Aspect-Based Sentiment Analysis Using BERT

Mickel Hoang
Chalmers University of Technology
Sweden
Hoangmicke@gmail.com

Oskar Alija Bihorac
Chalmers University of Technology
Sweden
Alija.bihorac@hotmail.com

Jacobo Rouces
Språkbanken, University of Gothenburg
Sweden
jacobo.rouces@gu.se

Abstract

Sentiment analysis has become very popular in both research and business due to the vast amount of opinionated text currently produced by Internet users. Standard sentiment analysis deals with classifying the overall sentiment of a text, but this doesn't include other important information such as towards which entity, topic or aspect within the text the sentiment is directed. Aspect-based sentiment analysis (ABSA) is a more complex task that consists in identifying both sentiments and aspects. This paper shows the potential of using the contextual word representations from the pre-trained language model BERT, together with a fine-tuning method with additional generated text, in order to solve out-of-domain ABSA and outperform previous state-of-the-art results on SemEval-2015 (task 12, subtask 2) and SemEval-2016 (task 5). To the best of our knowledge, no other existing work has been done on out-of-domain ABSA for aspect classification.

1 Introduction

Sentiment analysis, also known as opinion mining, is a field within natural language processing (NLP) that consists in automatically identifying the sentiment of a text, often in categories like negative, neutral and positive. It has become a very popular field in both research and industry due to the large and increasing amount of opinionated user-generated text in the Internet, for instance social media and product reviews. Knowing how users feel or think about a certain brand, product, idea or topic is a valuable source of information for companies, organizations and researchers, but it can be a challenging task. Natural language often contains ambiguity and figurative expressions that make the automated extraction of information in general very complex.

Traditional sentiment analysis focuses on classifying the overall sentiment expressed in a text without specifying what the sentiment is *about*. This may not be enough if the text is simultaneously referring to different topics or entities (also known as *aspects*), possibly expressing different sentiments towards different aspects. Identifying sentiments associated to specific aspects in a text is a more complex task known as aspect-based sentiment analysis (ABSA).

ABSA as a research topic gained special traction during SemEval-2014 (Pontiki et al., 2014) workshop, where it was first introduced as Task 4 and reappeared in the SemEval-2015 (Pontiki et al., 2015) and SemEval-2016 (Pontiki et al., 2016) workshops.

In parallel, within NLP, there have been numerous developments in the field of pre-trained language models, for example ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). These language models are pre-trained on large amounts of unannotated text, and their use has shown to allow better performance with a reduced requirement for labeled data and also much faster training. At SemEval-2016, there were no submissions that used such pre-trained language model as a base for the ABSA tasks. For this paper we will use BERT as the base model to improve ABSA models for the unconstrained evaluation, which permits using additional resources such as exter-

Proceedings of the 22nd Nordic Conference on Computational Linguistics (NoDaLiDa), pages 187–196
Turku, Finland, 30 September – 2 October, ©2019 Linköping University Electronic Press

nal training data, due to the pre-training of the base language model. More precisely, the contributions of this paper are as follows:

- It proposes the new ABSA task for out-of-domain classification at both sentence and text levels.
- To solve this task, a general classifier model is proposed, which uses the pre-trained language model BERT as the base for the contextual word representations. It makes use of the sentence pair classification model (Devlin et al., 2019) to find semantic similarities between a text and an aspect. This method outperforms all of the previous submissions, except for one in SemEval-2016.
- It proposes a combined model, which uses only one sentence pair classifier model from BERT to solve both aspect classification and sentiment classification simultaneously.

2 State-of-the-art

This chapter provides an overview of the techniques and models used throughout the rest of the paper, as well as existing state-of-the-art results.

Section 2.1 will cover the pre-trained model used in this paper, which has achieved state-of-the-art results in several NLP tasks, together with the architecture of the model and its key features. Thereafter, Section 2.2 will explain the ABSA task from SemEval-2016. Previous work with and without a pre-trained model will be briefly described in Section 2.3 and Section 2.4.

