InstructABSA: Instruction Learning for Aspect Based Sentiment Analysis

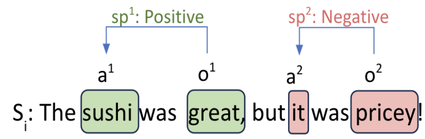
Kevin Scaria† **Himanshu Gupta**† **Siddharth Goyal**
Saurabh Arjun Sawant **Swaroop Mishra**♢ **Chitta Baral**
Arizona State University
{kscaria, hgupta35}@asu.edu

Abstract

We introduce InstructABSA, an instruction learning paradigm for Aspect-Based Sentiment Analysis (ABSA) subtasks. Our method introduces positive, negative, and neutral examples to each training sample, and instruction tune the model (Tk-Instruct) for ABSA subtasks, yielding significant performance improvements. Experimental results on the Sem Eval 2014, 15, and 16 datasets demonstrate that InstructABSA outperforms the previous state-of-the-art (SOTA) approaches on Term Extraction (ATE), Sentiment Classification(ATSC) and Sentiment Pair Extraction (ASPE) subtasks. In particular, InstructABSA outperforms the previous state-of-the-art (SOTA) on the Rest14 ATE subtask by 5.69% points, the Rest15 ATSC subtask by 9.59% points, and the Lapt14 AOPE subtask by 3.37% points, surpassing 7x larger models. We get competitive results on AOOE, AOPE, AOSTE, and ACOSQE subtasks indicating strong generalization ability to all subtasks. Exploring sample efficiency reveals that just 50% train data is required to get competitive results with other instruction tuning approaches. Lastly, we assess the quality of instructions and observe that InstructABSA's performance experiences a decline of $\sim 10\%$ when adding misleading examples [1].

1 Introduction

Aspect Based Sentiment Analysis (ABSA) plays a vital role in understanding the fine-grained sentiments expressed by users (Zhang and Liu, 2012). As illustrated in Figure 1, ABSA extracts aspects and classifies the aspect's sentiment polarity by extracting and understanding the author's opinions. Instruction learning paradigm (Mishra et al., 2022b; Wei et al., 2022; Gupta et al., 2023) has significantly improved the reasoning abilities of large language models (LLMs) and has shown impressive results across various tasks (Wang et al., 2022a; Lu et al., 2022). Owing to its previous success, we propose InstructABSA, instruction learning for aspect based sentiment analysis (ABSA). Our approach involves further instruction tuning of the Tk-Instruct model (Wang et al., 2022b) to address six subtasks of ABSA as shown in Fig. 1. We add instruction prompts specific to the downstream ABSA subtasks in the form of task definitions, followed by positive, negative, and neutral examples.



Subtask	Input	Output
Aspect Term Extraction (AOOE)	S_i	a^1, a^2
Aspect Term Sentiment Classification (ATSC)	S_i+a^1, S_i+a^2	sp^1, sp^2
Aspect Sentiment Pair Extraction (ATSC)	S_i	(a^1, sp^1), (a^2, sp^2)
Aspect Oriented Opinion Extraction (ATSC)	S_i+a^1, S_i+a^2	o^1, o^2
Aspect Opinion Pair Extraction (AOPE)	S_i	(a^1, o^1), (a^2, o^2)
Aspect Opinion Sentiment Triplet Extraction (AOSTE)	S_i	(a^1, o^1, sp^1), (a^2, o^2, sp^2)
Aspect Category Opinion Sentiment Quadruplet Extraction (ACOSQE)	S_i	(a^1, c^1, o^1, sp^1), (a^2, c^2, o^2, sp^2)

Figure 1: Illustration of the six ABSA subtasks where S_i is the i^{th} sentence, a^i are the aspect terms, sp^i are the sentiment polarities and o^i is the opinion terms.

We carried out extensive experiments on the SemEval 2014, 15, and 16 datasets (Pontiki et al., 2014, 2015, 2016), and the dataset by (Peng et al., 2020) for the AOSTE subask, which comprises the laptops and restaurants domain. Across the

[1]Experiments and results are available at https://github.com/kevinscaria/InstructABSA
† Currently in Amazon (The work was done prior to joining Amazon)
♢ Currently in Google Deepmind

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers), pages 720–736

subtasks in both domains, InstructABSA outperforms SOTA approaches. Specifically, for the 2014 ATE subtask, we obtain F1-score of 92.3 and 92.76 (Lapt14, Rest14), surpassing SOTA by 4.37% and 5.69% points respectively. For the ATSC subtask, InstructABSA attains an accuracy of 84.50 in the Rest15 dataset exceeding the previous results by 9.59% points. In the Rest14 dataset ATSC subtask, our approach gets a competitive accuracy score of 86.25 compared to the SOTA of 90.86. For the ASPE subtask, InstructABSA achieves F1-score of 79.34 and 79.47 (Lapt14, Rest14), outperforming SOTA by 3.37% and 1.4% points, respectively. We get competitive results on AOOE and AOSTE approaches as well (§4).

We conduct a thorough analysis along several lines of enquiry. We showcase sample efficiency of InstructABSA by achieving competitive scores using roughly 20% of training samples as compared to Varia et al. (2023)'s instruction tuning approach. We compare InstructABSA with fine-tuning methods such as Low-Rank Adaptation (LoRA) (Hu et al., 2021) to find that there is a sizebale gap of $\sim 20\%$. To understand the effect of different instructions for ABSA, we change the prompts on the lines of definition and task manipulation. We find that delusive examples roughly decrease the approaches results by $\sim 10\%$ giving a strong evidence of the impact of instructions on InstructABSA. We also provide evidence of cross-domain and joint-domain generalizations arising as part of our proposed approach.

Contributions:(a) we introduce InstructABSA, which achieves performance gains on ABSA subtasks of SemEval 2014,15 and 16 datasets, surpassing the previous SOTA models. (b) Despite using a 200M model, InstructABSA outperforms or get competitive results over the prior SOTA models with 1.5B parameters. (c) Finally, we provide an analysis of the impact of our method in terms of sample efficiency, adapter methods, effect of instruction and domain generalization.

