LLM-induced Rationales for More Compact Explainable Style Classification Models

Ahmad Aljanaideh
Bentley University
Waltham, MA, USA
aaljanaideh@bentley.edu

Saeb Ganideh
Toronto Metropolitan University
Toronto, ON, Canada
ganideh@torontomu.ca

Abstract

The complexity of recent natural language classification models led to interest in developing methods for improving the performance of explainable models (e.g. Logistic Regression). Existing methods focus on clustering word embeddings to discover fine-grained contextual features that can be used to train a linear model. While those methods help reduce the gap in performance between black-box models and explainable models, they are based on discovering a large number of features, and this affects interpretability. In this work, we propose a model that leverages Large Language Models (LLMs) and clustering algorithms to discover a compact set of interpretable features. The proposed model first uses GPT-4o mini to extract rationales (i.e. phrases which explain an item's label) from labeled text, and then clusters those rationales to obtain a compact, interpretable feature space. Across 3 Style Classification tasks, the resulting features achieve comparable performance to word-cluster baselines on most tasks, while reducing the number of features by 85–99%. These results highlight the potential of LLMs to improve the compactness of explainable AI models.

1 Introduction

As AI models become more complex, explaining their decision-making mechanisms becomes more difficult. This has led to the rise of Explainable AI (XAI), where the goal is to build models which can explain their decisions. XAI is crucial for fair and transparent adoption of AI (i.a. Meske et al., 2022; Abdul et al., 2018; Fernandez et al., 2019; Miller, 2019).

While most XAI methods focus on extracting explanations from complex models (i.a. Ribeiro et al., 2016; Lundberg and Lee, 2017; Simonyan et al., 2013; Bahdanau et al., 2014), an emerging line of research focuses on improving the performance of inherently explainable classifiers (e.g. Logistic Regression) via automatic discovery of interpretable features (Aljanaideh et al., 2020). This model focuses on clustering contextualized embeddings of each word in a vocabulary to obtain fine-grained context patterns of words which can be used as features to train a linear model. While this model helps in reducing the gap between linear models and more complex ones such as a fine-tuned BERT (Devlin et al., 2018), it leads to an extremely high number of features since embeddings of each word in a corpus have to be clustered to obtain multiple features for each word in the vocabulary, and this affects interpretablity. Moreover, this model does not leverage LLMs in the feature discovery process. More recently, Aljanaideh (2025) introduced an LLM-based approach that improves generalizability of explainable politeness classification by clustering theory-driven speech-act categories (e.g., gratitude, greeting). However, this approach assumes the availability of a predefined speech-act categories for the task, which may not be available or well-defined for many classification tasks.

In this work, we introduce a model for automatically discovering a compact set of features from labeled data using rationales extracted via LLMs. We define rationales as text spans which explain the label a text item received. The model first extracts rationale phrases from labeled data items using GPT-4o mini, and then clusters those rationales using the k-means algorithm. Features based on the rationale clusters are extracted and used to train an explainable classifier for the target task. By operating over rationale phrases rather than individual words, our method learns features that are both linguistically meaningful and significantly fewer in number.

We evaluate our approach on three Style Classification tasks, Subjectivity Detection, Shakespearean Style Identification, and Sentiment Analysis. We focus on those tasks since they differ substantially in the nature of rationales needed for evaluation.

Findings of the Association for Computational Linguistics: ACL 2026, pages 28571–28577

Results show the model produces a compact set of features that is competitive with word-cluster baselines in performance, while dramatically reducing the number of features (85-99% reduction). Our findings suggest that LLMs can help improve compactness and interpretability of explainable AI models.

2 Proposed Model

We propose a model for extracting a compact set of features that an explainable classifier (e.g. Logistic Regression) can be trained on. Aljanaideh et al. (2020)'s model clusters contextualized embeddings of each word in a training set to obtain fine-grained context patterns of words which can be used as features. In our case, we cluster rationale phrases which represent text spans that provide direct evidence for a label. Relying on rationale phrases instead of words offers multiple advantages. Unlike Aljanaideh et al. (2020)'s approach, semantically similar text portions that do not share a word in common (e.g. *good job!* and *great work.*) can be mapped to the same feature. Second, the number of learned features is reduced dramatically, since only one clustering is needed, whereas Aljanaideh et al. (2020) training N clustering models are needed where N is the size of the vocabulary. Specifically, our model consists of three steps: rationale phrase extraction, rationale clustering, and feature extraction. The features can then be evaluated for a downstream task. We describe those steps in detail below.

