University of Western Australia

Project Report

Aspect-Based Sentiment Analysis

Group ID - 6
Angela Jacinto - 23778435
Saif Ali Athyaab - 23801756
Wally Siraj - 23803313

Master of Data Science

Fulfilling requirements of

CITS4012 Natural Language Processing
Assignment

31 July 2024

Contents

Abstract

Aspect-Based Sentiment Analysis (ABSA) is a critical task in natural language processing, aiming to determine the sentiment polarity of specific aspects within a sentence. With the increasing volume of online reviews, ABSA provides granular insights into consumer sentiments, which are invaluable for businesses. Human reviews are often complicated for machines to understand due to the complex way of writing that involves sudden pivots when writing about something. This project explores ABSA using the MAMS dataset, which contains restaurant reviews annotated with multiple aspects and corresponding sentiments. Traditional methods often do not provide the best sentence polarity due to the presence of multiple aspects and their corresponding polarities. By breaking down the sentence into a number of aspects, our models are capable of assigning the right polarity to each aspect. Our objective was to design and evaluate three innovative models, each integrating aspect-specific information in unique ways within their architectures. We focused on optimizing model performance through hyperparameter tuning, including learning rate and batch size adjustments. The first model employed a Bi-directional Long Short-Term Memory (Bi-LSTM) network with concatenated word, aspect, and dependency parsing embeddings. Initial tests on this model demonstrated moderate success, achieving a test accuracy of 60.27%. The second model utilized a sophisticated Bi-LSTM with an encoder-decoder Seq2seq architecture and incorporated scaled dot product attention, achieving test accuracy of 52.16% . In this model, aspects were appended to the end of the sentences, and the sentences were augmented with part-of-speech tagging and dependency parsing. The final model consisted of adding an adaptive masking mechanism to a Bi-LSTM network by increasing attention to context around an aspect to emphasize certain parts of the sentence, achieving 42.62% accuracy.

Keywords: Aspect-Based Sentiment Analysis · Bi Directional Long-Short Term Memory

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a refined classification task in natural language processing that can provide complete and in-depth results Sun et al., 2020 by identifying the sentiment polarity attached to specific aspects within a sentence. Contrary to traditional sentiment analysis, in which overall sentence is assessed, ABSA allows for better insight as it captures the sentiment of each aspect separately. ABSA's detailed approach is useful in understanding specific sentiments about aspects such as food, service, or ambiance, which can highly influence business decisions and strategies. The ability of ABSA to extract precise sentiment details from unstructured text, such as online reviews, social media posts, and customer feedback, helps businesses gain actionable insights that can influence product development and customer service improvements. The complexity of human language, including ambiguity, sarcasm, and varied expressions, are significant challenges to ABSA. Unlike humans, neural networks lack the intuitive understanding needed to link sentiments to specific aspects, making this task more challenging.

In recent years, the application of Bidirectional Long Short-Term Memory (BiLSTM) networks has become increasingly prevalent in the field of Aspect-Based Sentiment Analysis (ABSA). Bidirectional Long Short-Term Memory (BiLSTM) networks are a powerful extension of traditional LSTM models, which are designed to process sequential data by capturing dependencies over long time spans. In BiLSTM networks, two LSTM layers are

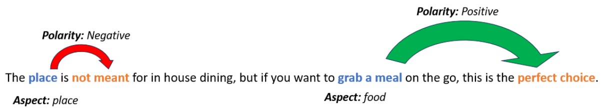


Figure 1: A sample of review from test dataset. It consists of multiple aspect terms and each aspect of the review contains its own sentiment polarity.

employed: one processes the input sequence from left to right (forward direction) and the other from right to left (backward direction). By leveraging both past and future context, BiLSTMs can more accurately determine the sentiment associated with specific aspects within a sentence, significantly improving performance over unidirectional models.

