University of Glasgow

Tuesday, 4 May 2021
Available from 09:30 BST
Expected Duration: 1 hour 30 minutes
Time Allowed: 3 hours
Timed exam within 24 hours

DEGREES of MSci, MEng, BEng, BSc, MA and MA (Social Sciences)

TEXT AS DATA H
COMPSCI 4074

(Answer all 3 questions)

This examination paper is an open book, online assessment and is worth a total of 60 marks.

1. This question is about tokenization and similarity.

(a) Consider the following piece of text:
["I placed an order (order# ZA10880) on 05-11-2020 for rear axle break discs for a BMW8 Coupe - part no. B342r. DH-D have lost the order. Please re-deliver ASAP."]
and a transformation of it that has been distorted:
["Iplacedanorder(order#ZA10880)on05-11-2020forrearaxlebreakdiscsforaBMW8Coupe-partno.B342r.DH-Dhavelosttheorder.Pleasere-deliverASAP."]

The result is a 'runaway token'.

(i) Describe a tokenization strategy to process the runaway token and split it into a series of "approximate" tokens as best as possible. No outside resources (e.g. vocabulary) are available. Justify your proposed tokenization method. [3]

(ii) *Name* and *discuss* three classes of tokens that are hard to identify, used in the example provided above. Discuss why each is challenging and provide a solution to handle each type of challenge appropriately. [3]

(b) You are designing a stock management system for 'GOLE' a manufacturer of children's building block sets. Each set contains several different types of bricks. The part codes and numbers of bricks are provided to you for each set: e.g. "3CD" $\times$ 10, "4DE" $\times$ 7, "1AB" $\times$ 5, "2BC" $\times$ 2.

Consider three different sets with the following collection of parts:

SetA: "1AB" $\times$ 6, "3CD" $\times$ 9, "5EF" $\times$ 17
SetB: "2BC" $\times$ 9, "3CD" $\times$ 10, "4DE" $\times$ 11, "5EF" $\times$ 3
SetC: "1AB" $\times$ 20, "2BC" $\times$ 12, "3CD" $\times$ 8, "4DE" $\times$ 7

(i) Create a vocabulary for the data and provide it. Describe any important decisions in the creation of the vocabulary. [3]

(ii) Using the vocabulary, provide a dense vector representation for each set (SetA, SetB, SetC). Justify the choice of set used. [3]

(iii) You are tasked with facilitating repurposing sets from one type to another. To do this you are asked to investigate which pairs of sets are most similar. Use Cosine similarity to calculate the similarity between each of the 3 pairs of sets (A&B, A&C, B&C) and identify which pair is most similar. [4]

(c) (i) Suggest another similarity measure that could be used instead of cosine similarity for the above problem. [2]

(ii) Discuss the Pros and cons of your suggested method. [2]

2. This question is about language modeling and classification.

(a) This task involves developing an order error corrector for a chain restaurant's breakfast order system. Below is a table of five separate order interactions transcribed from a mobile app.

forget it i wanna eat a croissant
no i wanna eat a croissant
do you serve granola?
i would like to have a bacon roll and a cappuccino
would you like coffee with that

Table 1: Five interactions for a burger restaurant ordering system.

Sample text collections statistics for a bigram model are below:

- $V = 27$ unique words (including reserved tokens)
- $N = 46$ tokens, including padding

(i) Use the text provided in Table 1 above to compute word unigram probabilities. In a list or table format complete the probability table with Laplace smoothing that has $K = 0.7$. Show your workings. Discuss the impact on the probability values of increasing or decreasing the value of K. Describe the effect of varying K when these probabilities are used in a spelling (error) correction task.

Word	Unigram Probability
granola	
roll	
croissant	

[5]

(ii) A larger collection of restaurant ordering data is collected. It has the following statistics: $N = 68237$, $V = 1892$ from a total of 8423 documents (utterances).
Compute the bigram probability of the following sequence:

```
[i would like some breakfast]
```

with Stupid Backoff smoothing with default values. Collection statistics for the required terms are provided below. Show your workings, including each bigram's probability. Describe how and why a smoothing method is used here.

