SBERTiment: A New Pipeline to Solve Aspect Based Sentiment Analysis in the Zero-Shot Setting

Matteo Muffo, Aldo Cocco, Edoardo Negri, Enrico Bertino
Indigo.ai
{matteo, aldo, edoardo, e}@indigo.ai

Devi Veena Sreekumar, Giorgio Pennesi
Whirlpool Management EMEA
{devi_veena_sreekumar, giorgio_pennesi}@whirlpool.com

Riccardo Lorenzon
Vedrai
riccardo.lorenzon@vedrai.com

Abstract

The field of Natural Language Processing is gaining increased attention for the Aspect Based Sentiment Analysis task due to its ability to provide fine-grained information. This paper introduces SBERTiment, a novel approach to perform Aspect Based Sentiment Analysis. The method extracts relevant topics along with their sentiments from the input text by using a 2-step pipeline. In the first step, a token classification model is used to identify the relevant aspect terms and their sentiments. In the second step, a Sentence-BERT embedding model maps each aspect term to a predefined aspect category. Our approach has been tested on benchmark datasets and has achieved scores that are comparable to the best-performing methods. The pipeline is also able to perform zero-shot classification, which means it can extract information in unseen domains without additional training. When evaluated on a dataset with unseen aspect categories, SBERTiment achieved the best score among benchmark approaches.

Introduction

The Aspect Based Sentiment Analysis (ABSA) task analyzes a text input and extracts the relevant aspects together with their associated sentiment. Over time, various subproblems emerged, providing different kinds of output and levels of detail. The taxonomy of such subproblems is well described in Zhang et al. (2022). Owing to the fine-grained level of analysis that an ABSA model can potentially provide, this field has acquired plenty of attention from Natural Language Processing (NLP) researchers in recent years.

This paper introduces SBERTiment, a novel approach to ABSA that performs well on benchmark datasets and can be adapted to different domains without additional training. The method extracts sentiments of relevant aspect categories belonging to a pre-defined set.

The pipeline of SBERTiment consists of two steps. In the first step, we use a BERT (Devlin et al. 2018) token classification model, referred to as the *aspect extractor*, to extract relevant aspect terms and their sentiments. In the second step, we use a Sentence-BERT (Reimers and Gurevych 2019) embedding model, referred to as the *topic matcher*, to map each extracted aspect term to a predefined aspect category by considering the entire sentence as context.

We train and evaluate SBERTiment on two popular benchmark datasets for ABSA: Semeval15 Restaurants (Pontiki et al. 2015) and Semeval16 Restaurants (Pontiki et al. 2016), which consist of reviews about the restaurant industry. Our method achieves results that are comparable to the best performing approaches. To test SBERTiment in a zero-shot setting, we use the Semeval15 Laptops (Pontiki et al. 2015) and Semeval16 Laptops (Pontiki et al. 2016) datasets, which are collections of laptop reviews containing different aspect categories compared to the Restaurants datasets. When evaluated on the Laptops data without any additional training, SBERTiment achieves the best scores compared to other benchmark approaches evaluated in the same way, making it a viable alternative for ABSA tasks in data-scarce environments or when no training data is available.

We make the data and code[1] used in these experiments publicly available to ensure reproducibility.

The paper is organized as follows: in the *Related work* section, we review related works in the field; in the *Method* section, we describe our pipeline; in the *Experiments and Results* section, we present the experiments we conducted and the results we obtained; in the *Conclusions and Future Works* section, we summarize our findings and discuss potential future work.

Related Work

The Aspect Based Sentiment Analysis (ABSA) problem aims to provide a detailed sentiment analysis by determining the sentiment of each relevant aspect in the input text. Following the terminology introduced in Zhang et al. (2022), an *aspect term* is a word or span of words that represent an opinion target, such as "salmon fillet" in the sentence "The salmon fillet was delicious". *Aspect categories* are a finite set of topics that each aspect term can be associated with (e.g. given the aspect categories {food, service, price}, the aspect term "*salmon fillet*" from the previous example belongs to the category *food*). As the field of ABSA has advanced, several subproblems have arisen, each with a differ-

[1] Code and data available at: https://github.com/mmuffo94/SBERTiment

ent level of detail in the output. A comprehensive survey of these subproblems and approaches is proposed by Zhang et al. (2022). Following their taxonomy, since our study primarily focuses on aspect categories and sentiments, we consider Aspect Category Sentiment Analysis (ACSA) methods to be related to our approach.

