Explaining a Deep Learning Model for Aspect-Based Sentiment Analysis Using SHAP

Kelvin Z. Yeung, Flavius Frasincar(✉)[0000−0002−8031−758X], and Finn van der Knaap

Erasmus University Rotterdam, Burgemeester Oudlaan 50, 3062 PA Rotterdam, the Netherlands
kelvin28.yeung@gmail.com, frasincar@ese.eur.nl, 573834fk@student.eur.nl

Abstract. State-of-the-art machine learning models have continuously improved over recent years, leading to increasingly high-performing models. Simultaneously, it has become increasingly difficult to infer insights in model behavior, as models continuously increase in complexity. This paper aims to increase the explainability of the HAABSA++ model, a state-of-the-art machine learning algorithm that combines a domain sentiment ontology with a deep learning model using contextualized word embeddings, designed for aspect-based sentiment analysis. The model is trained and evaluated on the SemEval 2016 restaurant reviews dataset. For model explainability we propose two SHAP approaches. The first approach, SHAP model 1, applies SHAP after BERT word embeddings are generated, measuring the contribution of each embedded word to the sentiment prediction. The latter approach, SHAP model 2, applies SHAP before BERT word embeddings are generated. SHAP model 2 appears to be better at finding words that individually contribute most towards a sentiment prediction since it generates new word embeddings for each subset of the original sentence.

Keywords: SHAP · HAABSA++ · Aspect-Based Sentiment Analysis · Explainability

1 Introduction

The mass integration of the Social Web into consumer markets has drastically altered interactions between businesses and consumers; specifically reviews have had a significant impact as a form of customer value co-creation [20]. Consequently, academic research into reviews has seen a huge increase in coverage, with sentiment analysis being one of the main fields of interest. Sentiment analysis makes it possible to automate the evaluation of review sentiments, enabling managers and researchers to derive insights on a large scale, which has had huge implications for various sectors [10].

The simplest application of sentiment analysis is the sentiment classification of sentences, paragraphs, or documents. This process, however, is unable to capture all the complexities of human language, as the meaning of language

is dynamic and influenced by factors such as context and presumptions. One way of dealing with context is Aspect-Based Sentiment Analysis (ABSA); here, the relevant entity or related aspect is identified and then classified according to sentiment [17]. Several varieties of ABSA models have emerged over the years, resulting in impressive performances.

As high-performing models often combine multiple techniques to increase accuracy, the transparency of such models decreases, resulting in black box models. The black box nature of a model is problematic, as it reduces the comprehension of why a model did, does, and will do something. Even if correct, the model prediction could be based on the wrong reasoning or predictors, potentially leading to decisions based on compliance, rather than truthful insights.

This research aims to increase the explainability of ABSA models – particularly to aid managers and researchers in comprehending model behavior – with the goal of creating trust and insights about how decisions are made. Aiding this purpose is the state-of-the-art ABSA model that has been developed for the SemEval 2015 and 2016 tasks and associated data, namely the Hybrid Approach for ABSA with BERT embeddings and hierarchical attention (HAABSA++) model proposed by [22].

The hybrid model uses an ontology-driven approach to predict sentiment, while the deep learning acts as a backup for when the ontology-driven approach proves inconclusive. The HAABSA++ model extends the LCR-Rot-hop model by adding a hierarchical attention layer to the deep learning algorithm, improving the model's flexibility. HAABSA++ also replaces the non-contextual word embeddings in HAABSA with BERT [5] word embeddings, which are able to take into account the context around each word. As the deep learning model benefits from multiple iterations of extensions, it has increased in complexity, further decreasing the explainability of the model's behavior.

Since the deep learning algorithms in the HAABSA++ model have reached a highly developed stage, it is of crucial importance that tools for interpretation stay up to par to be able to understand model behavior. Therefore, this paper focuses on the following research question: *To what extent can we correctly interpret and explain the behavior of the HAABSA++ model?*

To obtain model interpretability and explainability, post-hoc model-agnostic interpretation methods are used. Model-agnostic interpretation methods can be divided into global and local methods. Local interpretation methods concentrate on explanations of single instances, while global interpretation methods describe overall model behavior. Therefore, local interpretation methods are suitable for exploring the reason behind certain unexpected predictions, which gives insights into why a model predicts this outcome. On the other hand, global explanation methods are useful in summarizing the reasons for overall behavior, which makes it easier to identify trends or divergence points.

To explain HAABSA++ model behavior, a SHapley Additive exPlanations (SHAP) approach is proposed. SHAP uses the concept of Shapley values to capture the marginal contribution of features on the model prediction. In addition to local explanation, SHAP has the added benefit of aggregating to a global

interpretation representation, giving a clear depiction of which words contribute the most to the sentiment prediction on a global level.

