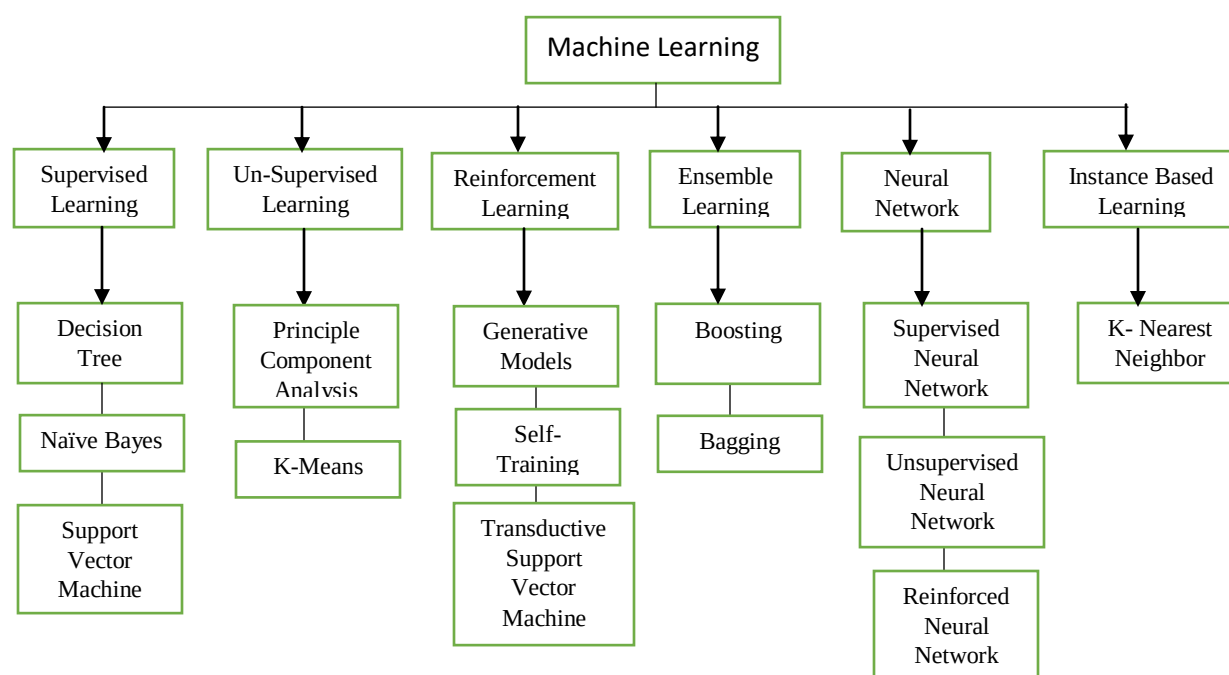


Fig.1 . Types of Learning [2]

1.1.1 Adaboost Classifier

Adaptive classifier or short we call it adaboost classifier. The original name of adaboost is adaboost.M1. But in recent time adaboost mostly used for classification problem rather than regression problem so it is now also referred as a discrete adaboost. One of the best uses of adaboost classifier is in the boosting the performance of decision tree classifier in the case of binary classification.

Adaboost classifier is an ensemble classifier like random forest classifier is one of the examples of ensemble classifier. Ensemble classifier are made-up of multiple machine learning classifier algorithm and output of the one classifier is input to another classifier and output of all classifier is the output of all those classifiers which are used in that ensemble classifier [3].

In the adaboost classifier all the weak classifier algorithm combines to form a one strong classifier. If we use single classifier algorithm, then there is possibility that may be this single classifier can classify the training object poorly. But on the other hand, if we combine multiple machine learning classifier algorithm to classify the object on the selected set of training example on every iteration and assign the right amount of weight in the classification of the training example of in the final voting of the classifier. So, through this approach we can have a good accuracy score for the classifier that we have used in our machine learning model.

Adaboost classifier algorithm retrained itself of the different-different training set and which training dataset it will select is depend on the accuracy of the previous training of the model.

In adaboost classifier the weight-age of every trained classifier algorithm is depend on the accuracy that we have achieved in training phase. After learning about the adaboost classifier we have two problem that are how we select the training dataset for each iteration and how do we assign the assign weight to each classifier after every iteration.

1.1.2 Decision Tree Classification

Decision Tree is one of the most popular and widely used machine learning algorithm. It is a supervised learning algorithm means in supervised learning we are given with the labeled dataset. Labeled data means we are given the answer of all the question of the dataset and on this data, we train our model and try to predict the output on different data. [4]

If we talk about the flow chart of decision tree is same as tree structure where every internal node of the tree represent test on the attributes of dataset, each edge or branch of tree represent the outcome of the test and each leaf node of the tree represent the class label.

