Explaining Pre-Trained Language Models with Attribution Scores: An Analysis in Low-Resource Settings

Wei Zhou[1], **Heike Adel**[3], **Hendrik Schuff**[4], **Ngoc Thang Vu**[2]
[1]Bosch Center for Artificial Intelligence, Renningen, Germany
[2]Institut für Maschinelle Sprachverarbeitung, University of Stuttgart, Germany
[3]Hochschule der Medien, Stuttgart, Germany
[4]Ubiquitous Knowledge Processing Lab, Technical University of Darmstadt, Germany
wei.zhou3@de.bosch.com, heike.adel@gmail.com,
hendrik.schuff@tu-darmstadt.de, thang.vu@ims.uni-stuttgart.de

Abstract

Attribution scores indicate the importance of different input parts and can, thus, explain model behaviour. Currently, prompt-based models are gaining popularity, i.a., due to their easier adaptability in low-resource settings. However, the quality of attribution scores extracted from prompt-based models has not been investigated yet. In this work, we address this topic by analyzing attribution scores extracted from prompt-based models w.r.t. plausibility and faithfulness and comparing them with attribution scores extracted from fine-tuned models and large language models. In contrast to previous work, we introduce training size as another dimension into the analysis. We find that using the prompting paradigm (with either encoder-based or decoder-based models) yields more plausible explanations than fine-tuning the models in low-resource settings and Shapley Value Sampling consistently outperforms attention and Integrated Gradients in terms of plausibility and faithfulness scores.

Keywords: explainability, attribution scores, low resource

1. Introduction

Recently, two paradigms of using pre-trained transformer models, such as BERT or GPT-2 (Devlin et al., 2019; Brown et al., 2020), have gained popularity: *fine-tuning* which adapts the weights of the model using task-specific training data, and *prompting* which defines or learns so-called prompts to retrieve knowledge from the model, often leaving the model's weights unchanged.

When deploying pre-trained models in real-world downstream applications, two challenges arise: (i) the need for *explaining the results* as the models are very complex (Madsen et al., 2022), and (ii) the need for adapting the models in *low-resource scenarios* as applications in special domains or languages typically do not provide many labeled training instances (Hedderich et al., 2021).

For challenge (ii), previous work has shown that fine-tuning models in low-resource settings is hard (or even impossible for zero-resource settings) while prompting can yield good performance in those cases (Brown et al., 2020; Schick and Schütze, 2021; Liu et al., 2022). In terms of challenge (i), there is a research gap of carefully analyzing the difference of fine-tuned models (FTMs) and prompt-based models (PBMs). Most methods that have been proposed to enhance models' explainability (Ribeiro et al., 2016; Lundberg and Lee, 2017) have so far only been studied in the context of FTMs (Atanasova et al., 2020; DeYoung et al., 2020; Ding and Koehn, 2021), e.g., to answer the question which attribution method works best for different models and tasks. To the best of our knowledge, no previous work has explored attribution scores from PBMs (neither encoder-based nor decoder-based models, i.a., large language models) nor compared their quality to signals extracted from FTMs.

In this paper, we thus address the following questions: (1) *How plausible and faithful are explanatory signals extracted from PBMs in comparison to FTMs?* While plausibility shows how plausible an explanation is according to human understanding, faithfulness measures to what extent the deemed important tokens are truly important for the predictions of the model. Thus, we evaluate explanations both from the perspective of humans and models, making the analysis comprehensive.[1]

In addition, we introduce a *new dimension into the analysis*, namely the number of training samples in order to carefully investigate the behaviour of different methods in low-resource settings.

In our second research question, we investigate the effects of different attribution methods: (2) *How well do different attribution methods perform in terms of plausibility and faithfulness?* We answer this question by comparing different methods

[1]Those two dimensions are also commonly studied in related work on models' explainability (Atanasova et al., 2020; Ding and Koehn, 2021).

LREC-COLING 2024, pages 6867–6875
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

(namely attention, Integrated Gradients and Shapley Value Sampling) using extensive statistical significance tests. We focus on explanations in the form of attribution scores that highlight the importance of different input parts since they are more closely related to the model input and output than, e.g., generated free-text explanations.

