Student ID Number: A0108517H

(Fill in your NUS Student ID here)

Institute of Systems Science
National University of Singapore

MASTER OF TECHNOLOGY IN ENTERPRISE BUSINESS ANALYTICS / INTELLIGENT SYSTEMS

Graduate Certificate Online Examination Semester I 2021/2022

Subject: *Practical Language Processing*

ANSWER BOOKLET SECTION A

Question	Marks
1	/30
2	/20
TOTAL	/50

Question 1 Answer

Question 1

1(a)

Architecture: End-To-End conversational modelling architecture are considered state of the art powerful modelling techniques. The problem is one need lot of data and model fine tuning. One need to create an AI corpus, receive user input, perform semantic analysis, pattern match keywords, extract the answer, and feed the results back to the chatbot.

Approach: GPT-2 is a powerful model can build intensive context information. However, specific domain training and fine tuning needed to boost model performance and obtain high quality results.

Training data:
Dialogue pairs of input and response extracted from Reddit discussion chains may not be the best as reddit discussions are wide and open, difficult to understand context. It will be hard to train the context logic.

Response generation using Beam Search:
The beam search is faster since it uses less memory. This is because storing all nodes in a queue isn't necessary and best, handful beam width candidates are selected. Challenge is it might never find a solution if it wont traverse well. Setting right threshold ensures fine trade off between performance and results quality. Even though the answer it delivers may not be optimal, it will undoubtedly save time for the user. A wider beam width can help with word selection, resulting in more accurate long phrases but requires more memory.

Model Evaluation:
Chat bot can be evaluated on given parameters:

1. SSA matrix to calculate perplexity can be used in order to check the quality of human responses.
2. Opting for human testers and set parameters
3. Effectiveness Performance, content, movie Domain coverage
4. Response time, waiting time, Logical Coherence, empathy in speech
5. Measuring satisfaction behaviour, ethics, profanity

1(b) (1) Identity: movie name, movie date release
Description: genre, reviews, imdb link
Scheduler: Location of theatre, timing
Booking: Payment details, Reservation confirmation

Reviews, genre, and an imdb link are all included in the information on the film.
Payment information and confirmation are required when making a reservation.

1(b) (2)

U: Hi - **Greeting_Intent()**
S: Good day, how can I help?

U: I wish to view the movie name Om Shanti Om (movie_name) at Yishun Golden Village(theatre_location)
S: Sure, what time ? **Booking_Intent()**

U: Is today 6 pm (timing) show ?
S: How many people are there accompanying? **booking_intent()** - booking_info

U: Three people (booking_info - no_of_seats)
S: Noted with thanks, please pay 30 dollars (payment_info) using credit card confirmation link.

U: Sure, I am paying
S: Alright, tickets confirmed for 6pm Wednesday at Yishun Golden Village for three people.

1(c) Squad is good for implementation. There are over 100,000 variety of questions on top of Wikipedia which is a huge quantity. Constituency regularly updated by Standford.

However, the downside is that squad may not have adequate detailed, context based information for specific movies like timings, location and seating details.

1(d) Content based recommender system can be used.

Dataset attributes to look at: Title, Director, Actors, Plot, Genre, Rating

Filter the keywords of all above attributes. Based on bag of words approach, need to create features based on frequency of words using Count Vectorizer.
Cosine Similarity which provides a similarity score between two vectors.
This is a dynamic way of finding the similarity that measures the cosine angle between two vectors in a multi-dimensional space. In this way, the size of the documents does not matter. The documents could be far apart by the Euclidean distance but their cosine angle can be similar.

1(e)(1)

The text is normalized by converting numbers, dates abbreviation and acronyms to standard unique word identified in the dictionary. This helps in maintaining correct pronunciation and recognition as it converts written-form words to spoken-form words. Eliminate diacritical marks. Convert words to lower case and lemmatise them. Remove stop words.

(e)(2) Lets look at the mistakes below for the 3 cases.
Case 1: tickets is mistaken as **takes** (Tickets is a booking domain term)
Case 2: age-rating is mistaken as **greeting** (Age-rating is movies domain term)

Case 3: two tickets mistaken as **two packets** for me. (two tickets is booking domain term)

Domain training for movies & booking is not accurate, ticket is recognized as a near word like take or packet depending upon where the emphasis was given while pronouncing the word.

It shows dataset is **not** trained on age-rating - movie keywords and domain. Surprisingly, other words (including capitalized words) and even punctuation are well recognised with high accuracy which implies overall generic model training is highly accurate.

