LabelFusion: Fusing Large Language Models with Transformer Encoders for Robust Financial News Classification

Michael Schlee[1], **Christoph Weisser**[2], **Timo Kivimäki**[3]
Melchizedek Mashiku[4], **Benjamin Säfken**[5]
[1]Centre for Statistics, Georg-August-Universität Göttingen, Germany
[2]Hochschule Bielefeld (HSBI) - University of Applied Sciences and Arts, Bielefeld, Germany
[3]Department of Politics and International Studies, University of Bath, Bath, UK
[4]Tanaq Management Services LLC, Contracting Agency to the Division of Viral Diseases
Centers for Disease Control and Prevention, Chamblee, Georgia, USA
[5]Institute of Mathematics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany
michael.schlee@uni-goettingen.de, christoph.weisser@hsbi.de, t.kivimaki@bath.ac.uk
melchizedek.mashiku@tanaq.com, benjamin.saefken@tu-clausthal.de

Abstract

Financial news plays a central role in shaping investor sentiment and short-term dynamics in commodity markets. Many downstream financial applications—such as commodity price prediction or sentiment modeling—therefore rely on the ability to automatically identify news articles that are relevant to specific assets. However, obtaining large labeled corpora for financial text classification tasks is costly, and transformer-based classifiers such as RoBERTa often degrade significantly in low-data regimes. Our results show that appropriately prompted out-of-the-box Large Language Models (LLMs) achieve strong performance even in low-data regimes. Furthermore, we propose LabelFusion, a hybrid architecture that combines the output of a prompt-engineered LLM with contextual embeddings produced by a fine-tuned RoBERTa encoder through a lightweight Multilayer Perceptron (MLP) voting layer. Evaluated on a ten-class multi-label subset of the Reuters-21578 corpus, LabelFusion achieves a macro F1 score of 96.0% and an accuracy of 92.3% when trained on the full dataset, outperforming both standalone RoBERTa (F1 94.6%) and the standalone LLM (F1 93.9%). In low- to mid-data regimes, however, the LLM alone proves surprisingly competitive, achieving an F1 score of 75.9% even in a zero-shot setting and consistently outperforming LabelFusion until approximately 80% of the training data is available. These results suggest that LLM-only prompting represents the preferred strategy under annotation constraints, whereas LabelFusion becomes the most effective solution once sufficient labeled data is available to train the encoder component. The code is available in an anonymized repository.

Keywords: multi-label text classification, large language models, fusion model, financial NLP, Reuters-21578

1. Introduction

Emotional factors play an substantial role in the decision-making processes of both institutional and individual investors. Such emotional signals exert a measurable influence on individual commodity returns, including assets such as oil or gold, and can even enable short-term predictive insights (Sinha and Shastri, 2020).

News articles on specific commodities can act as a primary source of these emotional signals; consequently, financial news represents a valuable resource for identifying market sentiment and potentially anticipating subsequent developments in commodity prices.

Transformer-based models such as FinBERT are commonly employed to extract sentiment from financial news by fine-tuning pre-trained language models on labeled financial text corpora (Araci, 2019). Different model families integrate these sentiment scores in different ways. Hybrid Long Short-Term Memory (LSTM) models use transformer-extracted sentiment as additional input features that vary in time along with price data (Chae and Choi, 2023; Yang et al., 2022; Nabipour et al., 2026). Transformer-based approaches incorporate sentiment through attention mechanisms that weight emotional signals according to their relevance to price movements (Chen et al., 2024).

An important preliminary step in news-based financial prediction tasks, such as forecasting sentiment in news to predict future gold or oil prices, is the filtering of relevant articles. Only news that is directly related to the specific commodity provides meaningful and informative training data for subsequent modeling tasks. This data filtering step is therefore as critical as the sentiment classification task itself. The extremely large volume of available financial news makes manual selection infeasible, creating a clear need for automated text classification methods capable of identifying documents that are relevant to specific commodities or financial assets.

