BANK CHAT BOT – An Intelligent Assistant System Using NLP and Machine Learning

Chaitrali S. Kulkarni, Amruta U. Bhavsar, Savita R. Pingale, Prof. Satish S. Kumbhar.

(Department of Information Technology, College of Engineering Pune, India.)

Abstract—Chat bots are intelligent systems that understand user's natural language queries and respond accordingly in a conversation, which is the focus of this study. It is more like a virtual assistant, people feel like they are talking with real person. They speak the same language we do, can answer questions. In banks, at customer care centers and enquiry desks, human is insufficient and usually takes long time to process the single request which results in wastage of time and also reduce quality of customer service. The primary goal of this chat bot is, customer can interact with mentioning their queries in plain English and the chat bot can resolve their queries with appropriate response in return. The proposed system would help replicate the customer service experience with one difference that the customer would be interacting with a bot instead of a real person and yet get the queries attended and resolved. It can extend daily life, by providing solutions to help desks, telephone answering systems, customer care centers. This paper explains the dataset that we have prepared from FAQs of banks websites, architecture and methodology used for developing such chat bot. Also this paper discusses the comparison of seven classification algorithms used for getting the class of input to chat bot.

Keywords—Chat bot, bank, classification, NLP, vectorization

I. INTRODUCTION

Banks play an important role in every country's economic development. In day-to-day life, everybody needs banks. But most of the people, especially the first-timers, struggle to know various procedures and processes required to get their work done at the bank and avail of its different products and services. Currently banks have their own web-sites, mobile applications and facilities like internet banking, mobile bank-ing but sometimes, these sources can be a bit overwhelming for most of the users who are either not well versed with technology or in some cases where the information is too scattered to search for easily. There are different types of platforms provided by different banks but people are facing problems accessing them (different GUIs, too much navigation). Although Customer Care centers are available, there are lot of wait times and redirection in some cases, leaving the customer with no choice but to experience considerable delays getting a simple informational query resolved. People have queries about various bank policies, loans, fixed deposits. This results in unnecessary crowd in banks for inquiry. Banks also face problems solving repeated queries of customers. This is time consuming and banking staff gets frustrated. Manpower and money gets wasted for separate inquiry counter.

A. Basics of Chat Bot

A chat bot is a conversational agent that interacts with users in a certain domain on certain topic with natural language

Sentences. Normally a chat bot works by a user asking a question or initiating a new topic. Chat bots can be called as software agents that simulate an entity usually a human. These are the software with artificial intelligence which allows them to understand users input and provide meaningful response using predefined knowledge base.

B. Chat Bot for Banks

Developing a chat bot will provide a smart solution to solve these queries, provide information as and when required, improve service and increase number of customers. It removes human factors included in organization and can give 24/7 hours service to increase productivity. We intend to provide a chat bot interface for customers which could be available on the web and on any hand-held devices. Customers can mention their queries in natural language and the chat bot can respond to them with correct answer. Proposed chat bot application is easily accessible to customer thereby solving redundant queries anywhere anytime. As there will be fast response for inquiry, this will be time saving for both bank and customers. The proposed system would be a stepping stone in having in place an intelligent query handling program which could in next phases not just respond but self-learn to improve itself thereby increasing not just the quality of customer service but also reducing human load, increase in productivity and of course increasing number of satisfied customers.

II. RELATED WORK

Current chat bots are developed using variety of methods like rule based where rules are hard-coded in code, AI based bots, pattern-based which can handle only mentioned patterns for retrieving answer. There are frameworks available for developing chat bots but they also use either rule-based or pattern-based techniques. In rule based chat bots which are easiest to build, one need to write rules like If X then Y else if A then B etc. So if there are 100 scenarios, developer needs to write 100 rules for each of the scenarios. The volume, variety and complexity of data makes such techniques inefficient. Its nearly impossible to write rules and/or patterns for massively available data. AI based bots are built on NLP and ML. They are based on human capability of learning information but with more efficiency. Natural Language Processing (NLP) can be used where predefined or static rules, patterns may not work.

