Diploma Programme
Programme du diplôme
Programa del Diploma

Computer science
Case study: The perfect chatbot

For use in May and November 2025

Instructions to candidates

- Case study booklet required for higher level paper 3.

5525 – 6401

8 pages

International Baccalaureate®
Baccalauréat International
Bachillerato Internacional

Scenario

An insurance company, *RAKT*, recently implemented a language model chatbot to handle customer queries. However, the chatbot has been performing poorly, with many customers reporting dissatisfaction with its responses.

The company has hired you as a student intern to investigate the issues and recommend improvements for the chatbot. Your supervisor has already analysed the customer complaints and identified the root cause of the issues. She wants you to explore six key areas that might improve the chatbot's performance.

Problems to be addressed

1. **Latency**: The chatbot's response time is slow and detracts from the customer experience.
2. **Linguistic nuances**: The chatbot's language model is struggling to respond appropriately to ambiguous statements.
3. **Architecture**: The chatbot's architecture is too simplistic and unable to handle complex language.
4. ***Dataset***: The chatbot's training dataset is not diverse enough, leading to poor accuracy in understanding and responding to customer queries.
5. **Processing power**: The system's computational capability is a limiting factor.
6. **Ethical challenges**: The chatbot does not always give appropriate advice and is prone to revealing personal information from its training dataset.

Latency

RAKT's customers expect the chatbot to provide accurate and timely responses to their queries. A complex natural language processing model was developed to handle the nuances of human-to-human interaction. However, this has increased the time the chatbot takes to respond, especially when the volume of queries is high.

It takes a vast network of machine learning models for conversational artificial intelligence (AI) to determine what to say next, with each model solving a small piece of the puzzle. One model might take the user's location into account, while another could analyze the history of user interactions, including the feedback provided by users on similar responses. Every model should add improvement but at the cost of increasing the system's latency.

That latency is a product of a decision algorithm known as the “critical path”, which is the shortest and most efficient sequence of linked machine learning models required to go from the input of the user's message to the output of the chatbot's response. It is known that changes to one model can impact larger machine learning networks with machine learning dependencies.

One way to overcome dependencies is to transform unstructured text into machine-actionable information. *Natural language understanding* (NLU) is a pipeline created through many machine learning models that carry out different functions to improve the chatbot's understanding of user input. This helps to identify and filter out unnecessary models, leading to a reduced latency.

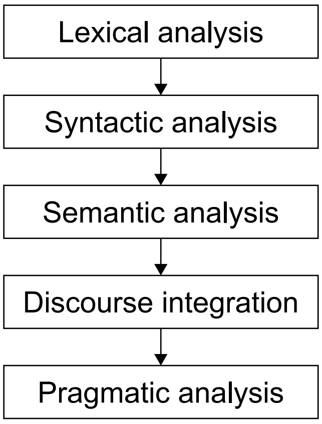
Training a chatbot is another step towards improving a chatbot's ability to understand text. To reduce response time, the training dataset needs to be large, accurate, classified, readable, domain specific and relevant.

Linguistic nuances

Chatbot language models can benefit from incorporating more human-like features, such as the ability to detect and respond to emotion, tone and context. Improving the language model allows the chatbot to generate responses that are more personalized and tailored to the individual needs of the customer, leading to a better overall experience.

Five stages of natural language processing are followed to ensure that chatbot responses are personalized (see **Figure 1**).