2.1 BERT

Pre-trained language models are providing a context to words, that have previously been learning the occurrence and representations of words from unannotated training data.

Bidirectional encoder representations from transformers (BERT) is a pre-trained language model that is designed to consider the context of a word from both left and right side simultaneously (Devlin et al., 2019). While the concept is simple, it improves results at several NLP tasks such as sentiment analysis and question and answering systems. BERT can extract more context features from a sequence compared to training left and right separately, as other models such as ELMo do (Peters et al., 2018).

The left and right pre-training of BERT is achieved using modified language model masks, called masked language model (MLM). The purpose of MLM is to mask a random word in a sentence with a small probability. When the model masks a word it replaces the word with a token [MASK]. The model later tries to predict the masked word by using the context from both left and right of the masked word with the help of transformers. In addition to left and right context extraction using MLM, BERT has an additional key objective which differs from previous works, namely next-sentence prediction.

Previous work

BERT is the first deeply bidirectional and unsupervised language representation model developed. There have been several other pre-trained language models before BERT that also use bidirectional unsupervised learning. One of them is ELMo (Peters et al., 2018), which also focuses on contextualized word representations. The word embeddings ELMo generates are produced by using a Recurrent Neural Network (RNN) named Long Short-Term Memory (LSTM) (Sak et al., 2014) to train left-to-right and right-to-left independently and later concatenate both word representations (Peters et al., 2018). BERT does not utilize LSTM to get the word context features, but instead uses transformers (Vaswani et al., 2017), which are attention-based mechanisms that are not based on recurrence.

Input Representaion

The text input for the the BERT model is first processed through a method called wordpiece tokenization (Wu et al., 2016). This produces set of tokens, where each represent a word. There are also two specialized tokens that get added to the set of tokens: classifier token [CLS], which is added to the beginning of the set; and separation token [SEP], which marks the end of a sentence. If BERT is used to compare two sets of sentences, these sentences will be separated with a [SEP] token. This set of tokens is later processed through three different embedding layers with the same dimensions that are later summed together and passed to the encoder layer: Token Embedding Layer, Segment Embedding Layer and Position Embedding Layer.

Transformers

Previous work in sequence modeling used the common framework sequence-to-sequence (seq2seq) (Sutskever et al., 2014), with techniques such as recurrent neural networks (RNNs) (Graves, 2013) and long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997).

The architecture of transformers is not based on RNNs but on attention mechanics (Vaswani et al., 2017), which decides what sequences are important in each computational step. The encoder does not only map the input to a higher dimensional space vector, but also uses the important keywords as additional input to the decoder. This in turn improves the decoder because it has additional information such which sequences are important and which keywords give context to the sentence.

Sentence Pair Classifier Task

Originally, BERT pre-trained the model to obtain word embeddings to make it easier to fine-tune the model for a specific task without having to make a major change in the model and parameters. Usually, only one additional output layer on top of the model was required to make the model more task-specific.

The Sentence Pair Classifier task deals with determining the semantic relations between two sentences. The model takes two texts as input, as described in Section 2.1, and outputs a label representing the type of relation between the sentences. This kind of task evaluates how good a model is on comprehensive understanding of natural language and the ability to do further inference on full sentences (Conneau et al., 2017). There is a benchmark that evaluates natural language understanding on models named general language understanding evaluation (GLUE) (Wang et al., 2018), which consists of several tasks such as multi-genre natural language inference (MNLI) (Williams et al., 2018), the semantic textual similarity benchmark (STS-B) (Cer et al., 2017) and Microsoft research paraphrase corpus (MRPC) (Dolan and Brockett, 2005).