One notable study by Tang, Qin and Liu, 2016 proposed a BiLSTM model with attention mechanisms to better capture the semantic correlations between aspect terms and their contexts. This model improved the accuracy of sentiment classification by allowing the network to focus on the most relevant parts of the input text. Similarly, Zeng, Ma and Zhou, 2019 introduced a position-aware Bidirectional LSTM model that integrates position information of aspect terms into the attention mechanism, further enhancing the model's performance in aspect-level sentiment classification. Another significant contribution by Lin, Yang and Lai, 2019 utilized a BiLSTM network within a multi-task learning framework to implicitly capture the relationships between aspect terms and opinion terms. Their approach demonstrated that BiLSTM could effectively model the dependencies and semantic relationships necessary for precise sentiment analysis.

These studies highlight the effectiveness of BiLSTM architectures, especially when augmented with attention mechanisms, in capturing nuanced contextual information crucial for accurate aspect-based sentiment analysis. In this project, we utilized the MAMS dataset, which contains detailed categorization of sentiments across different aspects in restaurant reviews. Our objective was to develop three innovative models capable of accurately predicting sentiment polarities while effectively integrating aspect-specific information. This project contributes to the current applications of aspect-based sentiment analysis and underscores the challenges and complexities of detailed sentiment analysis through innovative architectural developments.

2 Methods

Each model variant utilizes a Bi-LSTM as the sequence processing unit. The Bi-LSTM layer is particularly chosen for the reasons stated in the previous section.

2.1 Handling Variable Sequence Lengths

To manage varying input lengths across different data instances effectively, we integrate dynamic padding within the training pipeline. This approach adjusts the padding for each batch based on the longest sequence present in that batch, which optimizes memory usage and enhances computational efficiency. This ensures that the model only processes the necessary data for each batch. After this, we implement this dynamic padding through the collate function used in the data loader. The function ensures that all sequences in a given batch are consistently padded with the maximum sequence length in a particular batch before being passed to a model for processing.

2.2 Dataset Handling

To ensure that the training, validation and testing datasets are preprocessed, aligned and optimized for learning, we utilize the "SentimentDataset" class to transform and organize raw data into a usable format for the neural network models. The class integrates various specific embeddings into a single tensor, providing a rich feature set that enables the models to understand the context of each sentence in relation to a specific aspect.

2.3 Model Architectures

Model 1 integrates word embeddings, aspect embeddings, and dependency parsing label embeddings to create a comprehensive understanding of the context.

Word embeddings are created using the Word2Vec model.

$$\mathbf{w}_i = \text{Word2Vec}(\mathbf{token}_i)$$

Aspect embeddings are created using an embedding layer.

$$\mathbf{a} = \text{AspectEmbedding}(\text{aspect\_index})$$

Dependency label embeddings are created using an embedding layer.

$$\mathbf{d}_i = \text{DepEmbedding}(\text{dep\_index}_i)$$

The combined embeddings for each token are formed by concatenating the word embedding, aspect embedding, and dependency label embedding.