In this context, Hu et al. (2018) propose a Transformer neural network with regularized attention trained in a multitask framework to both extract the aspect categories in the sentence and predict the sentiment of all possible aspect categories. Wan et al. (2020a) study a BERT network trained to classify each possible aspect category-sentiment pair with labels *yes* or *no*, reconducting the problem to a binary classification. Ma, Peng, and Cambria (2018) propose a LSTM network (Hochreiter and Schmidhuber 1997) with hierarchical attention mechanism which leverages the SenticNet commonsense knowledge base (Cambria and Hussain 2015) to classify aspect category-sentiment pairs. Wan et al. (2020b) propose to train a binary BERT classifier (Devlin et al. 2018) to determine whether each possible aspect category-sentiment combination occurs in the input sentence. Cai et al. (2020) propose a Hierarchical Graph Convolutional Network which is able to leverage inner-relations among categories and inter-relations among categories and sentiments. Li et al. (2020) propose a model to jointly predict both aspect categories and sentiments, but they use a shared sentiment prediction layer in order to solve data deficiency problems which may arise for some aspect categories. Wu et al. (2021) leverage BERT to obtain feature vectors from aspect-sentence input pairs, use an LSTM network to model aspect and sentence representations and finally adopt a graph convolutional network to capture dependencies between aspect and sentence. Schmitt et al. (2018) use a convolutional neural network which receives as input a pair *(aspect category, sentence)* and provides as output a label from the set {*POSITIVE, NEGATIVE, NEUTRAL, N/A*} which determines if the considered aspect category has positive, negative, neutral polarity or is not present in the sentence. Liu et al. (2021) propose to use a sequence-to-sequence model to perform the ACSA tasks by providing a prediction score to pre-designed string templates containing output aspect categories and sentiments. Zhang et al. (2021) use a sequence-to-sequence T5 model (Raffel et al. 2020) to generate the whole predicted output as a string.

Compared to existing approaches, SBERTiment offers a new solution to ACSA by using a pipeline approach that involves a Sentence-BERT model in the second step. This allows SBERTiment to perform zero-shot ACSA without the need for generative approaches or architecture modification.

Our method is tested in a zero-shot setting, which makes the work by Shu et al. (2022) relevant to our study. In their paper, the authors propose a Natural Language Inference model that can perform ABSA tasks without any domain-specific training. However they study aspect term-related problems while our study focuses on aspect categories.

Method

In this work we propose SBERTiment, a new method to perform Aspect Based Sentiment Analysis that extracts aspect terms with relative aspect categories and associated sentiments from an input text. Both the set of possible aspect categories and sentiments are finite, problem-specific and pre-defined a priori. We will denote the set of possible aspect categories for a given problem with $\{C_1, C_2, \dots, C_m\}$ while we denote the set of possible sentiments with $\{t_1, t_2, \dots, t_n\}$. We underline that in this work we will analyse problems having {POSITIVE, NEGATIVE, NEUTRAL} as set of possible sentiments exclusively.

The approach that we propose consists of a 2-step pipeline that we describe in detail below. For the purpose of clarity, we report an illustration in figure 1.