Our third question concerns the choice of the underlying model, taking into account the new trend of using large language models: (3) *Do the results for PBMs also hold for decoder-based large language models?* We show that we get comparable results when extracting attribution scores from a large language model.

For the first time, our paper shows that prompt-based models yield more plausible explanations than fine-tuned models in low-resource settings and Shapley Value Sampling consistently outperforms attention and Integrated Gradients in terms of both plausibility and faithfulness scores. Thus, prompting pre-trained (either encoder-based or decoder-based) transformer models is better in low-resource settings than fine-tuning them, not only in terms of task performance but also when extracting attribution scores as explanations.

2. Extraction of Attribution Scores

We analyze attribution scores from three different kinds of models: encoder-only models (e.g., BERT and similar models) following either the prompt-based paradigm (called "PBMs" in the following) or the fine-tuning paradigm (called "FTMs"), and decoder-only models (e.g., large language models) following the prompt-based paradigm (called "LLMs"). We do not investigate encoder-decoder models as we want to avoid mixing effects from cross-attention and self-attention. In the following paragraphs, we describe how we extract attribution scores from the different model types.

Extraction from PBMs. We illustrate our approach for extracting attribution scores from encoder-only PBMs in Figure 1: The input is composed of the actual task input (yellow boxes), trigger tokens (i.e., tokens providing task information, shown in blue boxes), and a prediction token (i.e., the token that the model needs to predict to solve the task, shown in the pink box). Based on the input, the model computes probabilities for the prediction token. Given the predicted label tokens,[2] we then extract attribution scores for the actual task input. In particular, we use attention scores, Integrated Gradients and Shapley Value Sampling

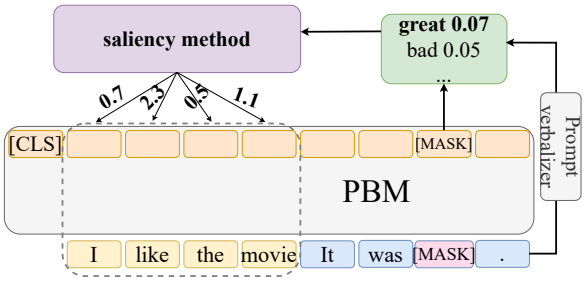


Figure 1: Extraction of explanatory signals from PBMs. Yellow boxes: actual task input. Blue boxes: trigger tokens. Pink box: prediction token. Orange boxes: last hidden representations of PBM. Green box: predicted label (converted by verbalizer, e.g., positive → great, negative → bad).

in our study. For attention, we extract attention scores from the last hidden layer of the [MASK] token, average them across different attention heads and normalize the attention scores over the actual task input. For Integrated Gradients and Shapley Value Sampling, we calculate attribution scores using the Captum package.[3]

Extraction from FTMs. For FTMs, the process is similar except that there are no prompts appended at the end of the sentences. Instead of using the language modeling head (the [MASK] token) for prediction, we use the default classification head (the [CLS] token) for FTMs and extract attribution scores for each token of the actual task input based on the predictions.

Extraction from LLMs. Extracting attribution scores from generative models is more challenging as they typically generate a whole sequence of output tokens and the position of the prediction token is not clear. To tackle this issue, we explicitly prompt the model to output only the verbalized class label.[4] Prompts can be found in Section A.2. Then we detect if the generated output corresponds to one of the verbalized class labels or not. If yes, we treat the class label as the prediction token. If not, we treat the first token in the generated output sequence as the prediction token. Finally, we extract attribution scores for the actual task input based on the prediction token, as we did in the extraction from PBMs or FTMs.

[2]We decided to use the tokens from the verbalizer instead of the true task labels as in Atanasova et al. (2020) because it cannot be assumed to have access to the true labels in real-world scenarios.

[3]https://captum.ai

[4]We chose the verbalizer such that the label name is part of the model's vocabulary and is not split into several subtokens.

3. Experimental Setup

Tasks and data sets. We use a sentiment classification (Tweet Sentiment Extraction (TSE)[5]) and a natural language inference dataset (e-SNLI (Camburu et al., 2018)) to cover tasks of different semantic depth and use their annotations of token-level explanations. Statistics of the datasets can be found in Table 1.[6] To create low-resource settings, we subsample the training sets into six low-resource sets, ranging from eight instances to the whole set.