2 InstructABSA: Instruction Learning for ABSA

We describe the mathematical formulation of ABSA subtasks and the proposed approach. Let S_i represent the i^{th} review sentence in the training sample, where:$S_i = w_i^1, w_i^2, ..., w_i^n$ with n as the number of tokens in the sentence. Each S_i contains a set of aspect terms denoted by $A_i = a_i^1, a_i^2, ..., a_i^m | m \leq n$ and the corresponding opinion terms, aspect category and sentiment polarities for each aspect term are denoted by $O_i = o_i^1, o_i^2, ..., o_i^m$ $C_i = c_i^1, c_i^2, ..., c_i^m$ and $SP_i = sp_i^1, sp_i^2, ..., sp_i^m$ respectively, where $sp_i^k \in [positive, negative, neutral]$ The ABSA tasks are described as follows:

$ATE : A_i = LM_{ATE}(S_i)$
$ATSC : sp_i^k = LM_{ATSC}(S_i, a_i^k)$
$ASPE : [A_i, SP_i] = LM_{ASPE}(S_i)$
$AOOE : o_i^k = LM_{AOOE}(S_i, a_i^k)$
$AOPE : [A_i, O_i] = LM_{AOPE}(S_i)$
$AOSTE : [A_i, O_i, SP_i] = LM_{AOSTE}(S_i)$
$ACOSQE : [A_i, C_i, O_i, SP_i] = LM_{ACOSQE}(S_i)$

In these equations, LM represents the language model, and the corresponding inputs and outputs are defined accordingly. As part of our approach, we instruction tune $LM_{subtask}$ by prepending task-specific instruction prompts $Inst$ to each input sample to arrive at $LM_{subtask}^{Inst}$. Here, $Inst = Definition + 2 \times PositiveExample + 2 \times NegativeExample + 2 \times NeutralExample$ For the LM, we use "'tk-instruct-base"' as the model. The $definition$ involves the task definition for each subtask. Contrary to the standard instruction tuning prompts proposed by (Wang et al., 2022b), $PositiveExample$ and $NegativeExample$ here represent examples that have a positive and negative sentiment example respectively. Additionally, we introduce $NeutralExample$ which is an example that has neutral sentiment respectively(§F).

3 Experimental Setup

We use the Tk-Instruct-base-def-pos [2] as the instruction-tuned model LM_{Inst}. We use two configurations of instructions as prompts for our experiments. InstructABSA-1 has the instruction prompt that includes the definition of the ABSA subtasks followed by 2 positive examples for the respective task. InstructABSA-2 has the definition followed by 2 positive, negative, and neutral examples.

Dataset: SemEval 2014,15 and 16 datasets are used for our experimentation. The dataset is used as a benchmark for ABSA tasks and has customer reviews from three domains; laptops (Lapt14), hotels (Hotel15), and restaurants (Rest14, Rest15, and Rest16). More details can be found in §C.

[2] https://huggingface.co/allenai/tk-instruct-base-def-pos

Model	Lapt14	Rest14	Rest15	Rest16
GPT2$_{med}$	82.04	75.94	-	-
GRACE	87.93	85.45	-	-
BARTABSA	83.52	87.07	75.48	-
IT-MTL	76.93	-	74.03	79.41
InstructABSA1	91.40	**92.76**	75.23	**81.48**
InstructABSA2	**92.30**	92.10	**76.64**	80.32

Table 1: ATE subtask results denoting F1 scores. GPT2$_{med}$, GRACE, BARTABSA and IT-MTL results are from Hosseini-Asl et al. (2022), Luo et al. (2020), Yan et al. (2021) and Varia et al. (2023) respectively.

Model	Lapt14	Rest14	Rest15	Rest16
ABSA-DeBERTa	82.76	89.46	-	-
LSAT	**86.31**	**90.86**	-	-
Dual-MRC	75.97	82.04	73.59	-
InstructABSA1	80.62	86.25	83.02	89.10
InstructABSA2	81.56	85.17	**84.50**	**89.43**

Table 2: ATSC subtask results denoting accuracy. ABSA-DeBERTa, LSAT and dual-MRC are from Marcacini and Silva (2021), Yang and Li (2021) and Mao et al. (2021) respectively.

Hyperparameters GPU: 1xNvidia Tesla P40, Train Batch Size: 16 for ATE and ATSC, 8 for other subtasks. Gradient Accumulation Steps: 2, Initial learning rate: 5e-5, Num of Epochs: 4

Evaluation Metric: Following previous approaches (Zhang et al., 2021; Luo et al., 2020), we use the micro F1-score for ATE, AOPE, AOOE, AOPE, AOSTE, and the accuracy for ATSC.

4 Results and Analysis

4.1 Sub Task Results

Tables 1 - 7 denotes the results of ATE, ATSC, ASPE, AOOE, AOPE, AOSTE and ACOSQE subtasks respectively. All the results reported are the average values from 5 runs for each experiment. For **ATE** subtask (Table 1), InstructABSA surpasses SOTA on Lapt14, Rest14, 15, and 16 datasets surpassing 7x larger models (Hosseini-Asl et al. (2022) uses GPT-2 with 1.5B parameters). For **ATSC** subtask, InstructABSA-2 achieves SOTA of Rest 15 while remaining competitive of Lapt and Rest 14 dataset. For the **ASPE subtask** (Table 3), InstructABSA acheives SOTA for all four datasets. In the **AOOE** subtask (Table 4) InstructABSA achieves an F1 score of 76.42 and 77.16 for the Lapt14 dataset, outperforming IOG and ONG.

In the **AOPE** subtask (Table 5), InstructABSA

Model	Lapt14	Rest14	Rest15	Rest16
GRACE	75.97	78.07	-	-
BARTABSA	67.37	73.56	66.61	-
IT-MTL	66.07	-	67.06	74.07
InstructABSA1	78.89	76.16	69.02	**74.24**
InstructABSA2	**79.34**	**79.47**	**69.39**	73.06

Table 3: ASPE subtask results denoting F1 scores. GRACE, BARTABSA and IT-MTL results are from Luo et al. (2020), Yan et al. (2021) and Varia et al. (2023).

Model	Lapt14	Rest14	Rest15	Rest16
IOG	70.99	80.23	71.91	81.60
ONG	76.77	82.33	78.81	86.01
BARTABSA	**80.55**	**85.38**	80.52	**87.92**
InstructABSA1	76.42	80.78	80.41	83.07
InstructABSA2	77.16	81.08	**81.34**	83.27

Table 4: AOOE subtask results denoting F1 scores. IOG, ONG and BARTABSA are from Fan et al. (2019), Pouran Ben Veyseh et al. (2020) and Yan et al. (2021) respectively.

Model	Lapt14	Rest14	Rest15	Rest16
Seq2Path	**74.29**	**77.35**	**71.84**	**79.09**
GAS	69.55	75.15	67.93	75.42
BMRC	67.45	76.23	68.60	76.52
InstructABSA1	60.75	70.46	60.31	72.04
InstructABSA2	61.74	71.37	62.59	70.06

Table 5: Results of the AOPE subask denoting F1 scores. Seq2Path, GAS and BMRC are from Mao et al. (2022), Zhang et al. (2021) and Chen et al. (2021) respectively.

suffers compared to the existing models. For the **AOSTE** subtask (Table 6), Seq2Path achieves the highest F1 scores for the datasets, however, our models achieve competitive results for Rest14. Finally, for the **ACOSQE** subtask, InstructABSA performs $\sim 1.1\%$ points more than the previous best. The performance of InstructABSA in AOPE, AOSTE, and ACOSQE is subpar as compared to ATE and ATSC due to exposure bias. For sentiment pair extraction tasks, the model had to decode only the aspect terms followed by sentiments that were constrained to positive, negative, and neutral labels. However, for the opinion pair extraction tasks and triplet extraction tasks, the model suffers higher exposure bias since the opinion terms are not grounded and could potentially be any word in the vocabulary (Zhang et al., 2020).