- **Aspect extraction:** in the first step we leverage a BERT (Devlin et al. 2018) token classification model (denoted as *aspect extractor*) to extract from the input text the relevant aspect terms together with their sentiment. Specifically, this model receives plain text as input and for each word provides a label in the set {`POS, NEG, NEU, O`}. While `POS`, `NEG` and `NEU` are the labels assigned to words which are relevant aspects with positive, negative or neutral sentiment, the label `O` is assigned to all words which do not correspond to a relevant aspect. In addition, if subsequent words are classified with the same label (different from `O`), they are considered to belong to the same aspect term. For example, given the input sentence *"The salmon fillet was delicious"* and the output labels {`O, POS, POS, O, O`}, *salmon fillet* is considered a single aspect term because both *salmon* and *fillet* have the label `POS`. This makes it an end-to-end approach to extract both aspect terms and sentiment with just one forward pass through the model. Formalizing what we described above and denoting with $S = \{w_1, w_2, \dots, w_q\}$ the input text, we have that the output of this first step of the pipeline is a list of *(aspect term - sentiment)* couples $(a_1, t_{j_1}), (a_2, t_{j_2}), \dots, (a_h, t_{j_h})$ with $a_i \subset S$ and $t_j \in$ {`POS, NEG, NEU`}. Irrelevant aspects with label `O` are excluded from the output. The number of extracted couples h depends on the input sentence S and can be equal to 0. We underline that the *aspect extractor* can be used in a generalized way: the same model trained on a particular dataset can extract aspect terms and sentiments for different problems and domains.
- **Topic matching:** in the second step we use a Sentence-BERT (Reimers and Gurevych 2019) embedding model (denoted as *topic matcher*) to map each aspect term $a_1, a_2, \dots, a_h$ extracted in the first step to one of the predefined aspect categories. Specifically, the input for this step is a string of the form `"{sentence} [SEP] {aspect term}"`, while the output is an aspect category. For example, the input *"The salmon fillet was delicious [SEP] salmon fillet"* provides as output the aspect category *Food*. Given an aspect term a_i and its original sentence S, at inference time the topic matching process is the following:
  - we use a Sentence-BERT model to encode all possible aspect categories C_i. This set of sentence embeddings is denoted as $\{\mathbf{c_1}, \mathbf{c_2}, \dots, \mathbf{c_m}\} \subset \mathbb{R}^d$, where d is the embedding dimension. These vectors can be stored for

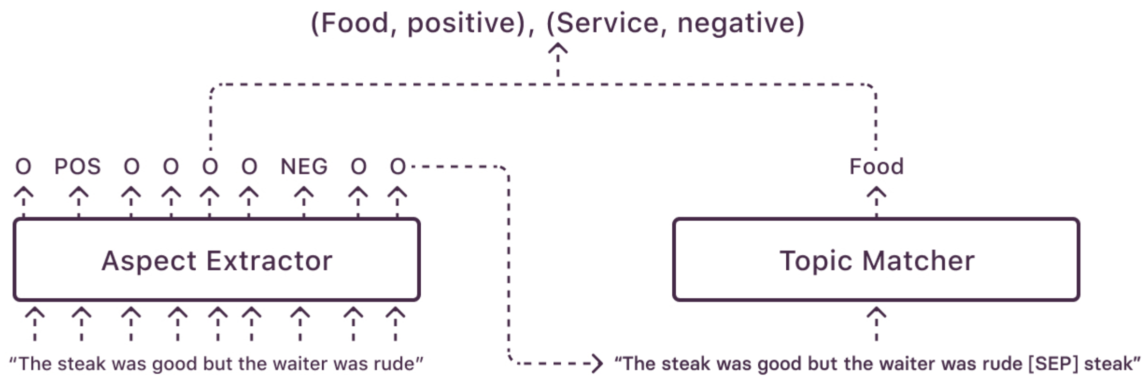

Figure 1: Illustration of the pipeline that we propose. In the example the input text is *"The steak was good but the waiter was rude"* and the set of possible aspect categories is {Food, Price, Service}

future use after being generated, so this process only needs to be done once;

– with the same model we compute a sentence embedding $\mathbf{q} \in \mathbb{R}^d$ for the input text;

– we get cosine similarity scores between the input sentence embedding and all the aspect categories sentence embeddings

$$s_i = \frac{\mathbf{q} \cdot \mathbf{c_i}}{\|\mathbf{q}\|\|\mathbf{c_i}\|}, \forall i \in \{1, \ldots, m\} \tag{1}$$

– given $s_{k_i} = max(\{s_1, s_2, \ldots, s_m\})$ as the highest cosine similarity score, we take C_{k_i} as the output aspect category predicted by the *topic matcher* model for the aspect category a_i.

We repeat this process for each aspect term $a_1, \ldots, a_h$ obtaining as output of the *topic matching* step a list of predicted aspect categories $C_{k_1}, C_{k_2}, \ldots, C_{k_h}$.

This second step can be used in a generalized way: once trained on a particular dataset, the same model can be used in different domains extracting unseen aspect categories. This is possible thanks to the sentence embedding approach adopted, which only requires the aspect category names to compute the problem-specific embeddings $\{\mathbf{c_1}, \mathbf{c_2}, \ldots, \mathbf{c_m}\}$.