Various researchers have attempted to interpret the HAABSA++ model and its predecessors. [14] analyzes local model predictions by focusing on surrogate models, as the authors create two sampling methods that feed an interpretability algorithm based on Local Interpretable Model-Agnostic Explanations (LIME). Furthermore, [6] attempts to increase global interpretation of the LCR-Rot-hop++ model using diagnostic classifiers, testing various hypotheses to break down whether certain layers of the model encode specific types of information. This paper builds upon previous research in two ways. First, the research introduces new interpretation methods that do not use surrogate models to analyze local model predictions. Second, in addition to local interpretation methods, this research builds a basis for direct global interpretation of the HAABSA++ model. The code of our proposed solutions is made publicly available at `https://github.com/kzyeung/SHAPHaabsa_plus_plus`.

This paper is structured as follows. Section 2 discusses previous work regarding ABSA and model-agnostic methods. Then, Sect. 3 illustrates the data, followed by, in Sect. 4, an overview of the HAABSA++ model and the proposed SHAP approaches. Next, Sect. 5 presents the obtained results. Last, Sect. 6 gives our conclusion and suggestions for future research.

2 Related Work

This section describes prior academic literature relevant to the topics in this paper. Section 2.1 gives an overview of the background and development of the state-of-the-art HAABSA++ model. Next, Sect. 2.2 goes into existing work on the relevance of understanding complex machine learning models.

2.1 Hybrid Approach to Aspect-Based Sentiment Analysis

As previously mentioned, the state-of-the-art model of interest offers a hybrid approach to ABSA. First, an ontology-based approach is considered to predict the sentiment value, as detailed in [23]. When the ontology-based approach proves inconclusive, the model makes use of a backup deep learning model, namely LCR-Rot-hop. HAABSA++ [22] extends the model by replacing the non-contextual GloVe word embeddings with deep contextual word embeddings using BERT [5]. The LCR-Rot-hop model is also updated by incorporating hierarchical attention, offering a high-level representation of the input sentence.

2.2 Understanding Black Box Models

Although an increase in complexity does not necessarily relate to an increase in accuracy [16], complex tasks where the sole purpose lies in maximizing the accuracy often do result in complex models with a lack of transparency, also known as black box models [11]. Black box models have been problematic in high-stakes

decision-making, as a lack of understanding can lead to undesired outcomes [16]. Although [16] states that black box models can still be used for the knowledge discovery process – which fits the scope of the utility of the HAABSA++ model – the mentioned problems are still relevant. A lack of understanding can lead to unjustified confidence in a model's external validity, complex decision pathways prone to human error, and a general lack of trust in the model's predictions. Hence, interpretability is often extremely important for a model to be usable, sometimes to the point that an increase in interpretability justifies a decrease in accuracy [15]. Model-agnostic interpretation methods aid in comprehending a model's behavior, often in exchange for some of its accuracy.

[8], however, argues that interpretability alone is insufficient; explainability is crucial for humans to gain trust in black box models. Although the differences between interpretability and explainability are often obscure, there are important reasons to distinguish between the two. The difference mainly lies in understanding model behavior, rather than just correctly predicting model behavior; i.e., interpretability refers to the ability to predict model outcomes, while explainability refers to understanding the relationship between output and input. Explainability ensures interpretability, but the opposite is not always the case. Consequently, this research intends to explore not only model interpretation but also model explainability. Therefore, global model-agnostic methods are explored on top of local model-agnostic methods. While local model-agnostic methods only focus on the vicinity of the instance one wishes to explain [15], global model-agnostic methods aim to understand the model in its entirety, enabling the possibility to summarize model behavior. Local interpretability is more readily applicable than global interpretability, as it only needs to stay faithful to the vicinity of the considered instance; thus, with global interpretability, predictive power is often exchanged for a more helpful overall explanation, instead of many different explanations for every possible instance. For this reason, both local and global model-agnostic methods are considered in this research.

The main method of our interest is based on Shapley values, which originate from game theory [19]. Shapley values calculate the marginal contribution of each feature to determine a fair payout. Various researchers have successfully adopted Shapley values in explaining machine learning models [4, 7, 12, 21]. Although precisely computing Shapley values is resource-intensive, many advances have been made in efficiently approximating Shapley values [1, 2]. Therefore, Shapley values present a viable method to calculate the contribution of each word to the sentiment. Additionally, [12] proposes SHAP; inspired by LIME, SHAP offers a united approach to explain any machine learning model's output using Shapley values. Furthermore, SHAP offers the benefit of being able to give global interpretability by aggregating all Shapley values.