[6]

Term	Count
i	2623
would	5
like	1522
some	323
breakfast	6
$\langle s \rangle$ i	2003
i would	0
would like	2
like some	25
some breakfast	3
breakfast $\langle e \rangle$	2

(b) The corrected trained LogisticRegression model has an accuracy of 92% on the training data, 87% on the validation data, and 45% on the test data. A dummy classifier has an accuracy of 50% on all three data subsets. Describe the model 'fit'. Suggest a way to fix this. [3]

(c) Below is a snippet of code to vectorize and classify text with Scikit-learn. Assume that `tokenize_normalize` and `evaluation_summary` have been defined, as we did in the labs. The input data has been pre-processed into a vector of unnormalized text documents (each a single string).

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import LogisticRegression

# Data processing
data = ... # Loads a vector of raw text documents
train_index = int(len(data) * 0.8)
test_index = int(len(data) * 0.2)
tmp_train = data[:train_index,:]
validation_split = int(train_split * 0.8)
train_data = tmp_train[:validation_split,:]
validation_data = tmp_train[validation_split:,:]
test_data = data[:test_index,:]

# Vectorization
one_hot_vectorizer = CountVectorizer(tokenizer=tokenize_normalize,
                                     binary=True,
                                     max_features=20000) # Reasonable number > 1k
one_hot_vectorizer.fit(train_data)
train_features = one_hot_vectorizer.transform(train_features)
validation_features = one_hot_vectorizer.fit_transform(validation_data)
test_features = one_hot_vectorizer.transform(train_data)

# Classification
lr = LogisticRegression(solver='saga', max_iter=500)
lr_model = lr.fit(test_features, test_labels)
```

```
evaluation_summary("LR Train summary",
lr_model.predict(train_features), train_labels)
evaluation_summary("LR Validation summary",
lr_model.predict(validation_features), train_labels)
evaluation_summary("LR Test summary",
lr_model.predict(test_features), test_labels)
```

Copy and paste the code above and fix its mistakes. Although there may be more, discuss **three** important mistakes with their consequence, one from each section (data processing, vectorization, classification). [6]

3. This question is about word embedding models and Natural Language Processing.

(a) Using your knowledge of Part of Speech (PoS) Tagging by means of HMMs, build emission and transition tables from the training sentences S1 and S2. Then utilise the Viterbi algorithm to figure out the most likely PoS tagging sequence for test sentence T1.

Note that smoothing is not required. I.e. let frequencies of 0 result in probability of 0.

Training sentences: [

S1: ⟨s⟩ Juliet (NN) loves (VB) people (NN) ⟨e⟩
S2: ⟨s⟩ People (NN) love (VB) Romeo (NN) ⟨e⟩

]

Testing sentence: T1: ⟨s⟩ Romeo loves Juliet ⟨e⟩

(i) Build a HMM emissions table [5]

(ii) Build a HMM transitions table [5]

(iii) Compute the most likely PoS tagging sequence using the Vitervi algorithm [3]

(b) Gaelic is a language spoken by a recorded 87,056 people mostly in Scotland. There are many books providing translations between english and gaelic. Below are some sample Gaelic-English translations:

Gaelic	Approximate English Translation
De an t-ainm a tha' oirbh?	What's Your Name?
Ciamar a tha sibh?	How are you?
Tha mi duilich	I'm sorry
Tha beatha ro ghoirid	Life is too short

Figure 1: Sample sentences in Gaelic

Imagine you are building a prototype translation software that relies on advanced NLP techniques such as BERT to produce high quality contextualised translations. Unfortunately you have not found any pre-trained BERT models online, so you decide to train a new model yourself on existing literature that has been published in both English and Gaelic

(i) Describe the process for pre-training BERT on Gaelic. Briefly describe what is required and any changes needed for the model. Be sure to include discussion of tokenisation considerations and training objectives. [7]