

000 HYBRID ATTRIBUTION PRIORS FOR EXPLAINABLE 001 AND ROBUST MODEL TRAINING 002

003 **Anonymous authors**
004

005 Paper under double-blind review
006

007 ABSTRACT 008

009 Small language models (SLMs) are widely used in tasks that require low latency
010 and lightweight deployment, especially classification. With the growing emphasis
011 on interpretability and robustness, explanation-guided learning offers an effective
012 framework by incorporating attribution-based supervision during training.
013 However, how to derive general and reliable attribution priors remains an open
014 challenge. Upon analyzing representative attribution methods in classification
015 tasks, we find that while these methods reliably highlight class-relevant tokens,
016 they tend to focus on common keywords shared by semantically similar classes.
017 Since these classes are already prone to confusion under standard training, the
018 attributions fail to provide sufficient discriminative cues, limiting their ability to
019 enhance model differentiation. To address this challenge, we introduce Class
020 Aware Attribution Prior (**CAP**), a novel attribution prior extraction framework
021 designed to guide language models in capturing fine-grained class distinctions,
022 thus producing more salient and discriminative attribution priors. Building on
023 this, we propose **CAP**_{Hybrid}, which integrates priors from CAP and existing attribution
024 techniques to form a more comprehensive and balanced supervisory signal.
025 By aligning the model’s self-attribution with these enriched priors, our approach
026 encourages the model to capture diverse decision-relevant features. Extensive ex-
027 periments across full-data, few-shot, and adversarial settings demonstrate that our
028 method consistently enhances both model interpretability and robustness.
029

030 1 INTRODUCTION 031

032 Small language models (SLMs), such as Bert (Devlin et al., 2019) and Roberta (Liu et al., 2019b),
033 have become indispensable, particularly for classification scenarios, due to their efficient inference
034 capabilities in resource-constrained environments and the ease of deployment on devices. SLMs
035 power a wide range of essential applications, such as intent detection in dialogue systems (Zhang
036 et al., 2023), sentiment analysis in social media (Wankhade et al., 2022), and topic categorization
037 on content platforms (Li et al., 2022b). In recent years, the performance of SLMs on standard
038 classification benchmarks has improved considerably (Wang et al., 2019), achieving competitive
039 accuracy with significantly smaller model sizes. The focus has gradually shifted to improving the
040 interpretability and robustness of SLMs so that they can meet the real-world demand for transparent
041 model decision-making mechanisms (Taha et al., 2024). Some researches focus on explanation-
042 guided learning (Liu & Avci, 2019), in which models are trained not only to predict correct labels but
043 also to align attribution priors to identify the most relevant input components for a given prediction,
044 thus enhancing interpretability and potentially improving robustness.

045 Current methods for obtaining attribution priors fall into two main categories. The first derives pri-
046 ors from the trained model itself or from larger, pre-trained variants of the same architecture using
047 techniques such as attention attribution or integrated gradients (Wu et al., 2023). These self-derived
048 priors reflect the model’s own insights into the data and knowledge, but can be biased, especially
049 in small models with limited capacity. The second category involves human-annotated priors that
050 encode linguistic or domain-specific knowledge (Jayaram & Allaway, 2021), offering greater trans-
051 parentency and generalizability. However, these priors are expensive to construct and lack scalability.
052 So there raises a natural question: *How to obtain attribution priors that are both reliable and scal-
053 able, which capturing both model-internal reasoning and external linguistic knowledge?* To explore
this question, we try to solve the following three problems:

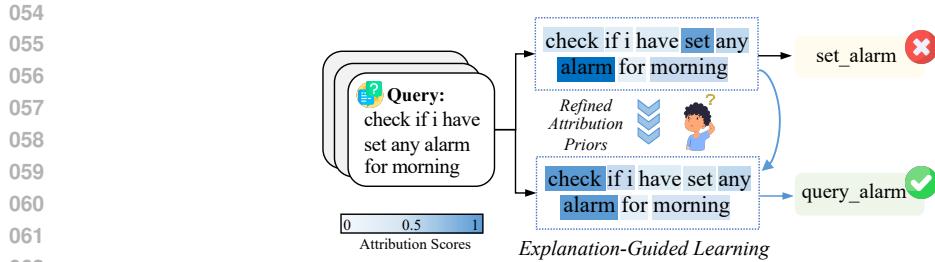


Figure 1: Training model with attribution produced by different methods for the same case.

Q1: Do existing attribution methods exhibit systematic limitations in SLM-based text classification?

Q2: Large language models inherently encode rich prior knowledge and possess strong language modeling capabilities. But how can we effectively extract high-quality attribution priors from them and integrate these priors with existing methods to enhance explanation-guided learning?

Q3: Will the refined attribution prior consistently achieve good interpretability and robust supervision across tasks in a variety of settings?

Specifically, we first analyze the characteristics of the existing typical attribution methods on classification tasks. For the traditional self-attribution method, such as integrated gradients, they inevitably inherit biases from the training data and cannot introduce novel and complementary knowledge. For the typical LIME and SHAP methods that can extract attribution scores from LLMs to guide SLMs, we observe two key phenomena: homogenization and class confusion. First, we find that both methods produce highly correlated attribution scores and consistently highlighting tokens with high affinity to the target label. Second, the class-level analysis of the top LIME-ranked words reveals that certain classes share many frequent keywords, showing clear word overlap among various classes. Importantly, these confused classes correspond to those most often misclassified by the vanilla-trained target model. Although LIME captures class-relevant cues, it fails to emphasize discriminative features that distinguish these ambiguous classes, thus directly using such priors to supervise training may amplify errors in confused classes. As shown in Figure 1, if the prior only captures the keywords *set* and *alarm* in the *set_alarm* category, it may lead to more serious misjudgments for close categories with similar keywords, such as the example case from *query_alarm*. Ideally, the attribution would fully comprehend the semantics, and assign a high attribution score to the keyword *check* for this case.