Pre-training tasks

Supervised machine learning tasks are solved training a model from scratch with training data. NLP is a diversified field that contains many distinct tasks for which only small sets of human-labeled training data may be available. It has been proven that a large amount of training data increases the performance of deep learning models, for instance in the computer vision field with ImageNet (Deng et al., 2009). The same concept can be applied to deep language models. The development of a general purpose language model uses large amount of unannotated text, which is called pre-training, and the general purpose for the language model is to learn the contextual representation of words.

Language Models are key components in solving NLP problems and learn word occurrence and word prediction patterns based on unannotated text data. A language model learns the context by using techniques such as word embeddings which use vectors to represent the words in a vector space (Mikolov et al., 2013). With the large amount of training data, the language model learns that representations of words, depending on the context, allows similar words to have a similar representation.

Masked Language Model BERT uses a mask token [MASK] to pre-train deep bidirectional representations for the language model. But as opposed to conditional language models that train left-to-right or right-to-left to predict words, where the predicted word is positioned at the end or at the start of the text sequence, BERT masks a random word in the sequence. The other reason for using a mask token to pre-train is that the standard conditional language models are only able to explicitly train left-to-right or right-to-left because the words can indirectly "see itself" in a multi-layered context.

Next Sentence Prediction is used to understand the relationship between two text sentences. BERT has been pre-trained to predict whether or not there exists a relation between two sentences. Each of these sentences, sentence A and sentence B, has its own embedding dimensions.

Sentence A : [CLS] The man went to the store . [SEP]
Sentence B : He bought a gallon of milk . [SEP]
Label : IsNextSentence

During training, half of the time sentence B is the follow-up of sentence A in half and the IsNextSentence label is used. The other half of the time, a random sentence is chosen for sentence B and the IsNotNextSentence label is used.

2.2 Aspect-Based Sentiment Analysis

ABSA is a more complex task than traditional text-level sentiment analysis. It focuses on identifying the attributes or aspects of an entity mentioned in a text, together with the sentiment expressed towards each aspect.

ABSA was first introduced in SemEval-2014 (Pontiki et al., 2014), which provided a dataset with annotated reviews about restaurants and laptops. The ABSA task in SemEval-2014 did not contain full reviews until SemEval-2015 (Pontiki et al., 2015) and the dataset for SemEval-2016 did not change from 2015 except for additional test data.

The goal of the SemEval-2016 ABSA task is to identify opinions expressed towards specific aspect for a topic within customer reviews. Specifically, given a text review about a certain topic, from the dataset (e.g. laptop, restaurant), the objective for SemEval-2016, the goal is to address the following tasks:

Aspect category classification aims to identify the topic and aspect pair, which an opinion is expressed in the text. The topic and aspect should be chosen from an already defined set of topic types (e.g. LAPTOP, RESTAURANT, FOOD) and aspects (e.g. PRICE, QUALITY) per domain.

Opinion target expression (OTE) is the task of extracting the linguistic expression used in the text input that refers to the reviewed entity, for each entity-aspect pair. The OTE is defined with one starting and ending offsets in the sequence. If no entity is explicitly mentioned, the value returned is ”NULL”.

Sentiment polarity classification has the objective of predicting the sentiment polarity for each identified topic and aspect pair. The sentiment polarity is a value within the set {positive, negative, neutral, conflict}.

Subtask 1: Sentence Level. The input consists of one sentence, usually obtained from the fully text level text.

Subtask 2: Text Level. The input is a full review, where several aspects can be mentioned simultaneously and also different opinions on the same aspect can be given.

2.3 ABSA without BERT

The submissions that performed best at the SemEval-2016 ABSA challenges used mostly machine learning techniques such as support vector machines (SVM) (Joachims, 1998; Hsu et al., 2003) or conditional random field classifiers (Lafferty et al., 2001). Even though deep learning models have shown to perform well in sentiment analysis (Kim, 2014), the submissions employing deep learning techniques performed poorly that year.

The features used with the SVM were usually contextualized word representations extracted using GloVe (Pennington et al., 2014) or word lists, which were generated by extracting the nouns and adjectives from the datasets.