At the end of both steps described above the output provided by our pipeline is a list of *(aspect term, aspect category, sentiment)* triplets $(a_1, C_{k_1}, t_{j_1}), (a_2, C_{k_2}, t_{j_2}), \ldots, (a_h, C_{k_h}, t_{j_h})$. At inference time, our pipeline only requires two forward passes, one for the *aspect extractor* and one for the *topic matcher*, thus offering a faster prediction time than other solutions such as Schmitt et al. (2018), which requires m forward passes (one for each possible aspect category).

In light of these observations, the proposed pipeline can be employed in a **zero-shot setting**, meaning that it can be trained on a given dataset and then, without the need for further training, it can be used to extract information in different domains with previously unseen aspect categories.

Experiments and Results

In this section, we present the experiments we conducted to evaluate our proposed pipeline and discuss their results.

Data

We conducted our experiments using four popular benchmark datasets containing reviews written in English. The first two are Semeval15 Restaurants (Pontiki et al. 2015) and Semeval16 Restaurants (Pontiki et al. 2016), both composed of reviews about restaurants. Each observation (review) is annotated with a varying number of aspect terms, entities (eg. food, drinks, location, etc.), attributes (eg. general, quality, prices, etc.) and sentiments (positive, negative or neutral). As stated in Pontiki et al. (2015), we use entity-attribute pairs as aspect categories (e.g. FOOD_GENERAL). The sets of possible entities and aspects are the same in both Semeval15 and Semeval16 Restaurants. We chose these two benchmark datasets for our pipeline since they are annotated with *(aspect term, aspect category, sentiment)* triplets, which is a necessity for training both steps.

The last two datasets are Semeval15 Laptops (Pontiki et al. 2015) and Semeval16 Laptops (Pontiki et al. 2016), which are composed of reviews about laptops. Each observation in these collections is annotated with a varying number of entities, attributes and sentiments, but aspect term annotations are not provided. Also in this case we use entity-attribute pairs as aspect categories. We emphasize that both Laptop datasets have the same set of aspect categories and that they differ from those in the Restaurant datasets. In our study, we use Laptop datasets to test SBERTiment on a domain with unseen aspect categories.

We report the number of observations in table 1 and a complete list of entities and attributes in tables 2 and 3.

Dataset	Train set	Test set
Semeval15 Restaurants	1120	582
Semeval16 Restaurants	1708	587
Semeval15 Laptops	-	644
Semeval16 Laptops	-	573

Table 1: Number of sentences in the studied datasets.

Experiments

Since we evaluate our pipeline for the Aspect Category Sentiment Analysis (ACSA) task, we want to extract a list $(C_1, t_{j_1}), (C_2, t_{j_2}), \ldots, (C_h, t_{j_h})$ of all the *(aspect category-sentiment)* couples contained in a text. Coherently to Pontiki et al. (2015) the metrics used to evaluate the tested methods are micro-Precision, micro-Recall and micro-F1 scores, computed comparing the predicted *(aspect category-sentiment)* couples with the respective gold pairs. We conducted two experiments:

- **Evaluation on benchmark datasets:** in this group of experiments we train the studied methods on the Semeval15 Restaurants and Semeval16 Restaurants train sets and we evaluate them on the relative test set. The objective of this group of experiments is to assess the performance of a method on common benchmark datasets and to determine its ability to perform ACSA in a classical context where train and test data come from the same domain and have the same labels.
- **Zero-shot evaluation on unseen aspect categories:** in this set of experiments we use the models previously trained on the Restaurants training sets and evaluate them on the Laptops test sets, without any additional training on the Laptops data. The purpose of this experiments is to evaluate the ability of a method to perform ACSA on a domain with unseen labels.

