Evaluating the Aspect-Category-Opinion-Sentiment analysis task on a custom dataset

Loris Di Quilio[1], Fabio Fioravanti[1]

[1]*DEc, University of Chieti-Pescara, Italy*

Abstract

In this work, we report the results of some experiments with Aspect Based Sentiment Analysis (ABSA) on a dataset consisting of user reviews of products of a manufacturing company operating in the packaging industry. We focus on one of the more challenging ABSA tasks, the Aspect Category Opinion Sentiment task, and compare the results obtained by using three different tools available in the literature. We have also performed experiments for assessing the improvements that could be obtained by using larger models and similarity measures.

1. Introduction

Sentiment analysis aims to determine and understand the opinion sentiment expressed in a text. The basic approach performs this analysis prediction at the sentence or document level, identifying the overall sentiment of the sentence or whole document. In this case, it is assumed that a single sentiment is associated with a single topic in the text, but that may not be always the case. For this reason, a fine-grained sentiment analysis named *Aspect Based Sentiment Analysis* (ABSA), has received increasing attention. In this task, the objective includes identifying which specific aspects or features the sentiments refer to.

Aspect-based Sentiment Analysis (ABSA) is a fine-grained sentiment analysis where the goal is to identify the aspects of given target entities and the sentiment expressed for each aspect [2, 3, 4]. Over the years the main research of ABSA includes various sub-tasks divided by the prediction characteristic of a single sentiment component or of several ones together[5, 6]. The sentiment components used by these tasks are the following[1]:

- **category (c)**: is a pre-defined category related to a specific domain of interest. For example, AMBIENCE, PRICE, FOOD can be categories for the *restaurant* domain.
- **aspect term (a)**: represents the specific opinion target explicitly mentioned in the provided text. For instance, in the sentence "*The pizza is delicious but the service is terrible*", the explicit aspects are "pizza" and "service". When this is implicit, as in the sentence "*it's very reasonably priced*", when the subject is not explicitly named, we use a "NULL" label.

NL4AI 2023: Seventh Workshop on Natural Language for Artificial Intelligence, November 6-7th, 2023, Rome, Italy [1]

loris.diquilio@studenti.unich.it (L. Di Quilio); fabio.fioravanti@unich.it (F. Fioravanti)

CEUR Workshop Proceedings (CEUR-WS.org)

[1] the nomenclature of components and tasks could differ in the various works in the literature.

- **polarity (p)**: characterizes the sentiment orientation expressed towards an aspect category or an aspect term. Sentiment polarity falls into one of three categories: positive, negative, or neutral indicating whether the sentiment is favorable, unfavorable, or neither, respectively.
- **opinion term (o)**: is the word or multiple words used by opinion users to convey their sentiment or feelings about the target entity or aspect. For example, in the sentence "*The pizza is delicious but the service is terrible*", "*delicious*" and "*terrible*" are opinions terms, expressing a positive and negative sentiment toward the pizza.

Among the tasks of Aspect-based Sentiment Analysis that aim to predict a single sentiment element, there are:

- **Aspect Term Extraction** (ATE);
- **Aspect Category Detection** (ACD);
- **Opinion Term Extraction** (OTE);
- **Aspect opinion co-extraction** (AOCE);
- **Aspect Sentiment Classification** (ASC).

The tasks where multiple sentiment elements are predicted include:

- **Aspect-Opinion Pair Extraction** (AOPE);
- **End-to-End ABSA** (E2E-ABSA);
- **Aspect Category Sentiment Analysis** (ACSA);
- **Aspect Sentiment Triplet Extraction** (ASTE);
- **Aspect Category Sentiment Detection** (ACSD);
- **Aspect Category Opinion Sentiment** (ACOS).

Following we show a summary of the tasks using the input sentence "The pizza is delicious but the service is terrible".

Task	Input	Output
ATE	sentence	pizza (a), service (a)
ACD	sentence	food (c), service(c)
OTE	sentence	delicious (o), terrible (o)
ASC	sentence, pizza sentence, service	positive(p) negative (p)
AOPE	sentence	{pizza (a), delicious (o)}, {service (a), terrible (o)}
E2E ABSA	sentence	{pizza (a), positive p)}, {service (a), negative (p)}
ACSA	sentence	{food (c), positive (p)}, {service (c), negative (p)}
ASTE	sentence	{pizza (a), positive (p), delicious (o)}, {service (a), negative (p), terrible (o)}
ACSD	sentence	{food (c), pizza (a), positive (p)}, {service (c), service (a), negative (p)}
ACOS	sentence	{pizza (a), food (c), delicious (o), positive (p)}, {service (a), service (c), terrible (o), negative (p)}

In this paper, we will focus our attention on the ACOS task which aims at predicting all the sentiment elements at once, namely category (c), aspect term (a), polarity (p), and opinion term (o). For the ACOS task, a relatively limited body of research and literature exists. Our primary objective is to establish an integrated framework that leverages multiple tools for efficient ACOS task execution.

2. Dataset and annotation tool

The dataset used in this work is based on user reviews about skincare and pharmaceutical products supplied by a manufacturing company. The reviews have been scraped from e-commerce sites and some of them have been annotated using an open-source tool named *Label Studio*[2]. The annotations have been curated by one of the authors of the article with a dual-stage revision process to ensure their reliability. These annotations exhibit variances when compared to datasets accessible in the literature due to the incorporation of numerous implicit aspects related to the product supplied and opinion terms, frequently composed of multiple words. The dataset (Table 2) comprises 756 sentences and 1038 annotations, with the possibility of each sentence having multiple annotations.

	Train	**Test**	**Total**
Sentences	623	133	756
Annotations	881	157	1038

Table 1
Number of sentences and annotations in the training and test datasets

The annotations appears to be composed in a balanced way with regards to sentiment polarity (p); neutral sentiment is not calculated because predicting neutrality is not of interest in this case. As regards the categories, 13 classes were identified, encompassing both general and specific aspects of product performance.

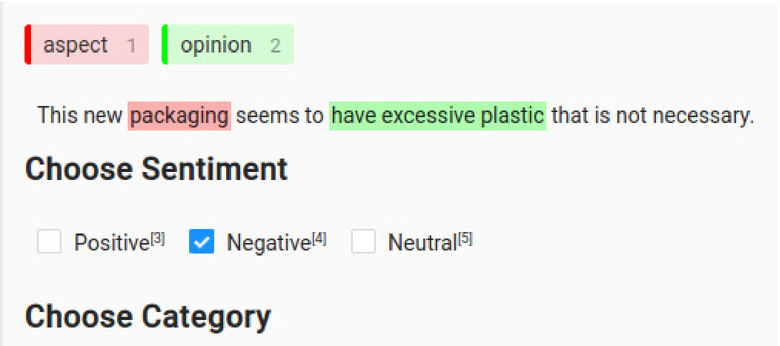
The distribution of classes is mostly balanced, with the exception of the category pertaining to "general satisfaction of the final consumer" which happens to be the most frequent one.

For this work, a custom template in *Label Studio* was built, which allows all elements to be annotated for each review. In Figure 1 we show an example of a sentence annotated on this annotation tool: the explicitly mentioned aspect and opinion elements can be directly selected in the text, while the polarity and the category, which is not shown, can be chosen from the predefined ones.

A translation module has been developed to convert the JSON encoding of the dataset exported from Label Studio to other formats, including those of the considered tools for the ACOS task, and the SemEval-2014 [3] and SemEval-2016 [4] formats.

The dataset and further details about the annotation process cannot be released due to a non-disclosure agreement.

[2] https://github.com/heartexlabs/label-studio



This new packaging seems to have excessive plastic that is not necessary.

Choose Sentiment

□ Positive[3] ☑ Negative[4] □ Neutral[5]

Choose Category

Figure 1: Example of a sentence annotated with Label Studio

3. Experimental evaluation

In this section, we present the details of the experimental evaluation we performed on our dataset using some tools that have been specifically built for the ACOS task. We have selected three tools that have stemmed from significant studies in this field and for which the source code is publicly available online. All the selected tools leverage the fine-tuning of pre-trained models, specifically T5 [7, 8] and BERT[9], as a crucial component of their functionality:

- **Paraphrase modeling** [10]: the model's objective is to generate a sequence of words, denoted as y, from an input sentence x. The sequence y should contain all the desired sentiment elements. Once the sequence y is generated, it's possible to recover the so-called "sentiment quads" $Q = (a, c, o, p)$. This approach aims to fully leverage the semantics of the sentiment elements represented by Q by generating them in natural language form within the sequence y. The pre-trained language model used is *T5-base*. This is the only tool among those we have considered that does not support implicit opinion terms;
- **Extract Classify-ACOS** [11]: This tool first performs aspect-opinion co-extraction, then predicts category-sentiment given the extracted aspect-opinion pairs. The tool uses the BERT model with AdamW optimizer[3] [12], so the data is transformed into a format suitable for it by delimiting each sentence using the CLS token.
- **PyABSA** [13, 14]: this tool is a variation of the original one, made for aspect-opinions pair extraction. There is no documentation about quadruple extraction because this feature is still experimental. The format of this tool was taken as a reference to transform the data once exported from the annotation tool. Also in this case *T5-base* is used as the pre-trained model.