2.4 ABSA with BERT

BERT has shown to produce good results on NLP tasks (Wang et al., 2018) due to the large amounts of text it has been trained on. For tasks such as ABSA, performance has shown to improve with the help of an additional training on Review text, called Post-Training (Xu et al., 2019). To solve an ABSA task, the Post-Training paper constructed ABSA as a question answering problem, together with a machine reading comprehension technique for reviews called “review reading comprehension”.

Solving ABSA as a sentence-pair classification task using BERT by constructing auxiliary sentence has been seen to improve the results, compared to the previous state-of-the-art models that used single-sentence classification (Sun et al., 2019).

3 Experiments

The models implemented in this paper are three: an aspect classification model, a sentiment polarity classification model, and a combined model for both aspect and sentiment classification. The aspect classification model, described in Section 3.4, uses sentence pair classification from BERT (Devlin et al., 2019). As it only predicts whether an aspect is related to a text or not, this model has the possibility to be used for out-of-scope aspects. The sentiment polarity classifier, described in Section 3.3, is a classification model that is trained to determine the sentiment labels (positive, negative, neutral, conflict) for a given aspect and text input. Finally, Section 3.5 explains the last model, which is a combination of both the sentiment and aspect classification models. It outputs a sentiment if the aspect is related, and otherwise it returns the unrelated label.

	Sentence Level			Text Level		
	Restaurant	Laptop	Both	Restaurant	Laptop	Both
Texts	2000	2500	4500	334	395	729
Unique Aspects	12	81	93	12	81	93
Aspects with Sentiment	2507	2908	5415	1435	2082	3517
Aspects without Sentiment	21493	199592	413085	2573	29913	64280
Total Aspects	24000	202500	418500	4008	31995	67797

Table 1: Distribution of data in each training dataset.

3.1 Pre-processing entity and aspect pairs for BERT

The format of the pairs in the SemEval-2016 dataset is originally structured in the form of ”ENTITY#ASPECT”. In order to fit better the BERT model when training and to be able have the pre-trained data in BERT to be applicable, we formatted it to have a sentence-like structure, so the pair ”FOOD#STYLE_OPTIONS” gets parsed into ”food, style options”. This text is what we use as aspect.

3.2 Data generation

The dataset used in our experiments is reused from SemEval-2016 - Task 5 (Pontiki et al., 2016). Each sample in the dataset contains text that has been annotated with a list of aspects and sentiment polarity which consists of ’positive’, ’neutral’, ’negative’ or ’conflict’. The annotations to be generated are those which have an aspect that are not related to the subject, for example, the text “The food tasted great!” and the aspect ’restaurant, ambience’ do not have any relations.

As the dataset has a fixed amount of aspects (e.g. the Restaurant dataset has 12 different unique aspects), we can assume that each aspect that has not been annotated for a specific text is unrelated to said text. The aspects, which are not related to the text will be added to the list of aspects for the text with an ’unrelated’ label instead of a sentiment label. Table 1 and Table 6 show the distribution of the original data and our generated data in the training and test dataset respectively.

Unbalanced data

The dataset from SemEval-2016 is originally very unbalanced, it becomes even more so when the unrelated data is generated, as seen in aspects without sentiment compared to aspects with sentiment in Table 1.

To compensate for the imbalance, we weight each label depending on how frequently they show up in the training set. The higher the frequency of a label, the lower the weight of the given label.

3.3 Sentiment Classifier

This is a model for predicting sentiment on a text, given a specific aspect. It is implemented using the architecture of the Sentence Pair classification model explained in Section 2.1, where the first input is the text to be evaluated, and the second input is the aspect that the text will be evaluated on. The output of this model will be one of the labels ‘positive’, ‘negative’, ‘neutral’ and ‘conflict’, where ‘conflict’ means that there are parts of the text where the aspect is judged positively and other parts where the aspect judged negatively.