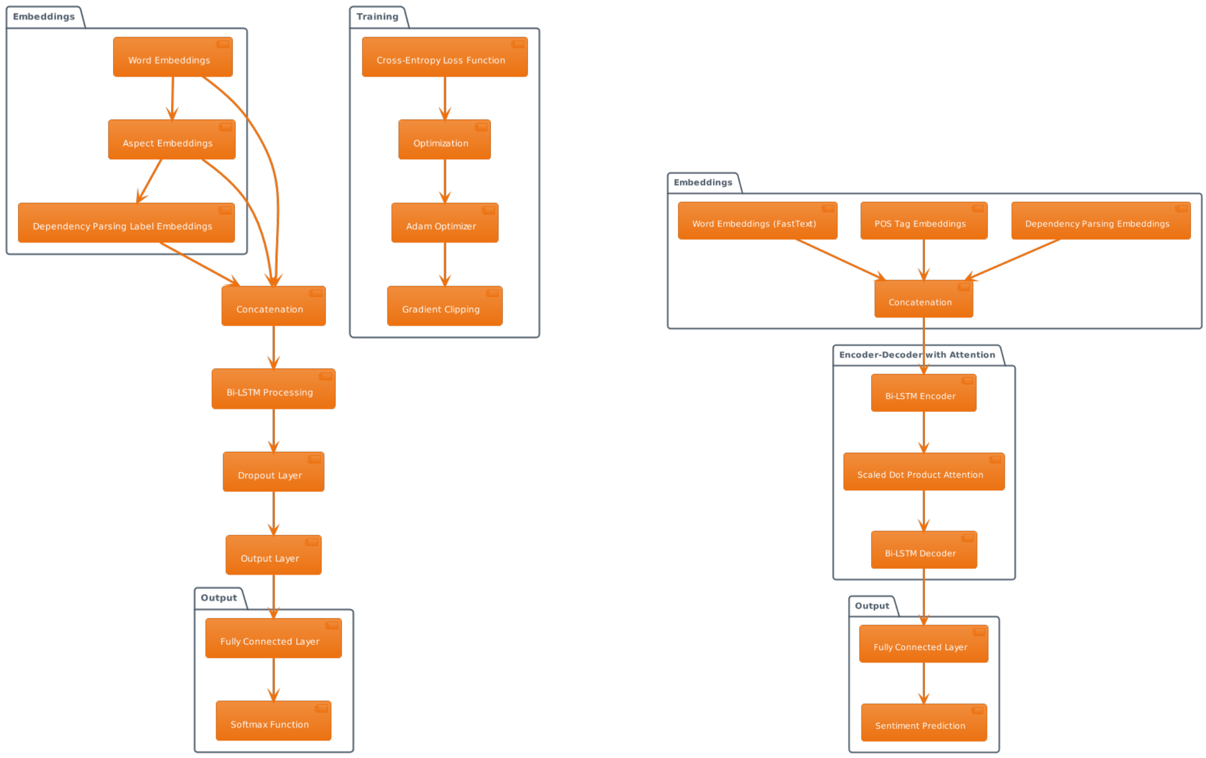
$$\mathbf{e}_i = [\mathbf{w}_i; \mathbf{a}; \mathbf{d}_i]$$

The BiLSTM processes the combined embeddings and generates hidden states.

$$\mathbf{h}_t^{\text{enc}} = \text{BiLSTM}(\mathbf{e}_t, \mathbf{h}_{t-1}^{\text{enc}})$$

The final hidden states from the forward and backward LSTM are concatenated and passed through a fully connected layer to produce the output.To handle overfitting, a dropout layer is integrated within the LSTM layers and after the LSTM output. The dropout randomly turns a fraction of features to values of 0 to learn more features and enhance generalization.

$$\begin{aligned}
\mathbf{h}_t^{\text{forward}} &= \mathbf{h}_t^{\text{enc}}[:\text{hidden_dim}] \\
\mathbf{h}_t^{\text{backward}} &= \mathbf{h}_t^{\text{enc}}[\text{hidden_dim}:] \\
\mathbf{h}_t^{\text{final}} &= [\mathbf{h}_t^{\text{forward}}; \mathbf{h}_t^{\text{backward}}] \\
\mathbf{o}_t &= \mathbf{W}_o \mathbf{h}_t^{\text{final}} + \mathbf{b}_o
\end{aligned}$$



(a) Model 1

(b) Model 2

Figure 2: Comparison of Model 1 and Model 2

Model 2 incorporates a Bi-LSTM with an encoder-decoder Seq2seq architecture and scaled dot product attention. This model also integrates aspect information by appending the aspect to the end of the sentences, which are augmented with part-of-speech tagging and dependency parsing. Unlike the first model, this model uses FastText with a Skip-gram approach for word embeddings. POS tag embeddings and dependency parsing embeddings are also generated.The word embeddings, POS tag embeddings, and dependency parsing embeddings are concatenated, forming a comprehensive input vector. The concatenated embeddings are fed into an encoder-decoder Seq2seq architecture with scaled dot product attention. The input to the encoder is a sequence of word embeddings, POS tag embeddings, and dependency embeddings. The BiLSTM processes the input sequence and generates hidden states.