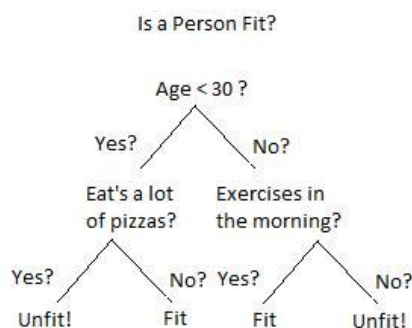


Fig.2. Representation of decision

tree Now, let's try to understand decision tree with this problem.

Let say we are given with the following data point. Now the problem is that how we can separate these data points as we can see that we are not able to separate these points. If want to classify these two classes, then we need more than two line to divide these two classes.

As you can clearly see that in order to divide these two class, we need two lines. In this way we can separate these two classes here one line separates the two classes according to threshold x_1 and other line separate the two classes according to threshold y_1 . Now, lets try to understand how decision tree try to separate these two class.

Firstly, decision tree classifier divides the area in subplots repetitively into subpart by identifying the lines. Decision tree try to divide the area repetitively because there may be possibility of distant region of same class.

2. Methodology and Implementation

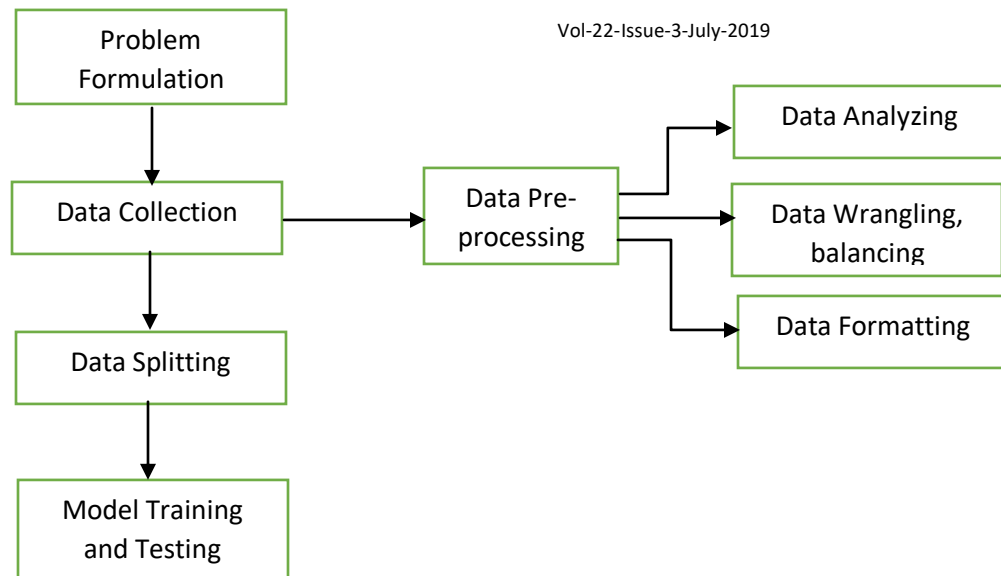


Fig.3. Flow Chart of Proposed Model

2.1 Data Collection

The accuracy rate of this proposed system is based on the considered data set for machine training. Entire successful rate is dependent on the significant database that would be required. [5] A large number of conversation logs are required for sequence to sequence models.

2.2 Data Pre-Processing

Data is analyzed, total data items are checked. Format of the data is set as required for training and testing purposes. Imbalanced data is set because imbalance dataset doesn't give better learning to machine. Missing values are identified and resolved.

2.2.1 Training dataset: In the training of adaboost classifier each weak learner classifier is trained using a random subset of training dataset or we can say that the dataset that each weak learner is trained is purely selected randomly. But it not purely 100 percent random or the random set that we are using is now actually that much random because when we are training a adaboost classifier, adaboost classifier assign a weight to each training set or week learner of that classifier. While assign a weight to the misclassified item what adaboost done is that it will assign higher weight to that misclassified item because of that all the misclassified item in this round appears with a higher probability in the next round of training subset.

2.2.2 Assignment of weight: When we train a adaboost classifier after each round of training the classifier need to assign weight to classifier based on the accuracy of the of previous training set. The Assignment of weight is done on the basis of accuracy of that classifier. The classifier with higher accuracy assigned with the higher weight so that it will have more impact on the final outcome of the model. The classifier that gives the accuracy of 50 percent will assign with 0 weight and the all the classifier with accuracy less than 50 percent is assign with negative weight.

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right)$$

(1)

$H(x)$ - is the output of adaboost classifier for input x .

$h_t(x)$ – is output of weak learner on the given input x .

α_t - is the weight assigned to each classifier.

Calculation of α_t :

$$\alpha_t = 0.5 * \ln((1-E)/E)$$

Here E -Error rate.

When the weak classifier is trained, we assign the weight, or we update the weight of each classifier.