3.4 Aspect Category Classifier

This is a model for aspect classification, with the structure of a Sentence Pair classifier described in Section 2.1, with the text and the aspect as input. This model is used to predict whether or not the aspect is related to the text or not, using labels ’related’ and ’unrelated’. With the aspect as input, it is possible to handle out-of-domain aspects, i.e. outside the set of aspects the model was trained on.

3.5 Combined model

This model is structured as a multi-class classifier for predicting both the aspect and the sentiment using the structure of a Sentence Pair classification, described in Section 2.1. The model also takes the text and the aspect as input and returns a sentiment label if the aspect is related to the text, and the unrelated label otherwise.

The model can be used as an entire ABSA structure. It has the possibility to behave as either an aspect category model by mapping the polarity labels to ’related’ or it can behave like a sentiment model by ignoring the value of the ’unrelated’ label or it can behave as both at the same time.

4 Evaluation

The evaluation is based on the SemEval-2016 Task 5, more specifically the subtasks: aspect categorization in subtask 1 & 2, slot 1 and sentiment polarity in subtask 1 & 2, slot 3. The results for each model implemented are presented in the tables in Table 2a to Table 5c, with the previous state-of-the-art models as baseline.

The Aspect Category Classifier, the Sentiment Classifier, and the Combined Classifier, have all been trained on each dataset described in Table 1. This results in 18 models, where each of these models have been tested on every dataset described in Table 6. However, the text-level Hotel dataset was generated by concatenating all the sentence-level input to a full text and labelling the text with all the aspects corresponding to the sentences, because the Hotel dataset only consisted of the sentence level.

For the results in the tables of this section, we only show the best performing model for model type, in-domain, out-of-domain, text-level and sentence-level model. The dataset in which these models has been tested on can be found in the description of the tables.

4.1 Aspect Category Models

In this section, we evaluate how well the aspect categorization works with our models, which are described in Section 3.4 and Section 3.5. Each is trained in all the different domains and levels described in Table 1. As the performance of aspect category classifiers is only measured with F1-score in SemEval-2016, all the result tables in this section are ordered by F1 score in descending order.

In the tables within this section, the 'Model' column represents which model type it is. The combined model is defined as 'COM' and 'ASP' is the Aspect Category Classifier. The other two columns, Domain and Level, denote which domain and text type it was trained on.

For aspect classification, the text-level datasets in Table 3 produce better results than the sentence-level datasets in Table 2. In both of these tables, the aspect classifiers always outperform the combined classifiers. In out-of-scope evaluations, aspect classification performs better with classifiers that have been trained on datasets with more unique aspects.

Model	Domain	Level	F1	PRE	REC	ACC
ASP	REST	SENL	**79.9**	80.2	79.5	96.3
COM	REST	SENL	77.4	75.9	79.0	95.8
ASP	REST	TEXL	55.5	41.0	85.9	87.4
COM	LAPT	SENL	35.7	30.0	44.1	85.5
Baseline: BERT-PT			78.0	-	-	-
Baseline: NLANGP			73.0	-	-	-

(a) Results of Aspect models on dataset: Restaurant, Sentence-Level. BERT-PT (Xu et al., 2019) and NLANGP (Toh and Su, 2016) as baselines

Model	Domain	Level	F1	PRE	REC	ACC
ASP	BOTH	SENL	**51.7**	40.7	70.6	98.4
ASP	BOTH	TEXL	39.0	27.5	66.7	97.5
COM	BOTH	SENL	38.7	25.5	80.7	96.9
ASP	REST	SENL	5.7	3.0	67.3	73.5
Baseline: NLANGP			**51.9**	-	-	-

(b) Results of Aspect models on dataset: Laptop, Sentence Level. With NLANGP (Toh and Su, 2016) as baseline.

Model	Domain	Level	F1	PRE	REC	ACC
ASP	BOTH	SENL	**34.4**	23.3	65.9	89.1
COM	REST	SENL	34.1	22.9	67.5	88.7
ASP	LAPT	TEXL	33.8	28.2	42.1	92.8

(c) Performance of Aspect models on the dataset: Hotel, Sentence-Level.

