Aspect Based Sentiment Analysis (ABSA)

Franco Meng – Franco.meng2@gmail.com

May, 2024

Abstract

Aspect Based Sentiment Analysis (ABSA), is an advanced NLP task that aim to identify aspect terms (entities that expressed in text) present in the given review sentence, and predict the sentiment (e.g. positive, negative, neutral) associated with those aspects (Punitha, et al. 2023).

This is more challenging comparing with the traditional sentences / article-based sentiment detection, where the task aims to predict an overall global sentiment only. While the overall sentiment is valuable, in many real practises, a single sentence (e.g. restaurant review) usually includes different sentiments addressing different aspects. It would be significantly more beneficial, by gaining a deeper understanding of the sentiment polarities, specifically to the aspect addressed (e.g. service, food, price etc).

This report has explored different recurrent neural network architectures with novel variants. The models were trained with MAMS (Multi-Aspect Multi-Sentiment) dataset (restaurant reviews), where each review sentence has at least two aspects mentioned with different sentiment polarities. The models aim to predict the sentiment polarities based on the corresponding review sentences, as well as the given aspects.

The best performing model architect utilised the attention mechanism that figured out the relevant significant words, which helps to determine sentiment polarities related to the aspect expressed. The quantitative and qualitative analysis have proved the attention mechanisms' effectiveness, compared to the simple aspect embedding techniques. The best proposed model achieved 76.91% accuracy score in testing dataset, with respectively well performing F1-score in all three target sentiment classes: 76.75% (Negative), 80.71% (Neutral), and 70.83% (Positive).

Introduction

With the release of ChatGPT in 2022, large language model has drawn tremendously 'attention' around the globe in the artificial intelligence industrial space. Similarly, this year 2024, Tesla's Full Self-Driving (FSD) system made another historical burst of major advances in the automotive industry.

It is not a coincidence where these major breakthroughs happening around the same time in such a short timeframe, thanks to the growth of GPU, enabled the boost in processing powers, the more complex, computational demanding tasks can now be completed concurrently in smaller batches. Meanwhile, the cloud computing has also revolutionized organizations and companies to processing large scale data efficiently, and cost effectively.

However, there is another most important core reason shared behind: the Transformer model, introduced in the paper 'Attention is All you Need', the first sequence transduction model based entirely on attention, replacing the recurrent layers most commonly used in encoder-decoder architectures with multi-headed self-attention. (Vaswani, et al. 2017). My report aims to examine the effectiveness of the attention machenism, in seq-to-seq recurrent neural network models, by experimenting different approaches to embed the aspect information.

Commonly, the task of Aspect-based sentiment analysis can be divided into two subtasks: Aspect Term Extract (ATE) and Aspect Term Sentiment Classification (ATSC) (Punitha, et al. 2023). It varies depending on the different datasets and objectives.

Kiritcheko et al., 2014, the top team in SemEval 2014, utilised surface-form features (ngram), lexicon features and parsing features, with traditional supervised machine learning techniques Support Vector Machines (SVMs) to build the model. Their ablation experiments demonstrated the significant impact of the lexical resources employed in the system, with a boost of 8.78 percentage point in accuracy, the majority comes from the sentiment lexicons, specifically from the in-domain Yelp Restaurant Sentiment lexicons. (Kiritchenko, et al. 2014)

Tang (Tang, et al. 2016) , discovered that modelling sentence with standard LSTM does not perform well on this target-dependent task. 'I bought a new camera. The picture quality is amazing, but the battery life is too short'. If the target words are 'picture quality' the sentiment polarity would be 'positive'.

Integrating this target information into LSTM could significantly improve the classification accuracy (Figure 1):

$\{w_{l+1}, W_{l+2}, ..., W_{r-1}\}$ are target words which relates to certain aspects ('picture quality' in above example), Vtarget is target representation. The model separates the sentence into two parts, with first part start with the first word in the review sentences, but ended on the last word of target words, where the second part started from the first word of the targe words, and end on the last word of the review sentences (Tang, et al. 2016).

