CS 475/675 Machine Learning: Homework 4

Due: Monday, November 7, 2022, 11:59 pm

50 Points Total Version 2.0

Make sure to read from start to finish before beginning the assignment. This assignment has undergone a major revision since the original release.

1 Introduction

This homework assignment has two parts:

1. **Analytical (30 points):** This part will ask you to consider questions related to the topics covered by recent lectures. These questions remain the same since the original release of the homework, except we have increased the point value to 30 points. Do not change the size of the answer boxes in the latex template. You may use either the original latex template or the updated template.
2. **Lab (20 points):** You will evaluate a given model and answer a series of questions in a Python notebook. This part of the homework has changed since the initial release.

2 Collaboration Policy

The course policy is that, unless otherwise specified, all work must be your own. See the course information Google document for more details.

For this assignment, you may work in groups of 1 to 3 students. You and your group will make a single submission to Gradescope. You will be able to indicate all students in your group when you submit. The entire group will receive the same grade, so please choose your group carefully.

You can only work in teams of one, two or three students; no more. Your group can include anyone from either section (01/02/03/04) or course (475 or 675) provided that all team members are enrolled in the class and taking it for credit (not audit). We highly recommend that you do every part of the assignment together instead of splitting it up. Our intention is to include questions on the exam that require an understanding of all parts of the homework. It is to your advantage to work together on all parts.

You can work on the same Overleaf document and think through the questions together. You probably want to work with the group for the semester (only for assignments where collaboration is allowed) but it is not a requirement. Please see the course policy document for an explanation of how late hours apply in a group.

3 Analytical

Please complete the Analytical portion of the homework by using the provided Latex Template. Do not change the size of the answer boxes in the latex template.

4 Lab

You will experiment with a decomposable attention model for Natural Language Inference (NLI) and examine the attention weights to understand how attention works. We will provide you with the code, a trained model (checkpoint) and data, which you will use to answer the questions described below.

We will consider the task of Natural language inference (NLI). In this NLI task, we want to infer the relationship between two sentences. Given a pair of sentences, hypothesis $\mathbf{a}$ of length l_a and premise $\mathbf{b}$ of length l_b, the model will predict whether a "hypothesis" is true (entailment), false (contradiction), or undetermined (neutral) given a "premise". We will use the Stanford NLI (SNLI) corpus, which contains pairs of sentences labeled with one of these three labels. For example, for the text "A man inspects the uniform of a figure in some East Asian country."" and the hypothesis "The man is sleeping" the correct label is "contradiction" (the man is not asleep.)

4.1 Model Implementation

We will experiment with the model described in this paper: `https://aclanthology.org/D16-1244.pdf`. We are providing you with an implementation of this model, and a checkpoint that contains a trained model (**ckpt20.pt**). Nevertheless, we encourage you to take some time to understand how the model works and how it uses attention.

Given hypothesis $\mathbf{a} = (a_1, ..., a_{l_a})$ and premise $\mathbf{b} = (b_1, ..., b_{l_b})$, we assume that each token $a_i, b_j \in \mathbb{R}^d$ is represented by a word embedding vector of dimension d. The data is a set of labeled pairs $\{\mathbf{a}^{(n)}, \mathbf{b}^{(n)}, \mathbf{y}^{(n)}\}_{n=1}^{N}$ where $\mathbf{y}^{(n)} = (y_1^{(n)}, y_2^{(n)}, y_3^{(n)})$ is an indicator vector encoding the label.

The model is a deep neural network composed of several layers that transforms the input (sentences) into the output (label). The central components of the model rely on attention. We will step through each set of layers of the network.

1) Embed. The first step is to convert the raw input text into word embeddings. Word embedding are vector representations of words in text. In this case, we'll utilize embeddings from a model pre-trained on a large corpora. Embeddings encode the meaning of words such that the words that are closer in the vector space are expected to be similar in meaning. We will use GloVe word embedding. For more information (optional) on GloVe: https://nlp.stanford.edu/projects/glove/.

2) Attend. Each of the inputs (word embeddings) are passed through a multi-layer Perceptron (MLP) F that transform the input into a new representation. F has two hidden layers and uses ReLU activation. A produce of the resulting representations is used to form unnormalized attention weights e_{ij} as follows:

$$e_{ij} = F(a_i)^T F(b_j). \tag{1}$$

Normalized attention weights are formed as follows:

$$\beta_i = \sum_{j=1}^{l_b} \frac{\exp(e_{ij})}{\sum_{k=1}^{l_b} \exp(e_{ik})} b_j \tag{2}$$

$$\alpha_j = \sum_{i=1}^{l_a} \frac{\exp(e_{ij})}{\sum_{k=1}^{l_a} \exp(e_{kj})} a_i. \tag{3}$$

3) Compare. The next step is to compare the aligned phrases $\{(a_i, \beta_i)\}_{i=1}^{l_a}$ and $\{(b_j, \alpha_j)\}_{j=1}^{l_b}$ using another MLP G:

$$\begin{aligned} \mathbf{v}_{1,i} &= G([a_i, \beta_i]) \quad \forall i \in [1, ..., l_a] \quad (4) \\ \mathbf{v}_{2,j} &= G([b_j, \alpha_j]) \quad \forall j \in [1, ..., l_b] \quad (5) \end{aligned}$$

where $[\cdot, \cdot]$ denotes concatenation in the argument of G.

4) Aggregate. We then first aggregate over the two sets of comparison vectors $\{\mathbf{v}_{1,i}\}_{i=1}^{l_a}$ and $\{\mathbf{v}_{2,j}\}_{i=1}^{l_b}$:

$$\mathbf{v}_1 = \sum_{i=1}^{l_a} \mathbf{v}_{1,i} \quad (6)$$

$$\mathbf{v}_2 = \sum_{j=1}^{l_b} \mathbf{v}_{2,j}. \quad (7)$$

5) Predict. The aggregated comparison vectors represent the interactions between the two sentences. These are given to H, a final stage formed of an MLP followed by a linear classifier:

$$\widehat{\mathbf{y}} = H([\mathbf{v}_1, \mathbf{v}_2]), \quad (8)$$

where $\widehat{\mathbf{y}} \in \mathbb{R}^3$ represents the predicted (unnormalized) scores for each class. The predicted class is given by $\widehat{y} = \text{argmax}_i \, \widehat{\mathbf{y}}_i$.

The model is trained to minimized cross-entropy loss, with dropout used to regularize the model parameters:

$$L = \frac{1}{N} \sum_{n=1}^{N} \sum_{c=1}^{3} y_c^{(n)} \log \frac{\exp(\widehat{y}_c)}{\sum_{c'=1}^{3} \exp(\widehat{y}_{c'})}. \quad (9)$$

Running the Model: While you could train a model from scratch, it takes some time (about 20 minutes on a good GPU). Instead, we provide you a checkpoint of a trained model which you can load and run in the provided notebook. This model has been trained for 20 epochs and has reasonable accuracy.

Even though we are giving you a trained model, we encourage you to use Google Colab or other GPUs for running the model.

4.2 Notebook

You will use the provided model code and checkpoint to complete the lab notebook. Your grade will be based on your answers to the questions below. Make sure your clearly answer each question, and mark those answers when you submit the PDF of the notebook to Gradescope.

Make sure you run the notebook with the correct versions of the libraries, as small differences may impact your results. The library version numbers are included in the beginning of the notebook.

You will begin the Lab by reading the model checkpoint, and then evaluating the trained model in a variety of settings. You should run the provided notebook and provide answers to each of the questions.

You will answer each of the following questions in the notebook. In each question, you will draw a figure that shows the attention weights e_{ij}. In each questions, your figure will be a matrix, where an entry for the ith row and jth column will visualize the weight e_{ij}. The first four questions you will use the provided trained model in the checkpoint. The last question you will train your own model as well.

Q1.1 (4 points): Draw a figure that shows the attention weights for a correctly classified example from test set for the entailment, contradiction and the neutral classes, respectively. Explain how the learned attention weights contribute to the correct predictions.

Q1.2 (4 points): Draw a figure that shows the attention weights for an incorrectly classified example from test set for the entailment, contradiction and the neutral classes, respectively. Describe the difference between the pattern in these attention weights with those of the correctly predicted examples above.

Q2.1 (2 points): Write your own example of an NLI instance (hypothesis and premise) such that the label is ambiguous (i.e. write a confusing/hard example!). Draw a figure of the attention weights and explain how the model's uncertainty is reflected in the weights.

Q2.2 (3 points): Make up two examples yourself, such that the two examples have similar words (but need not be identical) but the word order makes a difference in the label. Show the attention weights for each case. Compare the weights in each case and explain how they contribute to the model predictions.

Q3.1 (7 points): Construct a new model using the codes for building the model in the notebook. Call the **fit** function to start training the model from scratch. Train for 3 epochs. On a fast GPU, this will be 1 minute per epoch, but it may be slower on older GPUs. On CPU, it takes around 20 minutes per epoch. So we highly recommend you to run on colab. Select an example from the training set and plot the attention weights of the model after the third epoch versus those of the fully trained model. Draw a figure of each of the attention weights and describe the difference.

5 What to Submit

For this assignment you will submit the following items to Gradescope.

1. "Homework 4: Analytical" - a PDF of the Analytical homework based on the provided Latex template.
2. "Homework 4: Lab" A PDF of the Python Lab notebook with answers to the requested questions.

6 Questions?

Remember to submit questions about the assignment to Piazza: `https://piazza.com/class/l7542wgbgfu7a8`.