We also performed additional experiments using PyABSA. In particular, (i) we utilized the tool with a larger pre-trained model, *T5-large*, which comprises 770 million parameters; (ii) we applied a similarity threshold between true labels and those predicted by the model for one of the components: the opinion term (o); (iii) we evaluated the performance of PyABSA using

[3] AdamW optimizer: is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments with an added method to decay weights.

the T5-large model with the standard correctness criterion, without similarity, on some less complex ABSA tasks, namely ACSA, E2E ABSA, ACSD and ASTE.

The second experiment is motivated by the fact that sentences in our domain often contain implicit opinions, frequently composed of multiple words rather than single terms. So we established a relaxed correctness criterion for considering a prediction correct when it matches the gold standard in terms of aspect, category, and polarity, and when the similarity between the predicted opinion term and the real one is at least 70%. For computing string similarity we used the Python function, SequenceMatcher[4] that is based on an extension of the Ratcliff and Obershelp algorithm ("gestalt pattern matching") [15] and compares pairs of sequences by finding the longest common subsequence while excluding uninteresting elements, with a quadratic time complexity for the worst case. In this way, for instance, the prediction of the opinion "super practical to slip into my bag" can be considered correct even if the real opinion is "practical to slip into my bag".

In Table 2, we show the tool settings we used for the experiments, including the batch-size, which indicates the number of training examples used in each iteration, the learning rate, a parameter in controlling the step size at each iteration while moving towards the minimum of a loss function and the number of epochs, representing the complete cycles through the training dataset.

Tool	batch-size	learning rate	epochs
Paraphrase modeling	16	3e-4	20
Extract Classify-ACOS	32{a, o}, 16(p), 8(c)	2e-5{a, o}, 3e5(p),(c)	20
PyABSA	16	5e-5	20

Table 2
Tool settings used for training

3.1. Results

To measure the performance of the models on the data, we computed the most commonly used metrics to evaluate these types of tasks, namely *precision*, the fraction of relevant retrieved instances over all the retrieved instances, *recall*, the fraction of relevant retrieved instances over all the relevant instances, and *F1-Score*, the harmonic mean of precision and recall calculated as $(2 \cdot precision \cdot recall)/(precision + recall)$. The results are shown in Table 3. Please note that, with the exception of the last tool in the table where we used the similarity metric discussed above, the prediction of a quadruple is considered to be correct if and only if it is equal to the gold one in all its four components.

Among the tools with base pre-trained models (T5-base and BERT), the Paraphrase modeling tools seems to be the overall best, but the support for implicit opinion, lacking from this tool, could be important for some application domains. The Extract Classify-ACOS tool seems to be slightly better than Paraphrase modeling in terms of precision, but has a significantly lower value for recall. The last tool we considered, PyABSA, is not the best in terms of performance

[4] https://docs.python.org/3/library/difflib.html

Tool	Precision	Recall	F1-score
Paraphrase modeling (T5-base)	0.373	0.382	0.377
Extract Classify-ACOS (BERT)	0.384	0.205	0.268
PyABSA (T5-base)	0.323	0.310	0.311
PyABSA (T5-large)	0.414	0.409	0.409
PyABSA (T5-large with similarity)	**0.538**	**0.526**	**0.528**

Table 3
Results of the experiments on the ACOS task with different tools

but it turned out to be very well designed, allowing us to customize it for performing further experiments using a larger pre-trained model (T5-Large) and employing a similarity criterion for one of the components. By using the larger model the precision increased from about 32% to 41% using the standard correctness criterion, and to 54% using the relaxed correctness criterion based on similarity.

The results of the experiments using PyABSA with the T5-large model and the standard correctness criterion on some less complex ABSA tasks are reported in Table 4.

Task	Predicted elements	Precision	Recall	F1-score
ACSA	c,p	0.820	0.807	0.803
E2E ABSA	a,p	0.763	0.754	0.752
ACSD	c,a,p	0.662	0.654	0.652
ASTE	a,p,o	0.477	0.477	0.472
ACOS	c,a,p,o	0.414	0.409	0.409

Table 4
Results obtained by using the PyABSA tool with the T5-large model on some simpler ABSA tasks

From the obtained results, it is evident that the model used by PyABSA performs well in predicting tuples, both in Aspect Category Sentiment Analysis (ACSA) and End-to-End ABSA (E2E ABSA). Furthermore, the model demonstrates good performance in extracting triples for Aspect Category Sentiment Detection (ACSD). However, it performs less effectively than the ACOS model with opinion term similarity set at 70% in Aspect Sentiment Triplet Extraction (ASTE). This observation implies that, within the framework of this model and the provided dataset, the primary limitation appears to be in the accurate identification of opinion terms. These terms, as previously discussed and as one might intuitively expect, are frequently composed of multiple words, posing a significant challenge for the model to predict them with absolute precision.

4. Conclusion and future work

We benchmarked three ACOS systems available in the literature by applying them to a different domain, using a custom dataset we built. Additionally, we assessed the PyABSA tool's performance in handling ACOS subtasks to identify critical elements in this process, which in this application domain seems to be the identification of the "opinion terms".

In the future, we plan to experiment with additional ACOS tools and different similarity measures. We also would like to expand the dataset and improve the annotation process. Another direction for future research is comparing the effectiveness of ACOS tools that perform the prediction of all the sentiment components at once with respect to other approaches that combine the results of specialized tools on simpler tasks.

One of the goals of the research is to develop a unified framework that allows the execution of different ABSA tasks by running multiple tools on the same dataset. Adapters should be in charge of translating data into the appropriate format. Also, it should be possible to define a variety of experiments and to explore different scenarios through an automatic and controlled selection of test and train data, by defining constraints on data categories and polarities. We envision an integrated framework in which the predictions from these tools are used to automatically or semi-automatically enhance and expand the training data, improving both the efficiency and the overall quality of the sentiment analysis models.

References

[1] M. P. A. R. Elisa Bassignana, Dominique Brunato, Preface to the Seventh Workshop on Natural Language for Artificial Intelligence (NL4AI), in: Proceedings of the Seventh Workshop on Natural Language for Artificial Intelligence (NL4AI 2023) co-located with 22th International Conference of the Italian Association for Artificial Intelligence (AI* IA 2023), 2023.

[2] W. Zhang, X. Li, Y. Deng, L. Bing, W. Lam, A survey on aspect-based sentiment analysis: Tasks, methods, and challenges, CoRR abs/2203.01054 (2022). doi:10.48550/arXiv.2203.01054.

[3] M. P. et al., Semeval-2014 task 4: Aspect based sentiment analysis, in: P. Nakov, T. Zesch (Eds.), Proc. 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014, The Association for Computer Linguistics, 2014, pp. 27–35. doi:10.3115/v1/s14-2004.

[4] M. P. et al., Semeval-2016 task 5: Aspect based sentiment analysis, in: S. B. et al. (Ed.), Proc. 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016, The Association for Computer Linguistics, 2016, pp. 19–30. doi:10.18653/v1/s16-1002.

[5] M. M. Trusca, F. Frasincar, Survey on aspect detection for aspect-based sentiment analysis, Artificial Intelligence Review 56 (2023) 3797–3846. URL: https://doi.org/10.1007/s10462-022-10252-y. doi:10.1007/s10462-022-10252-y.

[6] G. Brauwers, F. Frasincar, A survey on aspect-based sentiment classification, ACM Computing Surveys 55 (2023) 65:1–65:37. doi:10.1145/3503044.

[7] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of Machine Learning Research 21 (2020) 140:1–140:67.

[8] S. V. et al., Instruction tuning for few-shot aspect-based sentiment analysis, in: J. Barnes, O. D. Clercq, R. Klinger (Eds.), Proc. 13th Workshop on Computational Approaches to

Subjectivity, Sentiment, & Social Media Analysis, WASSA@ACL 2023, Toronto, Canada, July 14, 2023, Association for Computational Linguistics, 2023, pp. 19–27.

[9] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis MN, USA, June 2-7, 2019, Vol 1, Association for Computational Linguistics, 2019, pp. 4171–4186.

[10] W. Z. et al., Aspect sentiment quad prediction as paraphrase generation, in: M. M. et al. (Ed.), Proc. of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Punta Cana, Dominican Republic, 7-11 November, 2021, Association for Computational Linguistics, 2021, pp. 9209–9219. doi:10.18653/v1/2021.emnlp-main.726.

[11] H. Cai, R. Xia, J. Yu, Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions, in: C. Z. et al. (Ed.), Proc. 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, Vol 1, August 1-6, 2021, Association for Computational Linguistics, 2021, pp. 340–350. doi:10.18653/v1/2021.acl-long.29.

[12] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[13] H. Yang, K. Li, A modularized framework for reproducible aspect-based sentiment analysis, CoRR abs/2208.01368 (2022). doi:10.48550/arXiv.2208.01368.

[14] H. Yang, K. Li, PyABSA, 2023. URL: https://github.com/yangheng95/PyABSA.

[15] J. W. Ratcliff, D. Metzener, et al., Pattern matching: The gestalt approach, Dr. Dobb's Journal 13 (1988) 46.