Research in AI-based chatbot implementation in ECTs Low code application.

Master's Internship proposal

Sandeep Kasinampalli Rajkumar
Applied Mathematics for Network and Data Sciences
Hochschule Mittweida

Internship Supervisor
Reny M John
Product owner at ECT
European Computer Telecoms AG (ECT)
Westendstrasse 160, 80339 Munchen

Education qualification of Supervisor:
Bachelor of Engineering in Computer Science
Master of Business Administration.

17 July 2023

1 Background

Chatbots have become increasingly popular in recent years for providing personalized customer service, automating repetitive tasks, and handling a wide range of user queries. They can be integrated into a variety of applications, such as e-commerce, healthcare, and finance, to improve user experience and increase efficiency. With the recent advancements in natural language processing techniques, chatbots have become more advanced and sophisticated, able to understand and respond to a wide range of user queries. The goal of this internship is to develop a chatbot using advanced bot-building tools, semantic search, and various natural language processing services, in order to provide accurate and personalized responses to user queries.

2 Objectives

The main objective of this internship is to develop a chatbot that can:

- Understand user intent and extract entities from user queries.
- Provide accurate and personalized responses to a wide range of user queries.
- Integrate with advanced bot-building tools, semantic search, and various natural language processing services.
- Be accessible and user-friendly, with the ability to understand and respond to voice commands.

3 Chatbot Components.

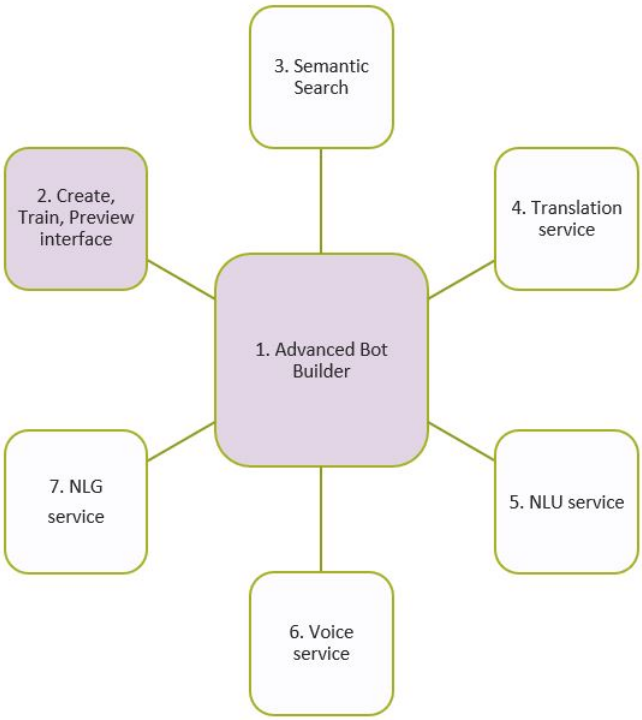

Figure 1: Chatbot Components[1].

4 Methods

4.1 Advanced Bot Builder

The advanced bot builder will be the backbone of the chatbot, taking care of dialog management and providing the base for adding intents, entities, and slot filling. It will also handle the storage and management of the chatbot's knowledge base, allowing for easy updates and modifications. The advanced bot builder will also provide the ability to test and preview the chatbot's performance, making it easier to identify and correct any errors or issues.

4.2 Create, Train, and Preview the Interface

The chatbot will be trained on a set of sample user utterances. One sample example: "Book me a flight to Rio next week", "Fly me to Rio on the 24th", and "I need a plane ticket next Sunday to Rio de Janeiro." The entities "From_location", "To_Location", and "Date of Journey" will be annotated in the training data. The train model view and test model view will be used to evaluate the chatbot's performance and make any necessary adjustments. The training data will be sourced from a variety of sources, such as customer service transcripts, social media posts, and online forums, in order to ensure that the chatbot can handle a wide range of user queries.

4.3 Semantic Search

The chatbot will use semantic search techniques to understand the meaning of user queries and provide relevant responses. It will be able to understand the intent behind a user query, even if it is phrased in a different way. For example, if a user asks "What is the weather like in New York City?", the chatbot will be able to understand that the user is asking about the current weather in New York City and provide an appropriate response.