Data set	TSE	e-SNLI
Train	11931	549367
Dev	2983	9842
Test	1449	9825

Table 1: Number of training, development, and test instances in TSE and e-SNLI.

Base models. For our main analysis, we focus on state-of-the-art encoder-based transformer models since running large language models (LLMs) on all our evaluation setups would have been infeasible due to extensive computational costs. In particular, we use BERT-base (Devlin et al., 2019), BERT-large, and RoBERTa-large (Liu et al., 2019). Nevertheless, we also perform a small comparative study with LLMs afterwards, namely with the Vicuna model (Chiang et al., 2023), a fine-tuned LLaMA version (Touvron et al., 2023).

Prompting methods. In our study, we focus on discrete prompts because they are more explainable than continuous prompts and also the standard input for LLMs. To be able to factor out possible differences stemming from the choice of the prompting method, we study three different methods: *Manual* uses a prompt from Schick and Schütze (2021) and fine-tunes all parameters of the model. *BitFit* uses the manual prompt but updates only the bias terms of the model during fine-tuning (Logan IV et al., 2022), and *BFF* automatically searches for a prompt (Gao et al., 2021) and fine-tunes all parameters with that prompt. Prompts and verbalizers are provided in Table 2.[7]

[5]https://www.kaggle.com/c/tweet-sentiment-extraction

[6]For TSE, we exclude data with the neutral label because their annotated explanations are mostly the whole sentence.

[7]For the LLM, we use manual prompts only.

Training details. We use 4-fold cross-validation to tune both PBMs and FTMs. The hyperparameters can be found in Section A.1.

Evaluation metrics. We evaluate the plausibility and faithfulness of the explanatory signals. Those two dimensions allow to investigate explanations both from the perspective of humans and models. They are also commonly used in related work on explainability (Atanasova et al., 2020; Ding and Koehn, 2021).

Plausibility indicates how plausible an explanation is according to human intuition. We quantify this with average precision (Atanasova et al., 2020).[8]

Faithfulness shows a model's ability to accurately represent its reasoning process. In related work, an established way of quantifying this is measuring the performance decrease when masking the most salient words (DeYoung et al., 2020; Atanasova et al., 2020). We follow Atanasova et al. (2020) and create several dataset perturbations by masking 0, 10, 20, ..., 100% of the tokens in the order of decreasing saliency. To calculate a single measure for faithfulness, the area under the threshold-performance curve (AUC) is used. However, this measure does not allow cross-model comparisons. Therefore, we normalize the AUC as the proportion of the area under the curve to the whole area (calculated as the highest possible performance multiplied by the number of thresholds). The lower the normalized score, the better the explanation is in faithfully showing the model's reasoning.

To investigate the *statistical significance* of our results, we apply Kruskal-Wallis tests and Dunn's Tests for pairwise differences.