Compared Approaches

We compare the performances of our pipeline with several ACSA systems. For what concerns the evaluations on Semeval15 and Semeval16 Restaurants datasets, we compare our pipeline with all the approaches reported in Cai et al. (2020):

- **Cartesian-BERT**: method proposed in Wan et al. (2020a) which classifies each possible aspect category-sentiment pair with labels *yes* or *no* and uses a BERT network as sentence encoder.
- **Pipeline-BERT**: a pipeline method evaluated in Cai et al. (2020) which first identifies aspect categories present in a sentence and then classifies their relative sentiments. Also in this case a BERT network is used as sentence encoder.
- **AddOneDim-LSTM**: method proposed in Schmitt et al. (2018).
- **AddOneDim-BERT**: method proposed in Schmitt et al. (2018) but using a BERT network as sentence encoder.
- **Hier-BERT**: approach proposed in Cai et al. (2020) which do not model aspect category inner-relations and category-sentiment inter-relations.
- **Hier-T-BERT**: method proposed in Cai et al. (2020) which models aspect category and sentiment relations with a Transformer (Vaswani et al. 2017) block.
- **Hier-GCN-BERT**: approach proposed in Cai et al. (2020) which models aspect category and sentiment relations with a Graph-Convolutional Network.

Moreover, we include in this evaluation the text generation method proposed in Zhang et al. (2021), denoted as **Seq2seq**. With the exception of Seq2seq and SBERTiment, all the scores reported in the *Results* section relative to this experiments are taken from Cai et al. (2020).

For what concerns the zero-shot evaluations on the unseen aspect categories of the Semeval15 and Semeval16 Laptops datasets we compare our pipeline with the AddOneDim-BERT and Seq2seq approaches. We chose to include AddOneDim-BERT in this analysis since it was the best method among those able to perform zero-shot ACSA, while we included the Seq2seq approach since we wanted to test a text generation model in this context. We couldn't include the Hier-BERT, Hier-Transformer-BERT and Hier-GCN-BERT methods since their architectures do not allow to perform zero-shot classification.

Training details

In this section we report the details of trainings conducted both for our pipeline and for the methods that we adopted as comparison.

Regarding SBERTiment, we train the aspect extractor BERT models using a classical token classification approach with a cross-entropy loss objective function. We adopt `bert-base-uncased`[2] as starting model checkpoint and we fine-tune an aspect extractor model for each of the two Resturants train sets described in section *Data* for 7 epochs, using an initial learning rate of $5 \cdot 10^{-5}$ and a batch size of 16. For the topic matcher, we train Sentence-BERT models by introducing negative examples pairs, namely *(sentence, aspect category)* couples with incorrect aspect categories, in the training data. We add negative examples in such a way that for each sentence we have a total of 5 observations divided among positive and negative examples. We adopt `all-mpnet-base-v2`[3] as starting model checkpoint and we fine-tune a topic matcher model for each of the two Restaurants datasets described in section *Data* for 6 epochs, with an initial learning rate of $2 \cdot 10^{-5}$ and a batch size of 16 using a contrastive loss objective function (Reimers and Gurevych 2019).

Similarly, for the *AddOneDim-BERT* approach we train one model on each of the Semeval15 and Semeval16 Restaurants train sets. As for the *topic matcher* we add negative examples, namely *(sentence, aspect category)* pairs with incorrect aspect category and "NA" label, in training sets in such a way that for each training sentence we have a total of 7 observations divided among positive and negative examples. We train each AddOneDim-BERT model for 8 epochs

[2]Available at: https://huggingface.co/bert-base-uncased

[3]Available at: https://huggingface.co/sentence-transformers/all-mpnet-base-v2

Semeval15 and Semeval16 Restaurants	
Entities	Attributes
RESTAURANT, FOOD, DRINKS, AMBIENCE, SERVICE, LOCATION	GENERAL, PRICES, QUALITY, STYLE & OPTIONS, MISCELLANEOUS

Table 2: Entities and attributes for the Semeval15 and Semeval16 Restaurants datasets. `ENTITY_ATTRIBUTE` couples are used as aspect categories for our experiments.

Semeval15 and Semeval16 Laptops	
Entities	Attributes
LAPTOP, DISPLAY, KEYBOARD, MOUSE, MOTHERBOARD, CPU, FANS & COOLING, PORTS, MEMORY, POWER SUPPLY, OPTICAL DRIVES, BATTERY, GRAPHICS, HARD DISK, MULTIMEDIA DEVICES, HARDWARE, SOFTWARE, OS, WARRANTY, SHIPPING, SUPPORT, COMPANY	GENERAL, PRICE, QUALITY, OPERATION & PERFORMANCE

Table 3: Entities and attributes for the Semeval15 and Semeval16 Laptops datasets. `ENTITY_ATTRIBUTE` couples are used as aspect categories for our experiments.

with an initial learning rate of $1 \cdot 10^{-4}$ and a batch size of 16. Also in this case we use `bert-base-uncased` as starting model checkpoint.