To solve these issues, we introduce Class-Aware Attribution Prior (CAP), a novel attribution prior extraction method that enriches the input text with task instructions and label space. By prompting LLMs in this way, CAP captures class distributions and generates more salient and discriminative attribution priors. Motivated by the complementarity among different attribution methods, we then fuse attribution priors from CAP and existing methods creating a more refined and reliable prior signal. By subsequently aligning the model's self-attributions with this rich prior knowledge during training, the SLM learns more reliable and general reasoning patterns. We evaluate our method across three diverse datasets on full-data training, few-shot learning, and adversarial settings. Extensive results demonstrate that the refined attribution prior consistently improves classification accuracy while significantly enhancing interpretability and robustness to adversarial perturbations. Our main contributions are summarized as follows:

- Through analysis of attribution scores from existing methods on text classification tasks, we identify two key limitations: homogenization and class confusion. This will lead to misclassification of semantically similar categories that share overlapping keywords under the attribution guided learning paradigm.
- We propose a class-aware attribution prior method that extracts salient and discriminative priors, along with a hybrid variant that captures more comprehensive attribution signals.
- We evaluate the effectiveness of our method across three experimental settings, focusing on accuracy, interpretability, and robustness. In addition to quantitative results, we also conduct qualitative analyses to further validate the impact of our method.

108
109

2 RELATED WORK

110
111

2.1 MODEL ROBUSTNESS AND INTERPRETABILITY

112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
Robustness and interpretability are critical components of any responsible and trustworthy models, where their black-box nature makes them vulnerable to adversarial attacks (Jyoti et al., 2022), especially in safety-critical or risk-sensitive domains (Tjoa & Guan, 2021; Kaur et al., 2023). Several studies have investigated adversarial attacks and defenses to bolster model robustness (Zhang et al., 2021; Hendrycks et al., 2020). Likewise, recent efforts have focused on enhancing the stability of explanation techniques (Wang et al., 2020; Li et al., 2023). Researchers have examined the connections between robustness and interpretability, highlighting the elicitation of more reliable explanations as a key pathway to enhance interpretative robustness (Li et al., 2022a). For interpretability, in recent years, many explanation techniques have been developed to better understand the decision making process of deep neural networks (Bach et al., 2015; Ribeiro et al., 2016; Selvaraju et al., 2017; Lyu et al., 2024). Integrated gradients (Sundararajan et al., 2017) addresses input-sensitivity by estimating each feature contribution along the straight-line path from a baseline input to the actual input. LIME (Ribeiro et al., 2016) generates perturbed samples by masking different subsets of features around the target instance, computes the predictions on these samples, and then fits a locally weighted surrogate model to attribute importance. SHAP (Lundberg & Lee, 2017) provides a unified interpretation framework that encompasses six prior methods, assigning each feature an importance score for the given prediction.128
129

2.2 ATTRIBUTION-DRIVEN LEARNING

130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
Recent studies have shown that integrating explanation signals can enhance model performance (Gao et al., 2022a;b; Rao et al., 2023). (Ross et al., 2017) regularize models by penalizing input gradients to match expert-defined attribution maps. e²KD (Parchami-Araghi et al., 2024) improves distillation faithfulness by aligning GradCAM explanations (Selvaraju et al., 2017) or B-cos model attributions (Böhle et al., 2022) between teachers and students. In natural language processing domains, several studies leverage human-annotated rationales as attribution priors. DBLP:conf/emnlp/JayaramA21 crowdsource word importance annotations on a small subset to construct attribution priors and apply them into mean attention weights attribution methods for detecting opinions and beliefs from text. (Liu & Avci, 2019) collect known toxic words and identity terms, and achieves better performance in text classification by setting an attribution target of 1 on toxic words and 0 for identity terms. DBLP:journals/corr/abs-1908-06870 propose to enhance faithful explanations of model attention for fine-grained sentiment analysis by aligning the model attention with human rationales. AD-KD (Wu et al., 2023) leverages the multi-view attribution distillation for model compression. DBLP:conf/naacl/PimparkhedeB25 propose to use main predicate in the text utterances along with the arguments of the main predicate to serve as explanation signals for intent classification. Although effective, these methods depend on costly human annotations or require parameter updates, limiting their scalability. In this paper, we propose a novel approach that automatically extracts comprehensive attribution priors from LLMs without any human effort or model fine-tuning.147
148
149

3 ANALYSIS OF ATTRIBUTION PRIORS

150
151
To answer the **Q1**, we analyze the characteristics of existing attribution methods in this section.152
153

3.0.1 INTEGRATED GRADIENT

154
155
156
157
158
159
160
161
As a typical white-box attribution method, Integrated Gradients (IG) computes token importance based on the model’s internal gradient information. Following (Wu et al., 2023), a common approach in attribution-prior guided training for small language models (SLMs) is to first train a model on the dataset, then extract attribution scores from this trained model over the training data, and incorporate these scores as priors during subsequent training. This method’s advantage lies in the strong alignment between the attribution priors and the model’s own decision process, enabling the model to reinforce its learned reasoning patterns. However, this approach also has limitations: since the attribution depends on the initially trained model, it is susceptible to biases present in that model; moreover, IG captures only local gradient-based information and may not effectively lever-

age semantic label information or external knowledge. This limitation is particularly pronounced in few-shot or fine-grained settings, where distinguishing between semantically similar classes requires richer and more discriminative attribution signals.

3.0.2 LIME AND SHAP

We investigate two representative black-box attribution methods, LIME and SHAP, which are capable of extracting rationale priors from large language models' predictions. These methods provide additional explanatory information from large models that can be used to guide the training of smaller models. Firstly, we take text as the input and the corresponding label as the target output to automatically extract priors by estimating the impact of each input token on the model target output, thus yielding the corresponding attribution scores. This leads to a question: what characteristics define high-quality attribution priors? To explore this, we analyze the attribution scores generated by LIME and SHAP on the Banking77 dataset and observe two key phenomena: homogenization and class confusion. First, both methods highlight words with high affinity to the target label. We also calculate the Pearson correlation coefficient between two scores, and find that it reaches 0.8, suggesting that the two methods focus on the similar subset of words. Second, we examine class-level behavior by identifying the words with higher LIME attribution scores for each class. By calculating the lexical overlap across classes, we observe that while many classes have distinct keywords, a subset exhibits substantial overlap in their attributed terms, called confused classes. Directly using such priors to supervise training may amplify errors in these categories, as they encourage the model to focus on overlapping words.

We further analyze the relationship between the misclassified classes from models trained solely with classification loss and the confused classes. Figure 2 shows the relationship between misclassified samples under standard training and confused classes of LIME. The horizontal axis shows the top 15 classes sorted by the number of misclassifications, with higher values indicating more frequent errors. The vertical axis represents the degree of keyword overlap between class pairs based on LIME attributions, where higher values indicate greater overlap. As shown in Figure 2, the majority of misclassifications occur in regions with high overlap levels, i.e., confused classes.