4. Results

4.1. Comparing PBMs and FTMs

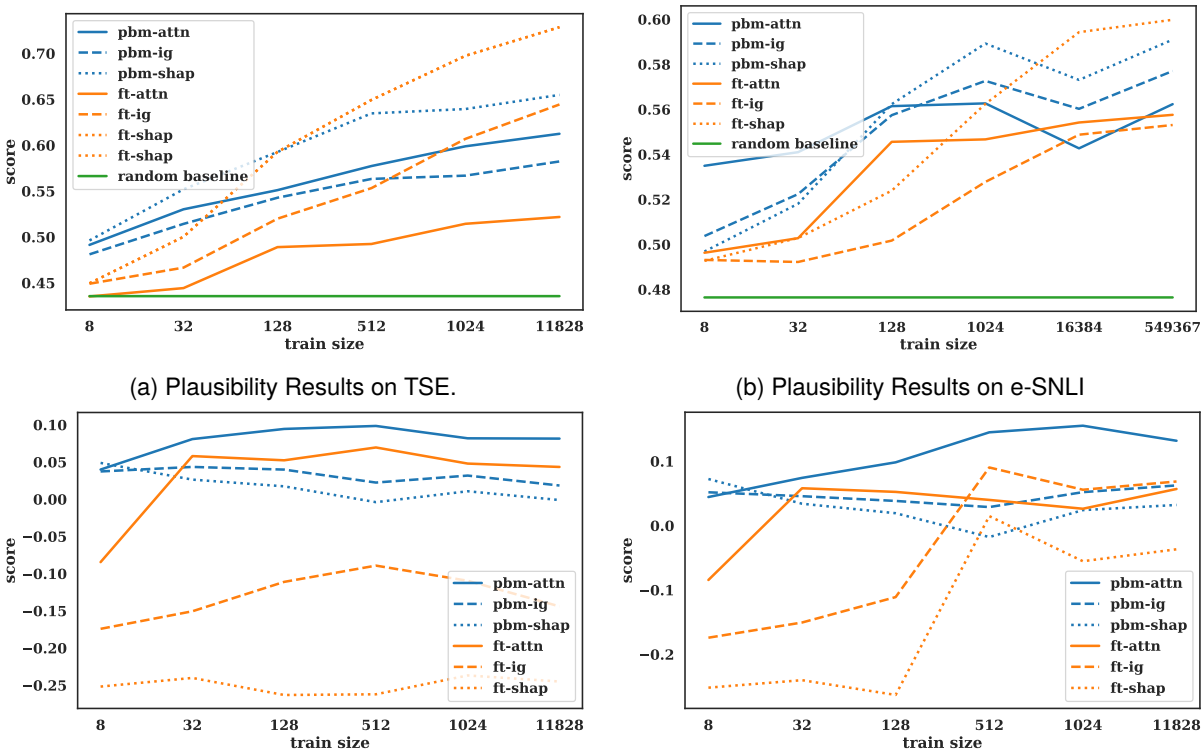
We extract attribution scores from all PBMs and FTMs and compute its plausibility and faithfulness scores for different training sizes, averaging out effects from different prompting methods and base models. Figure 2 shows the results. Individual results for base models and attribution methods can be found in Section A.4. We also report the task performance of the models in Section A.3.

Plausibility. For smaller training sizes, PBMs outperform FTMs but the trend reverses as the training size increases. To investigate whether the differences between PBMs and FTMs in the low/rich-resource settings are significant, we set

[8]sklearn.metrics.average_precision_score

Task	Prompt	Verbalizer	Setting
TSE	[S] It was [P]. This is [P]. [S]	terrible/great ragged/soldiers	Manual/Bitfit BFF
e-SNLI	[S1] ? |[P] , [S2] [S1] . [P] , no , [S2]	yes/no/maybe alright/except/watch	Manual/Bitfit BFF

Table 2: Prompts for TSE and e-SNLI in different settings. [S] stands for the sentence ([S1] and [S2] are the premise and hypothesis respectively), and [P] is the prediction token. For TSE, the verbalizers correspond to *positive/negative*. For e-SNLI, the verbalizers correspond to *entailment/contradiction/neutral*.



(a) Plausibility Results on TSE.

(b) Plausibility Results on e-SNLI

(c) Faithfulness Results on TSE

(d) Faithfulness Results on e-SNLI

Figure 2: Plausibility (the higher the better) and faithfulness (the lower the better) scores for different prompting methods and fine-tuning and different explanation methods, averaged across base models and seeds. The faithfulness results are shown as the difference between faithfulness scores of the resp. explanation method and the gold standard. *attn*: attention, *ig*: Integrated Gradients, *shap*: ShapSample.

up two bins for each task: we treat the two training sets with fewest data as low-resource and the two training sets with most data as high-resource. Within the low-resource part of the data, we find all comparisons to be statistically significant (TSE: $H(89)$=73.86, p<0.001, e-SNLI: $H(89)$=29.24, p<0.001). Within the high-resource part of the data, the differences are not significant. We also calculate the random baseline for plausibility scores (0.436 for TSE and 0.476 for e-SNLI) and find that explanations provided by PBMs achieve considerably higher plausibility scores than the random baselines in low-resource setting. Method-wise, we find that for both tasks, the plausibility scores of the explanations extracted by ShapSample are significantly higher than those from attention and Integrated Gradients.