LabelFusion addresses this gap by combining state-of-the-art LLMs with a fine-tuned RoBERTa encoder through an intelligent MLP-based voting mechanism. This approach enables effective multi-label text classification while reducing dependence on extensive manually labeled training data. Our contributions are as follows:

Proceedings of the 7th Financial Narrative Processing Workshop @ LREC 2026, pages 98–105

1. We show that out-of-the-box LLMs, when used with appropriate prompting, achieve high classification accuracy even when training data is scarce.

2. We introduce a hybrid fusion model that combines LLM predictions with RoBERTa embeddings via a trainable MLP. With sufficient training data (80%–100%), the approach improves the F1 score from 0.939 to 0.960 compared to a standalone fine-tuned RoBERTa model.

3. We present *LabelFusion*, a user-friendly software package that facilitates the integration and deployment of fusion-based classification models (Anonymous, 2025).

2. Related Work

The Reuters-21578 corpus (Lewis, 1997) has served as the standard benchmark for multi-label financial news categorization for decades, with classical classifiers such as Support Vector Machine (SVM), k-Nearest Neighbors (KNN), and Rocchio establishing strong baselines under severe label imbalance (Debole and Sebastiani, 2005). More recent work has applied Graph Convolutional Networks (GCNs) that propagate semantic information across document, word, and label nodes (Zeng et al., 2024), as well as attention-based architectures that explicitly model label correlations (Yuan et al., 2024; Ma et al., 2024). Despite their strong empirical results, all of these methods depend on substantial labeled training data and do not incorporate the broad language understanding that LLMs provide.

Brown et al. (2020) demonstrated that GPT-3 achieves competitive performance across a wide range of Natural Language Processing (NLP) benchmarks through in-context few-shot prompting without any gradient updates, an intuition formalized by Schick and Schütze (2021) through cloze-style Pattern-Exploiting Training (PET) and refined by Gao et al. (2021) via automatic prompt and verbalizer search. More recent evaluations confirm that instruction-tuned LLMs are effective zero-shot classifiers (Wang et al., 2023), although fine-tuned encoder models retain a competitive advantage when sufficient labeled data is available (Chae and Davidson, 2025).

Ensemble and fusion approaches that combine predictions from multiple pre-trained models consistently outperform individual classifiers (Abburi et al., 2023), and more tightly integrated architectures show that encoder embeddings and LLM-derived features contribute complementary information (Koloski et al., 2024; Gwak and Jung, 2025). To our knowledge, no prior work has explicitly fused prompt-based LLM predictions with fine-tuned encoder representations for multi-label financial news classification. LabelFusion addresses this gap by combining both sources through a trainable MLP voting layer, enabling robust performance across the full spectrum of labeled data availability.

3. Model Architecture

Let $\mathbf{x}$ denote an input text to be assigned labels from a predefined label set $\mathcal{Y} = \{1, \ldots, K\}$, where K denotes the total number of possible categories. The task is multi-label classification, meaning that multiple labels may be associated with a single input text. LabelFusion combines two complementary components — a prompt-based LLM and a fine-tuned RoBERTa encoder — whose outputs are fused by a trainable MLP voting layer.

3.1. Prompt-Based LLM Component

The input text $\mathbf{x}$ is inserted into a prompt template

$$p(\mathbf{x}) = \mathcal{T}(\mathbf{x}, \mathcal{Y}, \mathcal{E}),$$

where $\mathcal{T}(\cdot)$ denotes the prompt template, $\mathcal{Y}$ represents the list of predefined labels, and $\mathcal{E}$ denotes an optional set of demonstration examples. Three prompt regimes are considered: *zero-shot* ($\mathcal{E} = \emptyset$), *one-shot* ($|\mathcal{E}| = 1$), and *few-shot* ($|\mathcal{E}| > 1$). The prompt is processed by a LLM

$$f_{\text{LLM}} : p(\mathbf{x}) \mapsto \mathbf{z},$$

where $\mathbf{z} \in \{0, 1\}^K$ is a binary prediction vector. Each element z_k is defined as $z_k = 1$ if label k is predicted to be present and $z_k = 0$ otherwise.