Table 2: Best performance of aspect category classifiers in sentence-level datasets

Model	Domain	Level	F1	PRE	REC	ACC
ASP	REST	TEXL	**85.0**	84.2	85.9	88.7
COM	BOTH	TEXL	82.4	78.2	87.1	86.1
ASP	BOTH	SENL	78.8	81.9	76.0	84.7
COM	LAPT	TEXL	68.0	66.4	69.6	75.5
Baseline: GTI			84.0	-	-	-

(a) Results of Aspect models on dataset: Restaurant, Text-Level. Baseline: GTI (Álvarez-López et al., 2016).

Model	Domain	Level	F1	PRE	REC	ACC
ASP	BOTH	TEXL	**64.3**	60.9	68.1	92.3
COM	BOTH	TEXL	63.9	57.4	72.1	91.7
ASP	LAPT	SENL	61.0	58.7	64.6	91.6
COM	REST	SENL	21.6	12.3	87.4	37.0
Baseline: UWB			60.5	-	-	-

(b) Performance of Aspect models on the dataset: Laptop, Text-Level. UWB (Hercig et al., 2016) as baseline.

Model	Domain	Level	F1	PRE	REC	ACC
ASP	BOTH	TEXL	**60.8**	48.3	82.0	62.8
COM	LAPT	TEXL	59.4	53.7	66.4	68.0
COM	BOTH	SENL	58.8	45.2	84.0	58.5
ASP	REST	SENL	56.7	45.0	76.7	58.8

(c) Results of aspect category classifiers on the dataset: Hotel, Text-Level

Table 3: Best performance of aspect category classifiers in text-level datasets

4.2 Sentiment Models

In this section, we evaluate how well the sentiment classification performs with our models, which are described in Section 3.3 and Section 3.5. Each model trained on all the different domains and levels are described in Table 1. The F1 score measured on the tables in this section is a weighted average of the F1 on each label. As the performance of sentiment classifiers are only measured with accuracy in SemEval-2016, all the tables in this section is ordered by accuracy in descending order.

In the tables within this section, the 'Model' column represents which model type it is, 'COM' is the combined model, 'SEN' is the Sentiment Classifier. The other two columns, Domain and Level, is which domain and text type it was trained on.

For sentiment classification, in both Table 4 and Table 5, the combined classifiers always outperformed the sentiment classifiers. In out-of-scope scenarios, the classifiers which have been trained on sentence-level datasets outperform the classifiers which have been trained on the text-level datasets.

Model	Domain	Level	F1	PRE	REC	ACC
COM	BOTH	SENL	89.5	89.5	89.8	**89.8**
SEN	BOTH	SENL	89.2	89.6	89.5	89.5
COM	BOTH	TEXL	83.3	84.0	84.0	84.0
SEN	LAPT	SENL	81.6	84.0	81.2	81.2
Baseline: XRCE			-	-	-	88.1

(a) Performance of Sentiment models on the dataset: Restaurant, Sentence-Level. XRCE (Brun et al., 2016) as baseline.

Model	Domain	Level	F1	PRE	REC	ACC
COM	BOTH	SENL	83.2	83.6	82.8	**82.8**
SEN	LAPT	SENL	82.7	83.0	82.6	82.6
COM	REST	SENL	77.0	75.7	79.0	79.0
COM	BOTH	TEXL	76.2	76.1	76.7	76.7
Baseline: IIT-T			-	-	-	82.8

(b) Performance of Sentiment models on the dataset: Laptop, Sentence-Level. IIT-T (Kumar et al., 2016) as baseline.