Plausibility error analysis. We sample 20 instances per dataset for each attribution method to conduct a small error analysis in terms of plausibility of explanations. We find that Integrated Gradients tend to assign negative values to functional words. We also find that attention seems to encode sentence information into a single token, so a specific token can get high attribution scores.

Faithfulness. Figures 2c and 2d show that faithfulness scores are influenced by the attribution

Data	Model	Plausibility			Faithfulness		
		attn	ig	shap	attn	ig	shap
TSE	RoBERTa	.56	.57	.56	.02	**.00**	.01
	Vicuna	.47	.57	**.59**	.07	.06	.02
e-SNLI	RoBERTa	.53	.51	.50	.22	.09	.11
	Vicuna	.43	.51	**.55**	.02	.05	**.00**

Table 3: Plausibility (the higher the better) and faithfulness (the lower the better) scores of explanations obtained from Vicuna and RoBERTa.

methods. For instance, explanations extracted from FTMs with ShapSample are more faithful than explanations from PBMs independent of the number of resources. Explanations from PBMs with attention lead to the lowest faithfulness scores across all training sizes. For both datasets, we observe significant differences for all attribution method pairs except for ShapSample and gold. Thus, Shapley Value Sampling attribution scores are comparably faithful as gold annotations.

4.2. Studying LLMs

Given the increased relevance of large language models, we now investigate whether our findings hold for them as well and which plausibility and faithfulness scores we get for them compared to PBMs (i.e., encoder-based models used with prompting). Due to the large computational costs for obtaining attribution scores from LLMs, we limit the number of test instances to 100 for each data set and evaluate the 8-shot setting only. For the LLM, the 8 training samples are provided in each input prompt. The prompts can be found in Section A.2. For the PBM (we chose RoBERTa-Large with BitFit prompts which was the best performing individual model in our previous analysis), the 8 training samples are used to tune the bias terms of the model.

The results in Table 3 show that Shapley Value Sampling again leads to more plausible and faithful explanations for Vicuna. When comparing Vicuna with RoBERTa, we note larger performance gaps among the attribution methods. We further note that the plausibility scores of attention are even lower for Vicuna than for RoBERTa. A reason could be that LLMs encode a larger input context and, thus, information of tokens that are irrelevant to the prediction might also be encoded.

4.3. Discussion

Comparison of attribution methods. ShapSample consistently yields more plausible explanations than methods. We assume the reason for this lies in the calculation of Shapley Values: it takes in every permutation of features enabled to calculate a feature's importance. For instance, if we have a feature set "good", "day", the attribution score of the feature "good" is calculated by every permutation that contains it, i.e., "good" and "good day". Whereas for Integrated Gradients, this is not considered. We think taking each permutation to calculate feature importance is helpful in models like BERT, as context is of vital importance. Attention is the least plausible; this observation is in line with previous works, e.g., Bibal et al. (2022).

PBMs vs. FTMs vs. LLMs in low-resource settings. PBMs yield more plausible attribution scores than FTMs in low-resource settings. We think this might be because PBMs pick up task information quicker than FTMs in the low-resource settings, so the explanations given by PBMs are more plausible. Our study with LLMs shows that the trends of LLMs are comparable to the trends of PMBs, indicating the relevancy of our findings.

5. Conclusion

In this paper, we studied attribution scores extracted from prompt-based models in comparison to fine-tuned models, and compared different attribution methods w.r.t. plausibility and faithfulness scores. Our main findings were: (1) Prompt-based models generate more plausible explanations in low-resource settings. (2) Shapley Value Sampling outperforms other attribution methods, such as attention and Integrated Gradients across tasks and settings and is similarly faithful as gold annotations. (3) Our findings seem to be transferable to generative large language models.

Directions for future work are the investigation of soft prompts as well as a more extensive study of explanatory signals from large language models.

6. Bibliographical References

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.

Adrien Bibal, Rémi Cardon, David Alfter, Rodrigo Wilkens, Xiaoou Wang, Thomas François, and Patrick Watrin. 2022. Is attention explanation? an introduction to the debate. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), pages 3889–3900, Dublin, Ireland. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *NeurIPS*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

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.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Shuoyang Ding and Philipp Koehn. 2021. Evaluating saliency methods for neural language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5034–5052, Online. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. 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 3816–3830, Online. Association for Computational Linguistics.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2021. A survey on recent approaches for natural language processing in low-resource scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2545–2568, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2022. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Robert Logan IV, Ivana Balazevic, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2022. Cutting down on prompts and parameters: Simple few-shot learning with language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2824–2835, Dublin, Ireland. Association for Computational Linguistics.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.