This suggests that LIME attribution scores can not solve the error of standard trained model and relying solely on LIME priors may worsen inter-class confusion. This analysis highlights that high-quality attribution priors should not only exhibit strong affinity with the target label but also capture salient and discriminative features critical for distinguishing ambiguous classes.



Figure 2: Distribution of wrong predictions across overlap levels. Darker colors indicate more misclassified samples. Under standard training, wrong predictions tend to occur in classes with higher overlap in LIME attribution scores.

As shown in Figure 3, we introduce an explainable framework that generates comprehensive attribution priors to guide the target model toward a more faithful and robust learning process for answering **Q2**. To capture the salient information for each class, we propose a novel class-aware attribution extraction module to extract priors from large language models automatically. Further we conduct hybrid attribution prior fusion to obtain comprehensive multi-view priors. By leveraging their complementary characteristics, the fused prior highlights both high-affinity frequent tokens and discriminative class-boundary cues. Finally, during explanation-guided learning, we align the target model attribution scores with the fused priors, resulting in a more faithful and robust training.

4 THE PROPOSED METHOD

As shown in Figure 3, we introduce an explainable framework that generates comprehensive attribution priors to guide the target model toward a more faithful and robust learning process for answering **Q2**. To capture the salient information for each class, we propose a novel class-aware attribution extraction module to extract priors from large language models automatically. Further we conduct hybrid attribution prior fusion to obtain comprehensive multi-view priors. By leveraging their complementary characteristics, the fused prior highlights both high-affinity frequent tokens and discriminative class-boundary cues. Finally, during explanation-guided learning, we align the target model attribution scores with the fused priors, resulting in a more faithful and robust training.

4.1 CLASS-AWARE ATTRIBUTION PRIORS

To address the lack of discriminative attribution signals for ambiguous samples in difficult classes, we propose Class-aware Attribution Priors (CAP) that enhance saliency by emphasizing the key

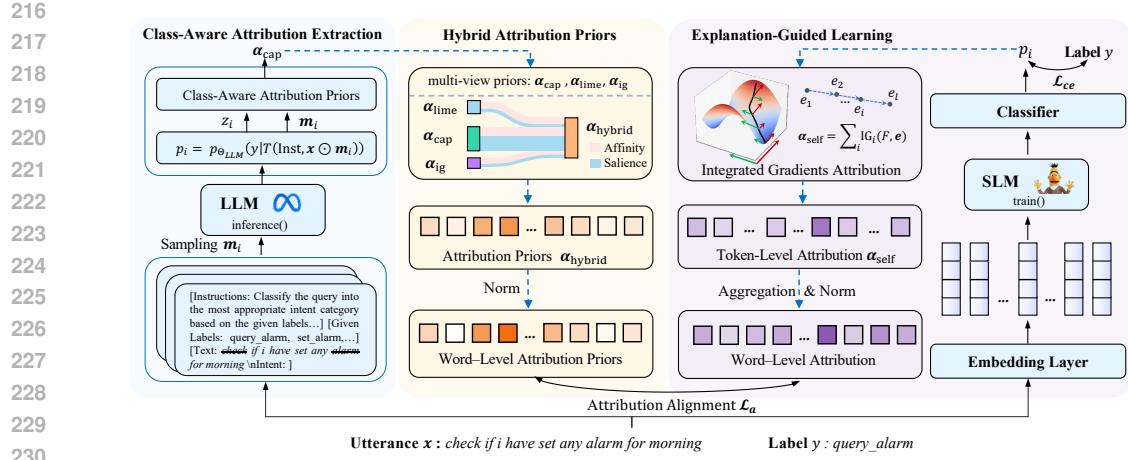


Figure 3: The illustration of our proposed framework. To extract salient priors for ambiguous classes, we first introduce class-aware attribution priors derived from large language models. We then perform hybrid attribution prior fusion to obtain more comprehensive multi-view priors. In the explanation-guided learning module, these priors guide the model toward greater faithfulness and robustness through attribution alignment.

features distinguishing one class from another. As illustrated in Figure 3, we enrich the input text with task instructions and label space, which prompts the LLM to capture the class distribution and produce more salient insights. We then randomly mask individual words and query the LLM for the true-label probability of each masked variant. By using an efficient and stable factorization of the linear equations between these probabilities and the attribution priors, we derive per-word scores.

Given a sentence $x = [w_1, w_2, \dots, w_d]$ with d words, we leverage the perturbation vector $\mathbf{m} \in \{0, 1\}^d$, where 0 and 1 indicate whether the corresponding word is removed or included, respectively. Due to the significant computational overhead for 2^d possible scenarios, we sample n perturbation vectors for predicting the attribution priors $\alpha \in \mathbb{R}^d$. Specifically, to construct \mathbf{m}_i , we first sample the number of masked words s from the range $[2, \lfloor \frac{d}{2} \rfloor + 1]$. Then we generate the binary mask containing exactly s zeros, where masked positions are chosen uniformly at random. For the specific case $s = 1$, each word is masked individually. By repeating this procedure n times, we obtain $\mathbf{m}_1, \dots, \mathbf{m}_n$ to predict and learn the attribution priors α .

$$\alpha = \operatorname{argmin} \left\{ \sum_{i=1}^n (z_i - \alpha^\top \mathbf{m}_i)^2 + \lambda \|\alpha\|^2 \right\}, \quad (1)$$

where z_i is the targeted score, which is positively correlated with the probability of answers, and λ is the regularization coefficient. Since probabilities for multi-token answers can be exceedingly small, we apply a logarithmic transformation. Specifically, we set $z_i = -\frac{1}{\log p_i}$, and p_i is computed as follows:

$$p_i = p_{\Theta_{LLM}}(y|T(\text{Inst.}, x \odot \mathbf{m}_i)), \quad (2)$$

where T is the instruction template, Inst. denotes the task instructions and label space, Θ_{LLM} is the parameters of the frozen large language model, and \odot denotes the element-wise masking operation.

