CSCI-GA-2590 Fall 2020 Midterm

Name and section: _____

# 1 True or False

1. (1 point) tanh can be used as an activation function in neural networks.
   ◯ True ◯ False
2. (1 point) Hinge loss cannot be used with recurrent neural networks.
   ◯ True ◯ False
3. (1 point) Logistic regression is a discriminative model.
   ◯ True ◯ False
4. (1 point) SGD can be applied to only convex functions.
   ◯ True ◯ False
5. (1 point) We can look at unlabeled test data during model development as long as the model does not have access to the test data.
   ◯ True ◯ False
6. (1 point) Higher held-out perplexity means a worse language model.
   ◯ True ◯ False
7. (1 point) Multinomial Naive Bayes models can be used to generate text.
   ◯ True ◯ False
8. (1 point) Conditional random fields are probabilistic models.
   ◯ True ◯ False
9. (1 point) The bag-of-words representation of a sentence is a vector of dimension $d$ where $d$ is the number of unique words in the sentence.
   ◯ True ◯ False
10. (1 point) Recurrent neural network language models use longer context than typical n-gram language models.
    ◯ True ◯ False

## 2 Multiple Choices

Note: There can be more than one correct answers; select all that apply! No partial credits: all correct answers must be checked.

1. (3 points) Consider a logistic regression model for text classification. If we find the model is overfitting, what should we try to fix it?

   ◯ Use higer-order n-gram features. ◯ Use $L_2$ regularization.
   ◯ Increase the amount of training data. ◯ Increase the number of training epochs.

2. (3 points) Suppose you are building a fake news detector using binary text classification (+: fake, −: real). You do not want to risk missing any potential fake news. The classifier should have

   ◯ high precision ◯ low precision ◯ high recall ◯ low recall

3. (3 points) You are training a neural language model on a corpus with a vocabulary size of 1000. Suppose you randomly initialized the neural network and printed the training loss in each step. Which of the following is most likely to be the loss you see in the first print? (Assume we're using natural log.)

   ◯ 0.1 ◯ 6.9 ◯ 3.8 ◯ 1000

4. Consider a toy training set (tokenized by white spaces) of sentiment classification:
   +1: *the thriller is captivating !*
   −1: *Jennifer was bored by it .*

   (a) (2 points) Which of the following feature templates can achieve zero training error?
   ◯ The first punctuation in the sentence.
   ◯ The number of tokens in the sentence.
   ◯ Adjectives in the sentence.
   ◯ The first word of the sentence.

   (b) (1 point) Which of the feature templates is more likely to give low test error?
   ◯ The first punctuation in the sentence.
   ◯ The number of tokens in the sentence.
   ◯ Adjectives in the sentence.
   ◯ The first word of the sentence.

5. (3 points) Consider building a POS tagger by predicting the tag for each word independently. Suppose we have two feature templates: $x_i$ and $y_i, y_{i-1}$. Given the following training set:
   The/DT old/ADJ man/VB the/DT boat/NN
   The/DT old/ADJ man/NN bought/VB the/DT boat/NN
   What's the best training error we can achieve?

   ◯ 3/11 ◯ 2/11 ◯ 1/11 ◯ 0

6. (3 points) Which of the following will increase the training perplexity of a language model?

   ◯ Increase $n$ in a $n$-gram language model.
   ◯ Increase the number of hidden units in a feed-forward language model.
   ◯ Use add-one smoothing.
   ◯ Remove words with frequency smaller than five from the vocabulary (i.e. replace them with a special symbol `UNK`).

7. (3 points) Which of the following is true about SGD?

   ◯ It makes more updates to the parameters than GD in the same amount of time.
   ◯ Each update does not necessarily move along the gradient of the objective.
   ◯ It may oscillate around a local optimum if the step size is fixed and too large.
   ◯ We can decide when to stop by checking the gradient norm.

8. (3 points) Suppose we have collected menus from some restaurants in NYC and find that there are 50 unique dishes out of 1000 items on the menus. Out of these, 5 dishes are offered by only a single restaurant. Using Good-Turing smoothing, what is the probability that we will see new dishes in future?

