CPSC 477/577
Natural Language Processing
Yale University
Spring 2021
Practice Final Solutions

Disclaimer: If no solution is available, but you need help with a particular question, please post on canvas and/or ask a TF.

Question 1

Represent the following sentences in first-order predicate calculus (FOPC). There may be multiple ways to represent each of them. Give only one representation for each sentence.

(a) Only one person understood the play.
(b) Exactly two people understood the play.

(a) $\exists$ x : Understood(x, thePlay) $\wedge$ ($\forall$ y: Understood(y, thePlay) $\rightarrow$ x=y)
(b) $\exists$ x,y : Understood(x, thePlay) $\wedge$ Understood(y, thePlay) $\wedge$ x $\neq$ y $\wedge$ ($\forall$ z: Understood(z, thePlay) $\rightarrow$ (x=z $\vee$ y=z))

Question 2

Give an example for each of the following verb subcategorization phrases.

1. NP
2. NP NP
3. Ø
4. V P_{to}
5. S

1. NP – run a race
2. NP NP – give my friend an apple
3. Ø – sleep
4. Pto – want to eat. (I think having “V Pto” was a mistake. I can't think of a verb with that frame.)
5. S – said he was fired.

Question 3

Consider the five sentences shown in Figure 1. Think of them as your training data to build a probabilistic context-free grammar (PCFG).

(a) Build a PCFG using the training data. For each non-lexical rule (e.g., S → NP VP), indicate its probability.

S -> NP VP	4/8	NP -> DT JJ NN	1/15
S -> WH S	2/8	VP -> VBZ NP PP	1/7
S -> AUX NP VP	2/8	VP -> VBZ NP PP PP	1/7
NP -> NNP NN CD	2/15	VP -> VBZ NP	1/7
NP -> NP PP	1/15	VP -> VB	2/7
NP -> NNP	4/15	VP -> MD VP	1/7
NP -> CD RB	1/15	VP -> VB NP	1/7
NP -> QP	1/15	PP -> IN NP	1
NP -> DT NNP NN	1/15	QP -> CD RB	1
NP -> DT NN	4/15	WH -> WRB	1

(b) Build a probabilistic lexicon for the PCFG. Give the probability of each lexical rule (e.g., NN → flight).

NNP -> Delta	2/7
NNP -> Toronto	1/7
NNP -> Atlanta	2/7
NNP -> Detroit	1/7
NNP -> Northwest	1/7
NN -> flight	6/8
NN -> meal	2/8
CD -> 411	¼
CD -> 6	¼
CD -> 29	1/4
CD -> 10	1/4
VBZ -> leaves	2/3
VBZ -> serves	1/3
IN -> for	1/3
IN -> at	2/3
RB -> PM	½
RB -> AM	1/2
DT -> that	1/6
DT -> a	2/6
DT -> the	1/6
DT -> this	2/6
VB -> serve	1/3
VB -> arrive	1/3
VB -> leave	1/3
WRB -> when	1
AUX -> does	1

JJ -> light	1
MD -> may	1

Question 4

What (maximum likelihood) formula is used to estimate trigram probabilities P(wn|wn-1,w n-2) using a corpus. Don’t worry about smoothing or backoff.

no solution given

Question 5

Why are “mildly context-sensitive grammars” like Tree Adjoining Grammars (TAG) and Combinatory Categorial Grammars (CCG) used in NLP? Give two reasons.

Most importantly, mildly context-sensitive grammars are able to model grammatical phenomena that context-free grammars are not able to. *For example, any of these grammars listed are able to model the language* $a^nb^nc^n$*, which is not context-free.*
Can capture cross-serial dependencies (it is enough to say “can capture some linguistic phenomena that CFG cannot”)

Question 6

A.
a. [[ice cream] soda]
b. [[science fiction] writer]
c. [[customer service] representative]
d. [state [chess tournament]]
e. [[Mars Rover] landing]
f. [plastic [water cooler]]
g. [[typeface design] report]

B. c. [[country song] [platinum album]]
A reality television show about people in space who must control the entire mission.
(a reality television show is a drama on television that is unscripted)

C. one plausible bracketing is
[[[family [board game] [togetherness effect]] [government study]] author]

D. This can be done recursively. 5 words can be broken down 4 ways:
5 → 1, 4 = 1 bracketing * 5 bracketings = 5 bracketings
5 → 2, 3 = 1 bracketing * 2 bracketings = 2 bracketings
5 → 3, 2 = 2 bracketings * 1 bracketing = 2 bracketings
5 → 4, 1 = 5 bracketings * 1 bracketing = 5 bracketings.

This is 14 total. To f(5) = 14.

f(6) can be computed similarly:
6 → 1, 5 = 1 * 14 = 14
6 → 2, 4 = 1 * 5 = 5
6 → 3, 3 = 2 * 2 = 4
6 → 4, 2 = 5 * 1 = 5
6 → 5, 1 = 14 * 1 = 14
So f(6) = 42

Question 7

Give first-order logic translations for the following sentences:

Vegetarians do not eat meat.
Not all vegetarians eat eggs

\forall x: Vegetarian (X) => ~Eats(x,Meat) - there may be other solutions
\exists x: Vegetarian (X) ^ ~Eats(x,Eggs) – there may be other solutions

Question 8

a) Answer = 0.0128

The probability of a given path that generates “baa” is the product of its transition and emission probabilities. To find the probability of generating “baa”, we need to sum over all the possible paths that can generate “baa”.