To solve Eq. (1), we set its derivative to zero and solve for α , yielding:

$$\sum_{i=1}^n (\mathbf{m}_i \mathbf{m}_i^\top) \alpha + \lambda \alpha = \sum_{i=1}^n z_i \mathbf{m}_i. \quad (3)$$

The detailed theoretical proof is provided in Appendix A.3. We then construct the matrix form. Let $M = [\mathbf{m}_1^\top, \mathbf{m}_2^\top, \dots, \mathbf{m}_n^\top] \in \mathbb{R}^{n \times d}$, $\mathbf{z} = [z_1, z_2, \dots, z_n] \in \mathbb{R}^d$, the Eq. (3) is:

$$(M^\top M + \lambda I) \alpha = M^\top \mathbf{z}. \quad (4)$$

We define $A = M^T M + \lambda I$, and $\mathbf{b} = M^T \mathbf{z}$. We show that A is a symmetric positive definite matrix, and provide the proof in Appendix A.4. Hence, A admits a unique Cholesky factorization (Press, 2007): $A = LL^T$, where L is a lower triangular matrix with real and positive diagonal entries. To solve $LL^T \boldsymbol{\alpha} = \mathbf{b}$, we then solve the two triangular systems in sequence $Ly = \mathbf{b}$ and $L^T \boldsymbol{\alpha} = y$. Since L is a lower triangular matrix, we solve y by forward substitution. Having computed y , we solve $\boldsymbol{\alpha}$ by backward substitution because L^T is an upper triangular matrix. This factorization avoids any explicit matrix inversion and yields $\boldsymbol{\alpha}$ efficiently and stably.

4.2 HYBRID ATTRIBUTION PRIORS

Each method derives importance scores via distinct mechanisms, yielding its own advantages and shortcomings, as also corroborated by DBLP:conf/nips/HanSL22. To leverage their complementary strengths and enhance overall explanation reliability and model robustness, we incorporate attribution priors from three sources: LIME, our proposed CAP derived from LLMs, and IG from the same model series. Specifically, LIME excels at identifying tokens with high affinity to the target label, CAP emphasizes salient class-boundary terms to distinguish similar classes, and IG captures the internal sensitivity of the model based on gradient information. For fusion, we explore two aggregation strategies: word-wise mean and word-wise max. Before aggregation, all attribution score vectors are normalized to ensure comparability. Formally, given attribution priors $\boldsymbol{\alpha}_{\text{lime}}$, our proposed CAP priors $\boldsymbol{\alpha}_{\text{cap}}$, and IG signals $\boldsymbol{\alpha}_{\text{ig}}$, we define the fused prior as

$$\boldsymbol{\alpha}_{\text{hybrid}} = \text{aggregate}(\{(\boldsymbol{\alpha}_{\text{lime}}), (\boldsymbol{\alpha}_{\text{cap}}), (\boldsymbol{\alpha}_{\text{ig}})\}) \quad (5)$$

where `aggregate` denotes either the element-wise mean or max operation. This results in a comprehensive word-level attribution prior that maintains full coverage while integrating multi-source rationale signals.

4.3 EXPLANATION-GUIDED LEARNING

After obtaining attribution priors, we conduct word-level attribution alignment between the priors and the target model self-attribution. Specifically, we leverage integrated gradients to attribute the prediction of the target model to input features. Formally, given an input sentence $\mathbf{x} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{d'}]$ consisting of d' tokens, $\mathbf{e}_i \in \mathbb{R}^l$ is the i -th token embedding encoded by the embedding layer, and l is the dim of token embedding. For each feature \mathbf{e} , \mathbf{e}' is a baseline feature, and the target model is $F(\cdot)$. IG estimates the contribution of each input by integrating the gradients of F along the straight-line path from \mathbf{e}' to \mathbf{e} as the integral path:

$$F(\mathbf{e}) - F(\mathbf{e}') = \sum_{i=1}^l \text{IG}_i(F, \mathbf{e}) = \sum_{i=1}^l [(\mathbf{e}_i - \mathbf{e}'_i) \times \int_{\mu=0}^1 \frac{\partial F(\mathbf{e}' + \mu \times (\mathbf{e} - \mathbf{e}'))}{\partial \mathbf{e}_i} d\mu]. \quad (6)$$

Then IG attribution scores $\boldsymbol{\alpha}_{\text{self}}$ is normalized for the subsequent alignment process. For clarity, let \boldsymbol{a}^p denote the normalized external attribution priors, and \boldsymbol{a}^s represent the normalized self-attribution scores from the target model. We use Mean Squared Error (MSE) as the distance metric for alignment:

$$\mathcal{L}_a = \|\boldsymbol{a}^p - \boldsymbol{a}^s\|^2. \quad (7)$$

For overall objective, we jointly optimize the standard vanilla cross-entropy loss between the model predictions and the ground-truth labels, and our attribution-driven objective to train the compact model. The overall objective is defined as:

$$\mathcal{L} = \mathcal{L}_{ce} + \beta \mathcal{L}_a, \quad (8)$$

where $\mathcal{L}_{ce} = -\frac{1}{N} \sum_{i=1}^N y_i \log p_{\theta_s}(y_i | \mathbf{x}_i)$ is the cross-entropy loss function, N is the number of training samples and β is the hyperparameter that balances attribution guidance against the standard classification objective.

5 EXPERIMENTS

5.1 DATASETS AND BASELINES

We evaluate on three intent classification datasets: HWU64 (Liu et al., 2019a), Banking77 (Casanueva et al., 2020) and Clinc150 (Larson et al., 2019). For the baseline, we select two methods

Methods	HWU64				Banking77				Clinc150			
	Acc	Acc _{AD}	Com↑	Suf↓	Acc	Acc _{AD}	Com↑	Suf↓	Acc	Acc _{AD}	Com↑	Suf↓
<i>Baseline</i>												
Roberta-Base	75.56	62.36	65.52	34.97	74.04	53.82	67.06	38.30	87.99	71.70	65.38	30.91
<i>Hybrid size = 1</i>												
IG	77.04	65.80	68.93	32.67	79.09	56.57	68.05	36.79	89.58	72.29	67.09	26.36
LIME	77.13	66.96	68.79	31.53	78.50	56.96	73.21	33.19	89.24	<u>74.49</u>	<u>71.23</u>	26.68
CAP	76.57	66.31	68.35	29.76	78.83	57.24	72.95	30.98	89.20	73.39	<u>65.69</u>	26.35
<i>Hybrid size = 2</i>												
IG+LIME	77.88	66.40	66.86	<u>25.49</u>	78.57	57.12	72.33	31.83	<u>89.64</u>	73.38	67.50	26.97
CAP _{IG}	<u>78.16</u>	66.91	66.56	31.16	<u>78.89</u>	57.17	69.42	37.02	89.58	74.16	69.95	27.76
CAP _{LIME}	77.50	<u>67.38</u>	<u>69.12</u>	24.55	78.31	<u>57.32</u>	<u>73.87</u>	31.70	89.00	74.19	73.66	24.34
<i>Hybrid size = 3</i>												
CAP _{Hybrid}	79.09	70.63	69.17	30.66	78.63	58.05	73.97	<u>31.57</u>	89.69	74.70	70.59	<u>25.53</u>

Table 1: Results on three datasets in the 5-shot setting. The *Hybrid Size* here represents the supervision attribution composed of several attributions combined together, e.g. CAP_{LIME} means that we merged attribution from CAP and LIME. The **best** and second-best results in each column are shown in **bold** and underlined.

with different principles commonly used in explanation guided learning for comparison: Integrated Gradients (IG) (Sundararajan et al., 2017) and LIME (Ribeiro et al., 2016). Further details are presented in Appendix A.2.