To determine the number of negative examples to include both for the *topic matcher* and the AddOneDim-BERT models we perform an hyperparameter exploration by splitting original training sets in train and validation sets, with ratio 9:1. We varied the total number of examples per sentence (divided among positive and negative examples) in the set $\{2, 5, 7, 10, 13\}$.

Lastly, we train two *Seq2seq* models on Semeval15 Restaurants and Semeval16 Restaurants train sets respectively. We use `google/t5-v1_1-small`[4] (Raffel et al. 2020) as starting model checkpoint and we train each Seq2seq network for 10 epochs with an initial learning rate of $1 \cdot 10^{-4}$ and a batch size of 9.

We use Adam (Kingma and Ba 2014) as optimizer and we rely on the Huggingface Transformers (Wolf et al. 2020) and Sentence Transformers (Reimers and Gurevych 2019) libraries as codebases.

Results

The results of SBERTiment on the Restaurants dataset, presented in Table 4, demonstrate its suitability for Aspect Category Sentiment Analysis in a traditional setting, where train and test datasets share the same domain and labels. While our method achieved slightly lower micro-F1 scores than the best performing approaches, Hier-Transformer-BERT and Hier-GCN-BERT, it can still be considered a viable alternative.

The performance of our pipeline on the zero-shot evaluation of the Laptops dataset is presented in table 5. SBERTiment achieves the highest micro-F1 scores on the Laptops test sets when using both Restaurants training sets. The AddOneDim-BERT model produces a significantly greater number of predicted *aspect category-sentiment* couples per input text, with an average of 10.05 compared to SBERTiment's average of 1.08. This explains the low precision and high recall scores that the AddOneDim-BERT method achieves in table 5, suggesting a tendency to overexpose in zero-shot settings. The Seq2seq model obtains 0.0 scores in all cases and on all metrics: although this text generation method can potentially perform the task in a zero-shot setting, all of the aspect categories it generates belong to the Semeval15 and Semeval16 Restaurants datasets since it was only trained on those. For this reason we omit Seq2seq scores in table 5. Hier-BERT, Hier-Transformer-BERT and Hier-GCN-BERT are excluded from this evaluation due to their inability to perform zero-shot ACSA. As their architectures are label-specific, they can only generate probability distributions on the labels found in the training set.

Conclusions and Future Works

This paper introduces SBERTiment, a novel approach to the Aspect Based Sentiment Analysis task that excels in both the classical setting, where the train and test data share the same domain, and the zero-shot setting, where the test data labels are unseen. The pipeline showed state-of-the-art results in the classical benchmark, outperforming the best existing methods in the zero-shot settings. This pipeline offers the ability to transfer knowledge between domains, making it a powerful tool in industrial contexts where it is challenging to collect quality training data. SBERTiment is therefore a promising solution for ABSA in all data scarcity scenarios.

We have shown that SBERTiment produces output triplets of *(aspect term, aspect category, sentiment)*. To emphasize its capacity to recognize unseen aspect categories, we have only assessed it on the Aspect Category Sentiment Analysis task, considering aspect terms as support information and not evaluating them directly. A promising avenue for future research is to evaluate our pipeline in the Aspect-Category-Sentiment Detection task, which takes into consideration the whole triplet.

[4]Available at: https://huggingface.co/google/t5-v1_1-small

	Semeval15 Restaurants			Semeval16 Restaurants		
Method	Precision	Recall	F1	Precision	Recall	F1
Cartesian-BERT*	72.02	49.15	58.42	74.96	63.84	68.94
Pipeline-BERT*	38.12	70.00	49.35	43.62	79.06	56.21
AddOneDim-LSTM*	54.33	28.44	37.32	61.56	42.82	50.05
AddOneDim-BERT*	68.84	55.86	61.67	71.75	67.95	69.79
Hier-BERT*	67.46	57.98	62.36	70.97	69.65	70.30
Hier-T-BERT*	70.22	59.96	**64.67**	73.72	73.21	73.45
Hier-GCN-BERT*	71.93	58.03	64.23	76.37	72.83	**74.55**
Seq2seq	51.60	38.69	44.03	66.95	52.33	58.74
SBERTiment	71.44	57.92	63.98	76.00	69.73	72.73

Table 4: Results obtained by models trained on the Semeval15 Restaurants and Semeval16 Restaurants train sets and evaluated on the respective test sets. Scores relative to methods with * are taken from (Cai et al. 2020).

