Aspect-Term Sentiment Analysis on Multi Aspect Multi Sentiment (MAMS) Dataset Using Variation of AE-LSTM and BERT Models

Clifton Felix, Heinrich, Jason Ciu Putra Sung, Jonathan Kwok, Vishandi Rudy Keneta
National University of Singapore
Singapore, 117543

Abstract

As an extension to sentence-level sentiment analysis, aspect-based sentiment analysis (ABSA) aims to find the sentiments of various aspects available in a sentence. With the development of human linguistics, each sentence tends to have more than one aspects being analyzed, which explains the increase in ABSA's popularity and applications. A subtask of ABSA, aspect-term sentiment analysis (ATSA), assumes the "aspects" to be some "terms" within the sentence. This paper reviews the effectiveness of AE-LSTM-based and BERT-based models for ATSA using the Multi-Aspect Multi Sentiment (Jiang et al., 2019) data set.

1 Introduction

The main goal of ATSA is to extract the polarity (positive, neutral, or negative) of each aspect available in a sentence. For example (taken from the MAMS dataset), given the sentence "The decor is not special at all but their food and amazing prices make up for it." and aspects "decor", "food", and "prices", the goal is to assign "decor" to negative, "food" to positive, and "prices" to positive (Jiang et al., 2019). The model's input is the whole sentence and each aspect, and the output is one of polarity. As seen from the example, this task is more robust and flexible than the traditional sentence-level sentiment analysis. Also, it is more relevant as many sentences tend to have multiple aspects to analyze. As ATSA's popularity has kept increasing these past few years, many datasets have been published explicitly for the ABSA task. Some popular ones include SemEval-2016 (Pontiki et al., 2016), SemEval-2014 (Pontiki et al., 2016), and Multi-Aspect Multi Sentiment (MAMS) (Jiang et al., 2019). This paper will use the MAMS dataset in all our experiments[1].

[1]All the codes that we used in this paper can be accessed on https://github.com/vishandi/CS4248-Team13

We will use an LSTM-based model with Aspect Embedding (AE-LSTM) (Hochreiter and Schmidhuber, 1997) with one recurrent layer as a baseline. We will then try the attention-based LSTM with Aspect Embedding (ATAE-LSTM) (Wang et al., 2016), which performs decently for the SemEval-2014 dataset (Pontiki et al., 2016). Subsequently, we try BERT-based models (Devlin et al., 2019) that are known to perform very well in ATSA tasks.

2 Models

2.1 AE-LSTM

2.1.1 Overview

Recurrent neural networks, or RNNs, have been one of the most powerful models in Natural Language Processing due to their ability to use their feedback connections to keep recent input representations in the form of activations, or "short-term memory", instead of "long term memory" with slowly updated weights. However, as the length of input begins to grow longer, and the time lag between signal and inputs becomes larger, RNN does not perform as well since the calculated gradient can explode (Exploding Gradient Problem) or disappear (Vanishing Gradient Problem) (Hochreiter and Schmidhuber, 1997). The former may lead to the gradient descent diverging continually to a set of unstable weights, while the latter can cause it to stop converging at a sub-optimal result.

Therefore, Long Short-Term Memory (LSTM) model was discovered. Using an efficient gradient-based learning algorithm, the model enforces a constant error flow through internal parts of specific units so that it will not vanish. It then consequently will be able to bridge significant time gaps between signal and inputs, even in the case of long, noisy, and dense sequences, without losing the "short term" aspect of it (Hochreiter and Schmidhuber, 1997). And to make the best use of aspect information, we add an embedding vector

for each aspect to make it an LSTM model with Aspect Embedding (AE-LSTM).

But even with that, more is needed for the model to keep track of the essential aspects, as all input sequences are encoded in a single vector, risking information loss. Instead, we can add an attention layer after the LSTM layer to create an Attention-based LSTM with Aspect Embedding (ATAE-LSTM). This allows the model to capture which parts of the sentence are important when considering different aspects (Wang et al., 2016).

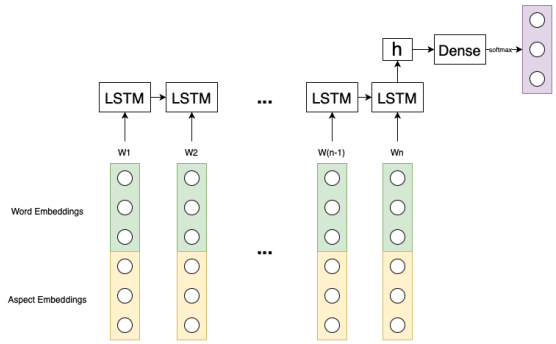

Figure 1: AE-LSTM

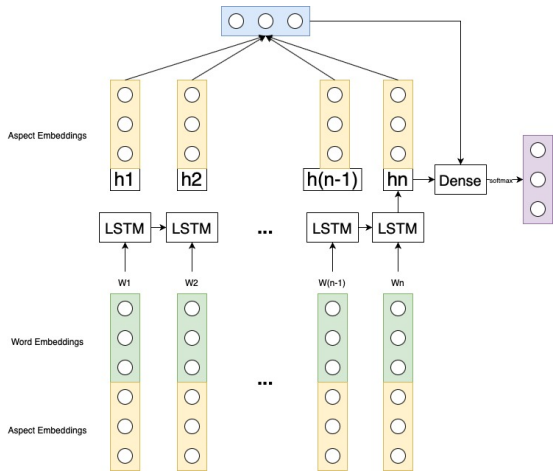

Figure 2: ATAE-LSTM

2.1.2 ATSA with AE-LSTM

In ATSA, it is a common practice that LSTM-based model are used as the baseline model, for example, in (Wang et al., 2016). This is because the model is not only straightforward but can also perform well on a long sequence of inputs due to its "Long Short-Term Memory" characteristics.

Given a sentence S and aspect term AT, the input to the model is simply the combination of concatenated embeddings of each word in S and AT, where the word embedding is constructed with the pre-trained Common Crawl 42 billion word vectors (Pennington et al., 2014).

2.2 BERT

2.2.1 Overview

While neural network models such as convolutional neural networks (CNN) or recurrent neural networks (RNN) have become popular for natural language processing, transformer models have surpassed them in performance for both language interpretation and natural language generation (Wolf et al., 2020). Unlike CNN or RNN, transformer models are able to solve sequence-to-sequence tasks while easily handling long-distance dependencies. They follow an encoder-decoder structure plus a self-attention mechanism without incorporating any recurrent or convolutional layer to generate an output.

In 2018, researchers at Google AI Language published a model called Bidirectional Encoder Representation of Transformer (BERT) which uses only the encoder part of transformers. BERT was pretrained on a large corpus of unlabeled text for two tasks: (1) Masked Language Modeling, where 15% of the tokens are masked and to be predicted based on the context, and (2) Next Sentence Prediction, where the model predicts whether the following sentence is probable given the first sentence.

To understand input correctly, BERT uses two special tokens $[CLS]$ and $[SEP]$. $[CLS]$ is a special classification token used to represent sentence-level classification, while $[SEP]$ token acts as a separator between 2 consecutive sentences in the same sequence inputs.

Lastly, BERT uses contextual word embedding to process input, which gives it an advantage over context-free models like Word2Vec. This helps BERT to learn information from both the left and right sides of a token's context during the training. With this contextual word embedding, BERT can differentiate the meaning of a word in two different contexts.

This pre-trained BERT model has shown strong performance on various downstream natural language processing tasks (Wang et al., 2018). The pre-trained model is then fine-tuned on the downstream tasks merely by adding only one final layer and training it for a few epochs to produce the results for the downstream tasks (Devlin et al., 2019).

2.2.2 ATSA with BERT

For downstream tasks such as ATSA, we construct auxiliary sentences with the aspect as the second sentence and transform the ATSA task into a sentence-pair classification task. This is effective in improving BERT model performance compared to single-sentence classification (Sun et al., 2019).