Model	Lapt14	Rest14	Rest15	Rest16
BMRC	59.27	70.69	61.05	68.13
Seq2Path	**65.27**	**75.52**	**65.88**	**73.67**
IT-MTL	-	43.84	52.94	53.75
InstructABSA1	60.67	70.50	60.63	68.15
InstructABSA2	61.86	71.17	59.98	70.72

Table 6: Results of the AOSTE subask denoting F1 scores. Seq2Path, IT-MTL and BMRC are from Mao et al. (2022), Chen et al. (2021), and Varia et al. (2023).

Model	Lapt14	Rest14	Rest15	Rest16
TAS-BERT-ACOS	27.31	33.53	-	-
ExtractClassify-ACOS	35.80	44.61	-	-
Seq2Path	58.41	-	-	42.97
InstructABSA1	57.21	56.32	57.56	59.87
InstructABSA2	**59.17**	**59.98**	**60.23**	**61.43**

Table 7: Results of the ACOSQE subtask denoting F1 scores. TAS-BERT-ACOS, ExtractClassify-ACOS and Seq2Path are from Wan et al. (2020), Cai et al. (2021) and Mao et al. (2022) respectively.

4.2 Analysis

In this subsection, we analyze InstructABSA on multiple line of enquiries.

Cross-Domain and Joint Domain Evaluation: In cross domain setting, we train the model on a train set from one domain and test on test set from another domain. In joint domain setting, the train data of the domains (laptops and restaurants) are combined to train the model, and it is evaluated on both test sets. Both experiments are performed on ATE, ATSC and ASPE subtasks for both instruction-tuned models (InstructABSA-1 & 2). Table 8 presents the cross domain experiment results. When trained on Lapt14 and tested on Rest14, InstructABSA-1 shows a drop in F1-score for the ATE and Joint Task compared to InstructABSA-2. For the ATSC task, similar trends were obtained with an accuracy of 75.53 from InstructABSA-1 and 80.56 from InstructABSA-2. The joint domain experiments are present in Table 9. The availability of additional training data for ATE subtask helps the language models as the proposed model surpasses the previously achieved SOTA. We also analyzed the performance of InstructABSA in a multi-task learning setup and find that our model achieves comparable results as presented in table 11.

Delusive examples reduce InstructABSA's performance We analyze the impact of instruction tuning along the lines of experiments proposed by Kung and Peng (2023), focusing on task definition and example manipulation. In task definition manipulation, we explore original, simplified, and empty definitions, but only use the empty configuration with vanilla T5 and Tk-instruct models. In task example manipulation, we study original, delusive, and empty examples, as well as additional configurations. Detailed results can be found in Figure 4 and Tables 15, 16, and 17. Notably, InstructABSA-1 and 2 outperform the vanilla models, highlighting the effectiveness of instruction tuning for most

Train	Test	Model	ATE	ATSC	ASPE
Rest14	Lapt14	InstructABSA-1	71.98	80.56	64.30
		InstructABSA-2	71.83	82.44	65.30
Lapt14	Rest14	InstructABSA-1	62.85	75.53	55.06
		InstructABSA-2	76.85	80.56	62.95
Rest15	Hotel15	InstructABSA-1	74.51	87.65	66.88
		InstructABSA-2	70.53	89.74	67.82

Table 8: Results of the cross-domain evaluation where the model is trained on Lapt14 and the test set is of Rest14 and vice versa. The results of the model trained on Rest15 and evaluated on Hotel15 is also reported.

Task	Model	ATE	ATSC	ASPE
Lapt14	InstructABSA-1	90.35	81.09	80.07
	InstructABSA-2	93.28	83.60	80.47
Rest14	InstructABSA-1	88.88	86.42	80.81
	InstructABSA-2	93.55	88.03	79.70

Table 9: Results of joint-domain evaluation where the model is trained on both Lapt14 and Rest14 datasets and evaluated on the respective test set.

Tasks	ATE		ATSC		ASPE	
	Lapt14	Rest14	Lapt14	Rest14	Lapt14	Rest14
LoRA 8	73.51	79.43	55.79	59.08	53.19	57.28
LoRA 16	73.57	78.32	54.30	59.16	52.30	57.19
LoRA 32	75.52	78.74	54.94	59.58	54.43	56.98
LoRA 64	71.61	76.93	55.87	58.64	55.87	58.64
InstructABSA-1	91.40	92.76	80.62	86.25	78.89	76.16
InstructABSA-2	92.30	92.10	81.56	85.17	79.34	79.47

Table 10: Results of LoRA PEFT and InstructABSA-1 and InstructABSA-2 across all subtasks. 8, 16, 32 and 64 in LoRA denote the rank of the adapter method.

Dataset	ATE	ATSC	ASPE	AOOE	AOPE	AOSTE	ACOSQE
Lapt14	93.41	82.33	80.89	79.12	62.94	62.31	62.43
Rest14	94.16	87.13	81.06	82.87	71.89	72.35	64.16
Rest15	78.53	86.67	71.31	82.78	64.52	61.13	64.23
Rest16	81.98	91.02	74.98	84.56	72.38	72.34	66.21

Table 11: Results of multi-task learning evaluation on the 4 datasets.

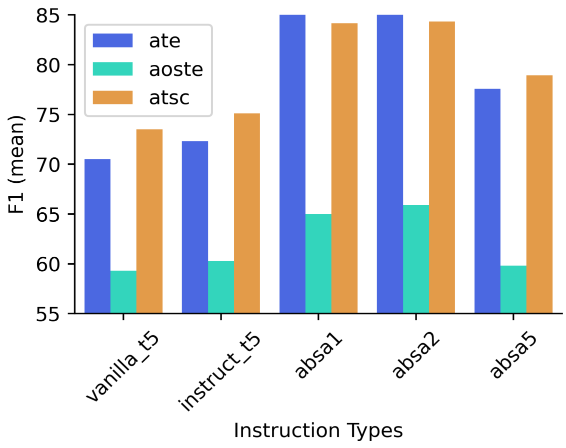


Figure 2: Comparison of various instruction configuration and its performance on ATE, AOSTE and ATSC subtasks. vanilla_t5 and instruct_t5 represent the base T5 model with and without instruction tuning on the dataset. absa1 includes a definition followed by 2 positive exemplars, absa2 includes a definition followed by 2 positive, negative, and neutral examples, and finally, absa5 is the delusive configuration with incorrect input and output mappings respectively.