3.2. RoBERTa Representation Component

The same input text is independently encoded by a fine-tuned RoBERTa model:

$$h = f_{\text{RB}}(\mathbf{x}) \in \mathbb{R}^{768},$$

where h corresponds to the contextual embedding of the `[CLS]` token, capturing rich task-specific semantic information from the input text.

3.3. Feature Fusion and Prediction

The outputs of both components are concatenated to form a fused representation:

$$u = [h\,;\, z] \in \mathbb{R}^{768+K}.$$

This fused vector combines the dense contextual embedding from RoBERTa with the discrete label

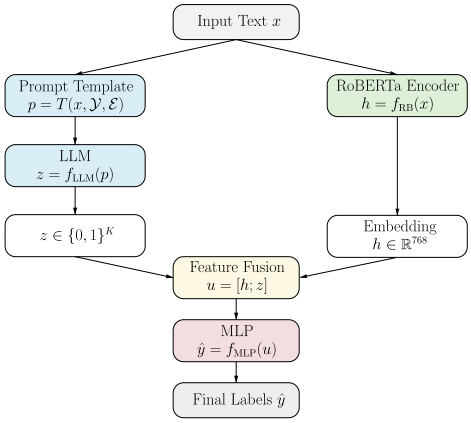

Figure 1: Architecture of *LabelFusion*. An Input text x is processed along two parallel branches: a prompt template $T(x, \mathcal{Y}, \mathcal{E})$ feeds a LLM to produce a binary label vector $z \in \{0,1\}^K$, while a RoBERTa encoder produces a contextual embedding $h \in \mathbb{R}^{768}$. The two representations are concatenated into a joint feature vector $u = [h\,;\, z]$, which is passed to a trainable MLP that outputs the final label predictions $\hat{y}$.

predictions from the LLM, and is passed through a MLP voting model:

$$\hat{\mathbf{y}} = f_{\text{MLP}}(u) \in [0,1]^K,$$

where each element $\hat{y}_k$ represents the predicted probability of label k. The architecture is illustrated in Figure 1.

4. Experimental Setup

4.1. Dataset

The dataset was constructed from the Reuters-21578 (R21578) corpus, which contains approximately 12,902 documents spanning around 135 topic categories with a highly skewed label distribution. To obtain a manageable and well-populated benchmark, only the ten most frequent topics were retained: *earn*, *acq*, *money-fx*, *grain*, *crude*, *trade*, *interest*, *ship*, *wheat*, and *corn*. Documents containing none of the selected topics were excluded, yielding 9,034 documents. Each article was represented using its full raw text, and a multi-label binary target vector of length ten was created for each document. Despite the filtering, the dataset remains challenging: label frequencies range from 3,964 documents for *earn* to only 237 for *corn*, and 9.2% of documents carry two or more topic labels simultaneously.

The corpus provides an official benchmark split into training and test data, which was preserved to ensure comparability with prior work. The test set (2,545 documents) was kept intact from the original ModApte split. From the training portion, a validation set of 647 documents (10%) was carved out using multilabel stratified sampling to maintain a consistent topic distribution across splits, leaving 5,842 documents for training. This procedure yielded three subsets: a training set used for model learning, a validation set for hyperparameter tuning and model selection, and a held-out test set used exclusively for the final evaluation of model performance.

4.2. Experimental Setup

To evaluate the proposed LabelFusion architecture, we conduct experiments with varying amounts of labeled training data. We construct training subsets comprising 5%, 10%, 20%, 40%, 60%, 80%, and 100% of the available labeled dataset (5,842 samples). Each subset is used to train the supervised components of the model (RoBERTa and the fusion MLP), while the LLM branch remains prompt-based. For each subset, the respective proportion of training data is incorporated into the prompt template, which is then sent to the LLM.