   ◯ 0.001 ◯ 0.005 ◯ 0.05 ◯ 0.1

9. (3 points) Which of the following models have convex loss functions (w.r.t. model parameters)?

   ◯ Skip-gram model
   ◯ Logistic regression with $L_2$ regularization
   ◯ Two-layer feed-forward neural network with tanh activation functions
   ◯ N-gram language model

10. (3 points) Which of the following have the largest impact on the training time of a CRF model for sequence labeling?

    ◯ Graph structure. ◯ Number of feature templates. ◯ Input length.
    ◯ Vocabulary size.

# 3 Short Questions

1. **Tricky examples.** In all questions, please construct **grammatical and sensical English** sentences. An example of grammatical but *nonsensical* sentence is "Colorless green ideas sleep furiously".

   (a) (1 point) Suppose we have trained a language model on New York Times news articles. Write a sentence that is likely to have much higher perplexity than the training perplexity of the model.

**Solution:**

(b) (2 points) Write two sentences with the same BoW representation but different meanings.

**Solution:**

(c) (3 points) Given the vocabulary $\{\text{the}, \text{movie}, \text{is}, \text{not}, \text{good}, \text{bad}\}$, construct a toy traing set for binary sentiment classification such that a multinomial Naive Bayes model (each feature is a unigram) cannot achieve a training error better than random guess.

**Solution:**

(d) (2 points) Consider the task of POS tagging. Construct a toy training set such that a bidirectional RNN model would achieve lower training error than a unidirectional (left-to-right) RNN model. Use the following format: The/DT fox/NN ran/VB.

**Solution:**

2. **Viterbi decoding.** Consider POS tagging with a tagset of size $m$ and a vocabulary of size $v$. Suppose a sentence has $n$ tokens. Given an HMM model:

$$p(y \mid x) = \prod_{i=1}^{n} p(x_i \mid y_i) p(y_i \mid y_{i-1}).$$

we can find the best tag sequence by Viterbi decoding.

(a) (1 point) What is the runtime of the Viterbi algorithm?

**Solution:**

(b) (2 points) Suppose each tag is only allowed to transit to $k (k \leq m)$ tags, i.e. all sequences that don't satisfy this condition have zero probability. We can modify the Viterbi algorithm to incorporate this constraint. What is the runtime of the modified algorithm?

**Solution:**

(c) (2 points) Suppose we switch to a model where tags are generated independently:

$$p(y \mid x) = \prod_{i=1}^{n} p(x_i \mid y_i) p(y_i).$$

We can improve the decoding runtime to

**Solution:**

3. **Bernoulli naive Bayes model.** In lecture 2 we talked about the multinomial naive Bayes model for text classification. Now consider a slightly different model. Let $x = (x_1, \ldots, x_v)$ where $v$ is the vocabulary size and $x_i$ is a binary indicator variable of whether the $i$-th word appears in the input sequence of words.

   (a) (1 point) Given the vocabulary {love, language, fun, is, the}, write the feature vector $x$ of the sentence "language is fun", assuming the index of words in the vocabulary is the same as their position in the list (i.e. "love" is the first word and "the" is the 5-th word).

   **Solution:**

   (b) (3 points) Learn a model with add-1 smoothing given the following word counts on the training examples:

| id | label | fun | love | boring | sleeping |
|---|---|---|---|---|---|
| 1 | +1 | 3 | 2 | 0 | 1 |
| 2 | +1 | 0 | 2 | 0 | 0 |
| 3 | -1 | 1 | 0 | 2 | 1 |
| 4 | -1 | 0 | 1 | 1 | 1 |

   What is the prediction for the sentence "the beginning is fun but it quickly became boring"? Note that you can ignore words not in the vocabulary ({fun, love, boring, sleeping}).

   **Solution:**

   (c) (2 points) Now learn a multinomial naive Bayes model using the same data with add-1 smoothing. Does its prediction agree with the Bernoulli naive Bayes model?

   **Solution:**

   (d) (2 points) Which model would you prefer in practice and why?

   **Solution:**