		Semeval15 Laptops			Semeval16 Laptops		
Train set	Method	Precision	Recall	F1	Precision	Recall	F1
	AddOneDim-BERT	3.05	36.88	5.64	2.43	38.20	4.58
Semeval15 Rest	SBERTiment	20.00	13.70	**16.26**	19.43	13.73	**16.09**
	AddOneDim-BERT	5.95	10.43	7.57	4.36	8.24	5.70
Semeval16 Rest	SBERTiment	21.04	15.81	**18.05**	20.85	15.98	**18.09**

Table 5: Results obtained by models evaluated on the Semeval15 Laptops test set in the zero-shot setting.

References

Cai, H.; Tu, Y.; Zhou, X.; Yu, J.; and Xia, R. 2020. Aspect-category based sentiment analysis with hierarchical graph convolutional network. In *Proceedings of the 28th International Conference on Computational Linguistics*, 833–843. Barcelona, Spain (Online): International Committee on Computational Linguistics.

Cambria, E., and Hussain, A. 2015. *SenticNet*. Cham: Springer International Publishing. 23–71.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9:1735–80.

Hu, M.; Zhao, S.; Zhang, L.; Cai, K.; Su, Z.; Cheng, R.; and Shen, X. 2018. Can: Constrained attention networks for multi-aspect sentiment analysis.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization.

Li, Y.; Yang, Z.; Yin, C.; Pan, X.; Cui, L.; Huang, Q.; and Wei, T. 2020. *A Joint Model for Aspect-Category Sentiment Analysis with Shared Sentiment Prediction Layer*. 388–400.

Liu, J.; Teng, Z.; Cui, L.; Liu, H.; and Zhang, Y. 2021. Solving aspect category sentiment analysis as a text generation task. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 4406–4416. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Ma, Y.; Peng, H.; and Cambria, E. 2018. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. *Proceedings of the AAAI Conference on Artificial Intelligence* 32(1).

Pontiki, M.; Galanis, D.; Papageorgiou, H.; Manandhar, S.; and Androutsopoulos, I. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 486–495. Denver, Colorado: Association for Computational Linguistics.

Pontiki, M.; Galanis, D.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S.; AL-Smadi, M.; Al-Ayyoub, M.; Zhao, Y.; Qin, B.; De Clercq, O.; Hoste, V.; Apidianaki, M.; Tannier, X.; Loukachevitch, N.; Kotelnikov, E.; Bel, N.; Jiménez-Zafra, S. M.; and Eryiğit, G. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 19–30. San Diego, California: Association for Computational Linguistics.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

Reimers, N., and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks.

Schmitt, M.; Steinheber, S.; Schreiber, K.; and Roth, B. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1109–1114. Brussels, Belgium: Association for Computational Linguistics.

Shu, L.; Xu, H.; Liu, B.; and Chen, J. 2022. Zero-shot aspect-based sentiment analysis.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need.

Wan, H.; Yang, Y.; Du, J.; Liu, Y.; Qi, K.; and Pan, J. Z. 2020a. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence* 34(05):9122–9129.

Wan, H.; Yang, Y.; Du, J.; Liu, Y.; Qi, K.; and Pan, J. Z. 2020b. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence* 34(05):9122–9129.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; Davison, J.; Shleifer, S.; von Platen, P.; Ma, C.; Jernite, Y.; Plu, J.; Xu, C.; Scao, T. L.; Gugger, S.; Drame, M.; Lhoest, Q.; and Rush, A. M. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. Online: Association for Computational Linguistics.

Wu, C.; Xiong, Q.; Yi, H.; Yu, Y.; Zhu, Q.; Gao, M.; and Chen, J. 2021. Multiple-element joint detection for aspect-based sentiment analysis. *Knowledge-Based Systems* 223:107073.

Zhang, W.; Li, X.; Deng, Y.; Bing, L.; and Lam, W. 2021. Towards generative aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 504–510. Online: Association for Computational Linguistics.

Zhang, W.; Li, X.; Deng, Y.; Bing, L.; and Lam, W. 2022. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges.