DCU Personal Assistant Chat Bot

Student: Ian Kelly (13480138)

Supervisor: Gareth Jones

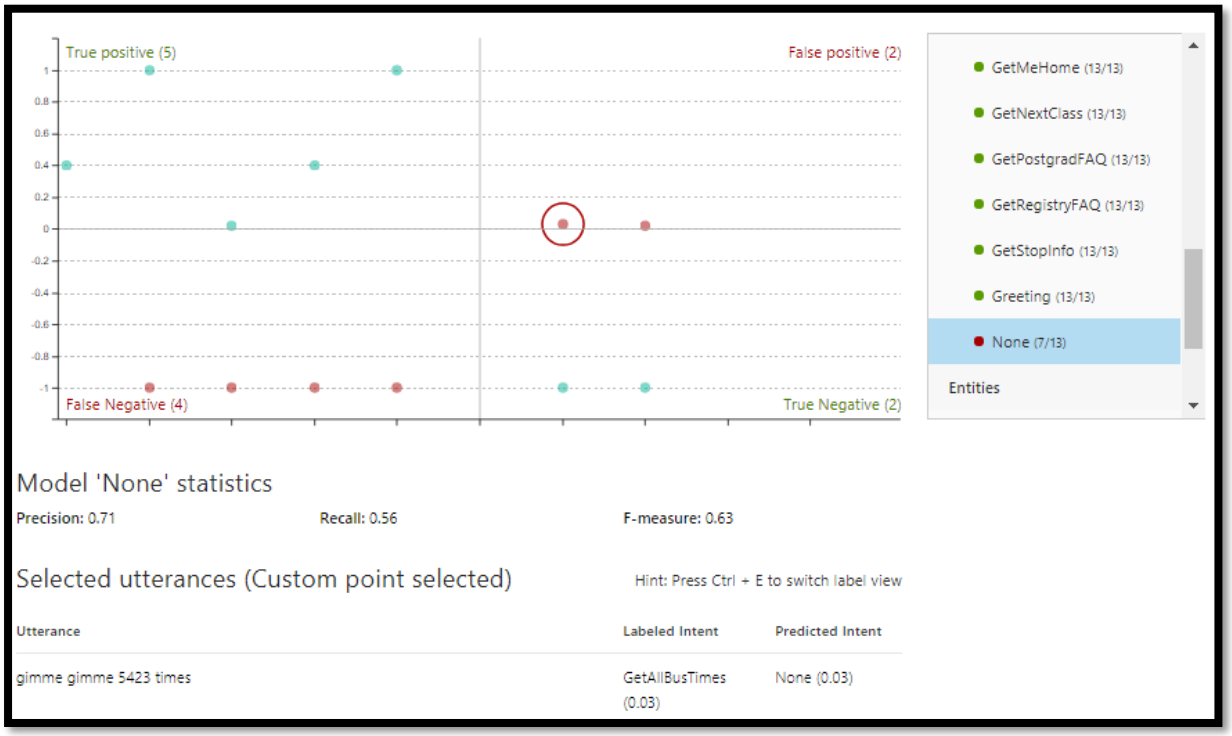
Testing

LUIS Batch Testing

Before any changes can be applies and published to the LUIS HTTP endpoint they must be tested. This means manually inputting example user sentences and ensuring that the correct intent is being called. As such, this testing is done continuously over the course of the bot development. This process can be automated with use of batch testing. The LUIS model is passed a json file containing dozens of user utterances and their expected intent and entities. This helps to improve intent prediction accuracy.

```
{
    "text": "\ ¸ ¶ ¥ £ ⅕ ⅙ ⅛ ⇤ ⇥ ⇦ ⇨ ⇩ ⇧ ⇧ ⇬ ⇬ ⇧ ⇬ ⇬ ⇨ ⇱ ⇲ ⇳ ⇷ ⇵ ⇶ ↔ ↔ ↔ ↔
    □ □ □ □ ",
    "intent": "None",
    "entities": []
},
{
    "text": "es",
    "intent": "None",
    "entities": []
},
{
    "text": "Present to me the total list of bus arrivals for stop 12",
    "intent": "GetAllBusTimes",
    "entities": [
        {
            "entity": "StopNum",
            "startIndex": 54,
            "endIndex": 55
        }
    ]
},
{
    "text": "bus times for 233 now",
    "intent": "GetAllBusTimes",
    "entities": [
        {
            "entity": "StopNum",
            "startIndex": 14,
            "endIndex": 16
```

As you can see from this image the batch test also tests bot input boundaries such as inputting characters and extremely long sentences. After the batch test has been run a graph is generated for each intent. Intents in the red indicate that there were errors with the utterance, indicating that the utterance was not processed correctly. As shown in the image below the 'None' intent has some false positives and false negatives. These results can then be used to alter the language model and a new batch test can be made.



User testing

The key to a successful chat bot is the ability to be able to answer most of questions that the user can ask with high accuracy. As such, user testing needs to be undertaken to find what users are looking for and in what form the users ask these questions.

To evaluate the successfulness of the DCU chatbot, 11 anonymous users were asked to test the chatbot on Kik and fill in a Survey Monkey form about their experiences. The users are given no prior knowledge of the bot's functionality apart from it being a chatbot for DCU students. This is to avoid potential bias in what questions are asked.

https://www.surveymonkey.com/r/H8JL37S

In the survey, the users are asked to answer 7 questions, with 6 of these questions asking the user to ask questions and record what they said and what the bot's response was. Below is a total list of what the users recorded, with some questions being left blank. Most of the bot responses we not specifically recorded so a mark of success or failure will be given instead, with a failure if the answer does not answer the use's questions.