We could use the Forward algorithm to do this, but since there are only 8 possible paths, it is maybe more straightforward to enumerate them and calculate their probabilities and sum the result:

State sequence	Transition probs		Emission probs		p(“baa”)
1 1 1:	.4 * .4 * .4 * .1	*	.3 * .7 * .7		0.0009408
1 1 2:	.4 * .4 * .5 * .2	*	.3 * .7 * .2		0.000672
1 2 1:	.4 * .5 * .6 * .1	*	.3 * .2 * .7		0.000504
2 1 1:	.6 * .6 * .4 * .1	*	.8 * .7 * .7		0.0056448
1 2 2:	.4 * .5 * .2 * .2	*	.3 * .2 * .2		0.000096
2 1 2:	.6 * .6 * .5 * .2	*	.8 * .7 * .2		0.004032
2 2 1:	.6 * .2 * .6 * .1	*	.8 * .2 * .7		0.0008064
2 2 2:	.6 * .2 * .2 * .2	*	.8 * .2 * .2		0.0001536
				Total	**0.0128**

b) Answer = 0 1 2 1 3 (or just 1 2 1)

Similar to the process for part (a), we can enumerate the possible paths that generate “aba”, find their probabilties, and see which path has though enumerating 8 possible paths the highest probability. We could also use the Viterbi algorithm, is possibly faster and less complicated:

State sequence	Transition probs		Emission probs	p(“aba”)
1 1 1:	.4 * .4 * .4 * .1	*	.7 * .3 * .7	0.0009408
1 1 2:	.4 * .4 * .5 * .2	*	.7 * .3 * .2	0.000672
1 2 1:	**.4 * .5 * .6 * .1**	*****	**.7 * .8 * .7**	**0.004704**
2 1 1:	.6 * .6 * .4 * .1	*	.2 * .3 * .7	0.0006048
1 2 2:	.4 * .5 * .2 * .2	*	.7 * .8 * .2	0.000896
2 1 2:	.6 * .6 * .5 * .2	*	.2 * .3 * .2	0.000432
2 2 1:	.6 * .2 * .6 * .1	*	.2 * .8 * .7	0.0008064
2 2 2:	.6 * .2 * .2 * .2	*	.2 * .8 * .2	0.0001536

Maximum of these, should be fairly obvious that is is 1-2-1
Best state sequence is 0-1-2-1-3

Question 9

A. Verbs, nouns, adjectives, adverbs

Common mistake: verb phrases, noun phrases, etc. (those are not parts of speech).

B. Prepositions, conjunctions, particles, interjections, pronouns, articles...

Question 10

No solution given.

Question 11

You are in a noisy bar diligently studying for your midterm, and your friend is trying to get your attention, using only a two words vocabulary. She has said a sentence but you couldn’t hear one of the words:

(w1 = hi; w2 = yo; w3 =???; w4 = yo)

Assume that your friend was generating words from this first-order Markov model:

p(hi|hi) = 0.7   p(yo|hi) = 0.3

$p(hi|yo) = 0.5$ $p(yo|yo) = 0.5$

Given these parameters, what is the posterior probability of whether the missing word is “hi” or “yo”?

This question is asking for $p(w_3|w_1,w_2,w_4)$. By the Markov assumption we can ignore w_1 completely, thus just $p(w_3|w_2,w_4)$. Next we want to manipulate this into a form where we can apply our model parameters, which specify $p(w_t|w_{t-1})$ for any pair of word types (those four numbers above). Our model does not tell us $p(w_3|w_2,w_4)$, nor does it tell us $p(w_3|w_4)$ ... but it does tell us $p(w_3|w_2)$ and $p(w_4|w_3)$. We can start to get the second form $p(w_3|w_2, {}_4)$ by applying Bayes Rule to flip w_3 and w_4. (This is an instance of background-conditional Bayes Rule p(a|bc) = p(b|ac)p(a|c)p(bc), which is like normal Bayes Rule except there's a "background" variable c always hanging on the right side.)

So we use Bayes Rule where the prior is $p(w_3|w_2 = \mathit{yo})$ (a function of w_3) and the likelihood is $p(w_4 = yo|w_3)$ (a function of w_3).

$P(w_3|w_2, w_4) = (1/Z)p(w_3|w_2)p(w_4|w_2, w_3)$ (1)
$P(w_3|w_2, w_4) = (1/Z)p(w_3|w_2)p(w_4|w_3)$ by Markov assumption (2)
$p(?? = hi) = (1/Z)p(hi|yo)p(yo|hi) = (1/Z)(0.5)(0.3) = (1/Z)0.15$ (3)
$p(?? = yo) = (1/Z)p(yo|yo)p(yo|yo) = (I/Z)(0.5)(0.5) = (I/Z)0.25$ (4)
$Z = 0.15 + 0.25 = 0.4$ (5)
$p(?? = hi) = 15/40$ (6)
$p(?? = yo) = 25/40$ (7)

I find it easiest to think of Z as summing over all possible versions of the denominator: Z = sumof w_3 $p(w_3|w_2=yo)p(w_4=yo|w_3)$. You could also start with $Z = P(w_3|w_4)$ then use the sum rule to work it out from there.

Question 12

One possible answer

Marina	NP	(or N)
often	(S\NP)/(S\NP)	
gives	((S\NP)/NP)/NP	
John	NP	(or N)
cold	NP/NP	(or N/N)
water	NP	(or N)

<table>
<tr><td>Marina
N</td><td>often
(S\NP)/(S\NP)</td><td>gives
((S\NP)/NP)/NP</td><td>John
N</td><td>cold
N/N</td><td>water
N</td></tr>
<tr><td>NP</td><td></td><td></td><td>NP</td><td colspan="2">N</td></tr>
<tr><td></td><td></td><td colspan="2">(S\NP)/NP</td><td colspan="2">NP</td></tr>
<tr><td></td><td colspan="3">(S\NP)/NP</td><td colspan="2">NP</td></tr>
<tr><td></td><td colspan="5">S\NP</td></tr>
<tr><td colspan="6">S</td></tr>
</table>

Question 13

What is the formula for the sigmoid function described in class f(z)? Why is it commonly used in Neural Networks?