3 Data

As our research builds upon the HAABSA++ model proposed in [22], the data and processing thereof are identical. Specifically, we utilize the SemEval 2016

contest data of restaurant reviews for aspect-based sentiment classification. The SemEval 2015 data, which was also used in [22], is a subset of the SemEval 2016 data. Therefore, we do not consider it.

The original data is in the .xml format and includes 350 and 90 reviews in the training and test sets, respectively. Reviews and sentences within reviews can contain multiple aspects, which totals to 2507 and 859 instances of sentiment-labeled aspects for the training and test sets, respectively. Each aspect is labeled a sentiment, namely 'positive', 'neutral', or 'negative'. A target word, if present, marks the word that indicates the aspect. A sentence can have multiple aspect categories, although not every aspect category has a target word. A target word is set to 'NULL' if this is the case.

Table 1. Distribution of sentiment classifications in the SemEval 2016 restaurant reviews data.

	Positive		Negative		Neutral		Total	
	N	%	N	%	N	%	N	%
Train data	1319	70.2	488	26.0	72	3.8	1880	100
Test data	483	74.3	135	20.8	32	4.9	650	100

To be able to conform to the HAABSA++ model requirements, the SemEval 2016 dataset is modified. First, all sentiment classifications without a target word are removed, as LCR-Rot-hop++ requires a target word to be able to separate the sentence in a left-center-right part. As a result, the dataset is reduced to 1880 and 650 instances for the training and test sets, respectively. Table 1 shows the distribution of each sentiment class for the remaining instances. The majority class is 'positive', representing 70.2% of the training data and 74.3% of the test data. The data is then processed using the NLTK platform [3]. Using the WordNet lexical database, the text is tokenized, tagged, and lemmatized [13].

Table 2. Distribution of sentiment classifications where LCR-Rot-hop++ is utilized.

Positive		Negative		Neutral		Total	
N	%	N	%	N	%	N	%
144	58.1	82	33.0	22	8.9	248	100

As explained in [22], the model first uses an ontology. Only when the ontology proves inconclusive, a backup method is used in the form of the LCR-Rot-hop++ mechanism described in [22]. Since the ontology is transparent in nature, our research focuses on explaining the backup method which is the black box component of the model. Out of the 650 sentiment-labeled aspects in the test

data, 402 are predicted with the ontology. The remaining 248 instances are undecided by the ontology, which is when the LCR-Rot-hop++ model is utilized. Table 2 shows the distribution of the sentiment classifications of the remaining 248 instances. Although positive classifications still account for the majority of the data, it has significantly decreased from 74.3% to 58.1%. This implies that the ontology is relatively better in predicting positive sentiment in comparison to neutral or negative sentiment.

4 Methodology

This section details the methods relevant to this research. First, Sect. 4.1 describes the HAABSA++ model. Although a detailed explanation of the model is given in [22], we provide a basic explanation needed to understand the fundamentals of our research. Next, the workings of the interpretation methods used to explain the HAABSA++ model are detailed in Sect. 4.2.

4.1 Hybrid Approach to Aspect-Based Sentiment Analysis

HAABSA++, which stands for Hybrid Approach for Aspect-Based Sentiment Analysis++, is a hybrid model designed to solve aspect-based sentiment classification problems using a combination of an ontology and a deep learning model. The model first attempts to classify the sentiment towards a target using an ontology. When the ontology proves inconclusive, the model switches over to a deep learning model called LCR-Rot-hop++ [22].

Ontology. The domain sentiment ontology is designed to find the possible sentiment expression of a word depending on the aspect. For instance, the word *'small'* implies a positive sentiment in the context of *'price'*, but often a negative sentiment in the context of *'portions'*. The model uses the NLTK platform to tokenize the text; each word is lemmatized based on the part-of-speech tagging using the WordNet lexical database within the NLTK platform. The ontology is manually constructed as explained in [18] using the *OntoClean* method [9]. Although [18] originally uses a support vector machine model as a backup method, a revised backup method proposed by [23] and then further improved by [22] proved to result in higher performance, leading to the final LCR-Rot-hop++ model.

LCR-Rot-hop++. LCR-Rot-hop++ utilizes BERT word embeddings to generate contextual word representations, as described by [5]. In LCR-Rot-hop++ the BERT tokenization proposed in [5] is used to tokenize the text data. Afterwards, the uncased base version of BERT containing 768 dimensions is used to generate word embeddings for our data.

LCR-Rot-hop++ then divides the sentence into three parts: left, center, and right. The target word(s) in the sentence are assigned as the center, while the

remaining parts of the sentence are assigned towards the left or right parts, depending on where they stand in relation to the target word. If the sentence starts or ends with the target word, the left or right parts remain blank.