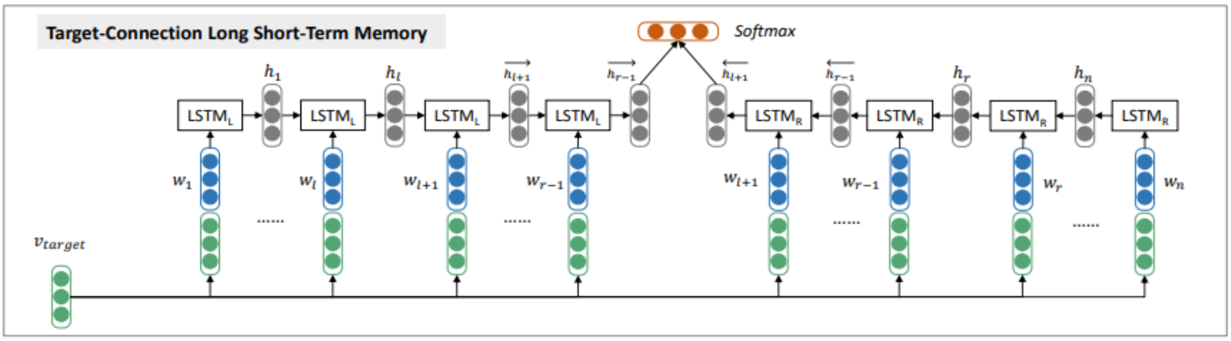

Target-Connection Long Short-Term Memory
Softmax
h_1
h_l
$\overrightarrow{h_{l+1}}$
$\overrightarrow{h_{r-1}}$
$\overleftarrow{h_{l+1}}$
$\overleftarrow{h_{r-1}}$
h_r
h_n
LSTM_L
LSTM_R
w_1
w_l
w_{l+1}
w_{r-1}
w_{l+1}
w_{r-1}
w_r
w_n
......
v_{target}

Figure 1

In Peng Chen, Zhongqian Sun (Chen, et al. 2017), proposed a more complex model which consists of five modules, input module, memory module, position-weighted memory module, recurrent attention module and output module.

The memory slices are weighted according to their relative positions to the target, so that different targets from the same sentence have their own tailor-made memories, after that, multiple attentions were paid on the position-weighted memory and nonlinearly combine the attention results with a recurrent network (Chen, et al. 2017).

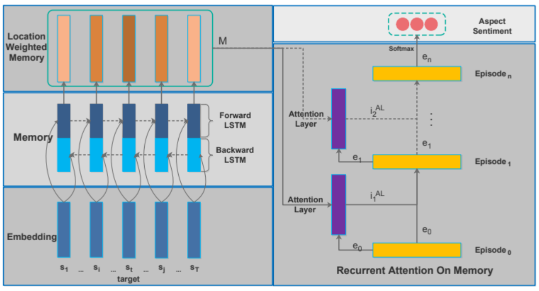


Figure 2

In this report, the MAMS (Multi-Aspect Multi-Sentiment) dataset, was used for training and testing, where each data instances includes three parts: **review, aspect, and sentiment**. The aspect has been given for each review, therefore the objective of my task will not be focused on Aspect Term Extract, more to the sentiment classification, based on the given aspect.

The MAMS dataset contains restaurant reviews, each review is relevant to 2 or more aspects, out of the total 8 aspects: **food, place, staff, service, price, menu, ambience, and miscellaneous**. The respective sentiment regarding different aspects could be: **positive, negative and neutral**, a 3-class classification task.

The challenging part with the MAMS dataset, is that each sentence contains **at least two aspects** with **different sentiment polarities**, therefore each review sentence will appear at least twice, with different aspects and sentiments associated with in my dataset. The given aspect word does not necessarily appear in the review sentence.

The main objective of my task: **how to embed the given aspect into different model variants ?**

Three different methods were proposed and conducted in the model experiments.

- Concatenate the aspect word embedding vector into the sentence word embedding representations, as the last word of the sentence.
- Concatenate the aspect word embedding vector into each word representation of the sentence, therefore the word representation vector dimension will be doubled.
- Attention mechanism, the aspect word representation will be used to calculate the attention weights for every word in the review sentence. The resulting attention output is expected to encompass a richer representation of the aspect-related section of the review sentence.

Multiple model designs and architectures have been attempted. The encoder – decoder sequence to sequence model, architected with Gated recurrent units (GRUs), and attention mechanism, outperformed other model variants, with an accuracy of 76.91% on the testing dataset. More specifically, the decoder of this model is only consisted of 1 GRU unit, where the inputs of the GRU unit are the given aspect word embedding, and the last hidden state of encoder. The output of the decoder will be used for attention mechanism to be multiplied with each hidden state of each word input in encoder. Finally, the attention output will be concatenated with decoder output for the final classification.