It is used as a non-linear activation function. It is differentiable, monotonic, and has values between 0 and 1, which makes it useful for computing probabilities.

Question 14

We first need to introduce a function that compares similarity between two sentences. Recall the definition for the cosine distance between two vectors x and y:

Part A:
f(s) = a vector that contains a feature for each word w that counts the number of times w is seen in s

Part B:

```
F(0,0) = 0
F(i,0) = 0
F(0,j) = 0
F(i,j) = max[  F(i-1, j-1) + s(i,j),
               F(i-1, j),
               F(i,j-1),
               F(i-2,j-1) + s(i-1,j) + s(i,j),
               F(i-1,j-2) + s(i,j-1) + s(i,j)]
```

Part C: F(0,0) = 0

```
F(i,0) = -ip
F(0,j) = -jp
F(i,j) = max[  F(i-1,j-1)+s(i,j),
               F(i-1,j)-p,
               F(i,j-1)-p,
               F(i-2,j-1)+s(i-1,j)+s(i,j),
               F(i-1,j-2)+s(i,j-1)+s(i,j) ]
```

Question 15

Assume in the contexts of HMMs that *S* refer to a sequence of states, *O* refers to a sequence of observations, and *M* is a particular HMM model. What is the algorithm that computes the value: argmax *P* (*S* | *O*, *M*)?

Viterbi

Question 16

Consider a logistic regression model used to classify documents into "sports" vs. "not sports". The weights θ = (−ln(4), ln(2), −ln(3)). A given document D is represented as the feature vector x = (1, 1, 1).

What is the probability that document D is about sports?

Please provide your answer in the form of a fraction.

1/7:
S = \theta * W = -ln(4) + ln(2) -ln (3) = ln(6)
Then we need a sigmoid function: 1/(1+e^(-s)) = **1/7** = 0.142857..

Question 17

What problem of simple recurrent neural network does LSTM solve? How does it solve the problem? What is another variant of SRN that solves the same problem?

Describe how an LSTM works. Draw a diagram and explain all gates and all forward propagation formulas.

LSTM solves the problem of vanishing and exploding gradients which prevents handling long-term dependencies. It uses memory cell to store information at each time step, and uses gates to control the flow of info thru the network. Specifically, the input gate protects the current step from irrelevant inputs. The output gate prevents the current step from passing irrelevant outputs to later steps. The forget gate limits info passed from one cell to the next. Another variant that solves the same problem is GRU.

Question 18

They're very fast, unlikely to overfit, and, if the features are independent, they can perform very well. They also don't require any hyperparameter tuning. Neural networks, on the other hand, can be slow and are very likely to overfit. They're also conceptually simple and we can prove things about their performance, which is much harder with neural networks. (I'm giving lots of possible things they could say.) However, in text, the features are unlikely to be independent.

P("good" | "ham", "cheddar", "mayo") = 0.8*0.8*(1-0.7)*(0-0.4) / P(sandwich) = .1152 / P(sandwich)

P("bad" | "ham", "cheddar", "mayo") = 0.2*0.2*0.7*0.4 / P(sandwich) = .0112 / P(sandwich)

Since we only have two classes, we normalize by the sum of the probabilities .1152 and .0112 and get that the probability the sandwich is good is 91.1% and that the probability it is bad is 8.9%. The assumption made here is that the influence of each ingredient on the tastiness of the sandwich is independent of what other ingredients are present, which is not true.

Question 19

What is BERT? What is the key innovation? Please give two examples of using BERT for NLP tasks?

BERT (Bidirectional Encoder Representations from Transformers) is a recent paper published by researchers at Google AI Language. It has caused a stir in the Machine Learning community by presenting state-of-the-art results in a wide variety of NLP tasks, including Question Answering (SQuAD v1.1), Natural Language Inference (MNLI), and others.

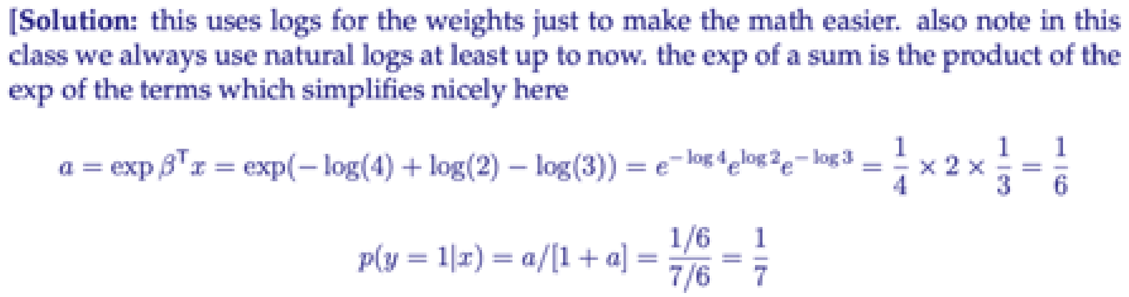
BERT (Bidirectional Encoder Representations from Transformers) is an open-sourced NLP pre-training model developed by researchers at Google in 2018. A direct descendant to GPT (Generalized Language Models), BERT has outperformed several models in NLP and provided top results in Question Answering (SQuAD v1.1), Natural Language Inference (MNLI), and other frameworks.

BERT's key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling. This is in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training. As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once. Therefore it is considered bidirectional, though it would be more accurate to say that it's non-directional. This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word).