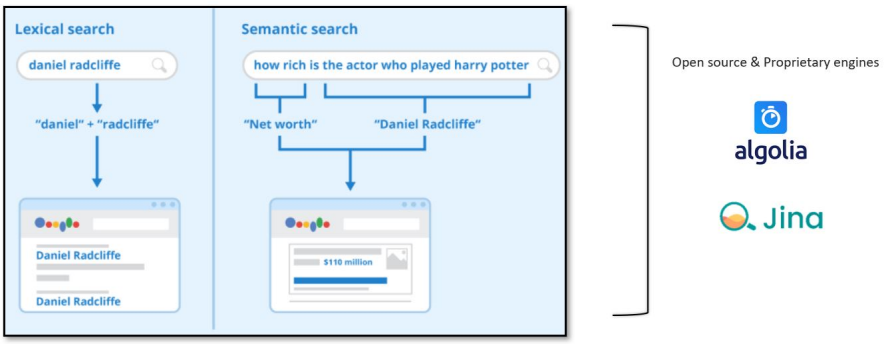

Figure 2: Semantic Search[1].

4.4 Translation Service

The chatbot will be able to understand and respond to user queries in multiple languages using machine translation.

4.5 NLU Service

The chatbot will use natural language understanding to extract entities and intent from user queries, allowing it to provide more accurate and personalized responses.

4.6 Voice Service

The chatbot will be able to understand and respond to voice commands, making it more accessible and user-friendly.

4.7 NLG Service

The chatbot will use natural language generation to provide responses in a human-like manner, making the conversation more natural and engaging.

5 High-Level Architecture.

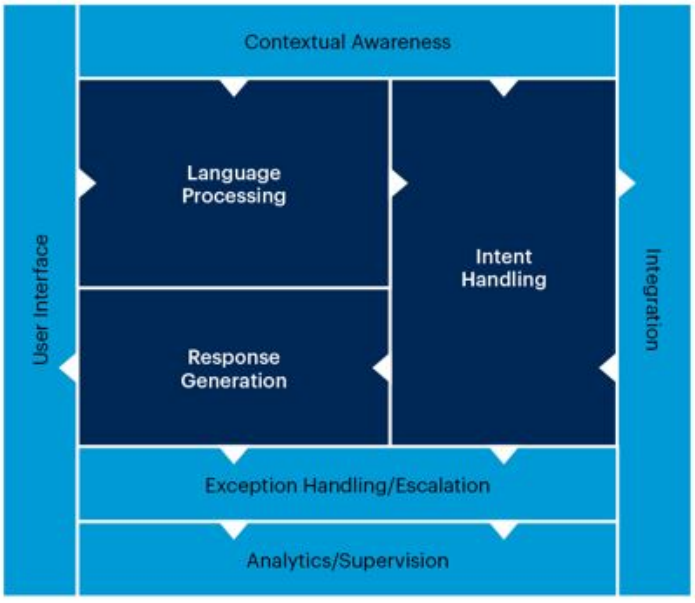

Figure 3: High-level-Architecture[1].

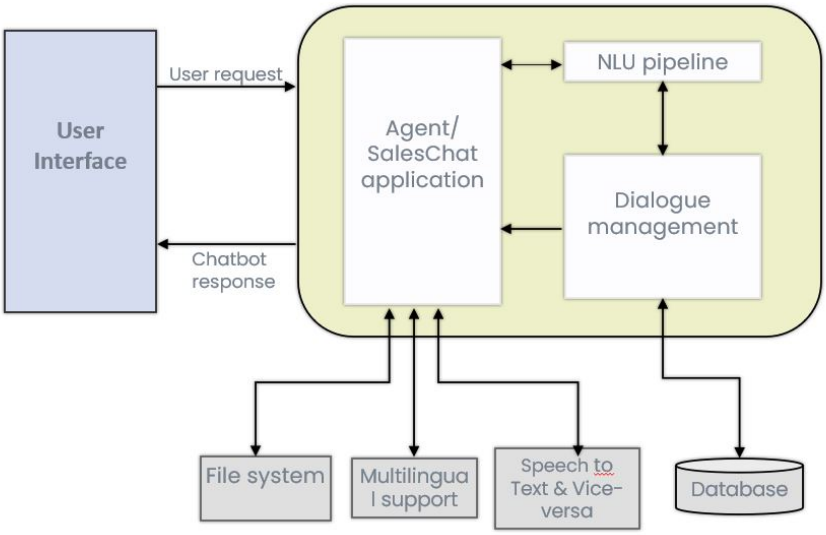

Figure 4: Architecture Diagram.