Without the attention mechanism, long short-term memory (LSTM) performed the best with an accuracy of 65.37% on testing dataset. The given aspect information was embedded through word processing, where the given aspect words were simply concatenated onto the review sentences as the last words.

Method

Comparing with the simpler sentence-based sentiment analysis tasks, where a single sentiment polarity classification is predicted based on the whole sentence, a recurrent seq-to-seq (m to 1) model would be sufficient. However, with the added complexity of the aspect, the focus should be on how to embed the aspect information most efficiently, to achieve a more accurate classification, in accordance with the given aspect.

Recurrent neural network (RNN), Long short-term memory (LSTM) and Gated recurrent unit (GRU) were used for different model implementation.

In figures 3 to 8 below:

- $\{w_1, w_2, ..., w_n\}$ represents context words representations.
- $\{w_a\}$ represents aspect word representation.
- $\{h_1, h_2, ..., h_n\}$ represents hidden states.

1. **RNN (Bi-directional):**

For the first simple RNN model, two different ways were used to embed the aspect information in the review sentence, as illustrated below.

- Variant 1: Concatenate the aspect word embedding vector into the sentence word embedding representations as the last word of the sentence.

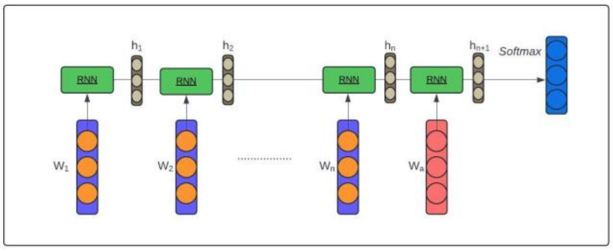


Figure 3

- Variant 2: Concatenate the aspect word embedding vector into each word representation of the sentence, therefore the word representation vector dimension will be doubled.

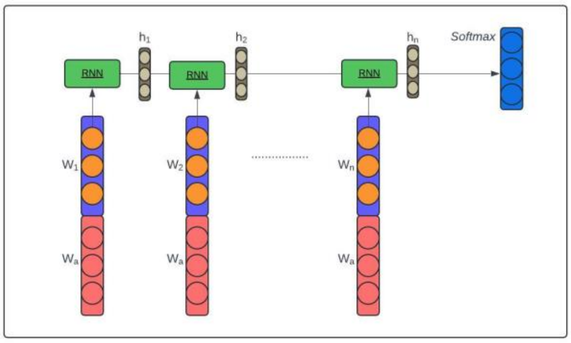


Figure 4

2. **LSTM (Bi-directional):**

For the LSTM Bi-directional model, the aspect word representations embedded into the sentence word embedding representations as the last word of the sentence.

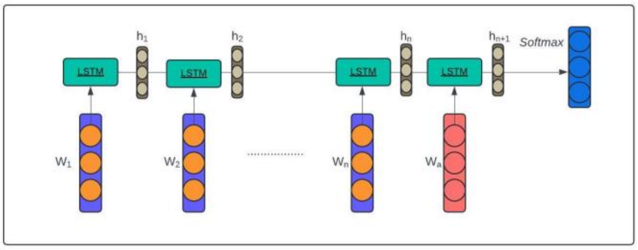


Figure 5

3. GRU (with attention mechanism):

- Variant 1: In this first experiment, the aspect word representations were fed into a simple linear layer, in order to get the same dimension with the GRU encoder hidden state. This liner layer output was used to calculate attention weights. The final attention output was concatenate with aspect linear layer output to make the final classification.