1. Classification tasks such as sentiment analysis are done similarly to Next Sentence classification, by adding a classification layer on top of the Transformer output for the [CLS] token.
2. In Question Answering tasks (e.g. SQuAD v1.1), the software receives a question regarding a text sequence and is required to mark the answer in the sequence. Using BERT, a Q&A model can be trained by learning two extra vectors that mark the beginning and the end of the answer.
3. In Named Entity Recognition (NER), the software receives a text sequence and is required to mark the various types of entities (Person, Organization, Date, etc) that appear in the text. Using BERT, a NER model can be trained by feeding the output vector of each token into a classification layer that predicts the NER label.

Question 20

Consider a logistic regression model with weights β = (−ln(4), ln(2), −ln(3)). A given document has feature vector x = (1, 1, 1). Now, please provide your answer in the form of a fraction a/b.



[**Solution:** this uses logs for the weights just to make the math easier. also note in this class we always use natural logs at least up to now. the exp of a sum is the product of the exp of the terms which simplifies nicely here

$$a = \exp \beta^{\mathsf{T}} x = \exp(-\log(4) + \log(2) - \log(3)) = e^{-\log 4}e^{\log 2}e^{-\log 3} = \frac{1}{4} \times 2 \times \frac{1}{3} = \frac{1}{6}$$

$$p(y = 1|x) = a/[1 + a] = \frac{1/6}{7/6} = \frac{1}{7}$$

Question 21

What is the purpose of an activation function in a neural network? What would happen if we didn’t have one?

Without an activation function, a neural network would be like a linear regression model, which has limited capabilities. The activation function introduces non-linearity, so that the network is more powerful in that it is capable of representing any function. That is, neural networks are Universal Function Approximators.

Which of the following is true of sigmoid?
Sigmoid overcomes the vanishing gradient problem.
Optimization is easier in sigmoid than in tanh.

The resulting outputs are in the range [0,1].

Sigmoid is preferred to ReLU.

Question 22

What does this figure represent?

a. neural attention
b. gated recurrent unit (GRU)
c. **long short-term memory network (LSTM)**
d. tree neural network
e. support vector machine

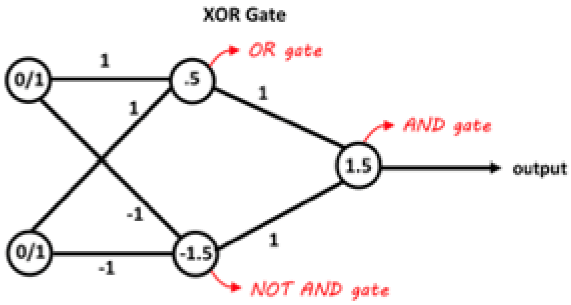
Question 23

What are the names of the four functions below?

Question 24

Draw a neural network with a hidden layer that can compute X XOR Y for the Boolean variables X and Y. Show all biases and other weights and explain why your network works. Note that there are many possible answers.

There are many possible answers. Here is one of them.



Question 25

What is the idea behind “retrofitting embeddings”, a method introduced by Manaal Faruqui (answer in one sentence)?

Making the automatically learned embeddings more consistent with manual ontologies such as WordNet.

Question 26

Give an example of a sentence that would be a valid Winograd schema. Do not use the examples shown in class.

No answer given.

Question 27

Which (zero, one, or more) of the following phrases are non-compositional?

a. white cloud
b. **red herring**
c. green tree
d. **black market**
e. blue box

Question 28

BERT is trained to predict 15% of the tokens. What does it do with these 15%? Pick all correct answers (zero, one, or more):

a. replaces the token with a mask
b. replaces the token with a random token
c. keeps the token unchanged
d. duplicates the token
e. writes the token backwards

Question 29

The formula below is used in what type of neural network:

a. Bidirectional LSTM
b. **Transformer**
c. Stacked LSTM
d. Pointer network
e. Convolutional neural network

Question 30

A. Semantic parsing conveys different information than semantic parsing; often, this information is *in addition* to the syntactic parse, such as in compositional semantics. In compositional semantics, we first parse the sentence syntactically, then associate some semantic information/representation with each word, and finally follow a parsing algorithm, combining the semantics of words and non-terminals recursively until reaching the root of the sentence. Semantic parsing is most helpful in situations where the syntactic information is not enough. One such example is the case of selectional restrictions; the verb “eats” must take an edible object as an argument; this is not imposed by the syntactic structure (in which the direct object simply must be a noun phrase), but it is imposed by the semantic selectional restrictions.

B. Semantics is often expressed with first order logic - *for all* children: like(children, toys). This is an objective and effective way to structurally represent the semantics of a sentence.

C. AMR uses a tree/graph structure that includes information about predicate-argument structure, named entity recognition, and coreference resolution. It is similar to (syntactic) dependency parsing, and makes use of the additional operations *swap*, *re-attach*, *replace head*, and *merge*.

D. Sentences are expressed as SQL queries, and a seq2seq network is trained on such examples. It involves less information about syntactic structure.

E. AMR currently lacks the ability to express quantifier scope, co-references *across* sentences, grammatical number/tense/aspect, some noun-noun or noun-adjective relations, etc. On the other hand, we can write SQL entries that express most such considerations; the effort to grow the scope of teh SQL-based approach is less. The only issue is that it requires a large amount of training data in a field where data is scarce.

Question 31

A. RNNs remember past information by keeping a hidden state and updating it at each timestep.

B.

   a. Short term memory means that RNNs cannot remember long-term dependencies. This is because the hidden states near the end of the sequence disproportionately encode more of the information at the end of the sequence and forget about information near the start of the sequence. Vanishing gradient problem means that as the gradient becomes extremely small and close to zero, which is caused by the unrolling of the network during backpropagation through time, where each derivative is < 1, and multiplying them leads to a small number. For exploding gradient, it's caused by a similar problem as that of vanishing gradient, except multiplying a whole lot of derivatives > 1.

   b. For short term memory, the impact is that the network can't capture long-term dependencies. For example, in predicting the next word of a sentence, the network may not correctly identify the subject of a verb and thus give the right verb form, if there's a long separation between the subject and the verb, e.g. due to a long relative clause. For vanishing gradient, the impact is that the network would stop learning. For example, in machine translation, while training on a very long sequence, the network would stop updating its weights. For exploding gradient, the impact is that the network has very unstabale performance, with large swings in its weight values.