5.2 ADVERSARIAL DATASET CONSTRUCTION

To assess the robustness of classifiers trained on the specific datasets, we automatically generate adversarial test sets using two complementary attack strategies: keyword addition and keyword replacement. *Keyword Addition*: Inject one or more keywords from an adversarial class into each original test example. The true label remains unchanged, so a robust model should still predict the original class. *Keyword Replacement*: Replace keywords from the original class with those from the adversarial class, aiming to preserve fluency while misleading the model toward the adversarial label. In the adversarial class selection, we first applied our method CAP to calculate the word-level attribution scores of training dataset for each class. These scores are subsequently aggregated to obtain a class-level keyword list. Pairwise difficulty is quantified by measuring the overlap of keyword distributions between classes. For every source class, we randomly sample n adversarial target classes; the value of n is determined by the total number of classes in the dataset (1 for HWU64, 2 for Banking77, and 3 for Clinc150). We construct this adversarial dataset by using GPT-4o, the generation prompt templates are provided in Appendix A.7.

5.3 IMPLEMENTATION DETAILS

For evaluation metrics, we report accuracy to assess overall performance for full-data, few-shot and adversarial settings. Additionally, following (DeYoung et al., 2020), we report comprehensiveness and sufficiency to quantify how well the model explanations reflect the decision process. Specifically, given a sample \mathbf{x}_i , the comprehensiveness measures the drop in confidence when the rationale \mathbf{r}_i is removed.

$$\text{Com} = \frac{p_{\theta_s}(y_i|\mathbf{x}_i) - p_{\theta_s}(y_i|\mathbf{x}_i \setminus \mathbf{r}_i)}{p_{\theta_s}(y_i|\mathbf{x}_i)}, \quad (9)$$

where y_i denotes the predicted label of \mathbf{x}_i , and $p_{\theta_s}(y_i|\mathbf{x}_i)$ is the probability that model θ_s assigns to y_i given input \mathbf{x}_i . A higher comprehensiveness score indicates that \mathbf{r}_i is critical to the model decision, whereas a lower score implies that the rationale has little influence. Conversely, sufficiency quantifies how well the extracted rationale alone supports the prediction.

$$\text{Suf} = \frac{p_{\theta_s}(y_i|\mathbf{x}_i) - p_{\theta_s}(y_i|\mathbf{r}_i)}{p_{\theta_s}(y_i|\mathbf{x}_i)}. \quad (10)$$

Method	Accuracy \uparrow		Com \uparrow	Suf \downarrow
	Test	Adversarial		
Base	92.63	70.29	56.73	35.87
IG	93.02	77.91	61.75	32.81
LIME	92.82	75.85	62.87	32.23
CAP	<u>93.41</u>	<u>78.20</u>	<u>68.00</u>	<u>24.30</u>
CAP_{Hybrid}	93.51	84.97	68.21	23.98

Table 2: Performance on Banking77 under the full-train setting. The **best** and second-best results in each column are shown in **bold** and underlined. *Com* and *Suf* stand for the comprehensiveness and sufficiency calculated on test set respectively.

A smaller sufficiency score, that is a minimal drop in confidence when using only r_i , indicates that the rationale by itself is sufficient for predicting y_i .

For parameter settings, we use RoBERTa-Base as the standard small language model for both training and extracting Integrated Gradients (IG) attributions. For extracting attribution scores using LIME and our proposed CAP method, we consistently adopt Llama-3-8B-Instruct as the large language model. Hyperparameters for training and the selection of aggregation can be found in the Appendix A.5 and A.6.

5.4 RESULT ANALYSIS

To answer **Q3**, we conduct comprehensive experiments and report the evaluation results in both *few-shot* and *full-data* training scenarios.

5.4.1 FEW-SHOT RESULTS

To evaluate whether our hybrid attribution prior can facilitate more reliable reasoning under low-resource conditions, we conduct experiments on 5-shot subsets of all three datasets. Table 1 reports the detailed results that on all of three datasets, single-prior (IG, LIME, and CAP) outperform the RoBERTa-Base baseline across all evaluation metrics. Each method exhibits unique strengths: IG consistently improves classification accuracy (e.g., 79.09% on Banking77), suggesting it helps the model focus on decision-critical tokens. LIME, by contrast, yields the highest comprehensiveness in most cases (e.g., 71.23% on Clinc150), indicating that its token-level perturbation captures prior more effectively. CAP achieves the best sufficiency score (29.76% on HWU64), showing that it guides the model to make predictions based on compact and essential evidence. This also illustrates our point of view that these attribution methods show different preferences in not only attribution mechanisms but also in the supervision of downstream tasks.

In particular, combining these priors leads to a synergistic effect. Hybrid variants generally outperform their individual counterparts, suggesting the complementarity of different attribution strategies. Among them, CAP_{+LIME} achieves the lowest sufficiency and high comprehensiveness, which validates the effectiveness of our proposed CAP. Our final method, CAP_{Hybrid} , which integrates CAP, IG, and LIME priors, achieves the highest overall accuracy (79.09%) in HWU64 dataset. Simultaneously, it maintains strong interpretability, with a comprehensiveness score of 69.12% and a sufficiency score of 30.66%, both ranking second among all methods. This suggests that CAP_{Hybrid} not only improves prediction accuracy but also promotes more faithful and stable reasoning processes. In the in-domain adversarial setting, the RoBERTa-Base baseline exhibits a substantial performance drop compared to the original test set, highlighting the vulnerability of models under standard training. In contrast, CAP_{Hybrid} significantly improves adversarial accuracy across all three datasets, with notable gains on HWU64 (8%) and Banking77 (4%). These results demonstrate that our method enhances robustness of models in more challenging settings.