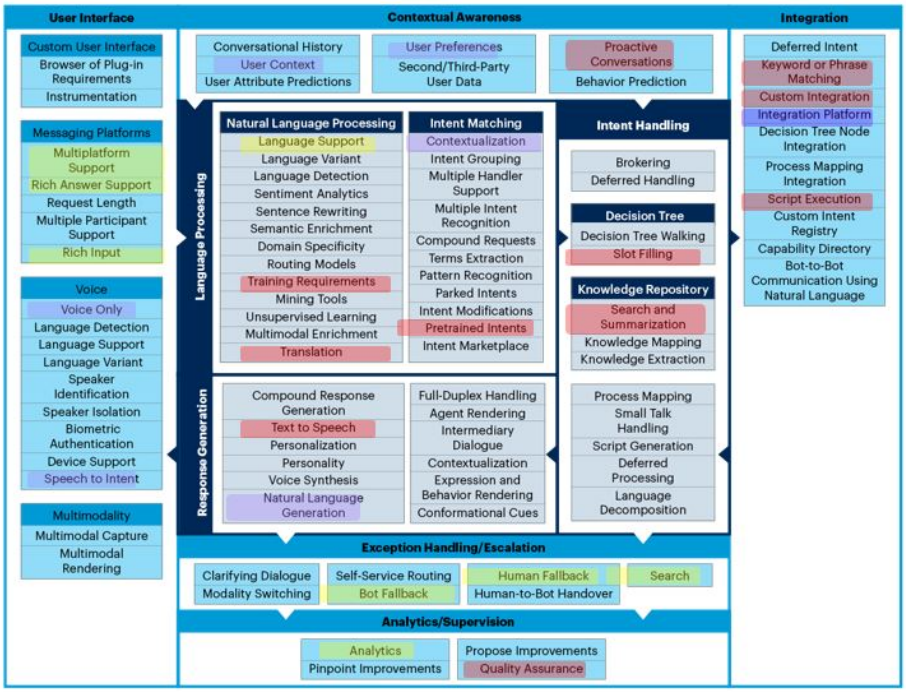

Figure 5: Detailed Architecture[1].

6 Problem Statement

The current chatbot technology provides personalized customer service, automates repetitive tasks, and handles a wide range of user queries. However, it lacks the ability to perform a semantic search on uploaded documents such as CSV and PDF files, which limits the scope of the chatbot's functionality. Therefore, the problem statement for this internship is to develop a chatbot with advanced features that can perform a semantic search on uploaded documents and provide accurate and personalized responses to user queries based on the information extracted from those documents. The chatbot should be able to understand the user's intent, extract entities from user queries, and provide relevant responses from the uploaded documents, making it a valuable tool for businesses and organizations that deal with large amounts of data.

The project requirements are:

- **Requirement 1:** Develop an AI chatbot where the owner uploads the FAQ CSV file with questions and answers to the portal, this will create the model, and the new chatbot is created.
- **Requirement 2:** Similar to the above the Owner uploads the documents which are in text document or pdf format.

7 Comparision among models

First, we see why we are using Transformers in NLP.

- **Convolutional neural network:**
  1. Good for image datasets.
  2. Haven't been as widely successful for language data.
  3. It does not store the previous step data.
- **Recurrent neural network:**
  1. Best for sequential data.
  2. Good with some of the NL data.
  3. But we need to move in order of the sequence, so there is a limit to how much we can parallelize training.

  RNN and CNN can process short sequences but are not efficient in processing large sequences. We can conclude that Transformers are the best model for the large sequence to process in NLP.

- **Transformer:**
  1. There is no limit to parallelized training.
  2. Sentences are processed as a whole rather than word-to-word.
  3. There is a newly introduces unit used to compute similarity scores between words in sentences.

Attention is all you need.[2]

8 Architecture of Transformer:

In this section, we look at a single block of the transformer. As in GPT-2, there are a large number of transformer blocks present. But for our better understanding, we will see the architecture of a single block of the transformer.

- Z - Embedding vector for i = 1 to n
- X - Inputs representation sequence for i = 1 to n
- Y - Output sequence for i = 1 to n

Transformer consists of 3 main parts:

- **Encoder:** Encoder are identical in structure. It consists of Multi-Self Attention Layer and Feed-Forward Network. The encoder's inputs first flow through a self-attention layer— a layer that helps the encoder look at other words in the input sentence as it encodes a specific word. The outputs of the self-attention layer are fed to a feed-forward neural network. The exact same feed-forward network is independently applied to each position.
- **Decoder:** Decoders are also identical in structure. It consists of Self Attention Layer, Encoder-Decoder Attention Layer, and Feed Forward Layer. These are to be used by each decoder in its "encoder-decoder attention" layer which helps the decoder focus on appropriate places in the input sequence.
- **Embedding:** Embedding is a numerical representation of words, usually in the shape of a vector. This vector will be all zeroes except one unique index for each word.