ABSA subtasks.

Competitive scores with just 50% train samples Gupta et al. (2023) showcased the effects of sample efficiency via instruction tuning. Following that work, we explore the performance of instruction tuning by using a smaller percentage of the training set. We carry out experiments to identify the sample efficiency gains for ABSA subtasks. The results are presented in Figure 3 and Table 18. We get competitive scores with our best scores when using roughly 50% train samples, demonstrating sample efficiency of InstructABSA. Figure 3 also showcases the performance of the vanilla T5 base model finetuned with the same number of samples. As shown in the figure, the vanilla model's performance is consistently lower compared to InstructABSA.

Hard Case Analysis: We analyze the performance of instruction tuning on hard samples (HDS), viz. samples that have more than one aspect with a different sentiment polarity. From table 12 it can be seen that InstructABSA achieves competitive performance in hard cases.

Adapter methods leading to poor performance We compare the performance of parameter efficient finetuning method Low-Rank Adaptation (LoRA)(Hu et al., 2021) with our instruction tuning approach InstructABSA. LoRA can lead to significant improvements in memory efficiency and computational efficiency, but it can also lead to a drop in performance. The experiment is performed on all the subtasks, and the results are presented in Table 10. As seen in the table a drop of 13.32% points in ATE, 26.8% points in ATSC and 19.8% points in ASPE. The drop in scores is significant to overlook when aiming to reap the advantages of a computationally optimized finetuning method.

Dataset	**AGDT**	**GCAE**	**IABSA1**	**IABSA2**
Rest14	51.3	56.73	56.21	57.13
Lapt14	60.33	47.06	52.36	53.01

Table 12: Results of the hard case analysis. AGDT and GCAE are from (Liang et al., 2019) and (Xue and Li, 2018) respectively.

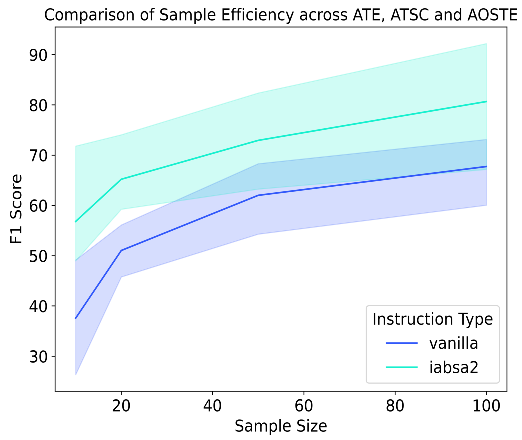


Figure 3: Comparison of sample efficiency on ATE, AOSTE and ATSC subtasks between InstructABSA-2 and vanilla model. Sample size is % of training data.

5 Conclusion

We proposed InstructABSA, an instruction-tuned modeling approach for all subtasks of ABSA. Our findings show that InstructABSA surpassed the previous scores on several tasks and achieved competitive scores on the rest using a significantly smaller model than previous approaches. We further analyzed the performance of the approach along several lines of enquiry revealing several interesting findings.

Limitations

Our study is limited to the Sem Eval 2014, 15, and 16 datasets, that are widely used in recent works. Future studies should include the extension of this work on other ABSA datasets to test the generalizability of our findings. We conducted our experiments using a 200M model, which may limit the applicability of our findings to smaller models. Future studies could consider using even smaller instruction-tuned models to analyze their performance. Our study was conducted using Tk-Instruct models for the English language. As a result, our findings may not be directly applicable to other languages. Future studies should include a multilingual dataset and a multilingual instruction-tuned model to investigate the model's performance across different languages.

Ethical Considerations

We acknowledge that the T5 model used in our experiments may have inherent biases due to the pre-training and instruction-tuning data used. While stress testing was not conducted, we believe that from our research no additional issues arise related to privacy, fairness, bias, and discrimination. We Our work directly contributes to the topic of aspect based sentiment analysis and we believe that our work will have a positive impact on the scientific community. We remain dedicated to advancing the responsible use of AI and will continue to prioritize ethical considerations in all our future research endeavors.

References

Ujjwala Anantheswaran, Kevin Scaria, Himanshu Gupta, Shreyas Verma, Chandrahaas Chintaboguda, Sabyasachi Bisoyi, Chitta Baral, and Swaroop Mishra. 2024. From booksmart to streetsmart: An adversarial approach to improve mathematical reasoning in llms. *ArXiv preprint*.

Hongjie Cai, Rui Xia, and Jianfei Yu. 2021. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 340–350, Online. Association for Computational Linguistics.

Shaowei Chen, Yu Wang, Jie Liu, and Yuelin Wang. 2021. Bidirectional machine reading comprehension for aspect sentiment triplet extraction. In *Proceedings of the AAAI conference on artificial intelligence*, 14, pages 12666–12674.

Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. 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 2509–2518, Minneapolis, Minnesota. Association for Computational Linguistics.

Himanshu Gupta, Saurabh Arjun Sawant, Swaroop Mishra, Mutsumi Nakamura, Arindam Mitra, Santosh Mashetty, and Chitta Baral. 2023. Instruction tuned models are quick learners.

Himanshu Gupta, Kevin Scaria, Ujjwala Anantheswaran, Shreyas Verma, Chitta Baral, and Swaroop Mishra. 2024a. Help me help you: Improving helpfulness of llms as agents. *ArXiv preprint*.

Himanshu Gupta, Kevin Scaria, Swaroop Mishra, and Chitta Baral. 2024b. Beyond the data bottleneck: Optimizing instruction tuning with difficulty-based exemplar selection. *ArXiv preprint*.

Himanshu Gupta, Neeraj Varshney, Swaroop Mishra, Kuntal Kumar Pal, Saurabh Arjun Sawant, Kevin Scaria, Siddharth Goyal, and Chitta Baral. 2022. " john is 50 years old, can his son be 65?" evaluating nlp models' understanding of feasibility. *arXiv preprint arXiv:2210.07471*.

Himanshu Gupta, Shreyas Verma, Tarun Kumar, Swaroop Mishra, Tamanna Agrawal, Amogh Badugu, and Himanshu Sharad Bhatt. 2021. Context-ner: Contextual phrase generation at scale. *arXiv preprint arXiv:2109.08079*.

Ehsan Hosseini-Asl, Wenhao Liu, and Caiming Xiong. 2022. A generative language model for few-shot aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 770–787, Seattle, United States. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Po-Nien Kung and Nanyun Peng. 2023. Do models really learn to follow instructions? an empirical study of instruction tuning. *arXiv preprint arXiv:2305.11383*.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. A novel aspect-guided deep transition model for aspect based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5569–5580, Hong Kong, China. Association for Computational Linguistics.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*.

