**COMP3074 – Human AI Interaction**

**Coursework 1 – General Knowledge Q&A Chatbot**

Oliver Unasing, psyou1, 20255470

psyou1@nottingham.ac.uk

## Introduction/Abstract

This report is explores the background, system design and evaluation of a General Knowledge NLP chat bot, built as the first coursework of COMP3074, Human AI Interaction, a level 3 module from the School of Computer Science at the University of Nottingham (academic year 2022/23). This project is built in 100% python and has had some functionality based on modified versions of code delivered in labs and tutorials as well as use of a general knowledge data set made available on the course. This report will discuss the project background, the proposed system (system structure, intent matching, small talk & general knowledge q&a), evaluation, discussion, and a conclusion on the achievements of the and future capabilities of final system.

## Background

The need for this system was made clear to me after I had conducted a discussion style study with students at the University of Nottingham, focusing on those who study heavily essay based degrees who are often using search engines to find facts and information. The outcome of this discussion what that the majority of these students found google sometimes challenging when searching for simple facts as often multiple links, webpages and ads provided as distractions from finding out the simple information they required. It was clear that there was a need for a simple to use system that allowed students to ask general knowledge questions without being overloaded with ads and additional data that only makes it more difficult to find the answer.

The solution for this problem flagged in discussion led to the decision to make this general knowledge chat bot. The goals then defined when evaluating the study were: easy to use GUI, no additional distracting features/data, familiar feel, and strong general knowledge question and answering.

## Proposed System

### System Architecture

The file structure of the chat bot is the figure below (fig1). To maintain clean coding practices I have divided the program into:  main.py, pre_processing.py, intent_matching.py, small_talk.py, question_answering.py and a readMe.

Main.py is where the while true chat bot loop is implemented, taking and pre-processing a query which then calls the  correct function (see intent matching) to generate response which runs until single stop words are detected in the query signalling the user is done with the system.

Pre-processing functions are stored in a file together as they are all called frequently by multiple other features. These functions contained are: textTokenize and multiple stemming and lemmatizing functions and work on all data, both query text and dataset pre-processing.

textTokenize() → Removes stop words from query

→ Converts text to lower case

→ Puts all characters in lower case

` →  Reduces all words to tokens

sbStemmer() → Uses snowball stemmer to reduce tokens to stems

pStemmer() → Uses porter stemmer to reduce tokens to stems

lemmitizer() → Lemmatizes tokens and maps the default tags to universal custom tagset

Data sets are then stored in the date file and reports stored in a report folder to maintain these good coding practices by not mixing file types in the same folder.

```
C:\Users\olive\OneDrive\Desktop\COMP3074\HAI_CW1>tree /f
Folder PATH listing for volume SSD
Volume serial number is AA8B-383C
C:.
    intent_matching.py
    main.py
    pre_processing.py
    question_answering.py
    ReadMe
    small_talk.py

├───data
        COMP3074-CW1-Dataset -QA.csv
        COMP3074-CW1-Dataset-ST.csv
        COMP3074-CW1-Intent-Dataset.csv

├───reports
        20255470_COMP3074_CW1_Report.docx

└───__pycache__
```
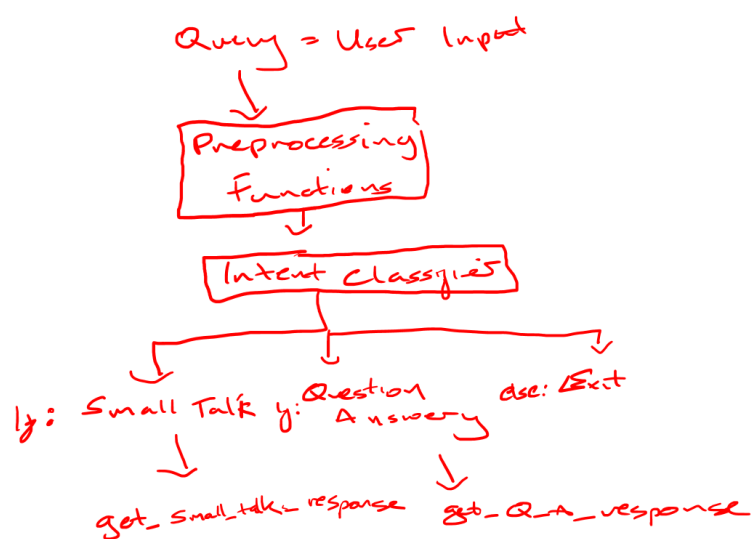fig 1.



fig 2.