Andreas Madsen, Siva Reddy, and Sarath Chandar. 2022. Post-hoc interpretability for neural nlp: A survey. *ACM Comput. Surv.*

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings*

of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 255–269, Online. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

A. Appendix

A.1. Hyperparameters

Table 4 provides the hyperparameters for training our models. The learning rate and accumulation steps are automatically selected based on the validation accuracy from the range of [1, 2, 4] for the accumulation steps and [1.0e-5, 5.0e-5, 1.0e-4, 5.0e-4] for the learning rate.

Task	Training size	Batch size	Epoch
TSE	8/32/128	2	30
TSE	512	4	15
TSE	2048/11828	8	5
e-SNLI	8/32/128	2	30
e-SNLI	1024	4	15
e-SNLI	16384/549367	32	3

Table 4: Hyperparameters.

A.2. LLM Prompts

- TSE: You will be given a target sentence and you will decide the sentiment of the sentence (Please return either yes or no only). Here are some examples: Input: {s1} Output: {l1}...Input: {s8} Output: {l8}
- e-SNLI: You will be given a pair of sentences and you will decide the relationship between the sentences (Please return yes for entailment, no for contradiction, maybe for neutral only.). Here are some examples: Input1: {s1} Input2: {s2} Output: {l1}...Input1: {s8} Input2: {s8} Output: {l8}

A.3. Task Performance

Figure 3 shows the task performances of PBMs and FTMs with regards to different training sizes.