Figure 1: The five stages of natural language processing

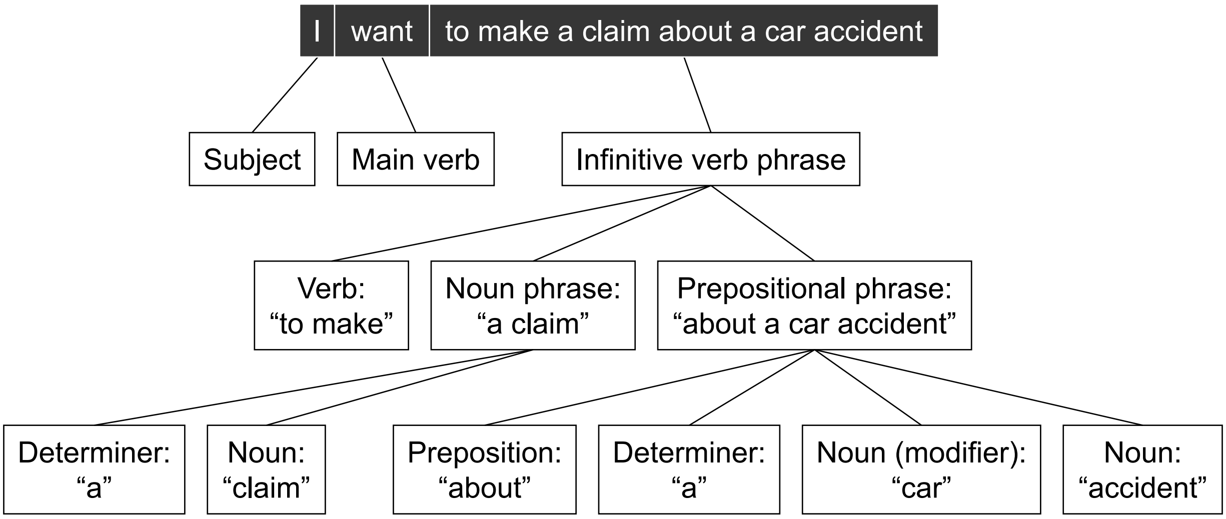


Lexical analysis and syntactic analysis focus on the structural aspects of language, such as breaking down text into individual words, identifying parts of speech, and analysing the relationships between words and phrases. Semantic analysis, discourse integration, and pragmatic analysis focus on the meaning of language, going beyond the surface-level structure.

Suppose a customer enters the following sentence into the chatbot: “I want to make a claim about a car accident”. Five-stage natural language processing interprets this sentence as follows:

1. **Lexical analysis**: The input text is broken down into words and sentences; for example, [“I”, “want”, “to”, “make”, “a”, “claim”, “about”, “a”, “car”, “accident”].
2. **Syntactic analysis (parsing)**: Interpreting the grammar and structure of the sentence; for example, that “I” is the subject of the sentence, “want” is the verb, and “claim” is the object (see **Figure 2**).
3. **Semantic analysis**: Analysing the meaning of the sentence and its individual words; for example, the sentence is about the customer’s desire to make a claim related to a car accident.
4. **Discourse integration**: Integrating the meaning of the sentence with the larger context of the conversation; for example, understanding that the user is a customer who would like to enquire about a claim.
5. **Pragmatic analysis**: Analysing the social, legal, and cultural context of the sentence; for example, understanding that the customer is a car driver who has had an accident and may or may not be hurt or have damaged the vehicle.

Figure 2: The chatbot's syntactic analysis of the grammar and structure of the sentence



Architecture

The natural language processing engine is the central component of a chatbot's architecture. It uses machine learning algorithms to determine the user's intent and match it to the action performed by the software.

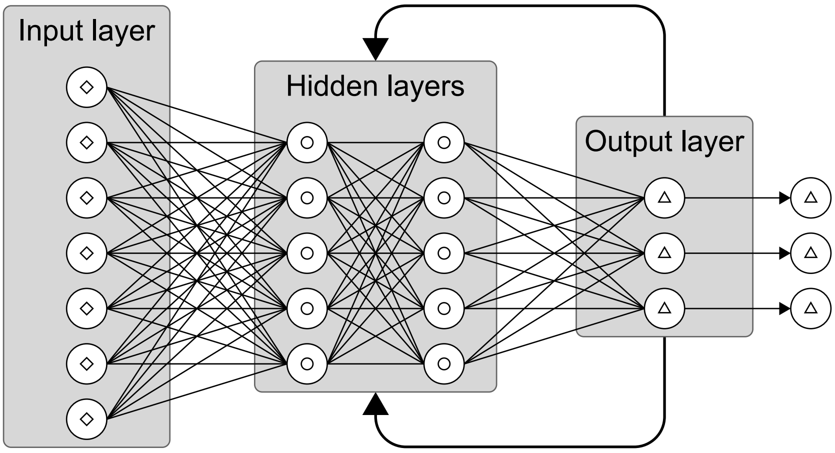
There are two types of deep learning models commonly used in natural language processing: *recurrent neural networks (RNNs)* and *transformer neural networks (transformer NNs)*.

Recurrent neural networks (RNNs)

The insurance company's chatbot uses an RNN, an artificial neural network designed to process sequential data.

An RNN has three *layers*. The *input* layer processes the sequence of data, with each element in the sequence fed as an input to the network. The *hidden* layers are responsible for maintaining the memory of the network. The *output* layer produces the final output of the network, which can be a prediction or a classification result (see **Figure 3**).

Figure 3: The input layer, hidden layers and output layer of a recurrent neural network (RNN)



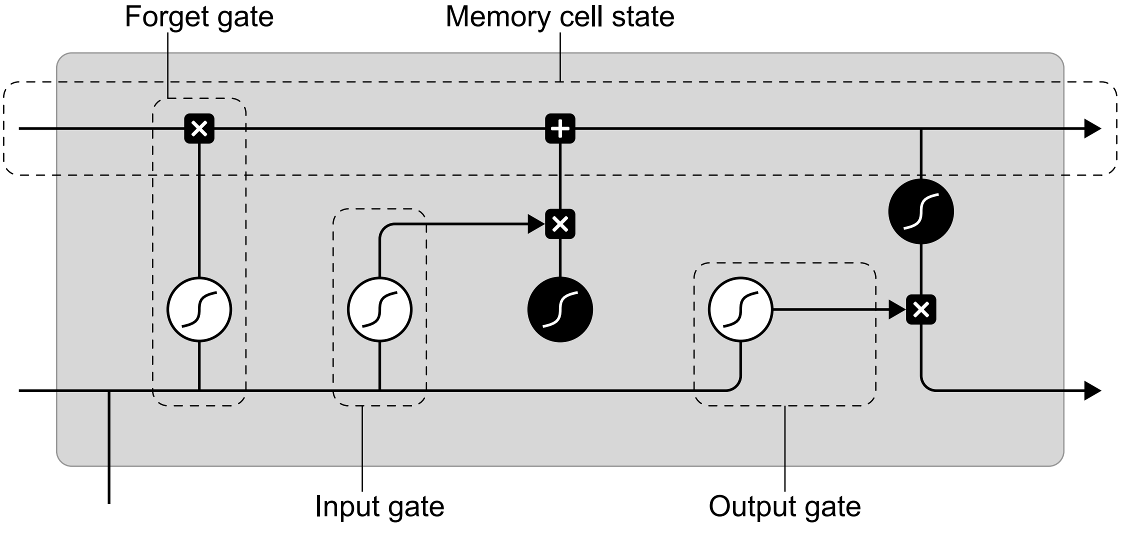
An RNN can process variable-length sequential data by maintaining a memory of previous inputs in the form of hidden states.

All RNNs need to be trained using an appropriate pre-processed dataset split into training, validation, and test sets. Hyperparameters, such as learning rate and the number of hidden layers, are set. *Hyperparameter tuning* involves finding the optimal values for these hyperparameters.

An RNN adjusts input, output and hidden layer *weights* during training using *backpropagation through time (BPTT)*. This technique computes gradients of the *loss function* for each weight at each time step, updating the weights accordingly. However, training RNNs using BPTT can cause the *vanishing gradient* problem, which makes it difficult to learn long-term dependencies in data.

Long short-term memory (LSTM) is a type of RNN designed to overcome the problem of vanishing gradients. It achieves this with a three-*gate* mechanism that controls the flow of information. The *memory cell state*, *input* gate, *forget* gate, and *output* gate enable LSTMs to selectively retain or forget information over time (see **Figure 4**).

Figure 4: The long short-term memory (LSTM) unit



Transformer NNs

A transformer NN is a powerful alternative to LSTM, with the key innovation being the application of a *self-attention mechanism*. The self-attention mechanism captures the relationships between different words in the input sequence by computing attention weights for each word based on its similarity to other words in the sequence.

An example of a large language model that uses self-attention mechanism is Generative Pre-trained Transformer 3 (GPT-3) created by Open AI.

Datasets

It is believed that the chatbot has not been using an unbiased, high-quality dataset. Datasets should be relevant to the domain in which the chatbot operates. These may include insurance claim data, customer service data, insurance policy data and medical data. Datasets can be created using real data and/or *synthetic data*.

It is important to ensure that these datasets are unbiased and diverse to avoid perpetuating any existing industry *biases*.

Several types of dataset biases may affect the accuracy and fairness of natural language processing models, including *confirmation*, *historical*, *labelling*, *linguistic*, *sampling* and *selection* bias.