## III. ARCHITECTURE

A. Bank Chat Bot

User will interact with the system using web application. He will enter his query in text box provided on front end of

---

this web application. Once he presses Enter button or submit the query, this request will be handled by bot controller logic.
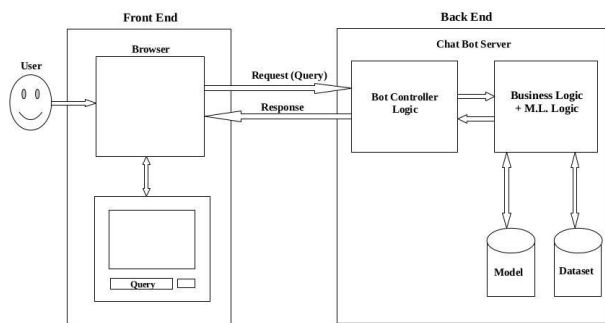
The bot controller logic contains implementation of Flask framework for handling user requests and sending answer to those queries as response. Then, the query will be sent to Business logic and Machine learning logic. Business logic contains pre-processing of user input query using Natural Language Processing (NLTK library) and its vectorization. NLP will tokenize the query, remove unnecessary spaces, stop-words and then extract lemmas for each token. Then this text-format query will be converted to vectorized format using vectorization. Now, using ML logic, classification algorithm will be applied to this transformed query to find the class it belongs to. Classification algorithm will be applied based on the previous saved model executed on train data. All questions from input data having class equal to retrieved class will be fetched and cosine similarity will be applied to these. According to similarity values we get, most similar answer will be returned to user as response.

B. Feedback System
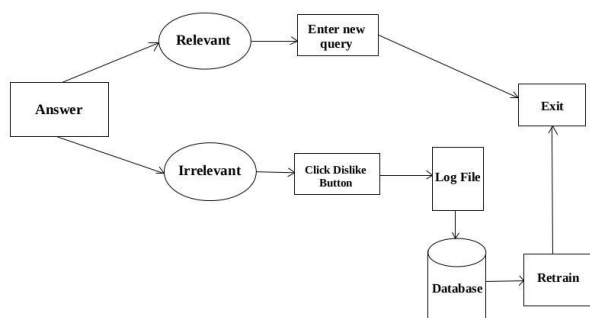
There are situations where chat bot cannot give right answer or cannot have answer to asked question because the question is out of database. For such circumstances, we have developed feedback mechanism for our chat bot. Web application will have Dislike button along with submit button. If in case user is not satisfied with the answer provided by system, he can press this Dislike button. Then log file will be generated for this query or the query will be inserted into already created log file. Developer comes into picture now for handling such situations. He will check classes for these queries, enter right answers and will retrain the classification model. So that whenever user enters the same query next time he will get correct answer. In this way chat bot will improve its accuracy and dataset.

IV. IMPLEMENTATION

1) **Preparing Data Set   :**   We have prepared our data set   as questions and answers that people usually ask to bank employees, at customer care centers or inquiry desks. We have referred different banks' websites and collected FAQs   as our   data. We have   used different web scrapping tools for this task. Distribution of questions   in Data-set :
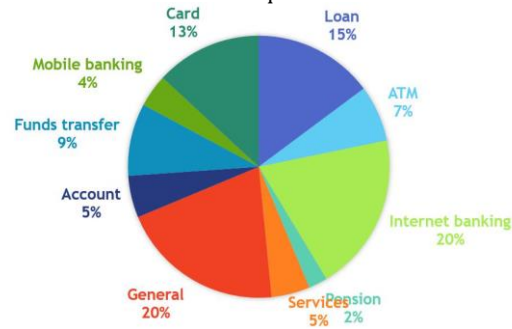
2) **Pre-processing :** We have used NLTK library for Natural Language Processing. As user input will be in English statement, to let machine understand this language we use Natural Language Processing. To decrease further processing and removing ambiguity caused due to use of same word in different forms, we have done this pre-processing. Steps included in this task are :

Removing punctuation marks and extra spaces

Tokenization - We have used tokenization to gen-erate sequence of words from user's input query.

Removing stop words - Most of the common words like 'want', 'are', 'can', which we don't need to be considered while processing are re-moved for improving the performance of system.

Lemmatization - We have used WordNet Lemma-tizer for getting lemma (root form of the word) of each token. e.g. 'processing' and 'process' should be considered equal while processing. So for getting 'process' from 'processing', lemmatization is used.