C. LSTMs solve these issues with regulating the cell state that transfer relative information down the network layers processing sequential data. LSTMs regulate the cell state through gates, which are composed out of a sigmoid neural net layer and a pointwise multiplication operation. The sigmoid layer outputs numbers between zero and one, describing how much of each component should be let through. A value of zero means “let nothing through,” while a value of one means “let everything through”. A LSTM has three of these gates: the Forget gate decides what is relevant to keep from prior steps; the input gate decides what information is relevant to add from the current step; the output gate determines what the next hidden state should be. LSTMs' architecture therefore enforces constant error through internal states of special units. This solves the vanishing and exploding gradient problem of RNNs, as well as retaining long-term dependencies.

D. GRU is different because it combines the forget and input gates of lSTM into a single "update gate". It also merges the cell state and hidden state.

E. Bidirectionality is implemented by concatenating two hidden states at each timestep, one hidden state generated by scanning data left to right, and the second by scanning right to left. Its advantage is it encodes contextual information both to the left and to the right of the current time step, and more context is helpful for creating a better encoding or representation of a word / token, during tasks such as language modeling or part of speech tagging.

F. Bidirectionality can be applied to the encoder but not the decoder. For the encoder, in tasks such as machine translation, having hidden states generated from both scanning left to right and right to left helps encoding contextual information both to the left and to the right of an input token at each timestep, helping to create a more contextualized and meaningful final context vector for feeding into the decoder. However, the decoder can't use bidirectionality since it can't know what is coming before it decodes the hidden state (i.e. right to left scanning is impossible).

Question 32

1) BERT’s key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling.

This is in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training. As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once. Therefore it is considered bidirectional, though it would be more accurate to say that it’s non-directional. This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word).

2)

1. Classification tasks such as sentiment analysis are done similarly to Next Sentence classification, by adding a classification layer on top of the Transformer output for the [CLS] token.
2. In Question Answering tasks (e.g. SQuAD v1.1), the software receives a question regarding a text sequence and is required to mark the answer in the sequence. Using BERT, a Q&A model can be trained by learning two extra vectors that mark the beginning and the end of the answer.
3. In Named Entity Recognition (NER), the software receives a text sequence and is required to mark the various types of entities (Person, Organization, Date, etc) that appear in the text. Using BERT, a NER model can be trained by feeding the output vector of each token into a classification layer that predicts the NER label.

3)
The General Language Understanding Evaluation benchmark (GLUE) is a collection of datasets used for training, evaluating, and analyzing NLP models relative to one another, with the goal of driving "research in the development of general and robust natural language understanding systems." The collection consists of nine "difficult and diverse" task datasets designed to test a model's language understanding, and is crucial to understanding how transfer learning models like BERT are evaluated.

Question 33

A. Recursive neural networks can be thought of as generalisations of recurrent NNs.
B. You'd use LSTMs or GRUs to try and fix the vanishing gradient problem.
C. You'd want to use a recursive neural network.
D. You'd want to use a recurrent neural network.
E. You unroll the RNN with respect to time and then do standard backprop.

Question 34

No answer given.

Question 35

Part A

P(line|e) = (1+1) / (7+5) = 2/12 = 1/6
P(flute|e) = (0+1) / (7+5) = 1/12
P(jazz|e) = (0+1) / (7+5) = 1/12
P(line|f) = (1+1) / (3+5) = 2/8 = 1/4
P(flute|f) = (1+1) / (3+5) = 2/8 = 1/4
P(jazz|f) = (1+1) / (3+5) = 2/8 = 1/4

Part B

$P(e|d5) \propto 2/3 * 1/6 * (1/12)^2 * 1/14 \approx 0.00006$
$P(f|d5)$ $\propto 1/3 * 1/4 * (1/4)^2 * 1/4 \approx 0.001$

Question 36

Use the following documents for this problem, where the frequency of word appearing (and not just the word's presence) in a document matters.

D1 = "cat, dog, fox"
D2 = "fish, tiger, cat"
D3 = "cat, fox, dog"
D4 = "fish, cat, fish"

You may find it helpful to transform the documents into frequency vectors using the table below:

	fish	cat	fox	tiger	dog
D1					
D2					
D3					
D4					

What is the Jaccard Similarity between D1 and D2? 0.2

What is the Euclidean Distance between D3 and D4?

Suppose we decide to use Cosine Similarity. Which Document is most similar to D2? D4

What are the ranges (in general, not for this particular problem) of the above similarity and distance functions: Jaccard Similarity, Euclidean Distance and Cosine Similarity? You may assume that all document vectors only consist of non-negative components.

Jaccard: [0, 1]
Cosine: [0, 1] (for vectors with non-negative values)
Euclidean: [0, Infinity)

Question 37

No solution given.

Question 38

Assume a PCFG trained on the Wall Street Journal portion of the Penn Treebank. Which of the following inequalities is/are likely to be accurate? Pick 0, 1, or more answers.

>

Question 39

Which one of the following can be considered as a "universal function approximator"?

a. **a two-layer neural network**
b. a hidden markov model
c. a push-down automaton
d. a finite-state automaton

Question 40

What is the difference between the Viterbi algorithm and the Forward algorithm for POS tagging? Give pseudo-code for both and also explain the difference in plain English.