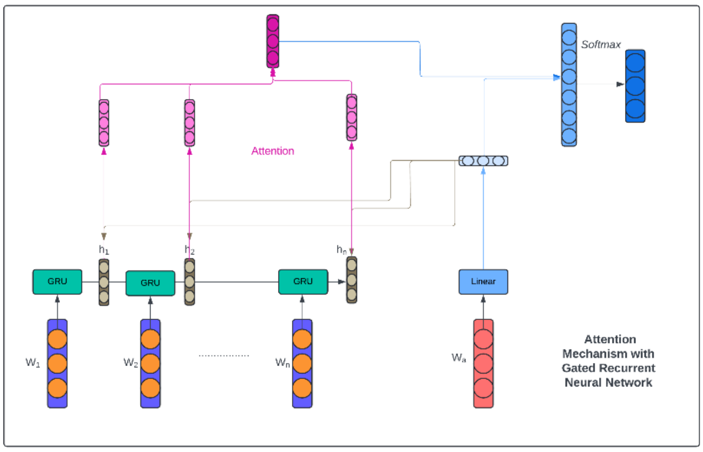


Figure 6

- Variant 2: a more traditional encoder – decoder architecture was used, where the aspect word embedding as an input of the decoder, along with the last hidden state of the GRU encoder, were fed into decoder GRU. The output of decoder was used for attention weight calculations, the attention output was concatenated with decoder output to make the final classification.

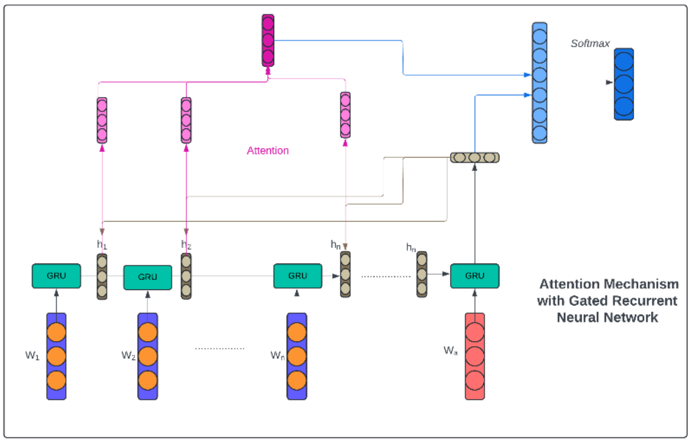


Figure 7

- Variant 3: The last variant is a hybrid version of variant 1 and 2, instead of using the output of encoder to concatenate with attention output for the final output, the aspect word representation was fed to another linear layer, the output of this linear layer was used to concatenate with the attention output for the final classification.

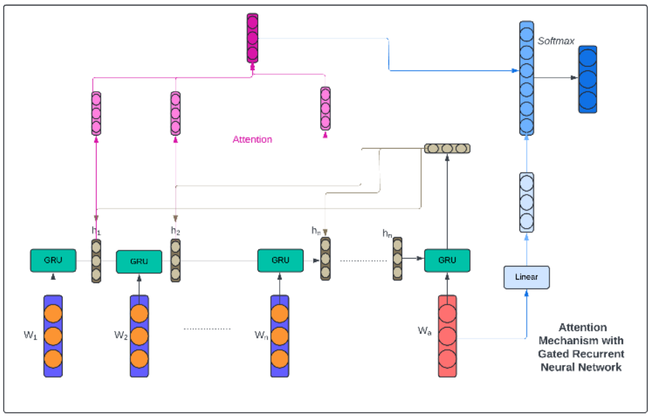


Figure 8

Experiments

Dataset Description: Multi-aspect Multi-sentiment (MAMS) dataset.

Each data instance in train, validation, and test sets are consist of 3 parts:

- Restaurant review sentence
- Aspects: food, place, staff, service, price, menu, ambience, or miscellaneous
- Sentiment polarities: positive, negative, or neutral

Each review sentence contains at least two aspects with different sentiment polarities. Below shows first 3 instances in training data before any processing:

[["It might be the best sit down food I've had in the area, so if you are going to the upright citizen brigade, or the garden, it could be just the place for you.", ***'food', 'positive'****],*

["It might be the best sit down food I've had in the area, so if you are going to the upright citizen brigade, or the garden, it could be just the place for you.", ***'place', 'neutral'****],*

['Hostess was extremely accommodating when we arrived an hour early for our reservation.', ***'staff', 'positive'****]]*

To capture more in vocabularies, the training set and validation set were combined to be used as a whole training set. The combined training set is an imbalanced dataset, 43.4% of review sentence and aspect combination had 'Neutral' as targeted class.

	Positive	Neutral	Negative	Total
Number of instances	2,170	3,465	2,343	7,978

Table 1

Further plots show sentence length distribution, as well as the aspect distribution.