Each of the three parts is translated to their embedding representation defined by BERT. The left, center, and right parts are then fed into separate bidirectional Long Short-Term Memory (bi-LSTM) networks, which produce three sets of hidden state vectors. Bi-LSTMs combine two LSTMs, processing the text input in both a forward and backward order, to avoid bias toward words near the end of their sequence. The hidden state vectors are then fed into a repeated two-step rotatory attention mechanism, rotating over a Target2Context and Context2Target mechanism for the left and right parts sequentially, until the desired number of iterations has been reached.

To compute the Target2Context vector, an average pooling layer first produces a vector representation of the target phrase. Second, an attention mechanism assigns attention scores to the context words, which are then normalized through a softmax function. The final context vector is computed using an attention weighted combination of the hidden states. The obtained context representation vector is used to adjust the target vector representation with a similar attention mechanism called Context2Target. Again, the computed attention scores are normalized through a softmax function, and using an attention weighted combination of the hidden states the final target vector is obtained. The four obtained vectors (target and context for both left and right sequences) are combined to calculate attention scores on the sentence level, scaling the context and target vectors one last time. The second iteration uses the obtained target vectors to replace the pooling layer in the first step, to reiterate the complete process for a total of three times. Lastly, the resulting vectors are concatenated to obtain the final sentence representation, which is fed into a multi-layer perceptron to compute the sentiment prediction vector.

4.2 Interpretation Methods

As the improved performance of the backup method has led to an increase in complexity, a need has arisen to explain the model in order to better derive insights from the model predictions. This research utilizes SHAP to understand the features that contribute to the model predictions.

Shapley Values. Shapley values are a concept originally developed in cooperative game theory [19], and have become more prominent in the field of machine learning since the introduction of SHAP [12]. The aim is to calculate the exact contribution of each player so that the value of each player can be distributed fairly. This concept has been adapted to the field of machine learning by interchanging players with features such that the most important features in a machine learning model can be identified. The Shapley value evaluates all possible combinations of features in different coalitions and calculates the output value of each coalition, which is then used to calculate the marginal contribution of each feature.

First, we define the original sentence as the text including the context phrase and target phrase $\{W, T\} \in X$ in the sentence, where W is the combination of all context words and T is the target phrase. Suppose we have a sentence with N context words, then W is the set of all context words w_n: $W = [w_1, w_2, \ldots, w_{N-1}, w_N]$. The nature of the LCR-Rot-hop++ model does not allow for target T to be removed, because it uses the context words to predict the sentiment on T. Hence, the features for the SHAP model consist of the context words w_n. Each subset S includes a number of features k between 0 and N (i.e., varying between $\varnothing$ and the original context W). The amount of total possible subsets S is 2^N. The power set of W is defined as all possible subsets S of W:

$$P(W) = \{S \mid S \subseteq W\}.$$

Using an example of a sentence with $N = 4$ context words defined as $W = [w_1, w_2, w_3, w_4]$, Table 3 depicts all subsets in $P(W) \mid S \subseteq W$. The sentiment class probability of each subset is calculated using the LCR-Rot-hop++ model described in Sect. 4.1. For every subset S we obtain three different p-values: $[p_1,\ p_0,\ p_{-1}]$ for positive, neutral, and negative sentiment respectively.

Table 3. Example of the power set $P(W) \mid S \subseteq W$, $N = 4$.

k:	0	1	2	3	4
	$[\varnothing]$	$[w_1]$	$[w_1,\ w_2]$	$[w_1,\ w_2,\ w_3]$	$[w_1,\ w_2,\ w_3,\ w_4]$
		$[w_2]$	$[w_1,\ w_3]$	$[w_1,\ w_2,\ w_4]$	
		$[w_3]$	$[w_1,\ w_4]$	$[w_1,\ w_3,\ w_4]$	
		$[w_4]$	$[w_2,\ w_3]$	$[w_2,\ w_3,\ w_4]$	
			$[w_2,\ w_4]$		
			$[w_3,\ w_4]$		

To calculate the Shapley value Φ_n of word w_n, (1) is used. Here, $p(S)$ is defined as the p-value of subset S. Within the summation, this accounts for all subsets S that do not contain w_n, as is denoted by $S \subseteq W \setminus \{w_n\}$. $p(S \cup \{w_n\})$ is the p-value of the subset where w_n is added. Specifically, we calculate the difference between all subsets including w_n, and all subsets excluding w_n, to obtain the marginal p-value of adding w_n to each subset. Then, the marginal value of the subset is divided by $\binom{N-1}{k}$ before being summed together, where k is the size of S. Thus we obtain Φ_n, which is the average marginal p-value of w_n.

$$\Phi_{w_n} = \frac{1}{N} \sum_{S \subseteq W \setminus \{w_n\},\ k=|S|} \frac{(p(S \cup \{w_n\}) - p(S))}{\binom{N-1}{k}} \tag{1}$$

For instance, if we want to calculate the Shapley value of w_1, we calculate the marginal p-value of adding w_1 to each subset S that does not contain w_1. Table 4 shows each subset S and corresponding $S \cup \{w_1\}$.