We compare LabelFusion against several baselines:

- RoBERTa, a transformer encoder used as a standalone classifier, is fine-tuned on the respective proportion of the training data.
- GPT-5-nano receives training data subsets embedded in our prompt template. Additionally, we evaluate the performance of the out-of-the-box model in a zero-shot setting, where no training data is used and the prompt template is not applied. Furthermore, we assess classification performance in ultra-low data scenarios, where only a single example is included in the prompt template.
- Term Frequency–Inverse Document Frequency (TFIDF) + Logistic Regression (LR), a classical linear baseline trained on the full dataset.

For each configuration, we report Accuracy, F1 score, Precision, and Recall. These metrics allow us to evaluate both overall prediction quality and the balance between false positives and false negatives.

5. Results & Discussion

Table 5 reports macro F1, accuracy, precision, and recall across all training data fractions and models. The classical TFIDF+LR baseline achieves 80.6%

Data	Model	F1	Acc.	Prec.	Rec.
0-shot	GPT-5-nano	75.9	83.4	89.2	70.5
1-shot	GPT-5-nano	**76.1**	84.1	92.0	68.0
5% (292)	Fusion	71.7	70.6	72.0	71.5
5% (292)	RoBERTa	37.2	0.0	27.6	71.3
5% (292)	GPT-5-nano	**93.0**	88.1	95.2	91.7
10% (584)	Fusion	67.1	67.0	67.2	67.1
10% (584)	RoBERTa	41.7	40.0	32.1	61.6
10% (584)	GPT-5-nano	**93.8**	88.5	96.2	92.6
20% (1168)	Fusion	75.2	72.0	76.9	74.5
20% (1168)	RoBERTa	53.4	67.3	46.5	64.3
20% (1168)	GPT-5-nano	**92.8**	88.6	95.1	92.3
40% (2336)	Fusion	88.6	83.6	89.3	88.9
40% (2336)	RoBERTa	83.6	82.0	85.8	85.0
40% (2336)	GPT-5-nano	**93.1**	87.9	95.2	91.7
60% (3505)	Fusion	93.2	85.5	92.9	95.0
60% (3505)	RoBERTa	90.7	83.4	90.6	94.5
60% (3505)	GPT-5-nano	**93.8**	88.4	95.9	92.4
80% (4673)	Fusion	**95.4**	90.2	95.4	96.5
80% (4673)	RoBERTa	94.3	88.8	93.0	96.6
80% (4673)	GPT-5-nano	93.4	88.0	95.1	91.8
100% (5842)	Fusion	**96.0**	92.3	96.7	96.1
100% (5842)	RoBERTa	94.6	89.0	93.2	96.6
100% (5842)	GPT-5-nano	93.9	88.9	96.3	92.7
100% (5842)	TFIDF+LR	68.2	80.6	95.4	56.9

Table 1: Performance comparison of LabelFusion, standalone RoBERTa, GPT-5-nano, and a TFIDF + LR baseline across varying amounts of labeled training data. GPT-5-nano is evaluated in one-shot mode throughout; the zero-shot row reports its performance without any training data. TFIDF + LR is trained on the full dataset and included as a classical reference. All metrics are macro-averaged and reported in percent (%) on the Reuters-21578 test set.

accuracy and a macro F1 of 68.2%, with high precision (95.4%) but low recall (56.9%), reflecting conservative prediction behavior.

Several clear patterns emerge from Table 5. Even in zero-shot mode, the standalone LLM already achieves a macro F1 of 75.9%, exceeding the TFIDF+LR baseline (68.2%) by a large margin, likely due to the broad general knowledge and natural language understanding acquired during pretraining. Moving to one-shot mode yields only a marginal gain (76.1%), suggesting that the LLM requires more than a single demonstration to benefit meaningfully from in-context examples.