Huaishao Luo, Lei Ji, Tianrui Li, Daxin Jiang, and Nan Duan. 2020. GRACE: Gradient harmonized and cascaded labeling for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 54–64, Online. Association for Computational Linguistics.

Man Luo, Sharad Saxena, Swaroop Mishra, Mihir Parmar, and Chitta Baral. 2022. Biotabqa: Instruction learning for biomedical table question answering. *arXiv preprint arXiv:2207.02419*.

Yue Mao, Yi Shen, Jingchao Yang, Xiaoying Zhu, and Longjun Cai. 2022. Seq2Path: Generating sentiment tuples as paths of a tree. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2215–2225, Dublin, Ireland. Association for Computational Linguistics.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13543–13551.

Ricardo Marcondes Marcacini and Emanuel Silva. 2021. Aspect-based sentiment analysis using bert with disentangled attention. *LatinX in AI at International Conference on Machine Learning 2021*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022a. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022b. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

Swaroop Mishra and Elnaz Nouri. 2022. Help me think: A simple prompting strategy for non-experts to create customized content with models. *arXiv preprint arXiv:2208.08232*.

Mihir Parmar, Swaroop Mishra, Mirali Purohit, Man Luo, Murad Mohammad, and Chitta Baral. 2022. In-BoXBART: Get instructions into biomedical multi-task learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 112–128, Seattle, United States. Association for Computational Linguistics.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8600–8607.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, Ion Androutsopoulos, Núria Bel, and Gülşen Eryiğit. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Amir Pouran Ben Veyseh, Nasim Nouri, Franck Dernoncourt, Dejing Dou, and Thien Huu Nguyen. 2020. Introducing syntactic structures into target opinion word extraction with deep learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8947–8956, Online. Association for Computational Linguistics.

Siddharth Varia, Shuai Wang, Kishaloy Halder, Robert Vacareanu, Miguel Ballesteros, Yassine Benajiba, Neha Anna John, Rishita Anubhai, Smaranda Muresan, and Dan Roth. 2023. Instruction tuning for few-shot aspect-based sentiment analysis.

Hai Wan, Yufei Yang, Jianfeng Du, Yanan Liu, Kunxun Qi, and Jeff Z. Pan. 2020. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9122–9129.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022a. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022b. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523, Melbourne, Australia. Association for Computational Linguistics.

Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for 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 1: Long Papers)*, pages 2416–2429, Online. Association for Computational Linguistics.

Heng Yang and Ke Li. 2021. Improving implicit sentiment learning via local sentiment aggregation. *arXiv preprint arXiv:2110.08604*.

Lei Zhang and B. Liu. 2012. Sentiment analysis and opinion mining. In *Encyclopedia of Machine Learning and Data Mining*.

Ranran Haoran Zhang, Qianying Liu, Aysa Xuemo Fan, Heng Ji, Daojian Zeng, Fei Cheng, Daisuke Kawahara, and Sadao Kurohashi. 2020. Minimize exposure bias of Seq2Seq models in joint entity and relation extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 236–246, Online. Association for Computational Linguistics.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 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)*, pages 504–510, Online. Association for Computational Linguistics.

Appendix

A Choosing Samples as Instruction Exemplars:

From Table 14, it can be noticed that the distribution of count of aspects across Lapt14, Rest14, Rest15, and Rest16 datasets is centered around one, two, and three aspects which account for 30%, 11%, and 4.5% of total aspects. Thus for our instruction exemplars, we randomly select samples that have aspects ranging between 1 and 3. We exclude these exemplars during evaluation.

B Instruction Effectiveness Study

To validate the effect of instruction tuning on the performance of various ABSA sub tasks, We analyse effect of instruction tuning along the lines of experiments proposed by Kung and Peng (2023). We carry out our analysis on two aspects: task definition manipulation and task example manipulation. In *task definition manipulation*, controlled experiments are conducted to examine whether models truly comprehend and utilize the semantic meaning of task definitions. Three levels of granularity was proposed viz. *original*, *simplified*, and *empty*. The simplified version removes all semantic components from the task definition, leaving only the output space information. The empty version eliminates the task definition altogether. However, as part of the task definition manipulation experiment we only conduct the empty configuration with vanilla_t5 and vanilla_tk where t5 is the T5-base model and tk is the Tk-instruct base model. In *task example manipulation*, the influence of task examples on model learning is investigated. Three types of task examples are compared: *original*, *delusive*, and *empty*. The original setup includes one/two positive example (absa1), while the delusive examples consist of negative examples with incorrect input-output mappings (absa6). The empty setup excludes task examples during training (task_def_only). We additionally carry out different configuration of task examples and call it additions, where we add 2 positive, negative and neutral examples (absa2), 2 negative (absa3), 2 neutral (absa4) and 1 positive, negative and neutral example (absa5). The detailed reports are presented in the Figure 4 and Tables 15, 16 and 17 . It is evident that for most ABSA subtasks, the instruction configuration of InstructABSA-1 and 2 yields the best performance. Additionally, it can be seen that both the vanilla models do not give the best results solidifying the effectiveness of further instruction tuning.

C Detailed Dataset Description:

Dataset	Split	Pos.	Neg.	Neut.
Lapt14	Train	987	866	460
	Test	341	128	169
Rest14	Train	2164	805	633
	Test	728	196	196
Rest15	Train	912	256	36
	Test	326	182	34
Hotel15	Test	163	45	7
Rest16	Train	1240	439	69
	Test	468	117	30

Table 13: Dataset Statistics for ATSC subtask denoting number of samples. Pos., Neg., and Neut. represent Positive, Negative, and Neutral, respectively

Table 14 displays the dataset description with respect to the count of aspect terms for all subtasks. For the training set, 1557 reviews in Lapt14 and 1020 reviews in Rest14 have no aspect terms and their corresponding polarities. Similarly, in the test set, 378 reviews in Lapt14 and 194 reviews in the Rest14 have no aspect terms and corresponding polarities. The dataset description for the ATSC subtask is presented in Table 13. To maintain consistency with the previous approaches for the ATSC task, we also ignore conflict labels.

D Extended Related Work

LMs and deep learning methods have been used for a plethora of downstream tasks for a long time. Several recent works have leveraged NLP methods and simple sampling methods for different downstream results The study of whether existing LMs can understand instructions has motivated a range of subsequent works. Mishra et al. (2022b); Gupta et al. (2024a); Anantheswaran et al. (2024); Gupta et al. (2024b) proposed natural language instructions for cross-task generalization of LMs. PromptSource and FLAN (Wei et al., 2022) were built to leverage instructions and achieve zero-shot generalization on unseen tasks. Moreover, Parmar et al. (2022) shows the effectiveness of instructions in multi-task settings for the biomedical domain. Mishra et al. (2022a) discussed the impact of task instruction reframing on model response. Gupta et al.