Table 4. All subsets $S \subseteq W \setminus \{w_1\} \mid N = 4$ and corresponding $S \cup \{w_1\}$.

k	$\binom{N-1}{k}$	$S \cup \{w_1\}$	S
0	1	$[w_1]$	$\varnothing$
1	3	$[w_1,\ w_2]$	$[w_2]$
1	3	$[w_1,\ w_3]$	$[w_3]$
1	3	$[w_1,\ w_4]$	$[w_4]$
2	3	$[w_1,\ w_2,\ w_3]$	$[w_2,\ w_3]$
2	3	$[w_1,\ w_2,\ w_4]$	$[w_2,\ w_4]$
2	3	$[w_1,\ w_3,\ w_4]$	$[w_3,\ w_4]$
3	1	$[w_1,\ w_2,\ w_3,\ w_4]$	$[w_2,\ w_3,\ w_4]$

SHAP. SHAP is an algorithm introduced by [12] to utilize Shapley values within the field of machine learning. As Shapley values offer an intuitive way to interpret the contribution of each feature, SHAP applies the concept of Shapley values to machine learning models that are difficult to interpret. As explained before, Shapley values show the average marginal p-value of each word w_n. Due to their nature, they are inherently easy to interpret, as the size and polarity of the value linearly translate to its contribution towards the final prediction. SHAP calculates Shapley values by masking parts of the original data input, thus creating a power set $P(W)$ of all the context words w_n. The sum of the computed SHAP values adds up to the difference in the base value $b(v)$ and the final predicted p-value. The base value $b(v)$ is the predicted p-value when all context words w_n are masked; in other words, a sentence that only consists of its target T.

In this research, we propose two versions of SHAP to understand the LCR-Rot-hop++ model to a greater extent. The first model (model 1) applies SHAP only to the LCR-Rot-hop++ model after word embeddings are generated using BERT. The subsets are created after BERT word embeddings are generated, measuring the contribution of each embedded word to the sentiment prediction. The second model (model 2) applies SHAP before BERT word embeddings are created, which means that new word embeddings are created for all subsets of the sentence W. The final SHAP values measure the contribution of each word on the final sentiment classification, accounting for a change in context as a result of SHAP masking part of the sentence. Figure 1 describes the steps involved in building SHAP models 1 and 2, as well as the differences in both models. X denotes the original sentence text, consisting of the context phrase W and target phrase T. S are the subsets belonging to the power set $P(W)$ as described before. The p-values $\{p_1, p_0, p_{-1}\}$ are the output of the LCR-Rot-hop++ model, referring to the positive, neutral, and negative sentiment probability, respectively. Lastly, $\{\phi_1, \phi_0, \phi_{-1}\}$ are the SHAP values that indicate the contribution of each word to the three sentiment class probabilities.