The Viterbi algorithm is used when you are given a sequence of symbols and a model and want to find the most likely sequence of states that produced the sequence. The forward algorithm is used when you are given a model structure and a set of sequences and want to find the model that best fits the data.

https://en.wikipedia.org/wiki/Viterbi_algorithm#Pseudocode
https://en.wikipedia.org/wiki/Forward_algorithm#Pseudocode

Question 41

Compute the derivative of the logistic function:
https://en.wikipedia.org/wiki/Logistic_function#Derivative

Question 42

S5: 2*1+3*1 = 5; 2*1 = 2

Question 43

choose the one with the probability |c)c) the largest, which is “dear”

Question 44

a. LCS (lowest common subsumer) the most specific concept which is an ancestor of both A and B. A mention of concepts, wordnet or an example is required for full credit.

b. confusion matrix for classification is a table which specifies predicted values vs actual values for binary classification and is useful for calculating statistics such as recall and precision; shows true/false positives and negatives

c. the principle of semantic compositionality – meaning of a whole is a function of the meanings of simpler parts and the way those parts are put together.

d. negative sampling (for word embeddings) a way of approximating the softmax by using a small number of randomly selected contexts for the denominator for efficiency in training word embeddings.

e. backpropagation over time for RNN – a method used in gradient descent for recurrent networks. It begins by unfolding the network over time and then applying backprogagation to find the gradient of the cost with respect to all parameters

f. HMM trellis represents the possible state sequences

g. softmax function takes in an input vector of real numbers and normalizes it into a probability distribution

h. horizontal markovization within lexicalization, label with the left siblings in your immediate tree. This takes into context where you are in your local tree structure.

i. labeled dependency accuracy takes into account heads and labels when computing accuracy

Question 45

A shift-reduce parser scans and parses input text in a single forward pass, building up a parse tree incrementally from left to right, accumulating a list of subtrees of the text that have already been parsed. Shift adds a single-node parse tree while reduce applies a completed grammar rule to recently parsed trees and joins them.

Question 46

$w(i + 1) = w(i) + y(i)x(i)$

Question 47

at each step, the CKY chart can only combine two items into one

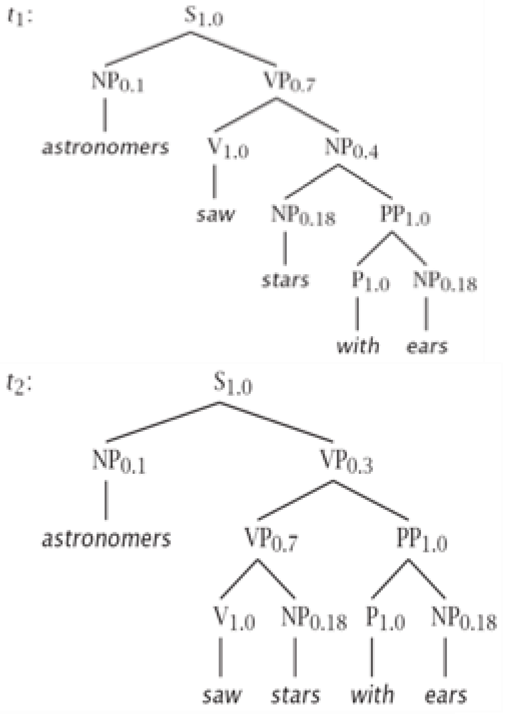
Question 48

No answer given.

Question 49

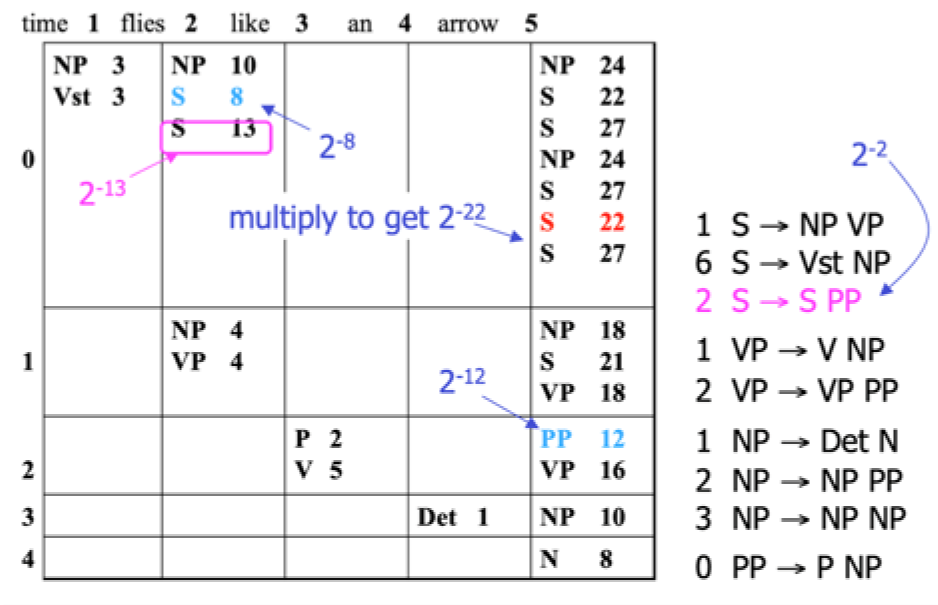
No answer given.