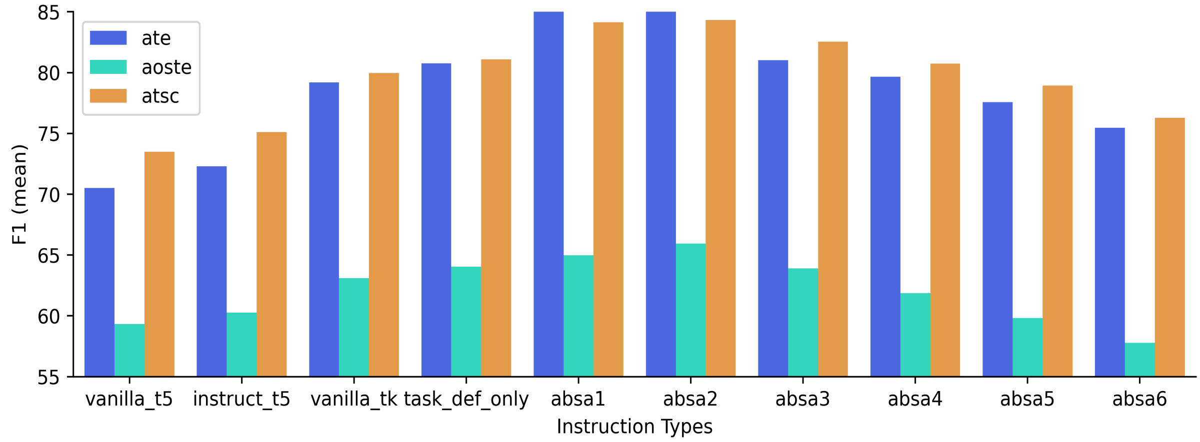


Figure 4: Comparison of various instruction configuration and its performance on ATE, AOSTE and ATSC subtasks. Vanilla_t5 and Vanilla_tk represent the models trained without any instruction. absa1, absa2, absa3, absa4, absa5 are different instruction configurations that include a definition followed by 2 positive, 2 positive, negative and neutral examples, 2 negative examples, 2 neutral examples, 1 positive, negative and neutral examples and finally examples with incorrect input and output mappings respectively. task_def_only only contains the task definitions.

Dataset	Split	#NO	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10+	#Total
Lapt14	Train	1557	930	354	140	43	10	6	3	1	-	1	3045
	Test	378	266	105	34	10	6	1	-	-	-	-	800
Rest14	Train	1020	1022	572	269	104	30	15	5	3	1	-	3041
	Test	194	290	186	80	30	14	3	2	-	-	1	800
Rest15	Train	482	576	174	58	22	2	-	-	1	-	-	1315
	Test	284	294	82	18	6	-	1	-	-	-	-	685
Hotels15	Test	98	135	23	7	2	1	-	-	-	-	-	266
Rest16	Train	766	868	258	76	28	2	1	-	1	-	-	2000
	Test	256	298	87	22	9	3	-	-	-	-	1	676

Table 14: Count of Aspects for the ATE, ASTE, AOOE, AOPE and AOSTE subtasks. $\#k$ is the count of samples that have k aspects/aspect-sentiment polarity pairs in them. #NO is the number of samples that have no aspect/aspect-sentiment polarity pairs in them.

(2022) showed that adding knowledge with instruction helps LMs understand the context better. Furthermore, several approaches have been proposed to improve model performance using instructions, including (Wang et al., 2022b; Luo et al., 2022; Mishra and Nouri, 2022) Several studies are present that show adding knowledge with instruction helps LMs understand the context better (Gupta et al., 2021).

E Additional Tables for Plots

The following section presents the absolute non aggregated numbers for the plots generated to analyse the instruction effectiveness (Figure 4) as well as the sample efficiency plots (Figure 3). The following analysis was conducted on the 3 subtasks viz. ATE, ATSC and AOSTE. This was based on the level of difficulty of the tasks. To balance out the analysis across tasks of various difficulties, we chose the easiest task which is just task extraction. It was followed by ATSC task which is more complicated since the model has to learn associations of the aspect term and its corresponding sentiment polarity. Finally the task with maximum difficulty was triplet extraction since the model has to extract all triplets given a sentence.

Table 15 presents the performance metrics in terms of F1 score for the ATE subtask for the 4 datasets when instruction tuned with various configuration of instructions as mentioned in §4.2. Similarly Table 16 presents the F1 scores for the ATSC subtask when instruction tuned with various configuration of instructions as mentioned in §4.2. Table 17 presents the F1 scores for the AOSTE subtask

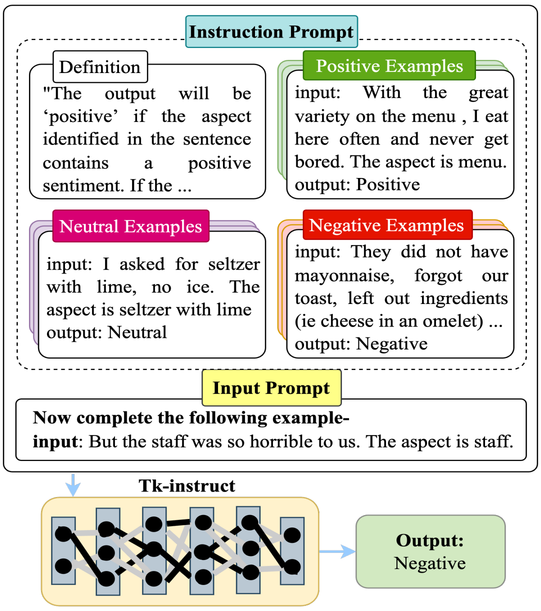


Figure 5: Formulation of InstructABSA for ATSC task. The input consists of an instruction prompt and a sentence. The output label is the sentiment polarity for the corresponding aspect.

when instruction tuned with various configuration of instructions as mentioned in §4.2. Finally, Table 18, describes the values for the sample efficiency plot. This plot presents the raw unnagregated numbers for ATE, ATSC and AOSTE.

F InstructABSA prompt examples

The instruction prompts for InstructABSA-1, and InstructABSA-2 are presented in detail for all three ABSA subtasks. Table 19, 20, and 21 presents the prompts provided for InstructABSA-2 model for the ATE, ATSC, and AOPE, respectively.

For the InstructABSA-1 model, the instruction prompts are similar, with the difference that negative and neutral examples are not provided in the instruction prompts.