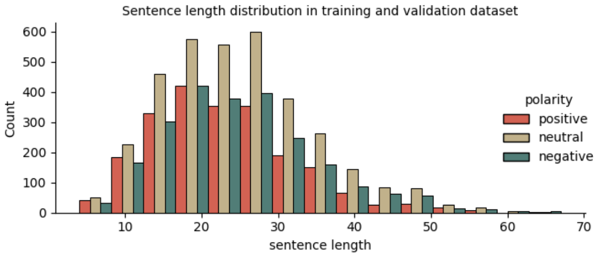


Figure 9

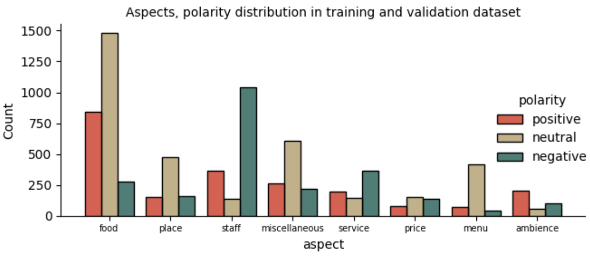


Figure 10

It is important to examine the data distribution, Figure 9 shows it may not be the best practice to pad all the sentence to the max length of 67 words, where majority of the review lengths are less than 50 words. Padding all sentences to the max length (which may also be skewed by outliers) may lead to vanish of the important information from the beginning of the sentence, also costing unnecessary processing time and computing power.

It is also important to understand data distribution for all aspects, for example, the aspect of 'Food' occurred most frequently in the training and validation set, we may expect a better performance in predicting sentiment over aspect of 'Food', where the model may struggle to identify negative sentiment in relation to the aspect 'Menu', simply due to lowest occurrences were available for training, and more importantly the cohort of sentiment words differ when describing the different aspect. E.g. 'bland' may be only used for aspect of 'Food', not 'Price'.

It may also not be a good idea to use the trained model on a similar review data, but majority of the review is about aspect 'Price'.

Experiment Setup:

For the text pre-processing:

- Case folding
- Contraction dealing
- Punctuation removal
- Word tokenization
- Sentence Padding (experimenting on the different padding length)
- Pre-trained word embedding: "glove-twitter-25", "glove-twitter-50", "word2vec-google-news-300"

For each model and each corresponding model variant (different ways to embed the aspect information). Below are the main hyper parameters experimented.

- Batch size: 48, 96, 128
- Total epoch: range between 15 to 40
- Hidden size: 96, 128, 256, 300, 600
- Number of layers: 2 - 5
- Dropout rate: 0.1 – 0.6
- Learning rate: 1e-1, 1e-2, 1e-3, 1e-4
- Loss function: Cross Entropy , NLLLoss

- Optimizer: Adam, SDG.
- Classification: Softmax
- Scoring: Accuracy, F1 score

Results

Quantitative Results:

Overall performance over different model architecture with different ways of aspect embedding:

As different model architecture may require different hyper parameters to achieve best performance suitable for that specific model architecture. To compare the model architecture in a more meaningful way, after multiple experiments, below hyperparameters were chosen for model architect comparison, where all models performed relatively well with this set of hyper parameter values. Table 2 reports the accuracy on the testing data for each model variants.

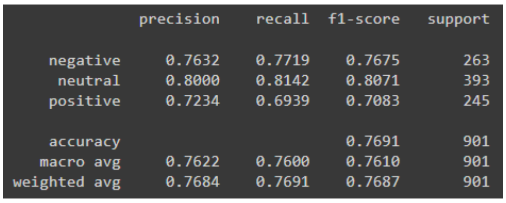
Batch size : 128 | Total Epoch : 30 | Loss function : Cross Entropy | Optimizer: Adam | Hidden size : 128 | pre-trained word embedding : word2vec-google-news-300 | Drop-out : 0.4

Model / Variant	Testing Accuracy	Model / Variant	Testing Accuracy
RNN (variant 1)	61.15%	RNN (variant 2)	55.60%
LSTM	65.37%		
GRU (attention variant 1)	73.03%	GRU (attention variant 2)	76.91%
GRU (attention variant 3)	76.14%		

Table 2

After fine tuning all hyper parameters on pseudo-validation set for GRU (attention variant 2), below reports the classification report on the testing set (901 instances), with accuracy reaching **76.91%,** and the precision/recall/F1-score all performing well across all sentiment classes.