As the team is planning to collect some data to improve the recognition accuracy of the speech recognition system, perform below mentioned tasks:

1. Collect movies related dataset for movie names, age, ratings, restriction
2. Collect ticket booking related dataset for generalizing the terms like seat, booking, tickets

(e) (3) Speaker verification is preferred choice as we need to one-to-one verification to understand whether the user claimed/calling is actually the one who is stored in our DB. One to one matching is to detect whether the voice belongs to the claimed speaker.
One to many matching (identification) is to find the speaker from a group of people which is not the case here as only one caller is calling.

Methodology:

1. Register - Store longer voice sample from the users, longer recording to get enough information with a few sentences of a min.
2. Extract the features from the user voice samples using x-Vector as long term features might be required. The x-vectors are embeddings extracted with DNN which is a little bit newer and more powerful method,
3. GMM-UBM or Supervector to adapt the speaker model and SVM classifier can be used to do the classification
4. Once a user is called the voice features embedding then can be matched against the stored ones and the one that is most similar should ideally be of the same user. Otherwise impersonation attempts can be flagged.

Question 2

(a) Named Entity Recognition is the process of extracting predefined entities from unstructured data which Spacy and Stanford NLP can achieve using pre-trained machine learning methods. Good thing is this reduces human effort in maintaining rules and dictionaries which is time consuming but one need to prepare set of annotated medical training data/corpus.

(b) Model is accurate but recall and precision are more comprehensive metrics. Precision is proportion of data instances of an event a model identified that really depict that event. Bob should ensure good precision to ensure better predictive performance rate of model (i.e those identified belonging to a class should belong to that class). Recall is proportion of total actual data instances of an event identified by model. Owing to class imbalances, important for Bob to track if all classes well identified or not.

(c) There’s room for improvement for ‘Diagnosis’ and ‘Treatment’ classes. Important to ensure stopwords removed as they contribute little impact to model performance. Remove special characters, punctuations and accents like diatrics. Lemmatisation, a practice of reducing different variations of similar word to its root form also help. N-grams features, practice of combining N number of words as features also improve model predictive power. Bigrams can be combined with parts of speech tags as well to create accurate features. Eliminate features with extremely low frequency. Increase predictive model class weights of least represented classes to make it sensitive to least occuring classes in original dataset.

(d)

The task that involves finding an answer in multiple documents is often referred to as open-domain question answering involving Medibert model.

Next, there are several stages for answer retrieval:

1. **Retrieval stage:** Need to find a set of relevant paragraphs in one document for Medibert at a time. The retrieval stage should be fast, so we care more about the recall and less about precision. Usually, the sparse retrieval (bigram TF-IDF or BM25) is used, but dense retrieval models like Universal Sentence Encoder (USE) also can be fast enough here.
2. **Re-rank stage:** Re-rank retrieval stage results to reduce the number of selected paragraphs. The retrieval and re-ranking stages can probably be reduced to a single one based on the number of documents, quality, and performance requirements.
3. **Document reader:** apply the MRC model to each document among multiple ones under consideration to find the answer.

(e) Tokenise paragraphs in a document into sentences. We can look at sentence level embeddings to compute similarities between asked question embeddings and sentence level embeddings as well using cosine similarity in Gensim library in python.

(f)

It is important to have rough definition and understanding of BERT model architecture.
Embedding Layer: The training dataset can be looked at individual document level which consists of paragraphs. Generate paragraph level embeddings.

Encoder Layer:
A bi-directional GRU/LSTM can be used at this layer that converts input sequence vectors. The output is a series of hidden vectors in the forward and backward direction and we concatenate them.

Attention Layer: It is the key component in the Question Answering system since it helps us decide, given the question which paragraphs in a document I must pay attention to.

Output Layer:
The final layer of the model is a softmax output layer that helps us decide the start and the end index for the answer span. We combine the context hidden states and the attention vector from the previous layer to create blended reps. Our loss function is the sum of the cross-entropy loss for the start and end locations and it can minimized using Adam Optimizer to generated predicted relevant paragraph.

BERT uses two methods for model training: MLM (Masked LM) and NSP (Next Sentence Prediction)

MLM (Masked Language Modelling)
In the sequence, we randomly mask some percentage of words, by replacing them with token mask for <information>. Normally, one can mask 15% of input words. It is trained to predict these masked words using the context from the remaining words

Next Sentence Prediction (NSP)
To understand the relationship between two sentences, BERT uses NSP training. The model receives pairs of sentences as input, and it is trained to predict if the second sentence is the next sentence to the first or not. The assumption is that the random sentence will be disconnected from the first sentence in contextual meaning.