Model	Domain	Level	F1	PRE	REC	ACC
COM	BOTH	SENL	90.0	91.0	89.5	**89.5**
SEN	BOTH	SENL	89.0	89.4	88.9	88.9
SEN	REST	SENL	87.0	86.9	87.3	87.3
COM	LAPT	SENL	86.2	86.0	87.0	87.0
COM	BOTH	TEXL	84.2	84.2	84.2	84.2
Baseline: lsislif			-	-	-	85.8

(c) Performance of Sentiment models on the dataset: Hotel, Sentence-Level. Lsislif (Hamdan et al., 2015) as baseline.

Table 4: Best performance of sentiment classifiers in sentence-level datasets

Model	Domain	Level	F1	PRE	REC	ACC
COM	BOTH	SENL	86.3	86.2	87.5	**87.5**
COM	BOTH	TEXL	84.7	84.1	86.6	86.6
SEN	REST	SENL	83.4	81.0	86.3	86.3
COM	LAPT	SENL	80.4	79.9	82.4	82.4
Baseline: UWB			-	-	-	81.9

(a) Results of aspect category classifiers on dataset: Restaurant, Text-Level. Baseline: GTI (Álvarez-López et al., 2016).

Model	Domain	Level	F1	PRE	REC	ACC
COM	BOTH	SENL	79.4	80.8	78.7	**78.7**
COM	REST	SENL	75.6	73.4	78.2	78.2
SEN	BOTH	SENL	77.1	76.7	77.8	77.8
COM	LAPT	TEXL	75.1	74.4	76.7	76.7
Baseline: ECNU			-	-	-	75.0

(b) Performance of Sentiment models on the dataset: Laptop, Text-Level. ECNU (Jiang et al., 2016) as baseline.

Model	Domain	Level	F1	PRE	REC	ACC
COM	BOTH	SENL	86.9	86.5	87.3	**87.3**
COM	REST	SENL	85.5	84.1	87.3	87.3
COM	BOTH	TEXL	85.4	84.9	86.4	86.4
SEN	BOTH	SENL	82.5	81.6	83.8	83.8
COM	LAPT	SENL	82.3	81.1	83.5	83.5

(c) Performance of Sentiment models on the dataset: Hotel, Text-Level

Table 5: Best performance of sentiment classifiers in text-level datasets

5 Discussion

Our proposed out-of-domain implementation performed well in the out-of-domain evaluation. In aspect category for hotels in Table 3c, which our aspect models have not been introduced to before, the model achieved a higher F1 score than the in-domain baseline for laptop F1 score in Table 3b. This shows the potential of using semantic similarities to find features for relations between aspect and a text input. However, to compare these models more in depth, a better measurement would be to look at both precision and recall, as the laptop domain has much more unique aspects, which in turn makes it more likely to predict more false positives which causes a lower precision.

For all the experiments and evaluation, we trained the models on each specific dataset and tried for the others. Our expectation was that the model would be able to improve the performance by using the combined dataset (restaurant & laptop) because it offers more features to use for the aspect classification task. This was not always the case, and we assume it has to

	Sentence Level			Text Level		
	Restaurant	Laptop	Hotel	Restaurant	Laptop	Hotel
Texts	676	782	226	90	80	30
Unique Aspects	12	81	28	12	81	28
Aspects with Sentiment	859	777	339	404	545	215
Aspects without Sentiment	7253	62565	5989	676	5935	625
Total Aspects	8112	63342	6328	1080	6480	840

Table 6: Data distribution in test datasets.

do with the difference between the amount of unique aspects in the domains. The aspect classifiers seem not to work well on the sentence-level test dataset. We suspect that the reason for this is that each sentence does not necessarily have enough information to validate whether an aspect is relevant for a text. A sentence-level text input example is "It wakes up super fast and is always ready to go", which is categorized as "LAPTOP#OPERATION_PERFORMANCE". In the out-of-domain and generalized model, this sentence does not provide the necessary information to make clear that the aspect is related to the sentence and instead can be applied to a lot of other aspects from other domains.

