TTIC 31210:
NLP

Homework 2:
Attentions

Yiyang Ou

May 1, 2019

1 Word Averaging Classifier

1.1

See section 1.1 in jupyter notebook for the implementation, experiment and accuracy.

1.2 Analysis

See jupyter notebook for the top15,low15 norm words. the I notice words with high norms have stronger emotions, such as ”worst,unfunny”. They’re able to explain the sentiment of a sentence (e.g. we can often guess a sentence to be bad emotion if ”worst” appears), so it makes sense they have large l2 norms. (Note because we take squares of each coordinate when computing l2 norms, so embeddings with very negative and very positive values both have high norms. This explains both good and bad emotion words appear here).

Words with low norms have little information regarding the sentiment of a sentence. For example, we see many words that describe objects, people or places, like Yimou, oatmeal, etc. I think these words are related to the content of the movie, but not with the sentiment of the review. Hence, these words have low norm.

2 Attention-Weighted Word Averaging

2.1

See section 2.1 in jupyter notebook.

2.2 Analysis: Word Embeddings and the Attention Vector

See jupyter notebook for the words. Words with high similarity (so higher attention weight) are either with strong emotions like ”bad” (so the model gives higher attention to them when predicting the sentiment), or structurally important for determing the sentiment. For example, words like ”not” ”n’t”, when combined with other abjectives like ”good”, decide the emotions of a sentence. So these words also have high attention weights.

On the other hand, words with low attention weights are generally words that don’t contribute to sentiment. Unlike in 1.2, where we saw many words describing objects, humans, we see here many conjunctions and punctuations. These words offer little or no information regarding sentiment, so they have very small weights. Also, this might relate to people’s habits when writing reviews: they aren’t very careful with grammar, punctuations and conjunctions use, but tend to use words with strong emotions to express sentiments.

2.3 Analysis: Variance of Attentions

See jupyter notebook for the words. We observe all these words are highly used words, yet because of this , these words have many different usages under different context. Thus it makes sense that their attentions would change depending their actual usages in the sentence. More specifically, on one hand, they themselves don't have absolute sentiment (positive or negative) like ”worst”; on the other hand, they're not like words in 2.2, that they still are related with the sentiments like ”quality”, ”interest” but I would expect their relations to sentiment would vary depending on the context of the sentence (like ”good quality” vs ”bad quality”). This inherent variability (in other words, expressing different sentiments within different sentence context), combined with their popular appearances in the training set, probably explains their large variance.

3 Simple Self-Attention

See jupyter notebook. The result shows residual connection slightly improves the accuracy.

4 Enriching

Implementation and result, see jupyter notebook. Formulation: I notice the model tends to overfit the training data (like over 90 percent), so I think it might be because we're only making use of words information in our model and we know some words in test/dev didn't appear in training set. Thus I tried to put words position and sentence length into the model (which are probably generalizable to new data), and hope this could help us better predict sentiment with unseen data. Specifically, I added a 40-dim representation of the position of each words, and 10 dim representation of the whole sentence length, and concatenate them with the word embeddings of each embedding. I then used this concatenated embedding to calculate attention. And for attention, I just used the self attention in part3 with the residual connection. In the end, unfortunately, I only saw minimal improvement on test set. (81 percent $->$ 82 percent)