2.1 Rationale Phrase Extraction

First we extract rationales from labeled training text items by prompting the GPT-4o mini model. Table 1 shows the prompt we used. The goal is to obtain cues to indicate why an item received a certain label. For example, if the item is *I watched the movie last night. It was great.* and received the label *positive*, the model is expected to extract *It was great* as the potential rationale.

2.2 Rationale Clustering

We cluster rationale phrases obtained in the previous step to obtain clusters of semantically similar rationale phrases. First, an embedding for each rationale phrase is obtained by averaging the pretrained BERT (Devlin et al., 2018) embeddings of its words. After that, those embeddings are clustered using the k-means algorithm. We select the k based on development-set classification accuracy.

You are given a text item and its gold label for a classification task. Your task is to identify the specific phrases in the text that most strongly support why the text item received this label. • A phrase may consist of one or more consecutive words. • List each phrase on a separate line. • Do not include explanations or any other output. **Task:** Classifying a text according to *{task_description}* **Possible labels:** *{labels}* **Assigned label:** *{label}* **Text:** *{text}*

Table 1: Prompt for extracting rationales from text items.

Task (Ref)	#K (Train / Dev / Test)
Subjectivity (Pang and Lee, 2004)	9.9 (6.0/1.2/2.7)
Shakespeare (Xu et al., 2012)	11.0 (6.0/2.2/2.7)
Sentiment (Socher et al., 2013)	12.1 (9.0/1.0/2.1)

Table 2: Style Classification datasets used in this work. The # column reports total dataset size, followed by train/dev/test splits.

2.3 Feature Extraction & Classification

The last step is generating features based on the discovered clusters. We follow a similar approach to Aljanaideh et al. (2020). For a training item, its features are the cluster ids of its rationale phrases. However, since our clusters correspond to phrases rather than individual words, we adopt a different strategy at inference time. For unseen items, we first compute contextualized BERT embeddings for each token, then form bigram and trigram representations by averaging embeddings of consecutive tokens. Each resulting n-gram embedding is assigned to its nearest rationale cluster, allowing the model to match phrases of varying lengths at test time without requiring explicit rationale extraction at inference. The resulting cluster ids of each item represent its features. Those features can be used to train a linear classifier.

3 Dataset

We evaluate our approach on three Style Classification tasks: Subjectivity Classification (objective vs. subjective), Shakespearean vs. modern English Classification, and Sentiment Classification (positive, negative, and neutral). Table 2 summarizes each dataset including its source and number of instances. We use the same train/development/test splits as Guo et al. (2023). To reduce the computa-

Task	UNI	WC	RC (ours)
Subjectivity	3282	4909	190
Shakespeare	1208	2494	150
Sentiment	2065	3552	515

Table 3: Number of features for each feature set. **UNI** indicates unigrams, **WC** indicates Word Clusters (Aljanaideh et al., 2020), RC indicates rationale clusters (this work). Features that appear less than 5 times in the training set are removed.

tional cost of LLM-based prompting, we subsample the training data to at most 3K instances per label for each task.[1]

4 Results

We apply the proposed model on the datasets described in the previous section. In this section, we perform cluster analysis on the discovered features. We also evaluate the perfornances of the features for each task.

4.1 Rationale Evaluation

First, we evaluate the rationales obtained via GPT-4o mini. We perform manual evaluation for a random sample of 100 items for each dataset, and provide a score out of 5 which reflects the quality of the rationale. The average scores for Subjectivity, Shakespeare and Sentiment are 3.6, 3.9 and 3.8, respectively[2].