In the ultra-low data regime, the standalone RoBERTa classifier struggles severely. With only 5% of the training data (292 documents), it achieves an accuracy of 0.0% and a macro F1 score of 37.2%, indicating that the model has not yet learned a reliable decision boundary. With 10% of the data (584 documents), performance improves to an F1 score of 41.7%, but remains far below the LLM. In contrast, the LLM achieves an F1 score of 93.0% at 5% and 93.8% at 10%, demonstrating that it is a highly effective standalone solution in low-data settings. LabelFusion in these regimes achieves F1 scores of 71.7% and 67.1% at 5% and 10%, which are better than standalone RoBERTa but inferior to the LLM alone. We interpret this as a consequence of the poorly trained RoBERTa component, which appears to introduce noise into the voting process and thereby degrades the overall fusion performance relative to the LLM in isolation.

In the low- to mid-data regime spanning 5% to 60% of the training data, the LLM consistently outperforms both standalone RoBERTa and LabelFusion with respect to macro F1 score. We further observe that using only 5% of the training data is sufficient for the LLM to achieve one of the highest F1 scores in the overall classification task. This finding highlights the potential of LLMs as a standalone approach. In the low- to mid-data regime, LabelFusion is still hampered by noise introduced by the insufficiently fine-tuned RoBERTa encoder, whose weak task-specific signal interferes with the more reliable LLM predictions in the voting layer. Nevertheless, a clear upward trend can be observed: as the proportion of training data increases, the RoBERTa component begins to contribute increasingly useful representations, leading to a steady improvement in LabelFusion's overall accuracy.

From 80% of the training data onward, this trend reaches a tipping point and the situation reverses decisively. LabelFusion surpasses both standalone methods, reaching an accuracy of 90.2% and a macro F1 score of 95.4% at 80%, and achieving the best overall performance with 92.3% accuracy and a macro F1 score of 96.0% when trained on the full dataset. These results confirm that once the RoBERTa component is sufficiently trained, the fusion mechanism effectively combines the task-specific discriminative power of the transformer with the broad language understanding of the LLM, resulting in a model that is more robust than either component in isolation.

6. Conclusion & Future Work

We introduced LabelFusion, a hybrid architecture that fuses the logits of a prompt-engineered LLM with embeddings from a fine-tuned RoBERTa encoder via a trainable MLP voting layer for multi-label financial news classification. Experiments on the Reuters-21578 corpus show that the two model families are complementary and that the optimal strategy depends on the available annotation budget: a carefully prompted LLM constitutes the strongest standalone solution in low- to mid-data regimes, while LabelFusion becomes the preferred choice once sufficient labeled data is available—approximately from 80% of the training data

onward—where the fine-tuned encoder provides reliable task-specific signals that the fusion layer can exploit effectively.

Future work will explore the integration of additional modalities into LabelFusion. The architecture is designed for the seamless integration of further input sources. In particular, we hypothesise that recent stock price time series of corresponding commodities significantly influence the probability that the commodity appears in financial news. The next logical step would therefore be the integration of historical commodity prices as an additional input modality. These time series could be processed by time-series transformer architectures, which are well suited for extracting short-term temporal patterns through attention mechanisms, and could thereby further improve the macro F1 score of the fusion model.

7. Acknowledgments

The work presented in this paper was conducted independently by the author Melchizedek Mashiku and is not affiliated with Tanaq Management Services LLC, Contracting Agency to the Division of Viral Diseases, Centers for Disease Control and Prevention, Chamblee, Georgia, USA.

8. Bibliographical References

Harika Abburi et al. 2023. Generative ai text classification using ensemble llm approaches. *arXiv preprint arXiv:2309.07755*.

Anonymous. 2025. Labelfusion: Learning to fuse llms and transformer classifiers for robust text classification.

Dogu Araci. 2019. FinBERT: Financial sentiment analysis with pre-trained language models.

Tom B. Brown et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Seung Chan Chae and Sun-Yong Choi. 2023. Forecasting the S&P 500 index using mathematical-based sentiment analysis and deep learning models. *Axioms*, 12(9):835.

Youngjin Chae and Thomas Davidson. 2025. Large language models for text classification. *Sociological Methods & Research*.