Instruction Type	Lapt14	Rest14	Rest15	Rest16
vanilla_t5	71.67	74.59	61.74	74.04
instruct_t5	73.02	77.25	63.90	75.04
vanilla_tk	83.07	85.23	70.40	78.04
task_def_only	85.60	86.78	72.31	78.32
absa1	91.40	92.76	75.23	81.48
absa2	92.30	92.10	76.64	80.32
absa3	88.06	89.19	72.31	74.52
absa4	87.25	87.78	71.81	71.81
absa5	85.58	86.00	70.35	68.33
absa6	83.91	84.21	68.89	64.85

Table 15: Tabular Results Instruction Effectiveness Plot for ATE

Instruction Type	Lapt14	Rest14	Rest15	Rest16
vanilla_t5	59.42	80.70	72.41	81.44
instruct_t5	62.56	81.30	74.03	82.54
vanilla_tk	71.98	83.10	78.91	85.86
task_def_only	74.56	83.27	80.12	86.45
absa1	79.37	85.15	82.98	89.09
absa2	80.84	84.47	83.37	88.66
absa3	79.01	82.34	81.67	87.12
absa4	77.18	80.21	79.97	85.58
absa5	75.35	78.08	78.27	84.04
absa6	70.12	75.95	76.57	82.50

Table 16: Tabular Results Instruction Effectiveness Plot for ATSC

Instruction Type	Lapt14	Rest14	Rest15	Rest16
vanilla_t5	53.53	66.48	64.53	52.73
instruct_t5	54.72	67.15	63.88	55.30
vanilla_tk	58.29	69.16	61.93	63.01
task_def_only	59.48	69.83	61.28	65.58
absa1	60.67	70.50	60.63	68.15
absa2	61.86	71.17	59.98	70.72
absa3	58.98	69.65	57.83	69.12
absa4	56.10	68.13	55.68	67.52
absa5	53.22	66.61	53.53	65.92
absa6	50.34	65.09	51.38	64.32

Table 17: Tabular Results Instruction Effectiveness Plot for AOSTE

Task	Sample Size	No Instruction	InstructABSA-2
ate	10	49.15	71.81
ate	20	56.12	74.06
ate	50	68.30	82.37
ate	100	73.13	92.20
atsc	10	37.24	49.67
atsc	20	51.23	62.34
atsc	50	63.45	73.21
atsc	100	70.06	82.65
aoste	10	26.34	48.98
aoste	20	45.78	59.24
aoste	50	54.29	63.25
aoste	100	60.05	67.16

Table 18: Tabular Results of Sample Efficiency Plots

Task	Aspect Term Extraction (ATE)
Definition	Definition: The output will be the aspects (both implicit and explicit) which have an associated opinion that is extracted from the input text. In cases where there are no aspects, the output should be noaspectterm.
Positive Example	Example Input 1: With the great variety on the menu, I eat here often and never get bored. Example Output 1: menu Example Input 2: Great food, good size menu, great service, and an unpretentious setting. Example output 2: food, menu, service, setting
Negative Example	Negative input 1: They did not have mayonnaise, forgot our toast, left out ingredients... Negative output 1: toast, mayonnaise, bacon, ingredients, plate Negative input 2: The seats are uncomfortable if you are sitting against the wall on wooden benches. Negative output 2: seats
Neutral Example	Neutral Input 1: I asked for a seltzer with lime, no ice. Neutral Output 1: seltzer with lime Neutral Input 2: They wouldn't even let me finish my glass of wine before offering another. Neutral Output 2: glass of wine
Input	Now complete the following example- input: My son and his girlfriend both wanted cheeseburgers and they were huge! output: cheeseburgers

Table 19: Illustrating InstructABSA-2 instruction prompting for the ATE sub task.

Task	Aspect Term Sentiment Classification (ATSC)
Definition	The output will be 'positive', 'negative' or 'neutral' if the sentiment of the identified aspect in the input is positive, negative or neutral respectively For the aspects which are classified as noaspectterm, the sentiment is none.
Positive Example	Example Input 1: With the great variety on the menu, I eat here often and never get bored. Aspect: menu Example Output 1: positive Example Input 2: Great food, good size menu, great service, and an unpretentious setting. Aspect: food. Example Output 2: positive
Negative Example	Example Input 1: They did not have mayonnaise, forgot our toast, left out ingredients (i.e., cheese in an omelet), below hot temperatures and the bacon was so overcooked it crumbled on the plate when you touched it. Aspect: toast Example Output 1: negative Example Input 2: The seats are uncomfortable if you are sitting against the wall on wooden benches. Aspect: seats Example Output 2: negative
Neutral Example	Example Input 1: I asked for a seltzer with lime, no ice. Aspect: seltzer with lime Example Output 1: neutral Example Input 2: They wouldn't even let me finish my glass of wine before offering another. Aspect: a glass of wine Example Output 2: neutral
Input	Now complete the following example- input: My son and his girlfriend both wanted cheeseburgers and they were huge! Aspect: cheeseburgers. output: positive

Table 20: Illustrating InstructABSA-2 instruction prompting for the ATSC subtask.

Task	Aspect Sentiment Pair Extraction (ASPE)
Definition	Definition: The output will be the aspects (both implicit and explicit), and the aspects sentiment polarity. In cases where there are no aspects, the output should be no aspect-tern: none.
Positive Example	Example Input 1: With the great variety on the menu, I eat here often and never get bored. Example Output 1: menu:positive Example Input 2: Great food, good size menu, great service, and an unpretentious setting. Example Output 2: food:positive
Negative Example	Example Input 1: They did not have mayonnaise, forgot our toast, left out ingredients (i.e., cheese in an omelet), below hot temperatures, and the bacon was so overcooked it crumbled on the plate when you touched it. Example Output 1: toast:negative Example Input 2: The seats are uncomfortable if you are sitting against the wall on wooden benches. Aspect: seats Example Output 2: negative
Neutral Example	Example Input 1: I asked for a seltzer with lime, no ice. Example Output 1: seltzer with lime: neutral Example Input 2: They wouldn't even let me finish my glass of wine before offering another. Example Output 2: glass of wine:neutral
Input	Now complete the following example- input: My son and his girlfriend both wanted cheeseburgers and they were huge! output: cheeseburgers: positive

Table 21: Illustrating InstructABSA-2 instruction prompting for the ASPE subtask.

Task	Aspect Oriented Opinion Extraction (AOOE)
Definition	Definition: The output will be the opinion/describing word of the aspect terms in the sentence. In cases where there are no aspects the output should be none.
Positive Example	Example Input 1: Faan 's got a great concept but a little rough on the delivery. Example Output 1: delivery:rough Example Input 2: it is of high quality , has a killer GUI , is extremely stable, is highly expandable. The aspect is GUI. Example Output 2: killer
Negative Example	Example Input 1: One night I turned the freaking thing off after using it , the next day I turn it on , no GUI , screen all dark,.. The aspect is GUI. Example Output 1: no Example Input 2: I can barely use any usb devices because they will not stay connected properly . The aspect is usb devices. Example Output 2: not stay connected properly
Neutral Example	Example Input 1: However, ..external mouse unnecessary. The aspect is external mouse. Example Output 1: unnecessary Example Input 2: ... extended warranty and they refused. The aspect is extended warranty. Example Output 2: refused
Input	Now complete the following example- input: My son ... cheeseburgers and they were huge!. The aspect is cheeseburgers. output: huge

Table 22: Illustrating InstructABSA-2 instruction prompting for the AOOE subtask.