Given a sentence S and an aspect term AT, we construct the auxiliary sentence as

$$AS = \text{"}[CLS]\ S\ [SEP]\ AT\ [SEP]\text{"}$$

For each token $t_i \subset AS$, we construct the input representation as

$$t_i^{tok} + t_i^{seg} + t_i^{pos}$$

where $t_i^{tok}, t_i^{seg}, t_i^{pos}$ are the token, segment, and positional embeddings of t_i.

3 Experiments

3.1 Dataset

We are experimenting using the ATSA subsection of the MAMS dataset (Jiang et al., 2019). The dataset consists of customer reviews on restaurants. Each sentence in the dataset contains at least two sentiments with different polarities. The dataset are designed in this way so that neural networks can be generalized better (Jiang et al., 2019).

3.2 Setup

We are utilizing two main models in this experiment, AE-LSTM, and BERT. For the AE-LSTM model, we are experimenting with a vanilla AE-LSTM, and ATAE-LSTM (Wang et al., 2016). Following (Wang et al., 2016), the batch size is 25, the number of epochs is 20, and the dimension of word vectors, aspect embeddings, and the size of the hidden layer are all 300. We use Adam optimizer with a 3e-5 learning rate, Xavier uniform initializer, and Cross Entropy Loss. Paddings are also used in the input to make the size consistent.

As for BERT, we are experimenting with BERT-base, and its variants such as BERT-SPC (Devlin et al., 2019), and BERT-CapsNET inspired by the Huggingface pre-trained model (Wolf et al., 2020). The BERT model is trained with a preset input size of 768, batch size of 32, and a number of epochs of 5. We also used Adam as our optimizer with a 2e-5 learning rate.

3.3 Main Results

The results are shown in both Table 1 and Table 2. Between AE-LSTM-based and BERT-based models, we observe that the BERT model always performs better than AE-LSTM. This could be due to a few reasons. First, BERT is a bidirectional model, allowing BERT to understand a sentence holistically. Furthermore, the difference between BERT and AE-LSTM is that BERT utilized its attention mechanism (which is not based on AE-LSTM base attention) to extract contextual features from a sentence (Vaswani et al., 2017). Hence, it is not surprising that the BERT-based models outperformed any LSTM-based model significantly, not only with an accuracy difference of more than 15% but also with precision and recall score differences of more than 15%.

Model	Accuracy	F1 score
AE-LSTM	67.14%	65.78%
ATAE-LSTM	67.37%	65.71%
BERT	80.99%	80.04%
BERT-SPC	80.99%	80.46%
BERT-CapsNET	**83.48%**	**82.25%**

Table 1: Experimental results of Accuracy and F1 score with MAMS datasets using AE-LSTM and BERT.

Model	Precision	Recall
AE-LSTM	66.25%	65.43%
ATAE-LSTM	65.96%	65.52%
BERT	80.04%	79.67%
BERT-SPC	80.45%	80.55%
BERT-CapsNET	**81.94%**	**82.66%**

Table 2: Experimental results of precision and recall score with MAMS datasets using AE-LSTM and BERT.

For the LSTM-based models, we found that the ATAE-LSTM model only improved slightly from the AE-LSTM model (by 0.2% in accuracy). This slight improvement must be traded with the increase in training time by approximately four times longer. This could be due to the nature of the datasets (containing multiple aspects) which may not be well captured by the additional attention layer. We also discovered similar phenomenon with the BERT base model. Thus, we concluded that simply concatenating the original datasets with the aspect term is sufficient to get the most out of the model. Additionally, we do not implement any LSTM-only model without modifying the input. This is because (Wang et al., 2018) found it to have a poor performance.

For BERT-based models, each variant performed

over 80% in accuracy. However, similar to LSTM-based models, each variant only slightly improves each other performances by approximately 1%. The performances of BERT and BERT-SPC are comparable, as the only difference lies in the model architecture. While BERT is trained through the pre-trained model, BERT-SPC creates a fully-connected layer out of the pooling layer output by the pre-trained BERT. We found that adding another fully connected layer has an insignificant contribution to improving the model performances. As for BERT-CapsNet, having an extra capsule layer significantly improves the performance as it helps the model understand the aspect term's position in the sentence, which leads to better sentiment prediction.