The combined model performs consistently better than the sentiment model in all domains. We believe that the reason for this is that the combined model is trained on a vast volume of "unrelated" data compared to the sentiment model, which allows it to learn to ignore redundant features when predicting the sentiment. However, the combined model performs worse than the aspect model in classifying relevant aspects. We conclude that the reason for this is that the combined model has to find what is relevant, which for this model is defined by the 4 sentiment polarity labels. This increases the complexity compared to the aspect model that was trained specifically on whether or not the aspect is relevant to the text.

A possible reason for why our model improves upon previous state-of-the-art models may be that it uses BERT for the word representation and can then employ the semantic similarities in the different word embeddings for the word, which captures the context, to find sentiments for an aspect in a text. Compared to the previous best models that generate one vector for each word, BERT uses positional word embeddings to generate different word embeddings for each word, depending on its position in the text. Another possible reason is the use of sentence-pair classification to compare the similarities of an aspect to a text instead of the previous best models that used single-sentence classification to determine what aspect is found in a text.

6 Conclusion

In this paper, we proposed an ABSA model that can predict the aspect related to a text for in-domain and out-of-domain. We achieve this by using the pre-trained language model BERT and fine-tuning it to a sentence pair classification model for the ABSA task. Moreover, we train the aspect classifier model with data that we generate, which consist of 'related' and 'unrelated' labels.

We further experimented with this approach for the sentiment classifier, by fine-tuning the model to find a relation between an aspect and a text and to make the model learn when the contextual representation showed a sentiment context. Furthermore, we proposed a combined model that can classify both aspect and sentiment using only one sentence pair classification model. Experimental results show that the combined model outperforms previous state-of-the-art results for aspect based sentiment classification.

References

Tamara Álvarez-López, Jonathan Juncal-Martínez, Milagros Fernández Gavilanes, Enrique Costa-Montenegro, and Francisco Javier González-Castaño. 2016. Gti at semeval-2016 task 5: Svm and crf for aspect detection and unsupervised aspect-based sentiment analysis. In *SemEval@NAACL-HLT*.

Caroline Brun, Julien Perez, and Claude Roux. 2016. XRCE at SemEval-2016 task 5: Feedbacked ensemble modeling on syntactico-semantic knowledge for aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 277–281, San Diego, California. Association for Computational Linguistics.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

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

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850.

Hussam Hamdan, Patrice Bellot, and Frederic Bechet. 2015. Lsislif: CRF and logistic regression for opinion target extraction and sentiment polarity analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 753–758, Denver, Colorado. Association for Computational Linguistics.

Tomáš Hercig, Tomáš Brychcín, Lukáš Svoboda, and Michal Konkol. 2016. UWB at SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 342–349, San Diego, California. Association for Computational Linguistics.

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

Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. 2003. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University.

Mengxiao Jiang, Zhihua Zhang, and Man Lan. 2016. ECNU at SemEval-2016 task 5: Extracting effective features from relevant fragments in sentence for aspect-based sentiment analysis in reviews. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 361–366, San Diego, California. Association for Computational Linguistics.

Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning*, ECML'98, pages 137–142, Berlin, Heidelberg. Springer-Verlag.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Ayush Kumar, Sarah Kohail, Amit Kumar, Asif Ekbal, and Chris Biemann. 2016. IIT-TUDA at SemEval-2016 task 5: Beyond sentiment lexicon: Combining domain dependency and distributional semantics features for aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1129–1135, San Diego, California. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. *Proceedings of the*

8th international workshop on semantic evaluation (SemEval 2014), pages 27–35.

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

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado. Association for Computational Linguistics.

Hasim Sak, Andrew W. Senior, and Franoise Beaufays. 2014. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH*, pages 338–342.

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

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Zhiqiang Toh and Jian Su. 2016. NLANGP at SemEval-2016 task 5: Improving aspect based sentiment analysis using neural network features. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 282–288, San Diego, California. Association for Computational Linguistics.

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

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

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, ukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis.