Query Routing using Artificial Intelligence (AI)

Yasminbanu Shaikh
Data Analytics of Business
St. Clair College
Windsor, Canada

Venkata Hanuman Sai Kumar Kaparaju
Data Analytics of Business
St. Clair College
Windsor, Canada

Urmi Patel
Data Analytics of Business
St. Clair College
Windsor, Canada

***Abstract* – The implementation of routing the complaints or queries of customer to the respected department is very cruicial. Banking is a key domain of customer applications. In the case of customer services, banking receives many customer requests for various matters during banking. Questions such as complaints, technical issues, general inquiries, etc. As these applications continue to grow, it is difficult for customer service departments to read and sort the questions appropriately for the relevant departments. This process is time consuming and economical. This problem has resulted in delays in customer inquiries, mishandling of calls to other departments and failure to get customer feedback. In addition, users need to repeat the same thing repeatedly. All these issues lead to consumer dissatisfaction. This leads to a lack of efficiency and productivity. As a result, customers choose to leave the banking service.**

Keywords – Natural Language Processing, Machine Learning, Deep Learning, Artificial Intelligence, Customer Service, streamlit, Long Short-Term Memory (LSTM) model

1. Introduction:

Customer care is the secret to success in business. Good customer experiences generate company satisfaction, boost sales and give the word of mouth beneficial benefits. It is no surprise that customers enjoy tradition, prompt and consistent customer service interactions in this era of technological advancement. In this sector, however, businesses are not able to handle all their customer demands because they have no consistent workflow.

Banking is a significant source of customer queries and requests. As far as customer services are concerned, the bank gets a number of customer inquiries for various issues during the banking period. The number of applications submitted by the bank includes queries such as inquiries, technological problems, general enquiries, etc. If these demands steadily escalate, the customer service teams find it difficult to correctly read and segregate the inquiries from the respective departments. This method takes time and is not cost-effective. Untill now the process was manual for handling customer ueries and requests which was inconvenient and if query doesnot belong to that department and transfereed to another department it was tkime and effort consuming to explain the problem again. As a result of the situation, customers are facing issues such as delays in their orders, improper routing of calls to other departments, and an inability to access the customers' point of view. In addition, the user is expected to illustrate the same dilemma several times. Any of these issues will lead to customer frustration. As a consequence, productivity and development suffer. As a result, customers opt to stop using the banking service. And change their serving organization/bank.

Query routing is a that can help to simplify query routing, which can be quick and effective. With the help of artificial intelligence (AI), you can analyse and transfer incoming inquiries into the most appropriate pool of agents focused on departments such as credit, debt, loan, mortgage, and general enquiries. This project shows how automatic query routing can help the customer service team to respond more efficiently and quickly. In most fields such as banking, healthcare, marketing etc, the query routing scheme can be applied. Most companies benefit from this system to enhance customer service and achieve customer loyalty.

2. Related Work:

The research paper titled **"Query Processing using NLP"** by Nipun Hedaoo, Mohd Nomaan Khan, Mahesh Shetty, Ashwin Kutemate, Dr. Preeti Patil discussed about query processing using chatbots implementation in Indian Railways. Chatbot application takes input and process the query, fetch the data from the API of Indian Railways and give asked information to the customers. They basically used the Natural Language Processing (NLP) and python packages and libraries such as ChatBot, SQL Alchemy, Natural Language Tool Kit (NLTK), FLASK and Kivy platform. It is a conversational application that interacts with user/customers and accept input as text/voice and system uses its specific dataset for frequently asked questions in English and Hindi. System uses the NLTK library for preprocessing of the queries. The algorithms such as Naïve Bayes, Recurrent Neural Network, Decision Tress, Support Vector Machine used in application. Authors claimed the built system can provide precise information in less time and has potential to replace inquiry offices in Indian Railways.

The titles paper **"Automated Query Processing Based on Natural Language Processing"** by Jeetu Kumar Gupta, Sohit Agarwal discussed about Natural Language Processing (NLP) and how it used in different sector with text processing/speech recognition. In NLP Natural Language Understanding (NLU) part helps machine to understand the text and Natural Language Generation (NLG) helps it to generate the response for query. Algorithms such as Long Short-Term Memory (LSTM), Sequence to Sequence model, Named Entity Recognition Model, User Preference Graph Model, Word Embedding model used for text processing and for speech recognition Word Recognition, Acoustic Modeling, Connectionist temporal classification, phased based machine translation, Neural machine Translation, and Google neural machine translation algorithms are used. NLP has different applications like Machine Translation, Sentiment Analysis, Information Retrieval, Text Categorization, Extracting data, Spellings and Grammars. In NLP text preprocessing includes Tokenization, Stemming, Lemmatization, Stop words, Bag of words, POS Tagging. In process of Natural Language Understanding system uses raw speech signal, syntactic analysis, Semantic analysis, and pragmatic analysis. NLU's deep learning made text preprocessing easy and reliable. Sectors like business, analysis of data product, marketing, sales, and advertising uses these applications of NLP. Tough there are problems with speech processing and cognitive, emotions, incentive, and skills. According to the needs of customer, system classifies the information using NLP and personalized it.

Papiya Mahajan1, Rinku Wankhade2, Anup Jawade3, Pragati Dange4, Aishwarya Bhoge5 titled the paper **"Healthcare Chatbot using Natural Language Processing"**. The purpose of this paper is to create a healthcare chatbot using NLP. The chatbot provides the type of disease based on user symptoms and provides doctor consulted medications and food to be consumed. The chatbot takes the customer's enquiry and marks as major and minor symptoms based on trained model using n-grams, tf-idf algorithms and gives a type of disease and suggests necessary medications and intake of food.

Abbas Raza Ali titled the paper **"Intelligent Call Routing: Optimizing Contact Center Throughput"**. The purpose of this paper to create a model that can significantly increase sales, improves customer satisfaction, and reduces average call handle time. It provides optimization based on underlying Customer and CSR demographics, psychographics, and historical performance information. By mapping hundreds of features and using advanced analytics, intelligent call routing helps in boosting overall contact center throughput.

The research named as **"Chatbot as an Innovation of Machine Learning"** by Saksham Bhambri, Muskan Ahuja, Vishakha Sehdev and Ankit Verma discussed about a new approach to artificial intelligence that has the idea of communicating to multiple users in different ways depending on the user's needs. A chatbot encourages a user to ask questions in the same way as a person does, and the chatbot would respond to such questions as if an expert is present within the computer or device and is the one addressing your queries. Chatbots can also connect you directly to a valued officer in the department of which you have a question, removing the need for a middleman, which helps both the customer and the business. Chatbots operate on the basic idea of hearing the user's message, searching the keywords of that query in their database, and then generating a suitable answer for that query. In this case, the system analyzes the user request and scans it in the stored database using the keywords in the request. This is a very complex procedure and if the machine takes an error in interpreting the keywords, the answer to the request will be different and inadequate for the customer. Here they perform

The Turing Test determine a chatbot's efficiency or intellect. Also, for fundamentals properties of intelligent machine they chose Arithmetic, Comparison, Logic and reasoning, Learning, Heuristics and memory, Senses, Perception and lastly Consciousness. They categorized the different types of intelligence using performance factor, Partially Intelligent Systems, and Completely Intelligent Systems.

The researched paper titled as **"Visual-textual Capsule Routing for Text-based Video Segmentation"** by Bruce Malntosh, Kevin Duarte, Yogesh Rawat and Mubarak Shah is focused on Integration of video and text for the task of actor and action video segmentation from a sentence. They proposed a Capsuled-based approach which performs Pixel -Level Localization based on a Natural Language Query Describing The actor of interest. They Explore the use of Capsuled to jointly encode and merge visual and textual information for the task of actor and action detection in videos. Gone through the related work on Vision and Language, Merging Visual and Textual Inputs and Capsule Networks, they finally decided to use Capsules to represent entities and used High-dimensional coincidence filtering for routing which is to learn part to whole relationship between these entities. This method allows the network to learn a set of entities, the capsule routing finds the similarity between the objects in the video and sentence inputs to generate a unified visual-textual capsule representation. Using A2D dataset containing 3782 videos and J-HMDB dataset with 928 short videos with 21 different action classes they implemented the network using PyTorch. The I3D used weights pretrained on kinetics and fine-tuned on Charades. The network was trained using Adam Optimizer. They perform Single-Frame Segmentation from sentence, Full video Segmentation from sentence, Image Segmentation Conditioned on Sentences and Ablation Studies. Their network has two failure cases, one where network incorrectly selects an actor not described in query and other where network fails to segment anything in video. Finally, by using visual-textual routing, their network successfully segments actors and actions in video, conditioned on textual query.

3. Problem Statement & Objective:

Problem Statement: Pleasant consumer experiences lead to company satisfaction, increase revenues and provide building jaws. Banking is one of the central areas of user demands. The customer service teams have difficulty interpreting and separating queries from the agencies in question with these demands growing slowly. It takes time and is not affordable. customers then prefer to abandon the business.

Well built communication with customers to answer concerns and complaints of the client. We build a question routing mechanism using AI to bridge the distance between customers and the company.

Analytical Objectives: Creating a Query routing scheme based on a Long Short-Term Memory (LSTM) neural network model. Exploratory data analysis (EDA) is being used to view the most often used terms and the number of questions in relation to their departments.Using Machine Learning techniques such as Natural Language Processing (NLP) and Deep Learning techniques such as Recurrent Neural Network (LSTM), which aid in text processing and categorizing customer requests and allocating them to the appropriate departments such as Credit, Debt, Loan, Mortgage, and General Queries.

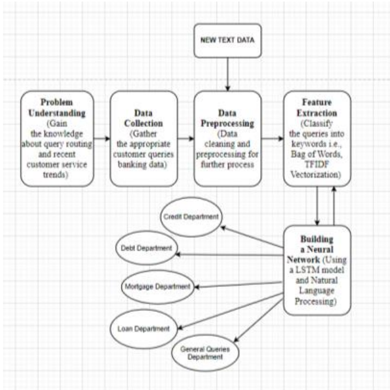
4. Proposed Solution/Method:

To build the Query routing model the necessary tools and effective methods are:

1. NLP Pipeline
2. Data Pre-Processing
3. Exploratory Data Analysis
4. Text Pre-Processing
5. Text Representation
6. Feature Selection and Extraction
7. A Supervised Classification Model
8. Deep Learning Model
9. Model Deployment

NLP Pipeline:

Problem Understanding (Gain the knowledge about query routing and recent customer service trends) → Data Collection (Gather the appropriate customer queries banking data) → Data Preprocessing (Data cleaning and preprocessing for further process) → Feature Extraction (Classify the queries into keywords i.e., Bag of Words, TFIDF) → Building a Neural Network (Using a LSTM model and Natural Language Processing) → Determining the accuracy of the Model → Finalize the result and route the queries to respective departments.

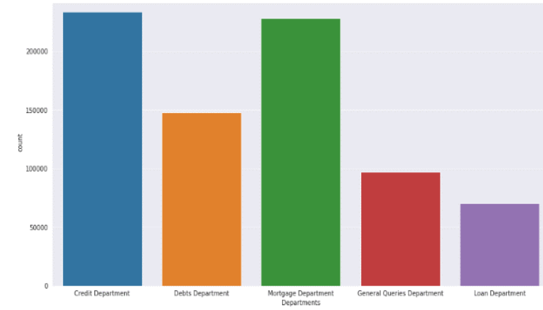


Data Pre-Processing:

To Pre-process our data, we are using Pandas Library. We used Banking Department Queries dataset and it contains 18 columns namely Date Received, Sub-product, Issue, Sub-issue, Consumer complaint narrative, Company public response, Company, State, ZIP code, Tags, Consumer consent provided, submitted via, Date sent to company, Company response to consumer, Timely response, Consumer disputed, Complaint ID, Product out of which we are using two columns Issue and Product. We eliminated other columns because it's irrelevant to our problem statement and have null values as well. We also replaced and re-named the product column to Departments as per our context. The departments are Credit, Debt, Loan, Mortgage, General Queries.

Exploratory Data Analysis:

The Seaborn and Matplotlib libraries were used to visualize our exploratory data analysis. Firstly, we displayed the count of queries with respect to their departments to check the data imbalance. From the above graph we can say that Credit and Mortgage departments have high number queries followed by Debt and General queries departments have reasonable amount of queries. Whereas, the Loan Department has the least number of queries. To overcome this problem, we used over-sampling methods namely SMOTE (Synthetic Minority Oversampling Technique).



Text Pre-Processing:

The NLTK and text_preprocessing libraries were used for text pre-processing. The methods used in text pre-processing are converting text to lowercase, expanding contractions (short form of words), removing stop words, and removing punctuation in the text.

Text Representation:

The WordCloud library is used for visualizing the most frequent words. A word cloud is built to represent the high frequent words with respect their departments.