The differences in F1 score between each class, have reflected our previous findings of imbalanced training data, around 43.4% of training data had sentiment class ‘neutral’. Hence the model performed better in classifying polarity ‘neutral’, compared with identifying ‘positive’.



```
              precision    recall  f1-score   support

    negative     0.7632    0.7719    0.7675       263
     neutral     0.8000    0.8142    0.8071       393
    positive     0.7234    0.6939    0.7083       245

    accuracy                         0.7691       901
   macro avg     0.7622    0.7600    0.7610       901
weighted avg     0.7684    0.7691    0.7687       901
```

Table 3

Ablation study: (numbered by the effectiveness and importance order)

1. **Aspect embedding methods:**
   In RNN (variant 1), the aspect word representation was concatenated to the whole sentence as the last word, comparing with concatenate to each word of review sentence (variant 2) (Figure 3 & 4). The model performed better with a over 5% boost in the accuracy to 61.15%.
   However, GRU (variants 1 – 3) (Figure 6, 7 & 8) shows using the attention mechanism to embed aspect improved model effectively. All three variants reached accuracies over 70%.

2. **Recurrent Neural Network variants:**
   By replacing RNN with LSTM (both bi-directional), is the only difference between RNN (variant 1) and LSTM model. (Figure 3 & 5). The accuracy on testing data was boosted by around 4%, this proves the effectiveness of LSTM in remembering sentence meaning.

   However, in GRU model variants with attention mechanism, if GRU units were replaces with LSTM units, the model performance score decreased (not listed in the table). This may be due to the nature, and effectiveness of attention mechanism. The transformer model solely relays on the attention mechanism and positioning embedding, rather than requiring LSTM as well which may cause adverse effect.

   The figure 6,7 & 8 clearly stated the difference in all three GRU with attention model variants. It seems that the encoder – decoder model still works the best with one GRU unit in the decoder, however the GRU variant 1, by using a simple liner layer for aspect word embedding in calculating attention weights, without a GRU unit and the last hidden state of encoder as decoder hidden input, still shows a strong performance. The only difference between variants 2 and 3 is using different vectors to concatenate with attention output vector for the final classification, both architectures performed similarly well.

3. **Pre-trained word embedding:**
   Comparing with the "glove-twitter-25", "glove-twitter-50", or the concatenation of both (dimension 75), the high dimension "word2vec-google-news-300" successfully improved the model performance by around 4%, especially in GRU with attention model variants, the models struggled to reach over 70 if the lower dimension word representations were used.

4. **Dropout Layer**
   The implementation of dropout layer also helped the model to generalise better, when setting the dropout layer rate to 0.4 – 0.5, it effectively reduced the model variance. It also prevents the model becoming overfitted where failing to 'learn' but to 'remember' the training data.

5. **Loss function / Optimizer / Learning Rate:**
   Overall, Cross Entropy was more effective for training the model comparing with Negative Likelihood Log Loss. The optimizer Adam is also more preferrable than SDG, in most cases, SDG required a higher learning rate, but the loss reduction may start to be very unstable when the learning rate is high.

6. **Hidden Size / Number of Layers:**
   The hidden size also played a pivot role in my ablation study, around 216 seems to be an optimal range, where the lower hidden size may not be enough to learn the pattern due to high word embedding dimensions (300). The models also benefit very minimum when the hidden size was increased to 300 to 800 range, instead it required far more computing power, and longer processing time when evaluating.
   The increase in number of layers also presented minimum improvement, costing unnecessary computational power and time.

Qualitative Results:

The main goal for qualitative results will be the visualisation of the attention weights. Here the attention weights extracted from the best performing model GRU (variant 2).

Example 1:

Index 157, 158 in testing data set, same review with different given aspects 'Ambience' and 'Food'. The model (visualised through attention weights) successfully paid more attention to the relevant part of the sentences, therefore predicting correct sentiment polarities. (Figure 10)