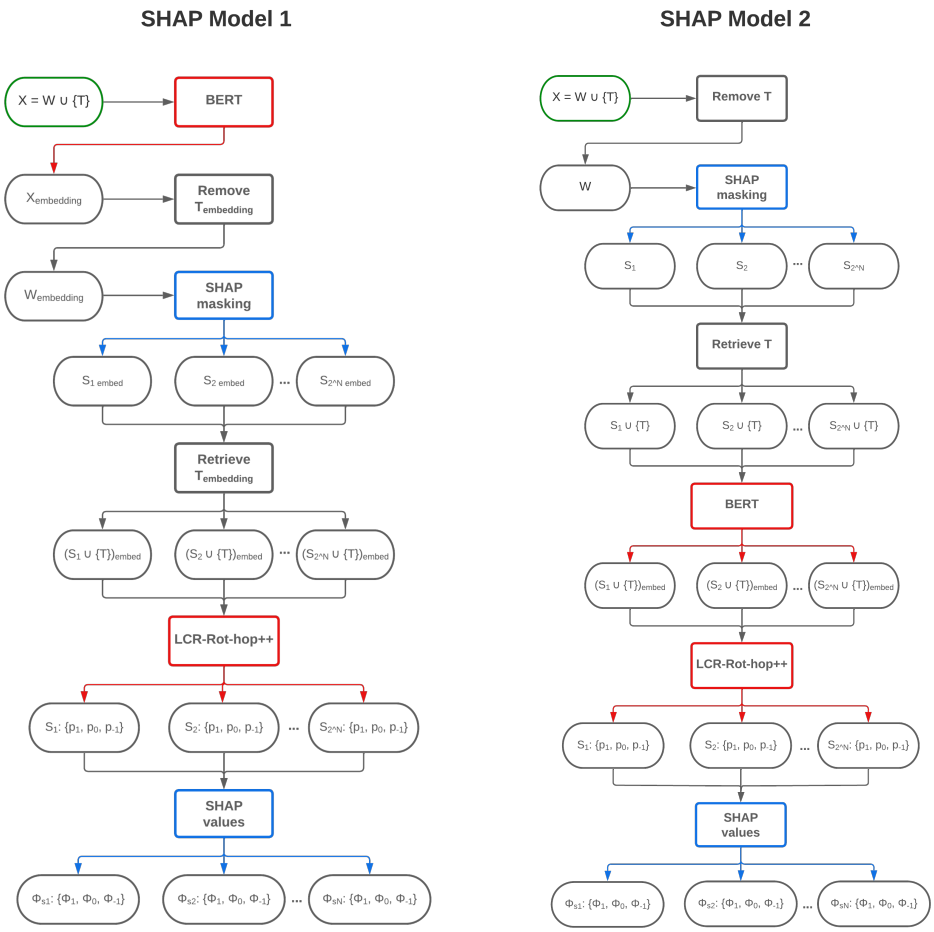

Fig. 1. SHAP integrated within LCR-Rot-hop++.

5 Results

This section presents the results of our SHAP models and the interpretation thereof. First, Sect. 5.1 demonstrates the use of SHAP on local instances, as sentiment classifications of single sentences are explained. We compare the results of SHAP model 1 and model 2. Section 5.2 continues to accumulate SHAP results to obtain global inferences on the LCR-Rot-hop++ model.

5.1 SHAP Local

Table 5 shows the results of SHAP model 1 and model 2 on a sentence that is incorrectly classified by the model, sorted by sentiment class prediction. As Table 5 reveals, sentence X_1: *'The bus boy even spotted that my table was shaking a stabilized it for me'* is classified as negative, even though the sentiment is positive in reality. The columns Φ_1, Φ_0, and Φ_{-1} show the SHAP values of each word, which indicates their contribution towards a positive, neutral, and negative sentiment predictions, respectively. The SHAP values for the target word *'bus boy'* is the base value which is the prediction when all context words are masked. Together they add up to the sentiment classification probabilities for each class.

Table 5. SHAP model 1 and model 2 results for sentence X_1.

The bus boy even spotted that my table was shaking a stabilized it for me

Target:	bus boy
Sentiment:	Positive
Prediction:	Negative

SHAP model 1				*SHAP model 2*			
w_n	ϕ_1	ϕ_0	ϕ_{-1}	w_n	ϕ_1	ϕ_0	ϕ_{-1}
the	-0.003	-0.054	0.057	shaking	-0.055	-0.009	0.064
spotted	-0.031	-0.007	0.037	the	0.015	-0.050	0.035
was	-0.016	-0.001	0.017	my	-0.021	-0.003	0.024
it	-0.008	-0.003	0.011	me	0.015	-0.035	0.020
table	-0.009	-0.001	0.010	it	-0.016	-0.003	0.019
shaking	-0.007	-0.001	0.009	a	-0.006	-0.010	0.016
even	0.003	-0.011	0.008	spotted	-0.009	-0.004	0.013
a	0.002	-0.007	0.006	was	-0.001	-0.008	0.009
stabilized	0.005	-0.008	0.003	stabilized	-0.008	0.006	0.002
that	0.014	-0.014	0.000	table	0.003	-0.001	-0.002
my	0.025	-0.004	-0.021	that	0.015	-0.007	-0.008
me	0.037	-0.011	-0.026	even	0.013	0.005	-0.018
for	0.036	-0.008	-0.029	for	0.064	-0.009	-0.055
bus boy	0.365	0.136	0.498	**bus boy**	0.403	0.133	0.463
P-value	0.412	0.005	**0.582**	P-value	0.412	0.005	**0.582**

Starting with SHAP model 1, Table 5 shows that the words *'the'*, *'spotted'*, and *'was'* contribute the most towards a negative sentiment prediction. This does not immediately make sense from a linguistic point of view, however, it can be explained by the fact that SHAP model 1 computes word embeddings from the complete sentence, where words contain information about the original context. The context of the word *'the'* is, therefore, most associated with a negative sentiment, as the methodology of SHAP model 1 implies that the embedding of *'the'* has captured a context that is similar to other contextual embeddings with a negative sentiment. Likewise, the embedding vectors for *'spotted'* and

'was' from the original sentence are the closest to other embedding vectors with negative associations.