4.2 Cluster Analysis

We leverage GPT-4o mini to perform analysis over the rationale clusters obtained with our proposed model[3]. Specifically, we provide examples from a cluster of phrase rationales, and ask GPT-4o mini to (1) provide a name for the pattern exhibited among the examples, (2) describe the pattern in a few sentences and (3) provide three example phrases which represent the pattern. Table 5 shows examples of cluster analysis for each task. For subjectivity, examples of objective language include explicit mentions of characters without any evaluative language. The model assigns phrases such as *francis doyle and tim sullivan* and *sergeant peter king* to the same cluster despite not having any words in common. In Aljanaideh et al. (2020) those two phrases cannot be in the same cluster because they do not share any lexical overlap. Examples of subjective patterns include subjective evaluation phrases such as *that's kinda what you'll get here* and *it isn't that funny*. Examples of Shakesperean patterns include archaic syntax and vocabulary such as *ere noon* and *till our coronation*. Modern language exhibits conventional structures (e.g. *I said anything*). For Sentiment, examples of positive language include humor conveyed in a positive way (e.g. *Sorvino glides gracefully*, *truly funny*), Negative language includes negative sentiment indicators (e.g. *shortcoming*, *lack of*), and Neutral include descriptive noun indicators *film entertainment* and *clash comedies*.

4.3 Classification Results

We next show the results of training a Logistic Regression classifier using features obtained with our proposed models, and compare those results to existing models. Table 3 shows the number of features for each model, and Table 4 shows the classification accuracy for each task across different combinations of feature sets. Rationale clusters (our work) provide comparable performance with Word Clusters (Aljanaideh et al., 2020), while using 85–99% fewer features. Combining Rationale Clusters with Unigrams and Word Clusters gives the best classification among explainable models for two tasks (Sentiment and Shakespeare). A fine-tuned GPT-4o mini provides the best classification performance for all tasks. However, this model is not linear and thus its decision making process is not inherently explainable. Figure 1 shows the development accuracy for different numbers of clusters (features) across the three tasks. We observed that initially, accuracy rises sharply as the number of features increases, and then stabilizes with additional features resulting only marginal gains.

5 Conclusions

In this work, we introduced a model which learns a compact set of features for improving the interpretability of explainable classification models. The model uses GPT-4o mini to extract rationale phrases from labeled text items, and then clusters those phrases to learn the features. Evaluations on three classification tasks showed the proposed model achieved comparable performances with ex-

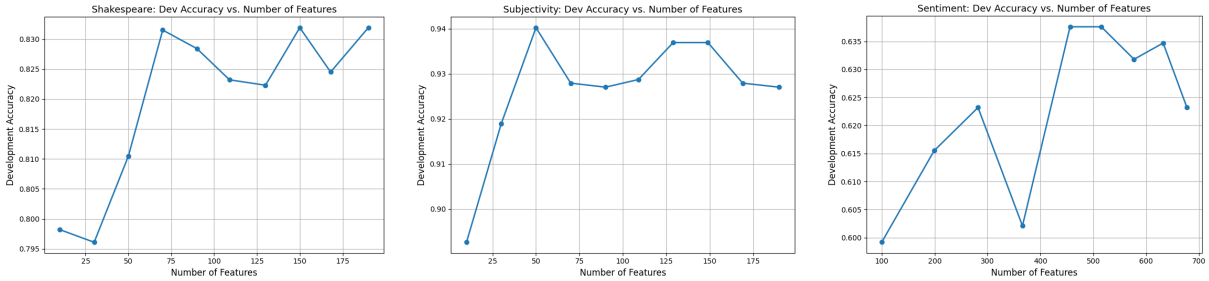
[1]We follow the original dataset releases and refer readers to the respective dataset papers for details on annotation procedures and annotator demographics (Pang and Lee, 2004; Xu et al., 2012; Socher et al., 2013).

[2]More details about the evaluation are shown in the Appendix A.1.

[3]Full prompt shown in Appendix A.3

Task	UNI	WC	UNI+WC	RC (ours)	UNI+RC	WC+RC	UNI+WC+RC	GPT
Subjectivity	89.7	94.7	93.9	92.7	93.6	93.9	94.3	**98.1**
Shakespeare	78.3	84.4	84.7	83.4	84.5	85.2	85.5	**93.4**
Sentiment	63.2	65.7	67.3	65.8	66.5	67.7	68.3	**83.1**

Table 4: Final test accuracy (%) by task and feature set. **UNI** indicates unigrams, **WC** indicates Word Clusters (Aljanaideh et al., 2020), RC indicates rationale clusters (this work). Bolded values indicate the best-performing configuration for each task. Underlined values indicate the best accuracy among explainable models. GPT indicates a fine-tuned GPT-4o mini (full prompt is shown Appendix A.4.)