D_t - is weight of classifier in the previous level of training.

Z_t – sum of all the weights.

In the implementation of this project we have use some of machine learning algorithm and few of techniques of natural language processing. Python is used as primary language for developing all the backend required of system. Frontend of our system is completely developed by html, CSS and JavaScript. The server which is used to connect both frontend and backend is flask. Now we will discuss all machine learning algorithms that are used in this project.

2.2.3 TfidfVectorizer

As Machine learning algorithms cannot work with raw string text directly. Rather, the text must be converted into some vectors of numbers. Scikit-learn's TfidfVectorizer aim to do the same thing, which is to convert a collection of text documents to a matrix of TF-IDF features. [6] When we try to build vector to word all the stop words are removed. Below is mathematical explanation of related term to TfidfVectorizer.

2.2.4 Term Frequency (TF)

Term Frequency is defined as the number of times a word appears in a document divided by the total number of words in the document. Every document has its own term frequency.

2.2.5 Inverse Data Frequency (IDF)

Inverse Data Frequency is defined as the log of the number of documents divided by the number of documents that contain the word w . Inverse data frequency determines the weight of rare words across all documents in the corpus. The IDF is computed once for all documents. Lastly, the TF-IDF is simply the TF multiplied by IDF.

3. Results and Discussion

Completion of a process is not complete until and unless errors are not found in a system while executing the different programs. Testing can detect the difference between the output which is present and the expected output. Basically, testing is done at the end of developing phase.

Validation: Validation is the process to examine whether the system satisfies the requirements. If the system fulfills the requirements, then it is validated. Like, this system should be able to take question of the users. It also should be able to take question from the users and able to send them to backend for the process. The users should be able to see their question on the conversation window while he/she is asking their question [7]. The admin should be able to manipulate the information regarding the question and answer and all the information regarding bot and also should be able to see all the conversation in an interface.

Proposed system is tested using unit testing to figure out if a unit of the system is working properly or not. It comes under white-box testing. It helps the developers to know if, that unit is able to fulfill the requirements or not.

If a user wants to ask a question, then he or she should be able to do so. After completing each of the modules, it is checked whether the information is correct or it is able to get the answer of his/her question. Whether a user is able to see his or her question and predicted answer from the bot, whether an admin is able to manipulate the data regarding the question and answer or not. In this part of unit testing we are checking whether the machine learning model is working properly and this function give the information regarding the accuracy of the model. This function is responsible for giving the answer of the user question as we can clearly see that this function is also working correctly, and our bot is able to give the answer of the question that we have asked. This function is responsible for running the server for backend. Through this function web application is able to fetch the answer of the question that user is going to ask. The functions used for creating frontend web page for the system. Html, Bootstrap, CSS are there for creating conversation window. As it is visible in unit testing of this function that it is working fine and input is given input to our bot. This JavaScript function is used to connect backend and with frontend application. As given the output of this function we are able to see that this function is also working fine.

4. Conclusion

This proposed work is implemented to find the solution of customer's problems specifically when they are looking for a quick solution to their problems. This system is implemented to resolve the issues of banking problems faced by customers only but this can be extended to other sectors in banks or in other fields like hospitals, industries, educational sectors wherever direct or indirect customers expect solution to their problems. Proposed system works on NLP based machine learning concepts and give good accuracy rate. Database can be extended more to have best accuracy rate.

5. References

- [1] Annina Simon¹ , Mahima Singh Deo, S. Venkatesan³ , D.R. Ramesh Babu, “ An Overview of Machine Learning and its Applications”, International Journal of Electrical Sciences & Engineering, pp. 22-24, 2015
- [2] M. Welling, “A First Encounter with Machine Learning”
- [3] Z. H. Zhou, “Ensemble Learning”, National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China.
- [4] X. Zhu, A. B. Goldberg, “Introduction to Semi – Supervised Learning”, Synthesis Lectures on Artificial Intelligence and Machine Learning, Vol. 3, No. 1, Pages 1-130, 2009
- [5] X. Zhu, “Semi-Supervised Learning Literature Survey”, Computer Sciences, University of Wisconsin-Madison, No. 1530, 2005
- [6] R. S. Sutton, “Introduction: The Challenge of Reinforcement Learning”, Machine Learning, 8, Page 225-227, Kluwer Academic Publishers, Boston, 1992
- [7] L. P. Kaelbling, M. L. Littman, A. W. Moore, “Reinforcement Learning: A Survey”, Journal of Artificial Intelligence Research, 4, Page 237-285, 1996
- [8] R. Caruana, “Multitask Learning”, Machine Learning, 28, 41-75, Kluwer Academic Publishers, 1997 [20] D. Opitz, R. Maclin, “Popular Ensemble Methods: An Empirical Study”, Journal of Artificial Intelligence Research, 11, Pages 169- 198, 1999