Task	Aspect Opinion Pair Extraction (AOPE)
Definition	Definition: The output will be the aspect terms in the sentence followed by its describing/opinion term.
Positive Example	Example Input 1: I charge it at night and skip taking the cord with me because of the good battery life. Example Output 1: battery life:good Example Input 2: it is of high quality , has a killer GUI , is extremely stable, is highly expandable,.. good applications,.. easy to use. Example Output 2: quality:high, GUI:killer, applications:good, use:easy
Negative Example	Example Input 1: A month or so ago , the freaking motherboard just died . Example Output 1: motherboard:freaking Example Input 2: I had always used PCs ....crashing and the poorly designed operating systems that were never very intuitive Example Output 2: operating systems:poorly designed, operating systems: never very intuitive
Neutral Example	Example Input 1: It has a 10 hour ... when you 're doing web browsing and word editing , making it perfect for the classroom or office, ... Example Output 1: web browsing:perfect, word editing:perfect Example Input 2: no complaints with their desktop , and maybe because it just sits on your desktop... which could jar the hard drive , or the motherboard Example Output 2: hard drive:jar, motherboard:jar
Input	Now complete the following example- input: Boot time is super fast , around anywhere from 35 seconds to 1 minute output: Boot time:superfast

Table 23: Illustrating InstructABSA-2 instruction prompting for the AOPE subtask.

Task	Aspect Opinion Sentiment Triplet Extraction (AOSTE)
Definition	Definition: The output will be the aspect terms in the sentence followed by their describing words and sentiment polarity.
Positive Example	Example Input 1: I charge it at night and skip taking the cord with me because of the good battery life. Example Output 1: battery life:good:positive Example Input 2: it is of high quality , has a killer GUI , is extremely stable, is highly expandable,.. good applications,.. easy to use. Example Output 2: quality:high:positive, GUI:kille:positive
Negative Example	Example Input 1: A month or so ago , the freaking motherboard just died . Example Output 1: motherboard:freaking Example Input 2: I had always used PCs ....crashing and the poorly designed OS that were never very intuitive Example Output 2: OS:poorly designed:negative, OS: never very intuitive:negative
Neutral Example	Example Input 1: It has a 10 hour ... when you 're doing web browsing and word editing , making it perfect for the classroom or office, ... Example Output 1: web browsing:perfect:neutral, word editing:perfect:neutral Example Input 2: no complaints with their desktop , and maybe because it just sits on your desktop... which could jar the hard drive , or the motherboard Example Output 2: hard drive:jar:neutral, motherboard:jar:neutral
Input	Now complete the following example- input: Boot time is super fast , around anywhere from 35 seconds to 1 minute output: Boot time:superfast:positive

Table 24: Illustrating InstructABSA-2 instruction prompting for the AOPE subtask.

Task	Aspect Opinion Pair Extraction (AOPE) - Task Definition Only
Definition	Definition: The output will be the aspect terms in the sentence followed by its describing/opinion term.
Input	Now complete the following example- input: Boot time is super fast , around anywhere from 35 seconds to 1 minute output: Boot time:superfast

Table 25: Illustrating Only Task Definition based prompting for AOPE subtask.

Task	Aspect Opinion Pair Extraction (AOPE) - 2 Negative Examples
Definition	Definition: The output will be the the aspect terms in the sentence followed by their describing/opinion term.
Negative Example	Example Input 1: A month or so ago , the freaking motherboard just died . Example Output 1: motherboard:freaking:negative Example Input 2: I had always used PCs ....crashing and the poorly designed OS that were never very intuitive Example Output 2: OS:poorly designed, OS: never very intuitive

Table 26: Illustrating Definition + 2 negative exemplars based prompting for AOPE subtask

Task	Aspect Opinion Pair Extraction (AOPE) - 2 Neutral Examples
Definition	Definition: The output will be the the aspect terms in the sentence followed by their describing/opinion term.
Neutral Example	Example Input 1: It has a 10 hour ... when you ’re doing web browsing and word editing, making it perfect for the classroom or office, ... Example Output 1: web browsing:perfect, word editing:perfect Example Input 2: no complaints with their desktop , and maybe because it just sits on your desktop... which could jar the hard drive , or the motherboard Example Output 2: hard drive:jar, motherboard:jar
Input	Now complete the following example- input: Boot time is super fast , around anywhere from 35 seconds to 1 minute output: Boot time:superfast

Table 27: Illustrating Definition + 2 neutral exemplars based prompting for AOPE subtask

Task	Aspect Opinion Pair Extraction (AOPE) - 1 Positive, Negative and Neutral Example
Definition	Definition: The output will be the aspect terms in the sentence followed by its describing/opinion term.
Positive Example	Example Input 1: I charge it at night and skip taking the cord with me because of the good battery life. Example Output 1: battery life:good
Negative Example	Example Input 1: A month or so ago , the freaking motherboard just died . Example Output 1: motherboard:freaking
Neutral Example	Example Input 1: It has a 10 hour ... when you ’re doing web browsing and word editing , making it perfect for the classroom or office, ... Example Output 1: web browsing:perfect, word editing:perfect
Input	Now complete the following example- input: Boot time is super fast , around anywhere from 35 seconds to 1 minute output: Boot time:superfast

Table 28: Illustrating Definition + 1 positive + 1 negative + 1 neutral exemplars based prompting for AOPE subtask

Task	Aspect Opinion Pair Extraction (AOPE) - Delusive Examples
Definition	Definition: The output will be the aspect terms in the sentence followed by its describing/opinion term.
Positive Example	Example Input 1: I charge it at night and skip taking the cord with me because of the good battery life. Example Output 1: motherboard:freaking
Negative Example	Example Input 1: A month or so ago , the freaking motherboard just died . Example Output 1: web browsing:perfect, word editing:perfect
Neutral Example	Example Input 1: It has a 10 hour ... when you ’re doing web browsing and word editing , making it perfect for the classroom or office, ... Example Output 1: battery life:good
Input	Now complete the following example- input: Boot time is super fast , around anywhere from 35 seconds to 1 minute output: Mac M1: fast

Table 29: Illustrating delusive instruction based prompting for AOPE subtask. In this task, the output labels of the examplars are mapped incorrectly with the inputs.