**Intent Matching**

Intent matching is arguably the feature that requires the highest level of accuracy as the response feature function (small talk or question answering) is called based on the prediction of the classifier. Fig 2 (above) is an initial diagram of the intent matching design. This classifier uses the logistic regression classifier. This classifier was utilised as it is recognised as one of the more suitable machine learning models used for text classification and is based on the mathematic graphical diagram below (fig 3).
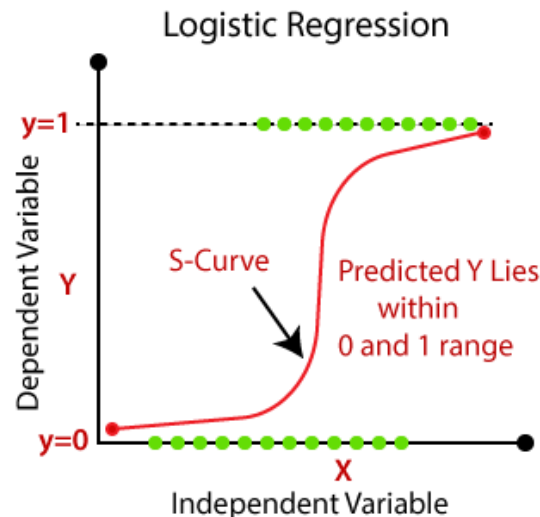


fig 3.

The data set used to train this model is custom made and a combination of the small talk and question answering datasets. The two columns in this dataset are [query, intent]. The query feature is filled with the example queries from the small talk and question answering datasets, which makes this dataset the most suitable for this specific problem; and the intent feature was manually filed with the dataset that query was from, which is its intent, either 'small talk' or 'question answering'.

This data was then split into X (queries) and y (intent), here the data goes through the pre-processing pipeline and was then further split into training and test sets (X_train, X_test, y_train, y_test) in a factor of 0.2 (80% training, 20% testing). The training data was then fed into the model which trained the intent classifier. Once trained the model can be used to predict the class/intent of any new queries using .predict(query).

Hence every time a user enters a query, intent matching function is called, feeding the model with the new, pre-processed query and returns either 'small talk' or 'question answering' which results in the system calling correct response function.

**Small Talk**

Small talk functionality was an area which was extremely important as this is what will gives users a feel of comfort when using the system. When discussing current methods of information retrieval (search engines) an issue was raised that they feel too robotic and 'stiff'. The small talk feature aims to break this down and prevent the system from feeling like a robot to the user, by allowing people to be able to converse with the chat bot a little before asking the questions. This feature was designed so the chatbot asks questions back to the user to provide a more natural conversation flow between user and chatbot.

Unlike intent matching, small talk uses similarity to link responses to the user query, based on a data set containing small talk question and answers from natural conversations. Similarity is calculated

using Cosine similarity (fig 4). This similarity is compared to a hardcoded threshold (set to a suitable value). If the similarity is above this threshold, the response is selected and outputted to the user.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum\limits_{i=1}^{n} A_i \times B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2} \times \sqrt{\sum\limits_{i=1}^{n} B_i^2}}$$

fig 4.

This feature deals with the initial greeting requests, asking how the user is and what they want to use the system for etc... Similar to a greetings function however this has been developed further to ask questions back at the user to encourage conversation. Creating a more personal feel between the user and system.

**General Knowledge Q&A**

Similar to the small talk functionality, the general knowledge question answering response system uses Cosine similarity (fig 4) to match queries to a response. This uses a small data set of questions and their answers made available on the Moodle (University of Nottingham) student platform as it provides a suitably sized set of data with a good breadth and depth of questions.