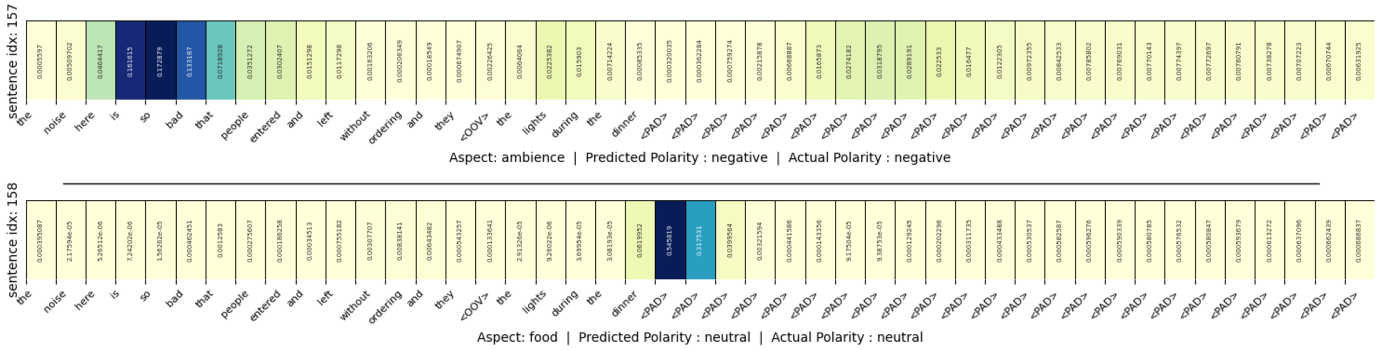


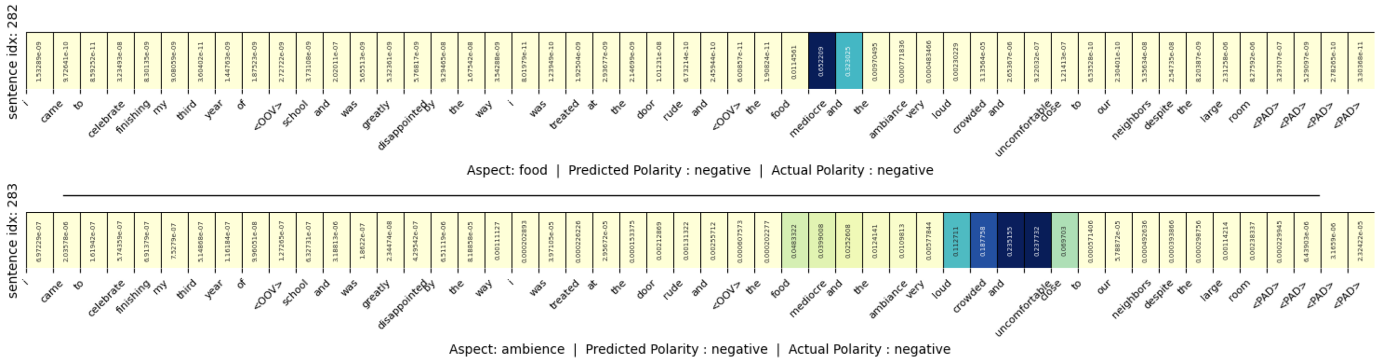
Figure 10

It is quite exciting to see when aspect '**Ambience**' was given, the model has paid attention to the beginning of the sentence. The word 'Ambience' didn't appear at all in the review sentence, but the model is still able to identify 'the noise here is so bad' is more relevant to the ambience, therefore predicting correct 'Negative' sentiment.

On the contrary, when aspect 'Food' was given, the model has ignored the beginning part of the sentence, but paid more attention to the words around food. Regardless the presence of the word 'Food', there was no sentiment / comment made towards the food, therefore the model has predicted 'Neutral' correctly.

Example 2:

Index 282, 283 in testing data set, same review with different given aspects 'Food' and 'Ambience', The model (visualised through attention weights) successfully paid more attention to the relevant part of the sentences, therefore predict relevant sentiment polarities. (Figure 11)