8.1 Modules in the transformer

Next, I will discuss the elemental components that comprise the original transformer architecture.

- Attention modules
- Position-wise feed-forward networks
- Residual Connection and Normalization
- Positional encoding

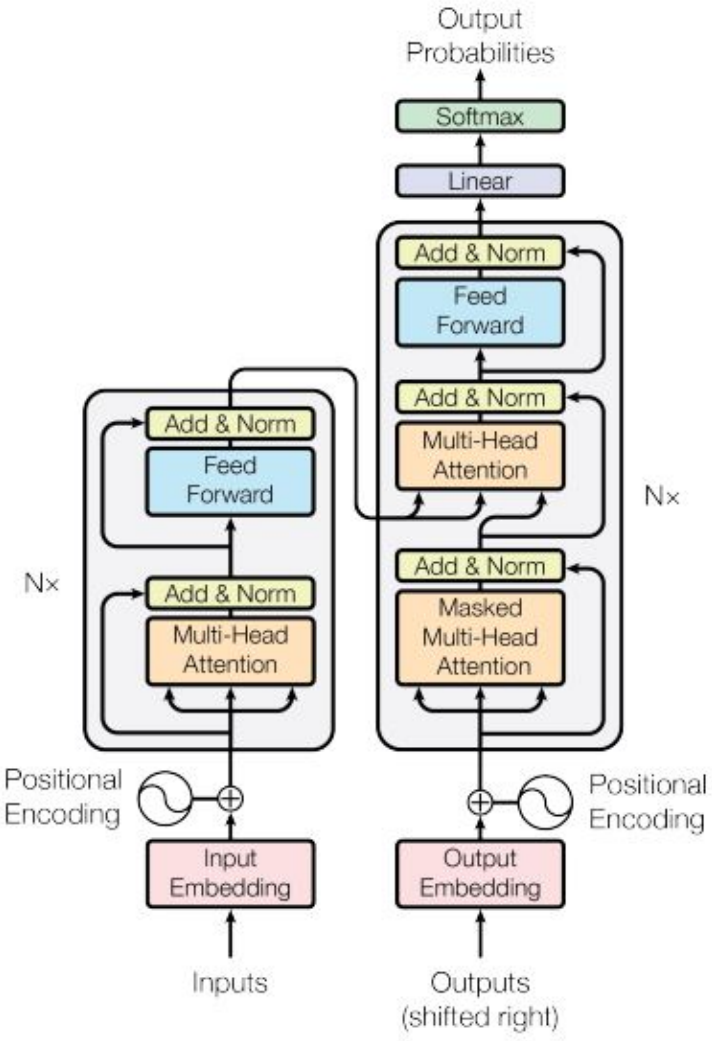

Figure 6: The Transformer - model architecture[2].

8.2 Attention modules

The transformer integrates Query-Key-Value (QKV) concept from information retrieval with attention mechanisms.

- **Scaled dot-product attention:**
  Here we are taking a single attention function.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{D_k}}) * V = AV = Z \qquad (1)$$

where,
Q = Queries
K = Keys
V = Values
D_K = Dimension of key vector.
A is called the attention matrix.

- **Multi-head attention:**
  Similar to above in Multi-head attention we are considering multiple simultaneous layers.

$$MultiHeadAttn(Q, K, V) = Concat(head_1, .., head_h)W^0 \tag{2}$$

$$head_i = Attention(QW_i{}^Q, KW_i{}^K, VW_i{}^V) \tag{3}$$

where,
$QW_i{}^Q$ = Q value is sent to the Neutral network with the weight $W_i{}^Q$
$KW_i{}^K$ = K value is sent to the Neutral network with the weight $W_i{}^K$
$VW_i{}^V$ = V value is sent to the Neutral network with the weight $W_i{}^V$

8.3 Feed-forward neural network (FFNN)

The feed-forward layer is the weight that is trained during training and the exact same matrix is applied to each respective token position.

Since it is applied without any communication with or inference by other token positions it is a highly parallelizable part of the model. The role and purpose are to process the output from one attention layer in a way to better fits the input for the next attention layer.

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{4}$$

8.4 Residual Connection and Normalization

Residual connections are a way to address the vanishing gradient problem that can occur in deep neural networks.

Normalization techniques are used to improve the stability and performance of neural networks. In transformers, the most commonly used normalization technique is layer normalization. Layer normalization normalizes the inputs to a layer across the feature dimension, which helps to reduce the effect of input variations on the outputs of the layer. This can improve the stability of the network and make it easier to learn.