The similarity between the two vectors is then compared to a hardcoded threshold (set to suitable value), which is higher than the small talk threshold, as queries are far more specific when asking questions and the response it's a lot more important to output the correct answer. This similarity is still low enough however to allow various ways of asking a specific question, whilst still outputting the correct response (fig 5)

```
:>>what is jk rowlings worth
['Question']
["The 2008 Sunday Times Rich List estimated Rowling's fortune at £560 million ($798 million), ranking her as the twelfth richest woman in the United Kingdom."]
:>>how much is jk rowling worth?
['Question']
["The 2008 Sunday Times Rich List estimated Rowling's fortune at £560 million ($798 million), ranking her as the twelfth richest woman in the United Kingdom."]
:>>what is the net worth of jk rowling
['Question']
["The 2008 Sunday Times Rich List estimated Rowling's fortune at £560 million ($798 million), ranking her as the twelfth richest woman in the United Kingdom."]
:>>
```
fig 5

**Evaluation**

Once the system had been developed, to test it comprehensively a round of user testing was conducted.

User testing instance #1:

```
C:\Users\olive\OneDrive\Desktop\COMP3074\HAI_CW1>python main.py
Hi nice to meet you, I'm S.A.M
:>>hello how are you?
['Small Talk']
[' I am good. And u?']
```

```
:>>i am very well thank you
['Small Talk']
[' glad to know ur well. do u need any help?']
```

```
:>>yes i want to know how antibodies work?
['Question']
I am sorry, I cannot help you with this one. Hope to in the future. Cheers :)
:>>how do antibodies work?
```

```
:>>how do antibodies work?
['Question']
['An antibody (Ab), also known as an immunoglobulin (Ig), is a large Y-shaped protein produced by B-cells that is used by the immune system to identify and neutraliz
:>>thank you
```

```
:>>thank you
['Small Talk']
I am sorry, I cannot help you with this one. Hope to in the future. Cheers :)
:>>bye
Chatbot: ByeBye
```

User testing instance #2:

```
C:\Users\olive\OneDrive\Desktop\COMP3074\HAI_CW1>python main.py
Hi nice to meet you, I'm S.A.M
:>>what is the number of stripes on the american flag?
['Question']
['The national flag of the United States of America , often simply referred to as the American flag, consists of thirteen equal horizontal
red to specifically as the "union") bearing fifty small, white, five-pointed stars arranged in nine offset horizontal rows of six stars (t
```

```
:>>whose face is on the US tne dollar bill?
['Question']
['The first U.S. Secretary of the Treasury (1789-95), Alexander Hamilton , is currently featu
:>>What is the role of the president of the USA
```

```
:>>What is the role of the president of the USA
['Question']
['The president leads the executive branch of the federal government and is the command
```

```
:>>bye
Chatbot: ByeBye
```

## Discussion

From these two instances of user testing have outlined many points of discussion.

Instance #1:

This instance highlighted areas of success and areas of improvement for the chatbot. This user engaged in small talk before asking any questions which was overall a success. The bot responded

well to questioning "how are you" and responded with a suitable follow up question "and u?" to keep the conversation fluid, as if it were two humans conversing, fulfilling the goal of maintaining good conversation throughout strong talk.

However, when the user combined their small talk response with a question (having two intents; replying to small talk and asking a question) the chat bot did get confused and responded with an error message.

Although in this instance the chat bot did handle the questioning well when asked directly. Generating the correct response to the question.

Stop-word detection worked as required.

Instance #2:

The second user was much for question driven. Not entertaining any small talk with the chat bot at the beginning. The user noted "this was good because there was nothing getting in the way of finding out answers" which is a positive comment as the system design allows users to have small talk, but does not force them into it unlike some existing systems.

The chatbot answered questions well and with correct responses for all the questions asked. Even handling spelling errors in the second line of questioning "whose face is on US *tne* dollar bill" generating the correct response.

Overall, this instance of user testing yielded positive results, displaying strong question and answering abilities, not miss labelling on intent at all.

Stop-word detection worked as required.

**Conclusion**

The chatbot has the ability to understand the intent of the users input and respond appropriately, thanks to its use of text classification for intent matching. This allows the users to provide relevant answers to the users questions and engage in meaningful/natural conversation. Additionally, the chatbot is able to handle small talk and prove appropriate responses through the use of cosine similarity.

An area where this could be improved is better handling of queries that contain both a reply to small talk and question (q&a), highlighted by user testing instance #1. Further the chat bot could have features such as name management to make the chatbot refer to the user personally rather than a general input box.

However, overall, the combination of the two techniques allow the chatbot to provide a natural and effective conversational experience whilst also being able to answer general knowledge questions.