Question One Do you use Kik Messenger regularly?

- 81.8% (9) of the users said they did not use Kik regularly.
- 18.2% (2) of the users said they used Kik regularly.

Question Two Without knowing the functionality of the bot, what is the first question you asked?

- Hey Pass
- Hi Pass
- Hello Pass
- How do you use this? Fail
- How do I use this? Fail
- What do you do? Fail
- Help Fail
- How can you help me? Fail
- Is my data safe? Fail
- What can you help me with? Fail

Question Three Now, ask it some random questions. What did you ask? And how did the bot reply?

- What time is the next bus home? Pass
- Is the restaurant open? Fail
- What time does spar close? Fail
- What are the buses into town from dcu? Fail
- What are the clubs and socs in dcu? Fail
- Can I make bookings? Fail
- What's on in the helix? Fail
- Tell me about dcu's sporting events. Pass
- When's my next class? Pass
- When is my report due? Fail
- When's my next exam? Fail
- What's the name of my lecturer? Fail
- When's the next bus?
- Where's xg23? Fail
- How do I get from O'Connell street to DCU? Fail

Question Four Now ask the bot some questions about Dublin Bus. What did you ask? And how did the bot reply?

- What bus gets me to town? Fail
- What bus gets me to dundrum? Fail

- What time's the next bus? Pass
- When's the last bus? Fail
- Get me a taxi Fail
- What bus gets me to town? Fail
- What bus gets me to a bus station? Fail
- What are the bus prices? Fail
- When is my next bus? Pass
- Is there a bus delay? Fail
- When's the next 15? Pass
- Get me the bus times for Whitehall. Fail

Question Five Now ask the bot some questions about DCU and the campus. What did you ask? And how did the bot reply?

- How many students are in DCU? Fail
- Does DCU have a rugby team? Fail
- Is there a ball coming up? Fail
- Where can I go to make friends? Fail
- What time does the library close at? Fail
- What time does the canteen close? Fail
- What's my next class? Pass
- When's the next sports event? Pass
- How do I join the football team? Fail
- How much is student accommodation? Fail
- Is nubar open? Fail
- How much is a pint in dcu? Fail
- Where can I get a coffee? Fail
- What are the library times? Fail
- Can I do a masters? Fail
- Is there a gym? Fail

Question Six Try get your DCU timetable using the bot. Were the results accurate?

- What time do I start? Fail
- What's my last lecture? Fail
- What time do I finish at? Fail
- Give me my timetable. Pass
- What's my next class? Pass
- Give me info on my next class. Pass
- Do I have anything due next class? Fail
- When's maths on? Fail

Question Seven Is there anything that the bot could not answer but you would like to see it be able to answer?

- What the last bus into town will be.
- To give info on clubs and socs
- Give me info on car parking
- Booking a study room.
- Finding my friends on campus.
- How to get a book from the library.
- Check if there's computers free.
- Find me a water fountain.

What I learned from the user tests

- The most common first question is a greeting. I can use this to provide the user with some details on the bot's functionality
- Without a knowledge of the bot's capabilities, users are likely to ask an extremely varied set of questions in multiple different ways.
- Some users have an unreasonable expectation on a bot's functionality. They ask vague questions that are almost impossible to answer without much for information. This could also be due to the lack of understanding about how a chat bot is built on the part of the user.

Unit and Integration Testing

Integration for this project is very minimal. This is due to each function being isolated in their own node module. When they are called in dialogs they are isolated to that specific dialog. This way function calls made outside of these dialogs have no effect on the function.

Each node module uses asynchronous callbacks to prevent interference from other function returns,

In relation to dialogs, only one dialog is called at a time and can only be interrupted by the user or when then dialog has finished completely.

Unit testing for the chat bot was done continuously over the course of development. For the chatbot to work correctly, the node modules need to output the correct data. So every time a node module was made or changes the functions it is called in are tested to ensure the correct response is produced.

There is also a test function in app.js that takes all of the node calls in one place and calls them using hardcoded variables. If they are not working correctly they will not produce the correct output for these variables.

```
bot.dialog('TestFunc', [
    function(session, args) {

        var intent = "What about sub judice?";
            dublinBus.getFOIFAQ(intent, function(err, qReply, answer)
            {
                session.send(qReply, session.message.text);
                session.send(answer, session.message.text);
            });

            dublinBus.getDCUSUEvents(function (err, result) {
                    session.send(result, session.message);
            });

        var building1 = "nursing";
            dublinBus.getBuildingLocation(building1, function(value)
            {
                session.send(value, session.message.text);
            });

        var building2 = "x";
            dublinBus.getBuildingLocation(building2, function(value)
            {
                session.send(value, session.message.text);
            });

        var courseID = 'case';
        var year = 4;
        var semester = 2;
            dublinBus.getTimetableDay(courseID, year, semester, function (reply) {
                session.send(reply, session.message);
            });
            dublinBus.getTimetableNext(courseID, year, semester, function (err, value) {
                session.send(value, session.message);
            });

        var stopNum = 4339;
        var routeNum = 42;
        dublinBus.getBusTimesSingle(stopNum, routeNum, function (err, reply) {
            if (routeNum != null) {
                session.send(reply, session.message.text);
            }
        });
```