(a) Shakespeare (b) Subjectivity (c) Sentiment

Figure 1: Development accuracy vs. number of clusters for each task.

Task	Pattern name	Label dist.	Description	Example phrases
Subjectivity	Character Identification Pattern	obj: 100% / subj: 0%	Explicit identification of characters or named individuals in a narrative context, conveying factual information without evaluative language.	*francis doyle and tim sullivan* *sergeant peter king* *charlotte and cecil*
	Temporal and Contextual Phrasing	obj: 100% / subj: 0%	Temporal or situational phrases that establish when or where events occur while maintaining an objective tone.	*after a really wild night* *a few days later* *they spend their afternoons*
	Subjective Evaluation Phrases	obj: 20% / subj: 80%	Subjective judgments framed as opinions or assessments, often expressing approval or disappointment.	*that's kinda what you'll get here* *it isn't that funny* *crazy as hell doesn't even have a great ending*
Shakespeare	Archaic Syntax and Vocabulary	modern: 0% / shakespearean: 100%	Archaic vocabulary and syntactic constructions characteristic of Shakespearean English.	*ere noon* *till our coronation* *till we shall meet again*
	Inverted Syntax and Archaic Vocabulary	modern: 0% / shakespearean: 100%	Inverted sentence structures and archaic lexical choices typical of Shakespearean dialogue.	*A tailor make a man?* *What horrible fancy's this?* *What is 't you do?*
	Conversational Modernity	modern: 90% / shakespearean: 10%	Contemporary conversational phrasing with modern syntax and accessible emotional expression.	*sounded like bulls, or lions* *a mug of beer* *I said anything*
Sentiment	Graceful Transition and Humor	neg: 0% / neu: 0% / pos: 100%	Positive sentiment conveyed through humor, wit, or light-hearted evaluative language.	*Sorvino glides gracefully* *truly funny* *without missing a beat*
	Negative Sentiment Indicators	neg: 100% / neu: 0% / pos: 0%	Strong negative evaluative expressions highlighting shortcomings or dissatisfaction.	*shortcomings* *a lack of* *the weaknesses of*
	Descriptive Noun Phrases	neg: 0% / neu: 70% / pos: 30%	Predominantly neutral noun phrases describing genres or entities without strong sentiment.	*film entertainment* *clash comedies* *fantasy world*

Table 5: Cluster analysis obtained with GPT-4o mini.

isting methods, while significantly reducing the number of features. This work shows explainable AI models can benefit from recent developments in LLM. In the future we plan to use LLMs to guide the unsupervised clustering of rationales to obtain higher performing features.

Limitations

There are several limitations to our work. First, LLM outputs can vary across runs which may affect reproducibility. Second, the datasets used in this work do not include ground truth rationales for the labels. Therefore, it is challenging to evaluate the rationales extracted by the LLM. Third, the model is evaluated on three Style Classification tasks only. We acknowledge there can be more types of tasks for application, but we reserve exploring those tasks in future work. One potential risk to this work is that over-reliance on LLM-generated rationales might result in rationales that appear convincing but do not faithfully reflect the true labeling rationale.

References

Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y Lim, and Mohan Kankanhalli. 2018. Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pages 1–18.

Ahmad Aljanaideh. 2025. Speech act patterns for improving generalizability of explainable politeness detection models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 18945–18954.

Ahmad Aljanaideh, Eric Fosler-Lussier, and Marie-Catherine De Marneffe. 2020. Contextualized embeddings for enriching linguistic analyses on politeness. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2181–2190.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. 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*.

Alberto Fernandez, Francisco Herrera, Oscar Cordon, Maria Jose del Jesus, and Francesco Marcelloni. 2019. Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to? *IEEE Computational intelligence magazine*, 14(1):69–81.

Ruohao Guo, Wei Xu, and Alan Ritter. 2023. Meta-tuning llms to leverage lexical knowledge for generalizable language style understanding. *arXiv preprint arXiv:2305.14592*.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Christian Meske, Enrico Bunde, Johannes Schneider, and Martin Gersch. 2022. Explainable artificial intelligence: objectives, stakeholders, and future research opportunities. *Information systems management*, 39(1):53–63.

Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and 1 others. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Wei Xu, Alan Ritter, Bill Dolan, Ralph Grishman, and Colin Cherry. 2012. Paraphrasing for style. In *Proceedings of COLING 2012: Technical Papers*, pages 2899–2914.

A Appendix

A.1 Rationale Evaluation

We provide additional detail on the manual evaluation of rationales generated by GPT-4o mini. For a random sample of 100 rationales, we evaluate each rationale on a 5-point scale reflecting its quality. The quality is measured by the extent to which words in the extracted rationale reflect the true label of the item. Table 6 shows examples of text

items, rationales and assigned scores for each category across the three tasks. The most common error observed is that GPT-4o mini often extracts additional unnecessary phrases which do not correspond to the actual label (e.g. Rows#1, #3, #4 and #5). Other examples include over-segmenting the text item (e.g. Rows#5 and #6)

A.2 Experimental Details

For clustering, we use the scikit-learn (Pedregosa et al., 2011) implementation of *k*-means. For the Subjectivity and Shakespeare tasks, we evaluate values of k from 10 to 200 with increments of 10, while for the Sentiment task we evaluate k from 100 to 1000 in increments of 100. For classification, we use scikit-learn's Logistic Regression with an L_2 penalty and the `lbfgs` optimizer, tuning the regularization parameter $C \in \{0.01, 0.1, 1, 10\}$. Results were obtained with a single run due to the computational cost of LLMs prompting.

A.3 Prompt for cluster analysis

Table 7 shows the prompt we used to perform cluster analysis shown in Table 5.

A.4 Prompt for finetuning GPT-4o mini

Table 8 shows the prompt we used to finetune GPT-4o mini.

A.5 AI Assistant Use

AI assistants were used for code debugging and language editing.

Row#	Full text item	Label	Rationale received	Score	justification
1	shaky close-ups of turkey-on-rolls stubbly chins liver spots red noses and the filmmakers new bobbed do draw easy chuckles but lead nowhere.	Subjective	*shaky close-ups of turkey-on-rolls / stubbly chins / liver spots / red noses / the filmmakers new bobbed do / draw easy chuckles / lead nowhere*	3	Over-extraction, for example, *red noses* is not necessarily subjective
2	Absorbing and disturbing – perhaps more disturbing than originally intended – but a little clarity would have gone a long way .	Objective	*absorbing and disturbing / more disturbing than originally intended / a little clarity would have gone a long way*	5	Extracted phrases correspond to subjective language (e.g. *absorbing and disturbing*)
3	Not until we part as friends.	Modern	*Not until we part / as friends*	4	*as friends* is not necessarily modern.
4	Oh, strange!	Shakespearean	*Oh, strange!*	3	While the *oh* part generally corresponds Shakespearean, *strange!* is not necessarily so.
5	part of the action, the wallpaper of his chosen reality . Here, thankfully	Positive	*the action / thankfully*	2	*action* does not correspond to positive sentiment.
6	Mad that I wasted 123 minutes and $9.	Negative	*mad that I wasted / wasted 123 minutes / $9.*	3	$9 is not necessarily positive by itself
7	A quiet family drama with a little bit of romance and a dose	Neutral	*A quiet family drama / a little bit of romance / a dose*	4	No reason for *dose* to be included by itself.

Table 6: Examples of text items, their labels and rationale phrases (separated by backslashes) extracted by GPT-4o mini. A score which reflects the quality of the rational is included with a justification.

You are analyzing a cluster of text items from a classification dataset.
Task: Classifying *{task_description}*

Label distribution of items in this cluster: *{label_distribution}*
Below are example phrases (highlighted between square brackets) from the cluster (one per line). Your job is to describe the common linguistic pattern shared by these phrases.
Return:

- A short name for the pattern (3–5 words).
- A 1–2 sentence description.
- Up to three example items with the highlighted phrases that represent the pattern.

Items:
{example_items}

Table 7: Prompt used to summarize cluster-level linguistic patterns from extracted phrases.

Role	Content
System	You are a text classifier. Return exactly ONE label from the allowed labels, with no extra words.
User	Allowed labels: *[$label_1$, $label_2$, . . .]* Text: *[input text]*
Assistant	*[gold label]*

Table 8: Prompt format used for supervised fine-tuning of GPT-4o mini on the classification task. The model is trained to output exactly one label given the input text and the set of allowed labels.