$$\begin{aligned} \mathbf{x}_t &= [\mathbf{w}_t; \mathbf{p}_t; \mathbf{d}_t] \\ \mathbf{h}_t^{\text{enc}} &= \text{BiLSTM}(\mathbf{x}_t, \mathbf{h}_{t-1}^{\text{enc}}) \end{aligned}$$

where: - $\mathbf{x}_t$ is the concatenated embedding vector at time step t consisting of word embedding $\mathbf{w}_t$, POS tag embedding $\mathbf{p}_t$, and dependency embedding $\mathbf{d}_t$. - $\mathbf{h}_t^{\text{enc}}$ is the hidden state of the encoder at time step t.

The attention mechanism computes attention weights using the scaled dot product of the query, key, and value matrices. This allows the model to focus on relevant parts of the input sequence when making predictions.

$$
\begin{aligned}
\mathbf{Q} &= \mathbf{W}_Q \mathbf{h}_t^{\text{dec}} \\
\mathbf{K} &= \mathbf{W}_K \mathbf{h}_s^{\text{enc}} \\
\mathbf{V} &= \mathbf{W}_V \mathbf{h}_s^{\text{enc}} \\
\mathbf{A}_{t,s} &= \frac{\exp(\mathbf{Q} \cdot \mathbf{K}^T)}{\sum_j \exp(\mathbf{Q} \cdot \mathbf{K}_j^T)} \\
\mathbf{c}_t &= \sum_s \mathbf{A}_{t,s} \mathbf{V}_s
\end{aligned}
$$

where: - $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ are learnable weight matrices for the query, key, and value transformations. - $\mathbf{A}_{t,s}$ is the attention weight between the decoder state at time step t and the encoder state at time step s. - $\mathbf{c}_t$ is the context vector at time step t.

The input to the decoder is the concatenation of the context vector and the previous hidden state. The BiLSTM processes this input to generate the hidden state and the output.

$$
\begin{aligned}
\mathbf{h}_t^{\text{dec}} &= \text{BiLSTM}([\mathbf{c}_t; \mathbf{h}_{t-1}^{\text{dec}}], \mathbf{h}_{t-1}^{\text{dec}}) \\
\mathbf{o}_t &= \mathbf{W}_o \mathbf{h}_t^{\text{dec}} + \mathbf{b}_o
\end{aligned}
$$

where: - $\mathbf{h}_t^{\text{dec}}$ is the hidden state of the decoder at time step t. - $\mathbf{o}_t$ is the output vector at time step t. - $\mathbf{W}_o$ and $\mathbf{b}_o$ are the weight matrix and bias vector for the output layer.

The output from the decoder is passed through a fully connected layer to predict the sentiment polarity for each aspect.

Model 3 is based on a masking mechanism (inspired by ASC task of Rafiuddin et al., 2024, they used BERT) to blur out irrelevant words using context based on aspects. The word **embeddings** are initialized using FastText with CBOW, trained on the dataset. Aspects are encoded using **LabelEncoder** and embedded similarly.