Question 50



$$\begin{aligned} P(t_1) &= 1.0 \times 0.1 \times 0.7 \times 1.0 \times 0.4 \\ &\quad \times 0.18 \times 1.0 \times 1.0 \times 0.18 \\ &= 0.0009072 \\ P(t_2) &= 1.0 \times 0.1 \times 0.3 \times 0.7 \times 1.0 \\ &\quad \times 0.18 \times 1.0 \times 1.0 \times 0.18 \\ &= 0.0006804 \end{aligned}$$

Question 51



	time 1	flies 2	like 3	an 4	arrow 5
0	NP 3 Vst 3	NP 10 S 8 S 13			NP 24 S 22 S 27 NP 24 S 27 S 22 S 27
1		NP 4 VP 4			NP 18 S 21 VP 18
2			P 2 V 5		PP 12 VP 16
3				Det 1	NP 10
4					N 8

1 S → NP VP
6 S → Vst NP
2 S → S PP

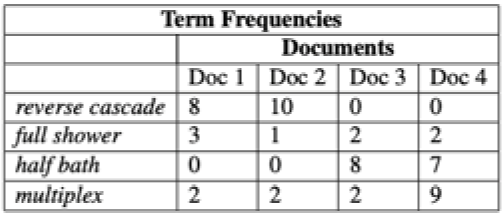
1 VP → V NP
2 VP → VP PP

1 NP → Det N
2 NP → NP PP
3 NP → NP NP

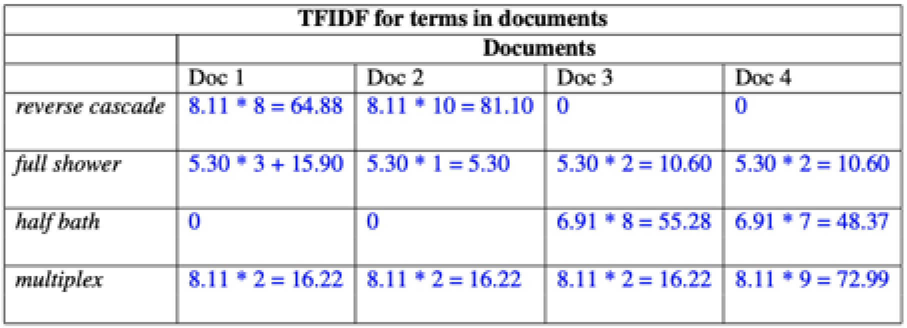
0 PP → P NP

Question 52

- **reverse cascade**: 3 IDF = log(10000/3) ≈ 8.11
- **full shower**: 50 IDF = log(10000/50) ≈ 5.30
- **half bath**: 10 IDF = log(10000/10) ≈ 6.91
- **multiplex**: 3 IDF = log(10000/3) ≈ 8.11



Term Frequencies				
	Documents			
	Doc 1	Doc 2	Doc 3	Doc 4
reverse cascade	8	10	0	0
full shower	3	1	2	2
half bath	0	0	8	7
multiplex	2	2	2	9



TFIDF for terms in documents				
	Documents			
	Doc 1	Doc 2	Doc 3	Doc 4
reverse cascade	8.11 * 8 = 64.88	8.11 * 10 = 81.10	0	0
full shower	5.30 * 3 + 15.90	5.30 * 1 = 5.30	5.30 * 2 = 10.60	5.30 * 2 = 10.60
half bath	0	0	6.91 * 8 = 55.28	6.91 * 7 = 48.37
multiplex	8.11 * 2 = 16.22	8.11 * 2 = 16.22	8.11 * 2 = 16.22	8.11 * 9 = 72.99

Question 53

Ex: John needs the deal. (Answers will vary)

Question 54

Consider the noun-noun phrases such as "cat food", "baby goat", "attorney general". What is/are some reasonable CCG parses for these:

(a) N/N N -> NP
(b) N N\N -> NP
(c) N/N N/N -> NP
(d) both a and b

Question 55

Consider the sentence "Bill drives a Honda from Albany to New York." Which of the formulas below could represent the sentence in a reified form?

(a) ∃ w,x,y: Driving(Bill, w,x,y)
(b) ∃ z: Driving(Bill, Honda, Albany, NewYork)
(c) ∃ w,x,y,z: Driving(Bill, Honda, Albany, NewYork)
(d) ∃ w,x,y,z: Driving(w,x,y,z)
(e) None of the above.

Question 56

Represent the following sentence "No person is immortal." in First-Order Predicate Calculus (FOPC):

(a) ¬∃ x: Person(x) ∧ Mortal(x)
(b) ¬∃ x: ¬Mortal(x)
(c) ∃ x: Person(x) ∧ Mortal(x)
(d) ¬∃ x: Person(x) ∧ ¬Mortal(x)
(e) ¬∃ x: Mortal(x)

Question 57

Adverbs are used to specify all of the following EXCEPT:
(a) place
(b) time
(c) manner
(d) degree
(e) agent

Question 58

The hypernym and hyponym relations in WordNet hold between which of the following notions (pick one):

(a) words
(b) lemmas
(c) stems
(d) synsets

Question 59

The dependency representation of the sentence "Jane has a cat" is the following (with the arrows pointing from parent to child node):

(a) has -> cat, has -> Jane, cat -> a
(b) cat -> has, Jane -> cat, a -> cat
(c) has -> Jane, Jane -> cat, cat -> has
(d) Jane -> has, has -> a, a -> cat
(e) has -> Jane, Jane -> cat, a -> cat

Question 60

What is the Levenshtein edit distance between "apples" and "pears"? Assume the following costs: insertion=1, deletion=1, substitution=1.

(a) 2
(b) 5
(c) 4
(d) 1
(e) 3

Question 61

Which of the following statements about syntactic constituents is false?

(a) Constituents are non-crossing.
(b) Each word is a constituent.
(c) If two constituents share one word, then one of them must completely contain the other.
(d) Constituents are continuous.
(e) Constituents cannot be nested.

Question 62

Assuming that exactly two constituents get combined at each iteration, a sequence of three nouns can be parenthesized in two different ways:

(a(bc)) and ((ab)c).

A sequence of four nouns can be parenthesized in five different ways:

((ab)c)d (a(bc))d (ab)(cd) a((bc)d) a(b(cd)).

In how many ways can a sequence of five nouns (or, alternatively, an adjective followed by four nouns) be parsed?