In combination, residual connections and normalization techniques can greatly improve the performance of neural networks used in NLP tasks such as language

modeling and machine translation. For example, the original transformer architecture used both residual connections and layer normalization to achieve state-of-the-art performance on a range of language modeling tasks.

8.5 Positional encoding

Positional embeddings are used to encode the relative positions of the words in a sequence.

A sequence of words is usually fed into the transformer model as a matrix, where each row corresponds to a word in the sequence and each column corresponds to a feature or embedding dimension. However, the transformer model does not have any inherent notion of word position, which is important for understanding natural language.

To solve this problem, positional embeddings are added to the input sequence of words before it is fed into the transformer model. The positional embeddings are learned during training and represent the position of each word in the sequence relative to the other words.

9 Goal and Objectives

This internship aims to utilize Natural language processing (NLP) to develop an advanced chatbot.

Following would be the tasks taken towards the completion of this project:

- **Data Pre-processing:** Collect the data from all the domains in question-and-answer format and save it in the Excel file, this will be used for training and testing, and later we are loading the large volume of text/vocabulary by using Spacy.

  SpaCy is a free, open-source library for advanced Natural language processing(NLP) in Python.
  SpaCy is designed specifically for production use and helps you build applications that process and "understand" a large volume of text. It can be used to build information extraction or natural language processing systems or to pre-process text for deep learning.

  GloVe (Global Vectors for Word Representation) is a technique for word embedding, which is a process of representing words as vectors of real numbers. GloVe is a type of unsupervised learning algorithm that learns vector representations for words by analyzing the co-occurrence statistics of words in a large corpus of text. The resulting vector representations capture semantic and syntactic information about words and can be used as input for various NLP tasks such as text classification, sentiment analysis, and machine translation.

- **Data Modeling:** Implementing the **Transformer**: State-of-the-art Natural Language Processing.
  A Transformer is a deep learning model that adopts the mechanism of attention, differentially weighing the significance of each part of the input data. It is used primarily in natural language processing (NLP) and computer vision (CV).

  BERT stands for Bidirectional Encoder Representations from Transformers. BERT's architecture is based on the Transformer model, which was introduced in the paper "Attention Is All You Need" by Vaswani et al. (2017). The Transformer model utilizes a self-attention mechanism to process input sequences, which allows it to learn dependencies between different parts of the sequence.

  BERT extends the Transformer model by training it on large amounts of text data in an unsupervised manner. The pre-training involves two tasks: masked language modeling and next-sentence prediction. In masked language modeling, BERT randomly masks some of the input tokens and trains the model to predict the masked tokens based on the surrounding context. In the next sentence prediction, BERT is trained to predict whether two input sentences are consecutive in the original text or not.

- **Testing:** Calculate the chatbot quality.

10 Project Milestones

The internship is intended to initiate from 1-August-2023, for the duration of 4 months (17 weeks) as per the required guidelines. The estimated time of completion of the tasks is as follows

Requirements	Estimated Completion Weeks	Tasks
Requirement 1: FAQ support.	Week 1 Week 2-3 Week 4-8 Week 9-10 Week 11-12	Understanding the project requirements Data Pre-processing. Data Modeling Integrate this to LCAP application. Deployment and testing.
Requirement 2: Support for word and pdf doc.	Week 13-14 Week 15-16	Data Modeling. Chatbot performance, deployment and Final testing
For both requirements	Week 17	Multilingual Support and Speech to text and Vice versa

11 Expected Outcomes

By the end of the internship, we expect to have developed a functional chatbot that can understand and respond to a wide range of user queries in multiple languages, using advanced bot-building tools, semantic search, and natural language processing services. The chatbot will be able to understand user intent, extract entities, and provide accurate and personalized responses, making it a valuable tool for Users.
Simultaneously we will focus on the openAI LLM, as there is a huge development in these technologies like prompt engineering and generative AI.

12 Conclusion:

In this internship proposal, I have outlined the development of a chatbot that utilizes advanced machine learning techniques, including advanced bot building, semantic search, translation service, natural language understanding (NLU) service, voice service, and natural language generation (NLG) service. Additionally, I have discussed the mathematical concepts that will be employed in the development of this chatbot. I am confident that this internship will provide me with valuable experience and knowledge in the field of machine learning and chatbot development.

References

[1] All the information and the diagrams are taken from the company[ECT - European Computer Telecoms AG] provided documentation.
https://www.ect-telecoms.com/

[2] Attention Is All You Need
https://arxiv.org/pdf/1706.03762.pdf