An embedding matrix is constructed from the FastText word vectors.

$$
\mathbf{E}_{i,j} = \begin{cases} \mathbf{w}_i & \text{if } \textbf{token}_i \text{ is in vocabulary} \\ 0 & \text{otherwise} \end{cases}
$$

The input embeddings are processed using two BiLSTM networks: one for the context and one for the aspect. The context BiLSTM processes the input sentence embeddings.

$$
\mathbf{H}^{\text{context}} = \text{BiLSTM}^{\text{context}}(\mathbf{E})
$$

The aspect BiLSTM processes the aspect term embeddings.

$$
\mathbf{H}^{\text{aspect}} = \text{BiLSTM}^{\text{aspect}}(\mathbf{E}_{\text{aspect}})
$$

The attention mechanism computes attention scores for the context embeddings, which are then softmax-normalized.

$$\mathbf{A}_t = \text{softmax}(\mathbf{W}_a \mathbf{H}_t^{\text{context}})$$

where: - $\mathbf{W}_a$ is the attention weight matrix. - $\mathbf{H}_t^{\text{context}}$ is the hidden state of the context BiLSTM at time step t.

The masked LSTM output is obtained by element-wise multiplication of the context embeddings with the attention weights.

$$\mathbf{H}_t^{\text{masked}} = \mathbf{H}_t^{\text{context}} \odot \mathbf{A}_t$$

The summed LSTM output is the sum of the masked LSTM outputs across the sequence.

$$\mathbf{H}^{\text{sum}} = \sum_t \mathbf{H}_t^{\text{masked}}$$

The final output is obtained by passing the summed LSTM output through a fully connected layer.

$$\mathbf{o} = \mathbf{W}_f \mathbf{H}^{\text{sum}} + \mathbf{b}_f$$

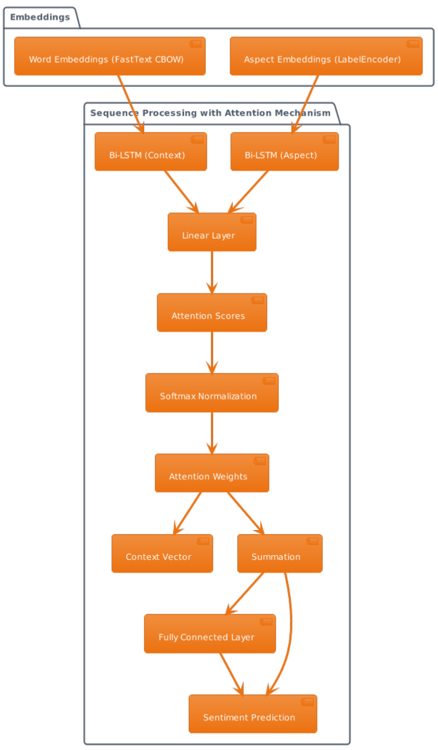


Figure 3: Model 3

3 Experiments

3.1 Dataset Description

The dataset used in this project is the Multi-Aspect Multi-Sentiment (MAMS) dataset, which contains sentences with at least two aspects, each having different sentiment polarities. Each instance includes a restaurant review, one restaurant aspect, and the sentiment polarity (positive, negative, or neutral) of that aspect. The dataset encompasses eight aspect categories: food, service, staff, price, ambience, menu, place, and miscellaneous.

3.2 Dataset Analysis

The initial analysis of the dataset revealed a higher occurrence of the 'neutral' polarity and the 'food' aspect across all sets (train, validation, and test). This imbalance suggests a potential bias in the model towards the more frequently occurring classes. We also counted the number of unique sentences in each set: 3149 in the training set, 400 in the validation set, and 400 in the test set.

3.3 Preprocessing

Preprocessing steps included case folding (converting all text to lowercase), expanding contractions using a predefined dictionary, removing punctuation, tokenizing the sentences, removing stopwords, and lemmatizing the tokens. These steps were applied uniformly across all three model variants. We created copies of the preprocessed dataset for use with different model variants.

3.4 Experiment Setup

All models were trained using cross-entropy loss and the Adam optimizer. We defined a hyperparameter search space with learning rates of [0.01, 0.005, 0.001], batch sizes of [64, 128, 256], hidden dimensions of [64, 128, 256], and the number of LSTM layers as [2, 3]. The models were initially trained on the training data for 50 epochs, and the hyperparameters corresponding to the best validation accuracy were saved. Subsequently, the models were retrained on the combined training and validation sets using these optimal hyperparameters and then evaluated on the unseen test set.