The p-values are identical for SHAP model 2, since both SHAP model 1 and 2 aggregate towards the embeddings of the full sentence. Nevertheless, the SHAP values are significantly different, as SHAP model 2 regenerates the BERT embeddings for every single subset $S \subseteq W \in X_1$, capturing information about a different context each time. This results in the word *'shaking'* as the highest contributor towards the negative sentiment class. Words weak in meaning like *'the'*, *'it'*, and *'a'* still score relatively high, which implies that adding these words to subsets where they do not exist has a relatively large effect.

Linguistically, the text *'table was shaking'* is the only part that has a negative connotation, which suggests that model 2 is more useful in terms of determining the individual words that lead to a negative prediction, rather than the context the word is captured in.

As SHAP model 2 is presumed to assign contribution more towards individual words, we present one more local interpretation using SHAP model 2. Table 6 shows the results of the model on a correctly predicted sentence X_2: *'For the amount of food we got the prices should have been lower.'* as negative. In this case, the word *'lower'* is attributed the highest contribution towards a negative sentiment. Following *'lower'*, the words *'we'*, *'been'*, and *'have'* are shown to have a significant contribution as well. These words are weak in meaning, as they act as function words. The fact that they still represent a large part of the contribution towards a negative sentiment implies that these words still add important information about the context, possibly defining how words are related to each other.

5.2 SHAP Global

To achieve global interpretation, SHAP simply calculates the mean SHAP values for all words to see their average contribution over the model predictions. The global interpretations are shown separately for each sentiment class.

Since SHAP generates all possible subsets for each sentence, the computational time is long. Therefore, 5% of the dataset is randomly sampled to obtain a segment of 12 instances to demonstrate the capabilities of SHAP. As SHAP model 2 is determined to be better at capturing the contribution of adding individual words, rather than the context that they originally consist of, we aggregate words only for SHAP model 2. The results are presented in Table 7.

Table 7 shows the words with the highest nine average SHAP values per sentiment classification. The last row contains the sum of the SHAP values of all remaining words. Unsurprisingly, the neutral sentiment barely has any contributing words, since neutral sentences are the clear minority in the dataset. The highest contributions belong to *'!'*, *'plenty'*, *'sure'*, *'not'*, whereas only *'not'* belongs to a negative sentiment classification. This is most likely the cause of positive being the majority class, which means there are more sentences and words that are positive.

Table 6. SHAP model 2 results for sentence X_2.

For the amount of food we got the prices should have been lower.

Target:			food
Sentiment:			Negative
Prediction:			Negative
w_n	ϕ_1	ϕ_0	ϕ_{-1}
lower	-0.031	-0.029	0.060
we	-0.053	0.008	0.045
been	-0.029	-0.010	0.039
have	-0.018	-0.004	0.022
of	-0.028	0.013	0.015
the1	-0.008	-0.004	0.012
the2	-0.001	-0.008	0.009
prices	-0.001	-0.006	0.007
got	0.009	-0.012	0.003
for	0.034	-0.023	-0.001
amount	0.047	-0.021	-0.026
should	0.075	-0.006	-0.069
food	0.389	0.118	0.491
P-value	0.387	0.007	**0.606**

Table 7. SHAP 2 global - averaged over a random sample with similar distribution containing 5% of the data.

Positive		Neutral		Negative	
Word	μ_ϕ	Word	μ_ϕ	Word	μ_ϕ
!	0.24	continued	0.02	not	0.11
plenty	0.12	of	0.01	continued	0.07
sure	0.10	we	0.01	though	0.06
pacific	0.09	stopped	0.01	over-	0.06
dinner	0.09	stabilized	0.01	no	0.06
should	0.08	either	0.00	shaking	0.06
yuppies	0.07	on	0.00	lower	0.06
old	0.07	take	0.00	removed	0.06
variety	0.06	tenderizer	0.00	rated	0.05
remaining	-0.39	**remaining**	-0.87	**remaining**	-0.3

6 Conclusion

The complexity of the LCR-Rot-hop++ model that acts as a backup method in the HAABSA++ method can lead to intricacies. Because the LCR-Rot-Hop++ model utilizes context-aware BERT embeddings, features are never completely separable from each other. In this paper, we propose two SHAP approaches, SHAP model 1 and model 2, to better capture the behavior of the LCR-Rot-hop++ model. SHAP is able to infer local and global interpretations, offering

the user insights into why a model did, does, and will do something. We conclude that SHAP model 1 is more likely to capture the contribution of specific contexts, since words maintain their information about the original context, even after being subsetted by SHAP. SHAP model 2 partially refutes this by generating new BERT embeddings for each subset of the original sentence. As a result, SHAP model 2 is better at capturing the contribution of specific words. Thus, words strong in meaning are more likely to be assigned a high contribution than function words that depend on their context. Future research could consider removing function words before feeding them to the SHAP model to avoid dilution of the meaning of other words.