- **Confirmation bias**: This form of bias occurs when the dataset is biased towards a particular viewpoint, such as training data that only include customer queries related to certain types of policies.
- **Historical bias**: This form of bias occurs when the training data do not reflect changes over time. For example, if the natural language processing model is trained on data from several years ago, it may not be able to accurately predict recent customer queries.
- **Labelling bias**: This form of bias occurs when the labels applied to the data are subjective, inaccurate or incomplete. For example, if the labels assigned to customer queries are too generic, the model may not be able to accurately predict the customer’s intent.
- **Linguistic bias**: This form of bias occurs when the dataset is biased towards certain linguistic features, such as dialect or vocabulary. For example, if a dataset is built on formal written language, the model may not be able to accurately interpret informal language.
- **Sampling bias**: This form of bias occurs when the training dataset is not representative of the entire population, such as training data that only include customer queries from one demographic.
- **Selection bias**: This form of bias occurs when the training data are not randomly selected but are instead chosen based on some criteria. A language model trained on data that suggest certain demographics may be more likely to file insurance claims that are biased towards people who fall under that category.

Processing power

The steps for preparing a machine learning model are:

1. *Pre-processing* the input data
2. Training the model
3. Deployment of the model

Pre-processing the input data

Pre-processing is the first step performed on raw data to make it understandable to the machine learning algorithms. Pre-processing involves cleaning, selection, transformation, and reduction of data to improve its quality and accuracy.

Presently, the chatbot's language model uses the *bag-of-words* algorithm to understand the client's input and extract relevant information. It does this by applying a vector representation of the text based on the frequency of words. The algorithm can be processor intensive when dealing with large datasets.

Training the model

Training the machine learning model is the most computationally intensive task, as massive amounts of text data are used. The training process can take weeks or even months to complete and requires powerful hardware, such as clusters of *graphical processing units (GPUs)* or *tensor processing units (TPUs)*, to speed up the computations.

Deployment of the model

Once trained, a large language model can be deployed on various hardware platforms. However, to achieve the best performance, specialized hardware, such as TPUs, is recommended. In addition to the hardware requirements, the running of LLMs also requires a significant amount of storage and memory to hold the model and its associated data.

Ethical challenges

The *RAKT* team have identified several ethical challenges that they will have to investigate. These include:

- Data privacy and security – protecting user data from misuse.
- Bias and fairness – avoiding discriminatory responses.
- Accountability and responsibility – assigning responsibility for the advice given.
- Transparency – explaining the decision making clearly.
- Misinformation and manipulation – preventing the spread of false information.

Challenges faced

The following challenges should be explored in more detail:

- Evaluating methods for reducing latency.
- Understanding the five stages of natural language processing.
- Determining whether a transformer NN could improve upon the performance of an RNN for natural language processing.
- Understanding what is required to create a large high-quality, unbiased dataset.
- Analysing the processing power requirements for a natural language processing model.
- Exploring the ethics of using a chatbot to provide advice.

Candidates are not required to understand the linguistic features of sentences such as in the subject, verb, object (SVO) model.

Additional terminology

- Backpropagation through time (BPTT)
- Bag-of-words
- Biases
  - Confirmation
  - Historical
  - Labelling
  - Linguistic
  - Sampling
  - Selection
- Dataset
- Deep learning
- Graphical processing unit (GPU)
- Hyperparameter tuning
- Large language model (LLM)
- Latency
- Long short-term memory (LSTM)
- Loss function
- Memory cell state
- Natural language processing
  - Discourse integration
  - Lexical analysis
  - Pragmatic analysis
  - Semantic analysis
  - Syntactical analysis (parsing)
- Natural language understanding (NLU)
- Pre-processing
- Recurrent neural network (RNN)
- Self-attention mechanism
- Synthetic data
- Tensor processing unit (TPU)
- Transformer neural network (transformer NN)
- Vanishing gradient
- Weights

Some companies, products, or individuals named in this case study are fictitious and any similarities with actual entities are purely coincidental.

Disclaimer:

Content used in IB assessments is taken from authentic, third-party sources. The views expressed within them belong to their individual authors and/or publishers and do not necessarily reflect the views of the IB.

References:

Figure 3 Sharma, V., 2019. Deep Learning – Introduction to Recurrent Neural Networks. [online] Available at: https://ailabpage.com/2019/01/08/deep-learning-introduction-to-recurrent-neural-networks/ [Accessed 9 May 2023]. SOURCE ADAPTED.

Figure 4 Rengasamy, D., Jafari, M., Rothwell, B., Chen, X. and Figueredo, G. P., 2020. Deep Learning with Dynamically Weighted Loss Function for Sensor-Based Prognostics and Health Management. *Sensors*, 20(723), pp.1–21. doi:10.3390/s20030723. SOURCE ADAPTED.