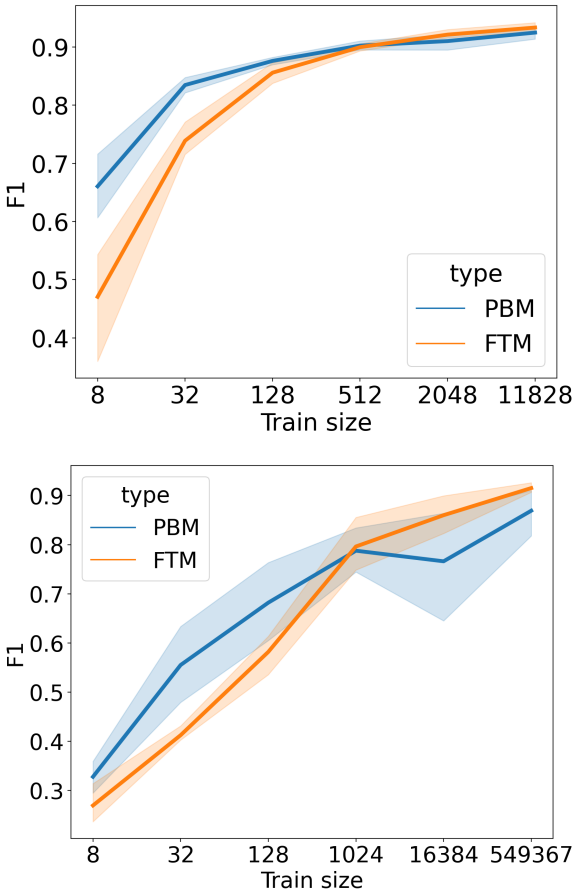
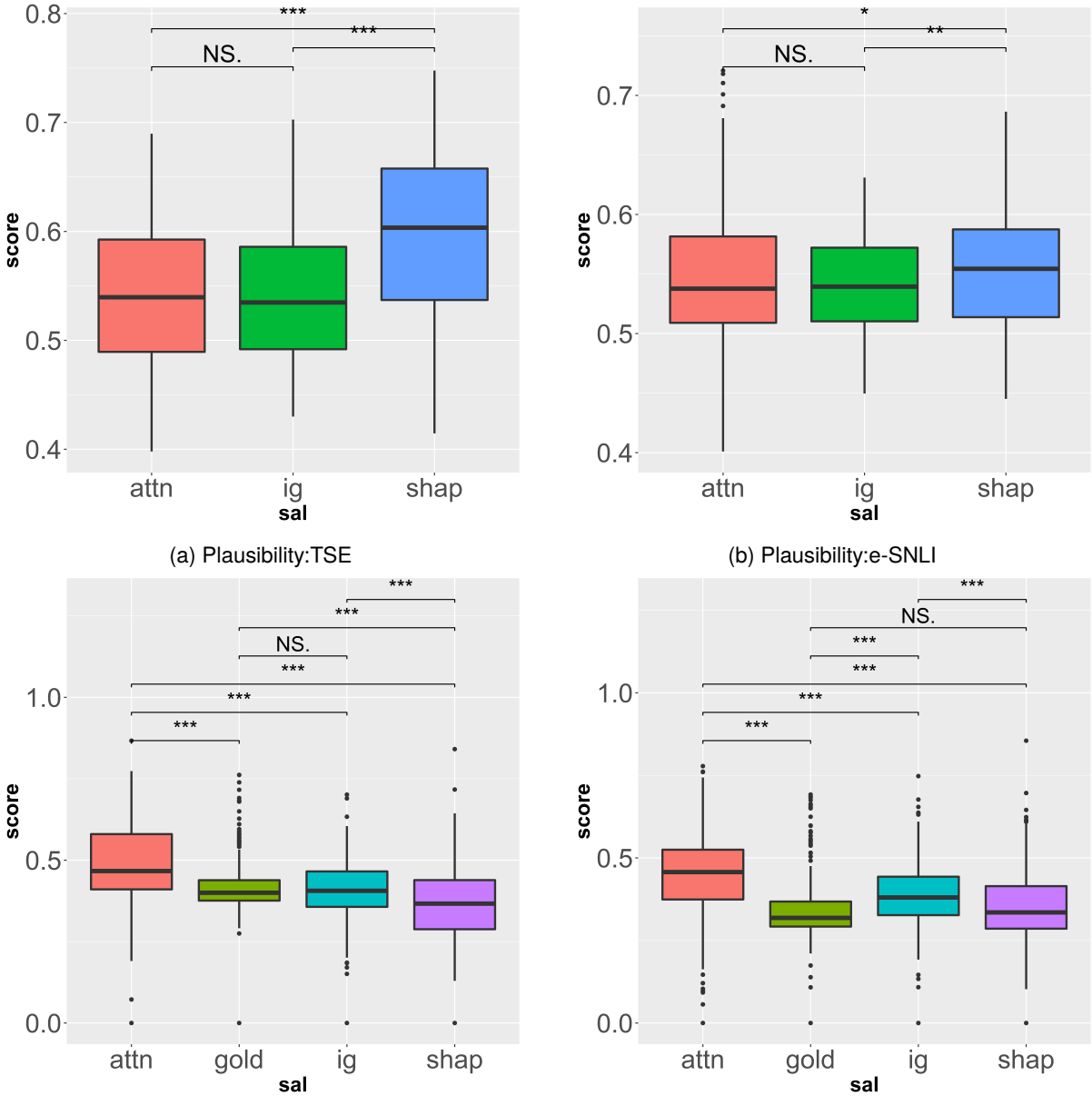


Figure 3: The F_1 scores of models trained with different sizes. From top to bottom: TSE and e-SNLI.

A.4. Comparing Saliency Methods and Base Models

Figure 4 illustrates the plausibility and faithfulness scores per saliency method, averaged across models, training sizes, prompting methods and seeds. Figure 5 illustrates the plausibility scores per base model, averaged across saliency methods, training sizes, prompting methods and seeds.



(a) Plausibility:TSE

(b) Plausibility:e-SNLI

(c) Faithfulness:TSE

(d) Faithfulness:e-SNLI

Figure 4: The Plausibility scores of explanatory signals, averaged across base models, training sizes and prompting methods. *attn* stands for attention. *ig* stands for Integrated Gradients and *shap* stands for Shapley Value Sampling. *gold* stands for the gold annotations. *NS* stands for the no significant difference. *, **, *** stand for *p*-value <.05, .01 and .001.