Yilin Chen, Xiaolong Li, and Yonghong Hu. 2024. Dual-attention transformer for financial news sentiment and stock price prediction. *Expert Systems with Applications*, 238:121134.

Franca Debole and Fabrizio Sebastiani. 2005. An analysis of the relative hardness of reuters-21578 subsets. *Journal of the American Society for Information Science and Technology*, 56(6):584–596.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *ACL*, pages 3816–3830.

Jungwoo Gwak and Yoonsang Jung. 2025. Layer-aware embedding fusion for llms. *arXiv preprint arXiv:2504.05764*.

Boshko Koloski, Senja Pollak, Roberto Navigli, and Blaž Škrlj. 2024. Automl-guided fusion of entity and llm-based representations. In *ICONIP*, pages 89–102.

David D. Lewis. 1997. Reuters-21578 text categorization test collection. Technical report, AT&T Labs Research.

Yilin Ma, Xin Gao, and Bowen Yang. 2024. LIAM: Label-interaction aware model for multi-label text classification. *Neurocomputing*, 580:127139.

Morteza Nabipour, Pejman Naeem, and Hamed Jabani. 2026. Federated learning for financial sentiment-augmented price prediction. In *Proceedings of the 2026 International Conference on Machine Learning and Applications (ICMLA)*. IEEE.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification. In *EACL*, pages 255–269.

Konstantinos Sechidis, Grigorios Tsoumakas, and Ioannis Vlahavas. 2011. On the stratification of multi-label data. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, pages 145–160, Berlin, Heidelberg. Springer.

Anand Sinha and Ravi Shastri. 2020. Impact of news sentiment on commodity returns. *Journal of Behavioral Finance*, 21(3):310–325.

Piotr Szymański and Tomasz Kajdanowicz. 2017. A network perspective on stratification of multi-label data. In *Proceedings of the First International Workshop on Learning with Imbalanced Domains: Theory and Applications (LIDTA)*, pages 22–35. PMLR.

Zhiqiang Wang, Yiran Pang, and Yanbin Lin. 2023. Large language models are zero-shot text classifiers. *arXiv preprint arXiv:2312.01044*.

Jianfeng Yang, Yufei Wang, and Xiang Li. 2022. Prediction of stock price direction using the LASSO-LSTM model combining technical indicators and financial sentiment analysis. *PeerJ Computer Science*, 8:e1148.

Hao Yuan, Weiguang Han, and Yucheng Li. 2024. LACN: Label-aware co-training network for multi-label text classification. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)*, pages 3412–3422. ELRA and ICCL.

Dingkun Zeng, Erxue Zha, Jian Kuang, and Yong Shen. 2024. Multi-label text classification based on semantic-sensitive graph convolutional network. *Knowledge-Based Systems*, 282:111107.

A. Dataset Construction and Processing

This section describes the construction of the dataset used in all experiments. We used the Reuters-21578 corpus as provided through the NLTK Reuters corpus interface and constructed a multi-label classification dataset based on the standard ModApte split.

A.1. Document Extraction

We restricted the label space to the ten most frequent Reuters topics commonly used in the ModApte setting: `earn`, `acq`, `money-fx`, `grain`, `crude`, `trade`, `interest`, `ship`, `wheat`, and `corn`.

Documents from both the original training and test partitions were filtered accordingly, and only those assigned at least one of the selected labels were retained in the final dataset.

For each retained document, the model input was constructed by concatenating the headline and the article body into a single text sequence.

A.2. Dataset Splitting Procedure

The original Reuters training and test partitions were preserved as defined by the corpus reader, where document identifiers beginning with `train` and `test` determine split membership. The filtered test partition was kept unchanged throughout all experiments as the held-out test set.

The filtered training partition was further divided into training and validation subsets using a fixed split ratio of 90/10. To preserve the multi-label distribution across the selected topic set, multilabel-stratified shuffle splitting based on the iterative stratification method was applied (Sechidis et al., 2011; Szymański and Kajdanowicz, 2017).