Feature Selection and Extraction:

We are using feature selection to bring out the words which are most corelated with each of the departments. Chi2 feature selection method is used to extract the words based on n-grams and tf-idf vectorization. We are using term frequency of words to filter-out unigrams and bi-grams. Since most classifiers and learning algorithms expect numerical feature vectors with a fixed size rather than raw text documents with variable length, they cannot directly process the text documents in their original form. Thus, the texts are translated to a more manageable representation during the preprocessing phase.

```
# 'Credit Issues':
  . Most correlated unigrams:
. features
. false
. expect
. existing
. exchange
. excessive
. escrow
. end
. embezzlement
. disputes
  . Most correlated bigrams:
. false statements
. existing problem
. disclosures info
. existing mortgage
. exchange rate
. excessive fees
. escrow account
. end loan
. existing issue
. wrong day
```

The bag of words model is a popular method for extracting features from text, a model in which the existence (and sometimes the frequency) of words is considered for each message, in our case a complaint narrative, but the order in which they appear is avoided. For each word in our dataset, we can compute a metric known as Term Frequency, Inverse Document Frequency, abbreviated to tf-idf. We will compute a tf-idf vector for each of the user complaint narratives using sklearn.feature extraction.text.TfidfVectorizer.

Supervised Machine Learning Model:

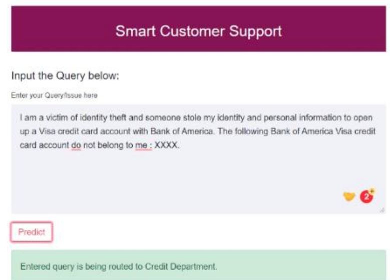
We are using both Multinomial naïve bayes classification and SVM (Support Vector Machine) classification algorithms to build our classification model. We began by converting the "Issues" into a vector of numbers to train supervised classifiers. We looked at vector representations like TF-IDF weighted vectors. After having the vector representations of the text, we can train the supervised classifiers to predict the "Department" from which they fall.

Deep Learning Model:

We are using TensorFlow deep learning framework to build our LSTM neural network. To vectorize the features we are using tensor flow text vectorization to convert text to numbers. We are using n-grams and tf-idf concept in text vectorization to generate vocabulary based on term frequency. The issues are first tokenized to words and these words are converted to sequences using text vectorization method. The converted sequences may have irrelevant shape in which we use sequence padding to balance the sequence by adding zeros after the text known as post sequence padding. We are imputing the converted features to Bi-directional LSTM neural network to train the 128 layered networks. We are compiling bi-directional neural network with adam optimizer and categorical cross entropy loss to train the model.

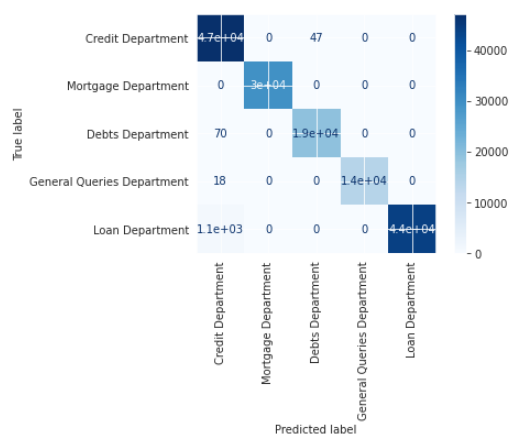
Model Deployment:

We are deploying our trained model on Streamlit application to process the real-time queries and route it to corresponding department.



5. Results:

Using the supervised classification model, it was able the predict the given query to it is appropriate department with 82% of accuracy and we built a confusion matrix to evaluate the predicted query being to its respected department. To improve the accuracy, we built a bi-directional LSTM neural network and achieved an accuracy of 89%. The model was able to predict with 62% of probability that the query being routed to its respective department.



6. Discussion:

When we decided to embark on a project to create a query routing system, our primary goal was to assist people in receiving reliable, timely, and appropriate responses to their queries or issues. One such interface will allow users to conveniently write their requests or queries and guide them to the respective bank's department for further procedure. To achieve the goal, we start working on the project. After we developed this routing system, we found that the queries are effectively move to it's respective department for next process.