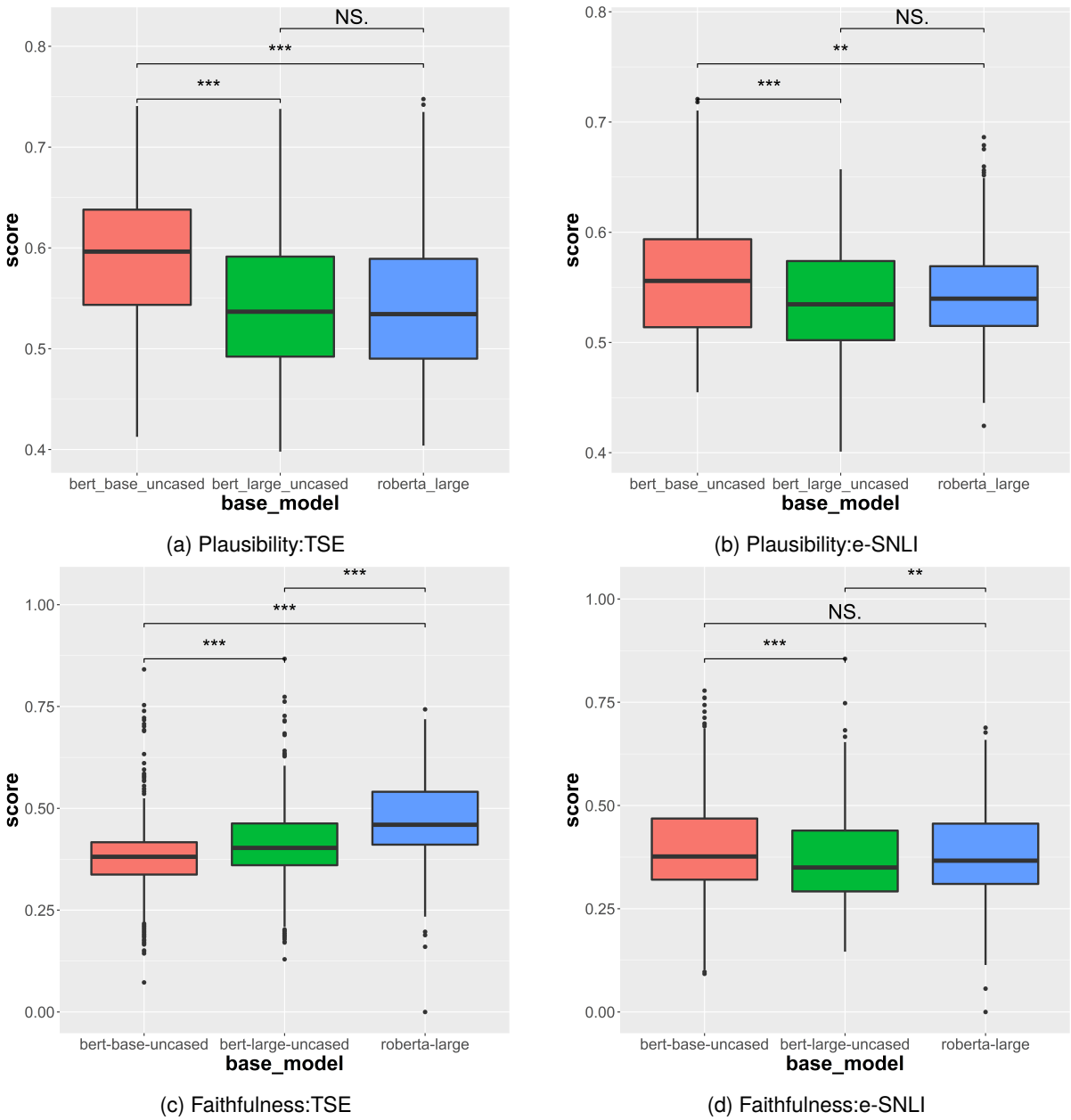


Figure 5: The Plausibility scores of base models, averaged across saliency methods, training sizes and prompting methods. *attn* stands for attention. *ig* stands for Integrated Gradients and *shap* stands for Shapley Value Sampling. *gold* stands for the gold annotations.*NS* stands for the no significant difference. *, **, *** stand for *p*-value <.05, .01 and .001.