5.4.2 FULL TRAINING RESULTS

Table 2 reports the results on Banking77 under the full-data training setting. As shown, our method continues to demonstrate strong performance. CAP_{Hybrid} achieves the highest test accuracy

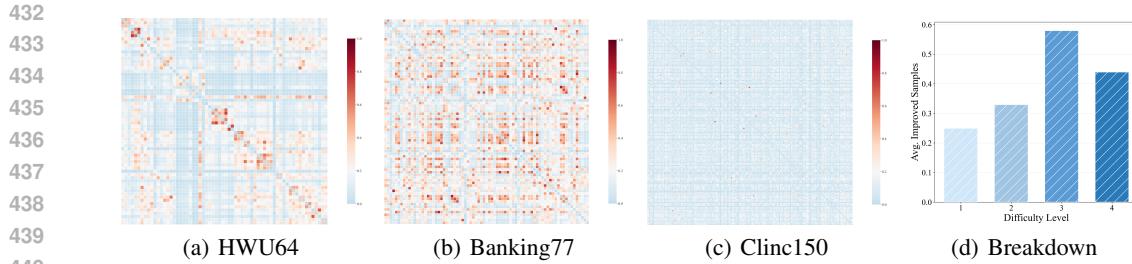


Figure 4: Subfigure (a)-(c) show the class similarity of HWU64, Banking77 and Clinc150 datasets. Darker colors denote higher similarities. Subfigure (d) shows the improvements performance breakdown across difficulty levels in 5-shot Banking77 dataset.

93.51%(vs 92.63%) and adversarial accuracy 84.97%(vs 70.29%), along with the best comprehensiveness 68.21%(vs 56.73%) and sufficiency 23.98%(vs 23.98%). These results confirm that our attribution-guided supervision remains effective even when ample training data is available.

5.5 QUALITATIVE ANALYSIS OF DATASET PROPERTIES AND METHOD EFFECTIVENESS

To further investigate the factors affecting the effectiveness of our method and to better understand the conditions under which attribution priors offer the greatest benefit, we conduct qualitative analyses centered on two key observations. First, Figure 4(d) illustrates the relationship between the number of corrected cases achieved by our attribution-guided training and the lexical overlap between the ground-truth labels and the model’s original incorrect predictions—a proxy for instance difficulty. The results reveal that our method is particularly effective in handling challenging examples (difficulty levels 3 and 4), where the confusing classes share a high degree of token-level similarity. This highlights the model’s enhanced ability to tease apart subtle semantic distinctions that are often lost under standard training.

Second, we analyze the overall structure of each dataset by computing the mean sentence embedding for each class and measuring pairwise class similarities. As shown in the similarity heatmaps Figure 4(a)-(c), Banking77 and HWU64 contain clusters of highly similar classes, indicating the presence of *inherent difficulty imbalance*. In contrast, Clinc150 exhibits more uniformly distributed and well-separated class representations, with less inter-class confusion. This distinction clarifies our empirical findings: on HWU64 and Banking77, baseline models frequently misclassify these fine-grained categories due to spurious token co-occurrence patterns. Our approach, by aligning predictions with richer and more discriminative attribution priors, mitigates this issue and enables the model to focus on features that are more indicative of class-level meaning rather than surface lexical overlap. In contrast, for Clinc150, even simple priors derived from LIME suffice to provide meaningful guidance, thereby limiting the room for further gains. Consequently, our method achieves marginal improvements in that setting. These findings suggest that our attribution-based supervision not only improves robustness to lexical ambiguity but also enhances the model’s capacity for deeper semantic reasoning, particularly in challenging scenarios involving closely related or fine-grained classes.

6 CONCLUSION

We identify a critical limitation in existing attribution guided learning: when categories exhibit high semantic overlap, models trained with surface-level priors often fail to distinguish fine-grained classes. To address this, we propose Class-Aware Attribution Priors (CAP), which inject task-specific and label-informed semantic constraints into model training, encouraging more meaningful feature attribution. We further introduce CAP_{Hybrid} , a fusion approach that integrates multiple complementary attribution sources to form richer, more balanced priors. Our method consistently enhances performance, interpretability, and robustness across various datasets, offering a promising direction toward building more transparent and reliable models.

486 REFERENCES
487

488 Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller,
489 and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise
490 relevance propagation. *PloS one*, 10(7), 2015.

491 Moritz Böhle, Mario Fritz, and Bernt Schiele. B-cos networks: Alignment is all we need for inter-
492 pretability. In *CVPR*, pp. 10319–10328, 2022.

493

494 Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient
495 intent detection with dual sentence encoders. In *The 2nd Workshop on Natural Language Pro-*
496 *cessing for Conversational AI*, pp. 38–45, 2020.

497 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep
498 bidirectional transformers for language understanding. In *NAACL-HLT*, pp. 4171–4186, 2019.

499

500 Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher,
501 and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *ACL*,
502 pp. 4443–4458, 2020.

503 Yuyang Gao, Tong Steven Sun, Guangji Bai, Siyi Gu, Sungsoo Ray Hong, and Liang Zhao. RES: A
504 robust framework for guiding visual explanation. In *KDD*, pp. 432–442, 2022a.

505

506 Yuyang Gao, Tong Steven Sun, Liang Zhao, and Sungsoo Ray Hong. Aligning eyes between humans
507 and deep neural network through interactive attention alignment. *Proc. ACM Hum. Comput.*
508 *Interact.*, 6(CSCW2):1–28, 2022b.

509 Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song.
510 Pretrained transformers improve out-of-distribution robustness. In *ACL*, pp. 2744–2751, 2020.

511

512 Sahil Jayaram and Emily Allaway. Human rationales as attribution priors for explainable stance
513 detection. In *EMNLP*, pp. 5540–5554, 2021.

514

515 Amlan Jyoti, Karthik Balaji Ganesh, Manoj Gayala, Nandita Lakshmi Tunuguntla, Sandesh Kamath,
516 and Vineeth N. Balasubramanian. On the robustness of explanations of deep neural network
517 models: A survey. *CoRR*, abs/2211.04780, 2022.

518 Davinder Kaur, Suleyman Uslu, Kaley J. Rittichier, and Arjan Durresi. Trustworthy artificial intel-
519 ligence: A review. *ACM Comput. Surv.*, 55(2):39:1–39:38, 2023.

520

521 Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill,
522 Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars.
523 An evaluation dataset for intent classification and out-of-scope prediction. In *EMNLP*, pp. 1311–
524 1316, 2019.

525

526 Dongfang Li, Baotian Hu, Qingcai Chen, Tujie Xu, Jingcong Tao, and Yunan Zhang. Unifying
527 model explainability and robustness for joint text classification and rationale extraction. In *AAAI*,
528 pp. 10947–10955, 2022a.