3) **Vectorization** : We have converted our text data to vectorized format using Bag Of Words(BOG) concept. BOG is a method for preparing text for input to our machine learning algorithm. BOG model develops a vocabulary from all of the documents and then model each document by counting number of times each word appearing in respective document.

4) **Classification** : As the data set increases, it takes more time to find similarity between user's query and the questions from large data set and return the answer. So

we have used classification to improve the efficiency by reducing the response time required to get the answer. We have used Scikit-learn library for implementing these classifiers. Scikit-learn is tool for data mining and machine learning in Python. As a part of literature survey and initial training we have chosen following subset of classifiers to choose the best performing one as the final classifier for the chat bot.

a) Decision Tree classifier
b) Bernoulli Naive Bayes Classifier
c) Gaussian Naive Bayes Classifier
d) K-nearest neighbor classifier
e) Multinomial Naive Bayes classifier
f) Random Forest classifier
g) Support vector machine

Also, for optimizing the algorithm's performance according to our data set, we have implemented parameter optimization. There are two approaches for implementing parameter optimization -

i.Grid search - Grid search is simply exhaustive searching approach. In grid search, it is needed to manually specify subset of hyper-parameter space of a learning model. Hyper-parameters are the parameters that are not directly learned within estimators.

ii. Randomized Search - As grid search is exhaustive and therefore expensive. So, in randomized search, it samples parameter settings a fixed number of times that are more effective. We have used Randomized Search approach in our case.

5) **Developing learning model :** In this phase, we have combined Natural Language processing, Vectorization and classification algorithms all together and save this model for further use. So, whenever the new query comes to system, we will just fetch this saved model,test this query on that model and get its class. In this way, we don't need to train model every time for each new query, thereby reducing the processing time.

6) **Testing model :** Checking for cross-validation score and precision and recall score of each classification algorithm, so that we can choose best for final use. Following is the table containing scores of each algorithm :

	Cross Validation Score	Accuracy Score	Precision Score	Recall Score
BernouliNB Classifier	0.6027	0.9252	0.9252	0.9252
GaussianNB Classifier	0.3893	0.8262	0.8262	0.8262
Multinominal Classifier	0.5966	0.9185	0.9185	0.9185
Decision Tree Classifier	0.5769	0.9845	0.9845	0.9845
Random forest Classifier	0.6187	0.9845	0.9845	0.9845
SVM Classifier	0.6524	0.9582	0.9582	0.9582
K Neighbour Classifier	0.3388	0.9845	0.9845	0.9845

7) **Choosing best approach** : According to scores of above table, 2 most accurate algorithms are - Random Forest classifier and Support Vector Machine classifier.
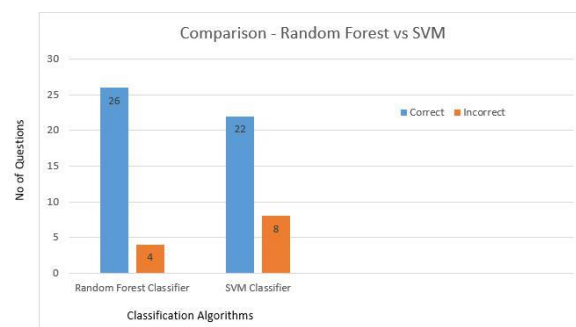

8) **Query mapping and getting answer(Using Cosine similarity) :** Once classifier gives us the class, we extract all questions that have this class from our data set. We check for cosine similarity of user's query with these extracted questions. Then answer of the most similar question is chosen as response to user's query and is returned to him.

This bot is closed domain i.e. restricted to bank only. We have set a threshold on values of cosine similarity measure for handling queries that are out of domain.

V. EXPERIMENTAL RESULTS

A. Experiment 1

We have tried different queries to validate the implementation of Bank Chat Bot. In this experiment, we have entered queries which are similar to the questions present in our data set. The analysis of the result is shown below :

SYSTEM PERFORMANCE
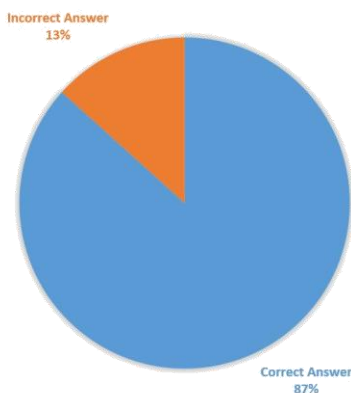
B. Experiment 2

In this experiment, we have tried same query in different forms. e.g. Query to open an account in bank can be asked in different ways like :