(a) 28
(b) 8
(c) 14
(d) 16
(e) 10

Question 63

Which of the following statements is true?

The operator ">" here means "strictly more expressive (powerful) than".

CCG = Combinatory Categorial Grammar, CSG = Context Sensitive Grammar, CFG = Context Free Grammar, TAG = Tree Adjoining Grammar, TSG = Tree Substitution Grammar.

(a) CSG > CFG > TAG
(b) CSG > TSG > CFG
(c) CSG > CCG > CFG
(d) CCG > CSG > CFG
(e) CSG > CFG > CCG

Question 64

Consider the corpus:

cat cat cat dog dog rat rat bat bat bat bat bat bat fox

Using "Add One" Laplacian smoothing, what is the estimate for P(rat)?

(a) 2/14
(b) 3/14
(c) 3/18

(d) 3/16
(e) 3/19

Question 65

No answer given.

Question 66

Which of the following statements about evaluation of relation extraction is *not* correct?

(a) Precision = #correctly extracted relations / #all extracted relations
(b) F1 = harmonic mean of Precision and Recall
(c) Recall = #correctly extracted relations / #all existing relations
(d) F1 = arithmetic mean (average) of Precision and Recall
(e) F1 = 2 x Precision x Recall/(Precision+Recall)

Question 67

Which of the following part of speech categories is open class?

(a) adjectives
(b) prepositions
(c) interjections
(d) articles
(e) conjunctions

Question 68

What does this logical expression mean in English?

$\exists$ e: Arriving(e) $\wedge$ Arriver(e, Speaker) $\wedge$ Destination(e, NewYork) $\wedge$ IntervalOf(e,i) $\wedge$ EndPoint(i,p) $\wedge$ Precedes(p, Now)

(a) I am arriving in New York
(b) He is arriving in New York
(c) I will arrive in New York
(d) I arrived in New York
(e) She will arrive in New York

Question 69

The CKY (Cocke-Kasami-Younger) parsing algorithm only works when...

(a) the grammar only includes a single production for any non-terminal.
(b) the grammar has been converted to Chomsky Normal form.
(c) the input sentence has exactly one verb.
(d) the input sentence is in English.
(e) the input sentence is not ambiguous.

Question 70

In the absence of any other relevant information, how should an out of vocabulary (OOV) word be tagged?

(a) determiner
(b) adverb
(c) noun
(d) verb
(e) adjective

Question 71

The sentence "Stolen painting found by tree" exhibits what phenomenon:

(a) prepositional phrase attachment ambiguity
(b) coordinating conjunction attachment ambiguity
(c) syntactic ambiguity
(d) all of the above
(e) none of the above

Question 72

What part of speech is *least likely* after an article?

(a) noun
(b) adjective
(c) verb
(d) numeral

Question 73

In the Bayes formula: $P(H|E) = P(E|H)P(H)/P(E)$, what do "H" and "E" stand for?

(a) H=hypothesis, E=evidence
(b) H=hyperparameter, E=evidence
(c) H=hyperparameter, E=estimate
(d) H=hypothesis, E=estimate
(e) none of the above

Question 74

In a large corpus, the frequencies of the three most frequent words are approximately 10%, X%, and 3%. What is the value of X, assuming a Zipfian distribution?

(a) 9
(b) 7
(c) 6
(d) 5
(e) 4

Question 75

In language modeling, the "add-1" method is an example of:

(a) linear interpolation
(b) backoff
(c) smoothing
(d) caching
(e) hypothesis testing

Question 76

An experiment was done to measure the perplexity of unigram, bigram, and trigram models on a news corpus corpus.

Which of the following sets of numbers makes the most sense?

(a) unigram 1000, bigram 200, trigram 100
(b) unigram 100, bigram 100, trigram 100
(c) unigram 100, bigram 200, trigram 1000
(d) unigram 100, bigram 0, trigram 0

Question 77

For a sentence with n words, the maximum number of boxes in the CKY table that can be nonempty is:

(a) n*n*n
(b) n
(c) n log n
(d) n*(n-1)/2
(e) n*(n+1)/2

Question 78

Which of the following sentences exemplifies "type coercion" in sentence parsing:

(a) I saw a cat in the park.
(b) I slept.
(c) I had a tea in the morning.
(d) I gave Mary a pretzel.
(e) Get out!

Question 79

Which of the following sentences is non-projective:

(a) The non-callable issue, which can be put back to the company in 1999, was priced at 99 basis points above the Treasury's 10-year note.
(b) John saw a dog yesterday which was a Yorkshire Terrier.
(c) Price details weren't immediately available.
(d) The collateral is being sold by a thrift institution.
(e) Ms. Haag plays Elianti.

Question 80

Which of the following relation(s) is/are symmetric:

(a) brother(X,Y)
(b) sister(X,Y)
(c) mother(X,Y)
(d) both a. and b. above
(e) none of the above

Question 81

Which of the following propositional logic statements is always true? The symbol "==" here is used to express equivalence.

(a) NOT (A AND B) == (NOT A) OR (NOT B)
(b) NOT (A OR B) == (NOT A) AND (NOT B)
(c) NOT A == NOT B
(d) A == NOT B
(e) more than one of the above

Question 82

Which of the following is true:

(a) English is an SOV language, Japanese is an SVO language
(b) English is an SOV language, Japanese is an SOV language
(c) English is an SVO language, Japanese is an SVO language
(d) English is an SVO language, Japanese is an SOV language
(e) English is an SOV language, Japanese is an VSO language

Question 83

The BLEU evaluation metric is essentially:

(a) n-gram recall with a penalty for brevity
(b) n-gram recall with a penalty for excess length
(c) n-gram precision with a penalty for brevity
(d) n-gram precision with a penalty for excess length
(e) none of the above