529

530 Dongfang Li, Baotian Hu, Qingcai Chen, and Shan He. Towards faithful explanations for text clas-
531 sification with robustness improvement and explanation guided training. *CoRR*, abs/2312.17591,
532 2023.

533

534 Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S. Yu, and Lifang
535 He. A survey on text classification: From traditional to deep learning. *ACM Trans. Intell. Syst.*
536 *Technol.*, 13(2):31:1–31:41, 2022b.

537

538 Frederick Liu and Besim Avci. Incorporating priors with feature attribution on text classification. In
539 *ACL*, pp. 6274–6283, 2019.

540

541 Xingkun Liu, Arash Eshghi, Paweł Swietojanski, and Verena Rieser. Benchmarking natural lan-
542 guage understanding services for building conversational agents. In *IWSDS*, volume 714, pp.
543 165–183, 2019a.

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

543

544 Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *NeurIPS*,
 545 pp. 4765–4774, 2017.

546

547 Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. Towards faithful model explanation in
 548 NLP: A survey. *Comput. Linguistics*, 50(2):657–723, 2024.

549

550 Amin Parchami-Araghi, Moritz Böhle, Sukrut Rao, and Bernt Schiele. Good teachers explain:
 551 Explanation-enhanced knowledge distillation. In *ECCV*, pp. 293–310, 2024.

552

553 William H Press. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge uni-
 554 versity press, 2007.

555

556 Sukrut Rao, Moritz Böhle, Amin Parchami-Araghi, and Bernt Schiele. Studying how to efficiently
 557 and effectively guide models with explanations. In *ICCV*, pp. 1922–1933, 2023.

558

559 Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining
 560 the predictions of any classifier. In *KDD*, pp. 1135–1144, 2016.

561

562 Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons:
 563 Training differentiable models by constraining their explanations. In *IJCAI*, pp. 2662–2670, 2017.

564

565 Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh,
 566 and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based local-
 567 ization. In *ICCV*, pp. 618–626, 2017.

568

569 Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In
 570 *ICML*, pp. 3319–3328, 2017.

571

572 Kamal Taha, Paul D. Yoo, Chan Yeob Yeun, Dirar Homouz, and Aya Taha. A comprehensive survey
 573 of text classification techniques and their research applications: Observational and experimental
 574 insights. *Comput. Sci. Rev.*, 54:100664, 2024.

575

576 Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (XAI): toward medical
 577 XAI. *IEEE Trans. Neural Networks Learn. Syst.*, 32(11):4793–4813, 2021.

578

579 Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman.
 580 GLUE: A multi-task benchmark and analysis platform for natural language understanding. In
 581 *ICLR*, 2019.

582

583 Zifan Wang, Haofan Wang, Shakul Ramkumar, Piotr Mardziel, Matt Fredrikson, and Anupam Datta.
 584 Smoothed geometry for robust attribution. In *NeurIPS*, 2020.

585

586 Mayur Wankhade, Annavarapu Chandra Sekhara Rao, and Chaitanya Kulkarni. A survey on senti-
 587 ment analysis methods, applications, and challenges. *Artif. Intell. Rev.*, 55(7):5731–5780, 2022.

588

589 Siyue Wu, Hongzhan Chen, Xiaojun Quan, Qifan Wang, and Rui Wang. AD-KD: attribution-driven
 590 knowledge distillation for language model compression. In *ACL*, pp. 8449–8465, 2023.

591

592 Chaoning Zhang, Philipp Benz, Chenguo Lin, Adil Karjauv, Jing Wu, and In So Kweon. A survey
 593 on universal adversarial attack. In *IJCAI*, pp. 4687–4694, 2021.

594

595 Feng Zhang, Wei Chen, Fei Ding, and Tengjiao Wang. Dual class knowledge propagation network
 596 for multi-label few-shot intent detection. In *ACL*, pp. 8605–8618, 2023.

594 **A APPENDIX**
595596 **A.1 THE USE OF LARGE LANGUAGE MODELS**
597598 During the preparation of this paper, we used large language models (LLMs) solely as general-
599 purpose writing assistants. Specifically, LLMs were employed to help refine the clarity, grammar,
600 and readability of our drafts, as well as to suggest alternative phrasings in English. Importantly,
601 all conceptual contributions including the design of research questions, development of methods,
602 execution of experiments, and interpretation of results were conceived and carried out entirely by the
603 authors. The authors carefully reviewed and edited all text suggested by LLMs to ensure accuracy
604 and originality, and we take full responsibility for the final content of the paper.
605606 **A.2 DATASETS AND BASELINES**
607608 We evaluate on three multi-domain intent classification datasets, each covering diverse real-world
609 scenarios: (1) HWU64 (Liu et al., 2019a) contains 64 intents drawn from 18 everyday domains (e.g.,
610 alarm, cooking, transportation), with utterances balanced across intents. (2) Banking77 (Casanueva
611 et al., 2020) comprises 13,083 labeled customer utterances spanning 77 fine-grained intents in the
612 banking domain. (3) Clinc150 (Larson et al., 2019) offers 22,500 user utterances evenly distributed
613 over 150 intents in 10 general domains, facilitating robust cross-domain evaluation.
614615 For the baseline, we select two methods with different principles commonly used in explanation
616 guided learning for comparison: Integrated Gradients (IG) (Sundararajan et al., 2017) We first train
617 a model using only the classification loss and then compute IG attribution scores on the training
618 set using this model. These attribution scores are subsequently incorporated as supervision signals
619 to train a new model from scratch. LIME (Ribeiro et al., 2016) We employ LIME to compute
620 attribution scores on the training dataset from large language model, and use these scores as external
621 supervision to guide the training of the smaller model.
622623 **A.3 PROOF OF EQ. (3)**
624625 This section describes the proof of Eq. (3). For clarity, we first restate our objective, i.e. Eq. (1):
626

627
$$\alpha = \operatorname{argmin} \left\{ \sum_{i=1}^n (z_i - \alpha^\top m_i)^2 + \lambda \|\alpha\|^2 \right\}, \quad (11)$$

628

629 where $\alpha \in \mathbb{R}^d$ is the attribution parameter vector to be estimated, each $m_i \in \mathbb{R}^d$ is a mask vector
630 with associated score z_i , and $\lambda > 0$ is the regularization coefficient. To find the minimizer, we
631 calculate its derivative. Specifically, let $\mathcal{L}(\alpha)$ be the function,
632