A.3. Dataset Assembly

All retained documents were aggregated into a tabular dataset in which each row corresponds to a single Reuters article. The final representation consisted of the document text and ten binary topic columns.

After filtering for the selected topic set, the dataset comprised 9,034 documents in total, including 5,842 training, 647 validation, and 2,545 test documents.

All preprocessing and split operations were deterministic to ensure reproducibility across experimental runs.

Topic	Training set	Validation set
earn	2877 (44.30%)	288 (44.51%)
acq	1650 (25.41%)	165 (25.50%)
money-fx	539 (8.30%)	54 (8.35%)
grain	434 (6.68%)	43 (6.65%)
crude	391 (6.02%)	39 (6.03%)
trade	369 (5.68%)	37 (5.72%)
interest	347 (5.34%)	35 (5.41%)
wheat	212 (3.26%)	21 (3.25%)
ship	198 (3.05%)	20 (3.09%)
corn	182 (2.80%)	18 (2.78%)

Table 2: Class distribution in the training and validation sets after multilabel-stratified splitting.

Parameter	Value
Base model	roberta-base
Maximum sequence length	256
Learning rate	2×10^{-5}
Number of epochs	2
Batch size	32
Task setting	Multi-label classification

Table 3: Hyperparameters of the standalone RoBERTa classifier.

A.4. Class Distribution

The distribution of topic labels in the training dataset reflects the natural class imbalance of the Reuters corpus. As shown in Table 2, the relative frequency of each topic remains nearly identical between the training and validation sets after multilabel-stratified splitting, demonstrating that the stratification procedure successfully preserved the overall class distribution across subsets.

B. Hyperparameter and Training Configuration

The configuration parameters for all model components and experimental settings are summarized in Tables 3–6. Unless stated otherwise, the same hyperparameters were applied across all experimental conditions.

B.1. Prediction Threshold

Predicted probabilities produced by the classification models were converted into binary label assignments using a fixed threshold of 0.5 for all topic categories. The same threshold was applied consistently across all models and experimental settings.

Parameter	Value
Model	GPT-5-nano
Temperature	0.1
Top-p	1.0
Maximum completion tokens	150
Task setting	Multi-label classification
Caching	Enabled

Table 4: Hyperparameters of the LLM classifier.

Parameter	Value
Fusion hidden dimensions	[64, 32]
Learning rate (RoBERTa in fusion)	1×10^{-5}
Learning rate (fusion MLP)	5×10^{-4}
Number of epochs	10
Batch size	8
Classification type	Multi-label

Table 5: Hyperparameters of the fusion ensemble model.

C. Prompt Template

For an input text x, the prompt presented to the language model follows a consistent structure consisting of task instructions, optional training examples, and an output format specification.

The initial role prompt is automatically generated from labeled training examples using a dedicated meta-prompting procedure. This design ensures that the prompt adapts to the characteristics of each dataset rather than relying on a fixed manually specified role description.

```
You are a financial analyst who
identifies whether a news article
belongs to one or more predefined
commodity-related categories.

The following examples show texts and
their classifications for labels:
earn, acq, money-fx, grain, crude,
trade, interest, ship, wheat, corn

Training Data Examples:
Example 1:
Text: U.S. grain exporters reported
increased shipments following strong
overseas demand.
Ratings: 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0
| 0 | 0

Example 2:
Text: The company announced plans to
acquire a regional competitor in a
stock transaction.
Ratings: 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0
| 0 | 0

Given the above information, make a
prediction for the following paragraph.

{test article}

The output format shall be:

    0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
0

Each value must be either 1 (label
present) or 0 (absent).
Multiple values may be 1, as this is a
multi-label classification task.
No additional text shall be shown --
output only the classification line.
```

Prompting regime	Number of examples
Zero-shot	0
One-shot	1
Few-shot	20

Table 6: Number of in-context examples used in each prompting regime.