Although SHAP offers a lot of potential to gain a better understanding of model behavior, it is not without its disadvantages. One of the main problems with SHAP is the required computational power. Additionally, although SHAP model 2 proves to be more effective at defining the contributions of individual words, the fact that new embeddings are created for every single subset adds another layer that is computationally intensive. The earlier suggestion to potentially remove function words could compensate for this by reducing the number of subsets, but it could also be worth exploring methods that only estimate Shapley values instead of using precise calculations [1, 2].

References

1. Aas, K., Jullum, M., Løland, A.: Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. Artificial Intelligence **298**, 103502 (2021)
2. Ancona, M., Oztireli, C., Gross, M.: Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In: 36th International Conference on Machine Learning (ICML 2019). vol. 97, pp. 272–281. PMLR (2019)
3. Bird, S., Klein, E., Loper, E.: Natural language processing with Python: Analyzing text with the natural language toolkit. O'Reilly Media, Inc. (2009)
4. Cohen, S., Ruppin, E., Dror, G.: Feature selection based on the Shapley value. In: 19th International Joint Conference on Artificial Intelligence (IJCAI 2005). pp. 665–670. Morgan Kaufmann (2005)
5. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: 2019th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019). pp. 4171–4186. ACL (2019)
6. Geed, K., Frasincar, F., Truşcă, M.M.: Explaining a deep neural model with hierarchical attention for aspect-based sentiment classification using diagnostic classifiers. In: 22nd International Conference on Web Engineering (ICWE 2022). LNCS, vol. 13362, pp. 268–282. Springer (2022)
7. Ghorbani, A., Zou, J.: Data Shapley: Equitable valuation of data for machine learning. In: 36th International Conference on Machine Learning (ICML 2019). pp. 2242–2251. PMLR (2019)
8. Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M., Kagal, L.: Explaining explanations: An overview of interpretability of machine learning. In: 5th International Conference on Data Science and Advanced Analytics (DSAA 2018). pp. 80–89. IEEE (2018)

9. Guarino, N., Welty, C.: Evaluating ontological decisions with OntoClean. Communications of the ACM **45**(2), 61–65 (2002)
10. Liu, B.: Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Cambridge University Press, second edn. (2020)
11. London, A.J.: Artificial intelligence and black-box medical decisions: Accuracy versus explainability. Hastings Center Report **49**(1), 15–21 (2019)
12. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: 31st Annual Conference on Neural Information Processing Systems (NIPS 2017). pp. 4765–4774. Curran Associates (2017)
13. Miller, G.A.: WordNet: An Electronic Lexical Database. MIT Press (1998)
14. Miron, V., Frasincar, F., Trusca, M.M.: Explaining a deep learning model for aspect-based sentiment classification using post-hoc local classifiers. In: 28th International Conference on Applications of Natural Language to Information Systems (NLDB 2023). LNCS, vol. 13913, pp. 79–93. Springer (2023)
15. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should I trust you?" Explaining the predictions of any classifier. In: 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2016). pp. 1135–1144. ACM (2016)
16. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019)
17. Schouten, K., Frasincar, F.: Survey on aspect-level sentiment analysis. IEEE Transactions on Knowledge and Data Engineering **28**(3), 813–830 (2015)
18. Schouten, K., Frasincar, F.: Ontology-driven sentiment analysis of product and service aspects. In: 15th Extended Semantic Web Conference (ESWC 2018). LNCS, vol. 10843, pp. 608–623. Springer (2018)
19. Shapley, L.S.: Quota solutions of n-person games. Rand Technical Report p. 343 (1952)
20. Shin, H., Perdue, R.R., Pandelaere, M.: Managing customer reviews for value cocreation: An empowerment theory perspective. Journal of Travel Research **59**(5), 792–810 (2020)
21. Štrumbelj, E., Kononenko, I.: Explaining prediction models and individual predictions with feature contributions. Knowledge and Information Systems **41**, 647–665 (2014)
22. Truşcă, M.M., Wassenberg, D., Frasincar, F., Dekker, R.: A hybrid approach for aspect-based sentiment analysis using deep contextual word embeddings and hierarchical attention. In: 20th International Conference on Web Engineering (ICWE 2020). LNCS, vol. 12128, pp. 365–380. Springer (2020)
23. Wallaart, O., Frasincar, F.: A hybrid approach for aspect-based sentiment analysis using a lexicalized domain ontology and attentional neural models. In: 16th Extended Semantic Web Conference (ESWC 2019). LNCS, vol. 11503, pp. 363–378. Springer (2019)