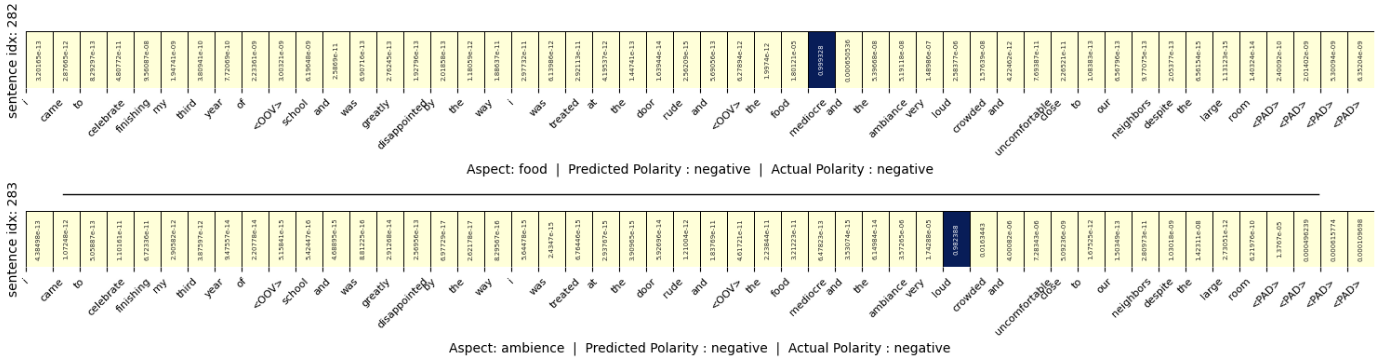


Aspect: food | Predicted Polarity : negative | Actual Polarity : negative

Aspect: ambience | Predicted Polarity : negative | Actual Polarity : negative

Figure 11

It is also interesting, and important to note that, dimensionality of word embedding, as well as the hidden size, will heavily affect the results of the attention weight distribution. Figure 12 shows attention weight when hidden dimension size **increased from 128 to 512**.



Aspect: food | Predicted Polarity : negative | Actual Polarity : negative

Aspect: ambience | Predicted Polarity : negative | Actual Polarity : negative

Figure 12

As a result of comparison between figure 10 or 11, the higher dimension of hidden size will lead the attention to focus more on a specific word, where then the dimensionality is lower, the attention focused more on group of words / nearby words as well.

This is another interesting evidence where the higher dimensionality in hidden size may not always be desired.

Conclusion

The main challenge of the aspect-based sentiment analysis is around how to embed the aspect information effectively. Based on my project and above results, 'attention is all I need'. The attention mechanism effectively injected the relevant aspect information into certain part of the sentence, it is more efficient and intuitive comparing with simply concatenating the aspect representations into the sentences.

This project mainly prioritizes and focus on the model design, seq-to-seq architect exploration with novelty, by experimenting and adapting different part of the model architecture to examine the effectiveness. Then, complex experiments and ablation studies were conducted for optimizing the hyper parameters. However, with the goal to achieve a larger vocabulary for a better generalisation, I've combined the training and validation set into a larger training set to train the model, the disadvantage of such practise is the difficulty in fine tuning hyper parameters, it is much preferable to utilise a real validation / OOB dataset, rather than relying on the pseudo-validation set (a small portion of training set) for hyper parameter optimization.

However, the model architect design weights greater importance in this project. There are also many other directions can be explored, including the different method to calculate the attention weights, as well as utilising pos-tag, and name entity techniques to filter the words syntactically and semantically, to focus on the only words may express sentiment polarities mainly (e.g. adjectives).

Meanwhile, due to the utilization of batch processing, all sentences were padded to the optimal length, which lead to many OOV padded into the sentence with shorter length. The LSTM or GRU may be effective to address this issue, but the model could perform even better when trained based on the original sentences.

References

Chen, Peng, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. “Recurrent Attention Network on Memory for Aspect Sentiment Analysis.” *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing.* Copenhagen: Association for Computational Linguistics. 452-461.

Kiritchenko, Svetlana, Saif M Mohammad, Xiaodan Zhu, and Colin Cherry. 2014. “NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews.” *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014).* Dublin: National Research Council Canada. 437-442.

Punitha, A, R Keerthi Prabu, P Devanathan, A Sanjai, and P Bala. 2023. “Aspect-Based Sentiment Analysis.” *International Journal For Multidisciplinary Research* 5 (3): Volumn 5, Issue 3.

Tang, Duyu, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. *Effective LSTMs for Target-Dependent Sentiment Classification.* Thesis, Harbin: Harbin Institute of Technology.

Vaswani, Ashish, Illia Polosukhin, Lukasz Kaiser, Aidan N. Gomez, Llion Jones, Jakob Uszkoreit, Niki Parmar, and Noam Shazeer. 2017. “Attention is All You Need.” *Advances in Neural Information Processing Systems 30 (NIPS 2017).* Long Beach California USA: NeurIPS Proceedings. 6000–6010. https://papers.nips.cc/paper_files/paper/2017.