While working with all new topic of natural language processing where we dealt with text data instead of numeric data, we faced number of challenges. Basically, to find a dataset that fulfill the needs of model and can feed to the model, because the dataset that we found was having to many null values or useless columns for the model and dataset. Also Lack of Keywords to route their department in training the model is major issue which leads improper routing. To increase the probability for predicting the query we need similar words to train the model. We have intended to write the entire code for the user interface framework in Python, which would be a daunting challenge. However, after extensive testing, we were able to complete the mission. As a result, the management team was successful in achieving the project goals.

7. Conclusion:

In conclusion, using the web-based application, the process of routing the query to the responsible department or person becomes fast, easy and fully automated. Apart from this will help the organizationto manage the queues and customer querires effciently. After building the query routing model we can improve the customer support to gain more profits and attract customers to improve the satisfaction with the companies as well.
From our analysis we observed, that we need more queries to train the model to work efficently.

8. Contributions:

We equally contributed to each task starting from Data collection (Gather the appropriate customer queries banking data), Data Preprocessing (Data cleaning and preprocessing for further process), Feature Extraction (Classify the queries into keywords i.e., Bag of Words, TFIDF), Building a Neural Network (Using a LSTM model and Natural Language Processing), Determining the accuracy of the Model. And we also developed a web interface using Streamlit framework. Except the technical part, we have done the whole documentation process from beginning to end including final report.

9. Refrences:

[1]: Introduction ticket routing with ai. (2019, October 16). Retrieved February 03, 2021, from https://monkeylearn.com/blog/ticket-routing/
[2]: Natural language toolkit¶. (n.d.). Retrieved February 03, 2021, from https://www.nltk.org/.
[3]: Word2Vec: TensorFlow Core. (n.d.). Retrieved February 03, 2021, from https://www.tensorflow.org/tutorials/text/word2vec
[4]: Recurrent neural networks (RNN) with Keras: TensorFlow Core. (n.d.). Retrieved February 03, 2021, from https://www.tensorflow.org/guide/keras/rnn.
[5]: *tf.keras.layers.experimental.preprocessing.TextVectorization*. TensorFlow. (n.d.). https://www.tensorflow.org/api_docs/python/tf/keras/layers/experimental/preprocessing/TextVectorization.
[6]: *Gensim: topic modelling for humans*. Word2Vec Model - gensim. (2021, April 16). https://radimrehurek.com/gensim/auto_examples/tutorials/run_word2vec.html.
[7]: *wordcloud*. PyPI. (n.d.). https://pypi.org/project/wordcloud/.
[8]: *sklearn.svm.SVC*¶. scikit. (n.d.). https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.
[9]: *1.9. Naive Bayes*¶. scikit. (n.d.). https://scikit-learn.org/stable/modules/naive_bayes.html
[10]: *sklearn.metrics.plot_confusion_matrix*¶. scikit. (n.d.). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.plot_confusion_matrix.html.
[11]: *tf.keras.preprocessing.sequence.pad_sequences : TensorFlow Core v2.4.1*. TensorFlow. (n.d.). https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/sequence/pad_sequences.
[12]: *tf.keras.preprocessing.text.Tokenizer : TensorFlow Core v2.4.1*. TensorFlow. (n.d.). https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/text/Tokenizer.
[13]: *tf.keras.layers.Bidirectional : TensorFlow Core v2.4.1*. TensorFlow. (n.d.). https://www.tensorflow.org/api_docs/python/tf/keras/layers/Bidirectional.
[14]: *Word2Vec : TensorFlow Core*. TensorFlow. (n.d.).

https://www.tensorflow.org/tutorials/text/word2vec?hl=en.
[15]: *sklearn.feature_extraction.text.TfidfVectorizer¶*. scikit. (n.d.). https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html.
[16]: Brownlee, J. (2021, March 16). *SMOTE for Imbalanced Classification with Python*. Machine Learning Mastery. https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/.
[17]: *6.2. Feature extraction¶*. scikit. (n.d.). https://scikit-learn.org/stable/modules/feature_extraction.html?highlight=nlp.
[18]: *sklearn.feature_selection.chi2¶*. scikit. (n.d.). https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.chi2.html?highlight=feature+selection#sklearn.feature_selection.chi2.
[19]: Turkel, W. J., & Crymble, A. (2012, July 17). *Keywords in Context (Using n-grams) with Python*. Programming Historian. https://programminghistorian.org/en/lessons/keywords-in-context-using-n-grams.
[20]: *Streamlit • The fastest way to build and share data apps*. https://streamlit.io/. (n.d.). https://streamlit.io/.