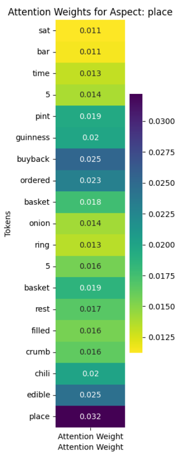
4 Results

4.1 Quantitative Results

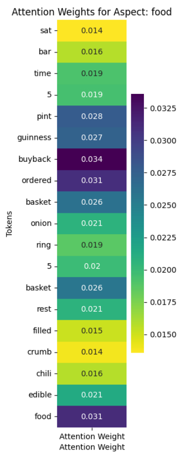
Table 1: Ablation Study

Model	Word Embeddings	Attention Mechanism	Batch Size	Learning Rate	Number of Layers	Number of Hidden Layers	Test Accuracy
Model 1	Word2Vec	No	128	0.005	3	128	60.27%
Model 2	FastText	Yes	128	0.005	3	128	52.16%
Model 3	FastText	Yes	256	0.01	3	256	42.62%

In the ablation study, the results indicate that a simpler architectural approach using Word2Vec embeddings achieved the highest accuracy of 60.27%. This could be due to



(a) Model 2 Neutral polarity sample



(b) Model 2 Negative polarity sample

Token	Attn. Score	Masked
sat	-0.0106	Yes
bar	-0.0352	Yes
time	-0.0520	No
5	-0.0447	No
pint	-0.0374	Yes
ordered	-0.0456	No
basket	-0.0444	No
onion	-0.0436	No
5	-0.0347	Yes
basket	-0.0174	Yes
rest	-0.0254	Yes
filled	-0.0379	Yes
chili	-0.0368	Yes
food	-0.0397	No
food	-0.0490	No
food	-0.0601	No
Total:	-0.6145	
Mean:	-0.0384	

(c) Model 3

Figure 4: Heatmap Visualisation of attention scores

the straightforward nature of the model, which effectively captured the semantic features necessary without the added complexity of more advanced mechanisms like attention. On the other hand, the second model, which incorporated FastText embeddings and an attention mechanism demonstrated a decrease in performance, achieving a 52.16% accuracy. This reduction could be due to the added complexity from the attention mechanism, a lower learning rate, and increased hidden layers. These factors potentially introduced too many variables for the model to effectively learn. Lastly, the third model, with an increase in batch size, use of FastText and attention, and masking showed a drop in performance of 42.64%. The decrease in accuracy could come from several factors, such as potential overfitting or less effective gradient updates due to the larger batch size, which can impact the model's ability to generalize well in unseen data. These findings suggest that the simpler model performed the best, which suggests that the dataset did not benefit from FastText and attention mechanisms or that further tuning might be necessary to optimize these advanced features. Moreover, it highlights the importance of balancing model complexity, training configurations and the characteristics of the dataset.

4.2 Qualitative Results

In **Fig 4a**, highest attention weight is given to the word "place" itself, while the word "bar," which should be more relevant to the aspect of place, has a significantly lower attention weight. This indicates that the model fails to correctly capture the contextual relevance of the tokens related to the aspect of place. The polarity is neutral, and the misalignment in attention might contribute to the model's inability to correctly interpret the sentiment related to the aspect. In **Fig 4b**, the highest attention weight is observed on the word "buyback," which is not directly related to the food aspect. Although the word "edible" receives a medium attention score, it is not sufficient to accurately capture the sentiment of the aspect. The full sentence indicates that the food was "not even edible," suggesting a negative polarity, which the model fails to correctly associate due to the inappropriate attention distribution. The removal of stopwords might have contributed towards this. Tokens like "sat", "bar", "pint", and "5" are masked, indicating the model considers them less relevant to the aspect being analyzed. In the Masking model **Fig 4c**, tokens such as "time", "ordered", "basket", "onion", and "food" (multiple instances) are not masked, indicating they are deemed significant.Tokens such as "sat", "bar", and "pint" are masked, suggesting that the model does not consider them highly relevant to the aspect. Despite negative scores, the model is able to to some extent mask the unimportant words.