3.4 Further Analysis

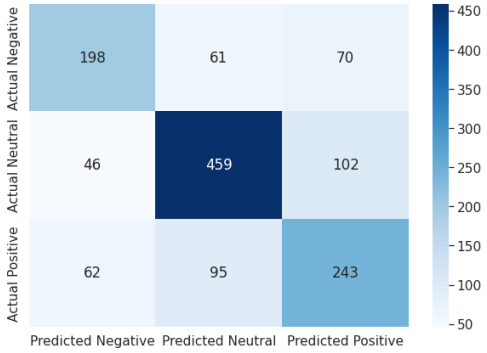

Figure 3: ATAE-LSTM Confusion Matrix

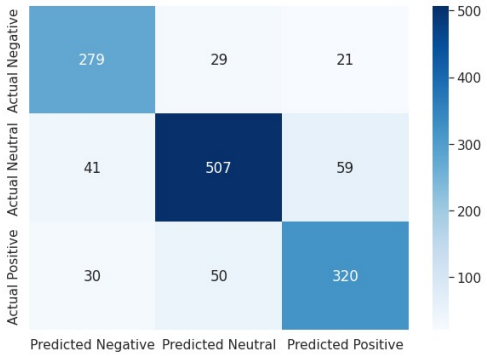

Figure 4: BERT-CapsNET Confusion Matrix

From the confusion matrices of the ATAE-LSTM and BERT-CapsNet models (refer to Figure 3 and Figure 4), we observed that BERT-CapsNet is superior to the ATAE-LSTM model.

In particular, due to the imbalanced label distribution in the training dataset in which there are more Neutral polarities than Negative and Positive polarities, ATAE-LSTM model precision for

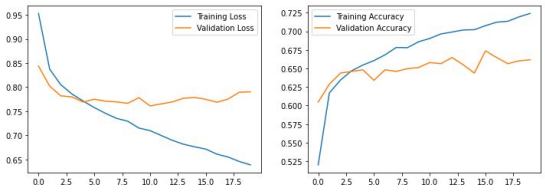

Figure 5: ATAE-LSTM Training Curve

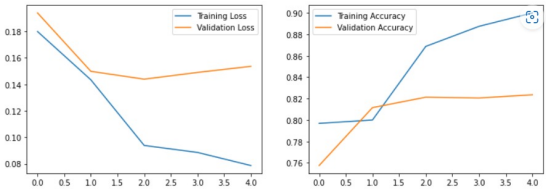

Figure 6: BERT-CapsNET Training Curve

Neutral polarity is the highest (0.76) compared to Positive and Negative polarities, 0.60 and 0.61 respectively.

On the other hand, the BERT-CapsNet model can generalize well enough that it is almost equally powerful in predicting all polarities; the precisions for Positive, Neutral, and Negative polarities in the BERT-CapsNet model are 0.85, 0.84, and 0.80.

Furthermore, from the train and validation loss and accuracy curves of the ATAE-LSTM and BERT-CapsNET models, we could see that at the last few epochs, the validation loss and accuracy were quite stagnant and did not improve that much anymore. Hence, training the models for a few more epochs may cause overfitting which leads to insignificant improvement to the validation accuracy/ loss

4 Conclusion and Future Work

After comparing some variations of LSTM-based models and BERT-based models, we found that the BERT-based models are superior to the LSTM-based models in all aspects: accuracy, F1 score, precision, and recall.

We acknowledge that we could probably further improve the result for each model by tuning the hyperparameters to get the most out of each architecture. However, due to the reason we discussed in the **3.4 Further Analysis** section, the comparison is still valid.

In the future, we will (1) do hyperparameter tuning to really get the best out of each model and (2) do a mathematical analysis to prove the superiority of the BERT-based models.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285, Hong Kong, China. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385, Minneapolis, Minnesota. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.