633
$$\mathcal{L}(\alpha) = \sum_{i=1}^n (z_i - \alpha^\top m_i)^2 + \lambda \alpha^\top \alpha. \quad (12)$$

634

635 The derivative of $\mathcal{L}(\alpha)$ with respect to α is as follows,
636

637
$$\nabla_\alpha \mathcal{L} = -2 \sum_{i=1}^n (z_i - \alpha^\top m_i) m_i + 2\lambda \alpha. \quad (13)$$

638

639 To find the minimizer, we set $\nabla_\alpha \mathcal{L} = 0$, yielding
640

641
$$\sum_{i=1}^n (\alpha^\top m_i) m_i + \lambda \alpha = \sum_{i=1}^n z_i m_i. \quad (14)$$

642

643 Next we transform the left-hand side of Eq. (14). For $(\alpha^\top m_i) m_i$ item, let $t = \alpha^\top m_i$, which is a
644 scalar. Then $t m_i = m_i t$ and $\alpha^\top m_i = m_i^\top \alpha$. We have
645

646
$$\begin{aligned} (\alpha^\top m_i) m_i &= m_i t \\ &= m_i (m_i^\top \alpha) \\ &= (m_i m_i^\top) \alpha. \end{aligned}$$

647

648 Substituting $(\alpha^\top \mathbf{m}_i) \mathbf{m}_i = (\mathbf{m}_i \mathbf{m}_i^\top) \alpha$ into Eq. (14) gives
 649

$$650 \quad 651 \quad 652 \quad \sum_{i=1}^n (\mathbf{m}_i \mathbf{m}_i^\top) \alpha + \lambda \alpha = \sum_{i=1}^n z_i \mathbf{m}_i, \quad (15)$$

653 which is exactly Eq. (3).
 654

655 A.4 PROOF OF POSITIVE DEFINITE MATRIX A

656 We now show that
 657

$$658 \quad A = M^\top M + \lambda I, \quad (16)$$

659 is both symmetric and positive definite, where I is the identity matrix and $\lambda > 0$.
 660

661 Symmetry.
 662

$$663 \quad A^\top = (M^\top M + \lambda I)^\top \\ 664 \quad = (M^\top M)^\top + (\lambda I)^\top \\ 665 \quad = M^\top M + \lambda I^\top \\ 666 \quad = A.$$

667 Positive Definiteness. For any nonzero real column vector \mathbf{x} , we have
 668

$$669 \quad x^\top A x = x^\top (M^\top M + \lambda I) x \\ 670 \quad = x^\top (M^\top M) x + x^\top (\lambda I) x \\ 671 \quad = (x^\top M^\top) (M x) + \lambda x^\top x \\ 672 \quad = \|M x\|^2 + \lambda \|x\|^2.$$

673 Since $\|M x\|^2 \geq 0$ for all \mathbf{x} and $\lambda \|x\|^2 > 0$ whenever $\mathbf{x} \neq \mathbf{0}$, it follows that
 674

$$675 \quad x^\top A x > 0, \quad \forall \mathbf{x} \neq \mathbf{0}, \quad (17)$$

676 thus A is positive definite.
 677

678 A.5 TRAINING PARAMETER SETTING

679 When calculating our class-aware attribution priors, we set the $\lambda = 0.1$ and $n = 100$ in Eq. (1).
 680 And when training the Roberta-Base model, we set the learning rate to 5e-5, batch size to 16, and
 681 use early stopping with a patience of 10. Additionally, we apply gradient norm clipping with a
 682 maximum norm of 1.0. For our attribution-aligned training, we experiment with attribution loss
 683 weighting coefficient $\beta = 1$. While both *mean* and *max* aggregation strategies were considered for
 684 hybrid prior methods, we report only the test results corresponding to the combination that achieved
 685 the best validation performance. The comparison experiment can be found in Appendix A.6.
 686

687 A.6 THE COMPARISON OF DIFFERENT AGGREGATION OPERATION

688 We compare different aggregation strategies in terms of their impact on classification accuracy. As
 689 shown in Table 3, the mean aggregation achieves the best predictive performance in most cases.
 690

691 A.7 ADVERSARIAL DATASET GENERATION PROMPT

692 In this section, we present all the prompts utilized for constructing the adversarial datasets.
 693

694
 695
 696
 697
 698
 699
 700
 701

702
703
704
705
706
707
708
709
710

Dataset	Banking77		Clinc150		HWU64	
	Aggregation	Mean	Max	Mean	Max	Mean
IG+LIME	78.57	76.91	89.64	89.44	77.88	77.53
CAP+IG	78.81	78.89	89.58	89.31	78.16	78.04
CAP+LIME	77.89	78.31	89.00	88.71	77.50	77.29
CAP _{Hybrid}	78.63	78.47	89.69	88.98	77.41	79.09

711 Table 3: The comparision of accuracy for different aggregation methods on the 5-shot setting.
712
713
714
715
716

717 Task: Generate a keyword addition adversarial example for the following
718 text by adding keywords from a adversarial class without changing the
719 classification result.
720 Original text: {original text}
721 Original class: {original label}
722 Adversarial class: {adversarial label}
723 Adversarial class keywords: {adversarial keywords}
724 Adversarial class example sentences: {adversarial sentence}
725 Requirements:
726 1. Preserve the original meaning as much as possible
727 2. Add 1{2 keywords from the source class, but avoid making the intent
overly explicit
728 3. Ensure the new text is still classified as the target class
729 4. Ensure the text remains natural and fluent
730 5. Only return the modified text, nothing else
Adversarial example:

731 Table 4: The template of the prompt we used for generate *Keyword Addition* adversarial examples.
732
733
734
735
736

737 Task: Generate a keyword replacement adversarial example by replacing
738 original class keywords in the following text with keywords from a
739 adversarial class.
740 Original text: {original text}
741 Original class: {original label}
742 Adversarial class: {adversarial label}
743 Adversarial class keywords: {adversarial keywords}
744 Adversarial class example sentences: {adversarial sentence}
745 Requirements:
746 1. Replace target keywords with suitable original-class keywords, and
optionally add new ones
747 2. Preserve the original sentence structure (e.g., questions,
conjunctions)
748 3. Ensure the new text remains semantically coherent and realistic
749 4. Ensure the modified text is classified into the new target class
750 5. Only return the modified text, nothing else
Adversarial example:

752 Table 5: The template of the prompt we used for generate *Keyword Replacement* adversarial exam-
753 ples.
754

755