5 Conclusion

Our project aimed to explore different novel network architectures in tackling Aspect-Based Sentiment Analysis using the MAMS dataset, focusing on distinctively integrating aspect information within the architectures of three models. The main finding is that simpler models might be more effective for this specific dataset since the simplest model variant, employing a Bi-LSTM with Word2Vec embeddings but no attention mechanism, achieved the highest accuracy at 60.27%. The introduction of more complex mechanisms like attention in the second model yielded high train accuracy(90+) but lower test accuracy indicating ooverfitting. Moreover, the attention mechanisms did not successfully highlight relevant tokens for determining sentiment about specific aspects, as evidenced by the qualitative analysis where obvious aspect-related tokens did not receive higher weights. Although the models were expected to benefit from the incorporation of attention mechanisms due to their potential to enhance focus on relevant textual features, they did not perform as expected when added to the models.

To potentially achieve the expected results in future studies, proper tuning and training of attention mechanisms are needed to make them effective and sensitive enough to focus on the most impactful words related to the aspect being analyzed. Despite the mixed results, the project successfully designed and evaluated three distinct models, each integrating aspect-specific information differently. This process allowed the team to develop a deeper understanding of how different architectures and integrations can influence model performance in complex sentiment analysis tasks. Furthermore, the project addressed the challenge of varying sentiment polarities within the same sentence. Some limitations observed include the presence of more neutral reviews, which suggests an imbalance in the dataset. The imbalance might have contributed to biased model outcomes and relatively lower accuracy. Additionally, the preprocessing of tokens through POS tagging and dependency parsing might have reduced some contextual clues necessary for effective sentiment analysis. All in all, this project has contributed valuable insights into the capabilities and limitations of different approaches to the ABSA task. It highlights the potential for future improvements that could lead to more accurate sentiment analysis.

6 Team Contribution

Each team member equally contributed to the whole project.

7 References

Lin, Peiqin, Meng Yang and Jianhuang Lai (Aug. 2019). 'Deep Mask Memory Network with Semantic Dependency and Context Moment for Aspect Level Sentiment Classification'. In: DOI: 10.24963/ijcai.2019/707. (Visited on 24/09/2023).

Rafiuddin, S M et al. (Apr. 2024). 'Exploiting Adaptive Contextual Masking for Aspect-Based Sentiment Analysis'. In: *Lecture notes in computer science* 14650, pp. 147–159. DOI: 10.1007/978-981-97-2266-2_12. (Visited on 18/05/2024).

Sun, Chengai et al. (Dec. 2020). 'Deep Interactive Memory Network for Aspect-Level Sentiment Analysis'. In: *ACM transactions on Asian and low-resource language information processing* 20, pp. 1–12. DOI: 10.1145/3402886. (Visited on 19/05/2024).

Tang, Duyu, Bing Qin and Ting Liu (Nov. 2016). *Aspect Level Sentiment Classification with Deep Memory Network*. ACLWeb. DOI: 10.18653/v1/D16-1021. URL: https://www.aclweb.org/anthology/D16-1021/.

Zeng, Jiangfeng, Xiao Ma and Ke Zhou (2019). 'Enhancing Attention-Based LSTM With Position Context for Aspect-Level Sentiment Classification'. In: *IEEE Access* 7, pp. 20462–20471. DOI: 10.1109/access.2019.2893806. (Visited on 22/05/2022).

[1]

Notes

[1]ChatGPT has been used to collate the report, helping to convert the equations to LaTeX code, for initial model ideation, and understanding the core methods of research papers in the field. OverLeaf was used as a collaborative means to write the reports in LaTeX.