1) steps for opening account
2) process for opening account
3) how to open an account
4) i want to open account
5) opening account
6) procedure for opening an account in bank

Even if we enter same query in different forms, chat bot can respond same answer accurately to each form.
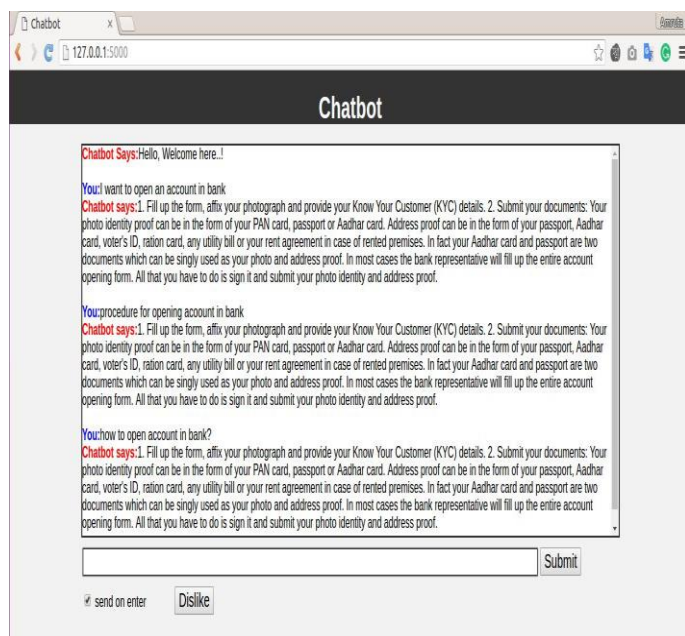
VI. CONCLUSION

The proposed system would be a stepping stone in having in place an intelligent query handling program which could in next phases not just respond but self-learn to improve itself thereby increasing not just the quality of customer service but also reducing human load, increase in productivity and of course increasing number of satisfied customers.

VII. FUTURE SCOPE

- Widening the domain
- Intelligent answers constructed by combining not just the existing list of FAQs but also from various other sources like internet, databases and other sources of data
- Providing close suggestions
- Intelligent representation of response images, links
- Combining semantic similarity along with cosine similarity
- Showing account related information using Bank's Gate-way

REFERENCES

[1] Automatic Extraction of Chatbot Training Data from Natural Dialogue Corpora, Bayan AbuShawar, Eric Atwell

[2] Chatbot Evaluation and Database Expansion via Crowdsourcing, Zhou Yu, Ziyu Xu, Alan W Black, Alexander I. Rudnicky

[3] AI BASED CHATBOT, Prof.Nikita Hatwar, Ashwini Patil, Diksha Gondane

[4] Data Mining: Concepts and Techniques   Jiawei Han and Micheline Kamber

[5] H. Kopka and P. W. Daly, Data Mining Practical Machine Learning Tools and Techniques Ian H. Witten Eibe Frank Mark A. Hall

[6] https://www.cse.iitb.ac.in/ bibek/WriteUP2016:pdfhttp $:$ $==cs224$d:stanford:edu=

[7] https://scikit-learn.org

[8] http://www.nltk.org

[9] http://www.wikipedia.org

[10] https://chatbotsmagazine.com/the-complete-beginner-s-guide-tochatbots-8280b7b906ca.i2zgql2op

[11] https://www.quora.com/What-is-the-best-way-to-learn-and-write-a-AI-Chat-bot

[12] https://chatbotslife.com/ultimate-guide-to-leveraging-nlp-machine-learning-for-you-chatbot-531ff2dd870c.5mcveo57b

[13] https://apps.worldwritable.com/tutorials/chatbot/

[14] http://machinelearningmastery.com/

[15] http://flask.pocoo.org/

[16] pandas.pydata.org

[17] https://www.pandorabots.com/

[18] http://www.wildml.com/2016/04/deep-learning-for-chatbots-part-1-introduction/

[19] https://stanfy.com/blog/advanced-natural-language-processing-tools-for-bot-makers/