Review of Attention Based LSTM for Aspect-level Sentiment Classification

Jon Stewart

I reviewed a paper titled, *Attention Based LSTM for Aspect-level Sentiment Classification*. The researchers were Yequan Wang, Minlie Huang, Li Zhao, and Xiaoyan Zhu. The paper focused upon using LSTMs for the purposes of Sentiment Classification. The specific area they concentrated upon when there was more than one sentiment expressed in the same sentence. Their first example was a sentence such as “The appetizers are ok, but the service is slow.”. This sentence has two separate sentiments expressed. The sentiment expressed towards the appetizers would probably be classified as neutral, while the sentiment expressed regarding the service would probably be classified as negative.

Before describing their technique, the researchers talked about previous methods. The most described one they talk about is tree LSTMs. In this network structure, input sentences are first classified and branched to different LSTMs based upon aspects, and then classified, this is done in a recursive manner, hence why it is described as a tree. While this works well, it is problematic in several ways. First, if aspects are not correctly identified, then the actual classification of those aspects in terms of sentiment are very unlikely to be correctly identified. Secondly, networks based upon conditional branching can be difficult to build, as it requires dynamic graph generation, and due to variability in the number of aspects, can be potentially very memory intensive.

The researchers proposed to use an LSTM network with attention. Attention can be used in a number of ways for both recursive and feed forward networks. The base idea is that if two related neurons are both highly activated at the same time frequently, those activations are probably correlated in some way, even if their occurrences in the sentence are very different. Typically, this is done by some form of multiplying or adding activations together. Oftentimes, at least one of those activation functions being multiplied together will be a sigmoid, or other bounded function, which will act as a gate on the other neuron.

One unique aspect they took is to use vector embedding focused upon aspect. The method they used was the following. For each word processed, the LSTM layer produces a hidden layer of the size of the specified LSTM width by 1. Within the LSTM block, each of those hidden layers is also sent to a vector embedding which has been trained for aspect. The output of these embeddings are then sent to a layer to determine attention. Afterwards, both the sentiment and aspect vectors are used in the output. The aspect vectors are used to determine attention, and the LSTM outputs are then weighted by that attention vector, which is one of the outputs. The attention layer is concatenated onto the attention vectors. This lets the final decision layer decide how to treat the combination of the LSTM output and the aspect output, once each word is classified by both.

The authors tested out variations of this model, all by altering the specific LSTM blocks. For the initial embedding, a Glove embedding was used. Other than the different network designs, the same parameters (learning rate, hidden layer output size) was used throughout.The most basic version simply obtains the embedding for the word, then concatenates that to the initial input to the LSTM block. The second version takes the aspect

embedding after the regular LSTM block, but does not compute attention. Rather, it just concatenates the output of the aspect embedding and the LSTM output at the end of the LSTM block. The third version is the researcher's full version, where an attention layer is used. For the sake of making a simplistic example in order to test out their version of the model, the researchers only used one LSTM block throughout all variants of the model tested. Additionally, the researchers also tested it against a tree LSTM and several other variants, including a baseline LSTM.

Testing revealed that the version of the model that explicitly computes the aspect embedding after the initial LSTM model, and also uses attention, had superior results. The plain LSTM with only Glove embedding, unsurprisingly, had the worst results, as it is unable to separate aspects or use attention mechanisms. The researcher's version of the model showed superior performance, including having a better result than tree-LSTMs.

Conclusion

The researchers designed a workable model that incorporated attention mechanisms and specific aspect embedding in order to improve the ability of classifiers to model more than one sentiment expressed in a given sentence. This improved upon tree based LSTMs, and although not the main emphasis of the paper, was also more memory efficient. The researcher's design was interesting, as they added variations to the actual LSTM unit, rather than just change the overall network architecture for a given LSTM unit. The idea of adding attention, in particular, improved their results. One further consideration would be to change how the attention mechanism is used. In particular, gating is also an important part of attention. Oftentimes, this will use some bounded function, typically a sigmoid activation or similar, and multiply a pair of inputs. Then, if the bounded function or the other activation is near zero, the resulting output will only be weakly activated, while if both are high, the activation is stronger. This let's the network focus upon the most pertinent parts of the sentiment being expressed. Overall, I thought the article was well done, and both the concept and specific procedure used were well expressed throughout the paper.