MyMy at SemEval-2025 Task 9: A Robust Knowledge-Augmented Data Approach for Reliable Food Hazard Detection

Ben Phan
Department of Computer Science
and Information Engineering,
National Cheng Kung University
Tainan City 701, Taiwan
ben@iir.csie.ncku.edu.tw

Jung-Hsien Chiang
Department of Computer Science
and Information Engineering,
National Cheng Kung University
Tainan City 701, Taiwan
jchiang@mail.ncku.edu.tw

Abstract

The Food Hazard Detection (SemEval-2025 Task 9) advances explainable classification of food-incident reports collected from web sources, including social media and regulatory agency websites, to support timely risk mitigation for public health and the economy. This task is complicated by a highly imbalanced, long-tail label distribution and the need for transparent, reliable AI. We present a robust Knowledge-Augmented Data approach that integrates Retrieval-Augmented Generation (RAG) with domain-specific knowledge from the PubMed API to enrich and balance the training data. Our method leverages domain-specific knowledge to expand datasets and curate high-quality data that enhances overall data integrity. We hypothesize that Knowledge-Augmented Data improves Macro-F1 scores, the primary evaluation metric. Our approach achieved a top-2 ranking across both subtasks, demonstrating its effectiveness in advancing NLP applications for food safety and contributing to more reliable food hazard detection[1].

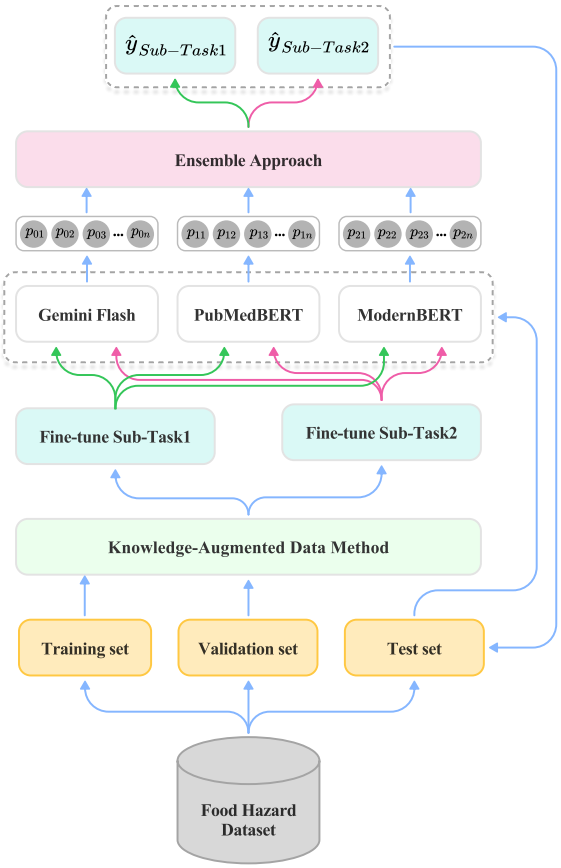

Figure 1: Overall System Architecture for Food Hazard Detection Challenge Using the Knowledge-Augmented Data Method.

1 Introduction

The increasing volume of food incident reports from various online sources highlights an urgent need for automated detection systems. These reports come from social media and official food agency websites and reflect economic and public health risks associated with foodborne illnesses and contamination. SemEval-2025 Task 9 addresses these challenges by developing systems that classify food incident reports and predict potential hazards. Current methodologies, particularly data augmentation from large language models (LLMs), face hurdles such as hallucination, which complicates the development of reliable, scalable solutions. These challenges are further exacerbated by class imbalance in datasets and the need for transparent, explainable AI.

To address these issues, we introduce a Knowledge-Augmented Data approach that integrates RAG (Lewis et al., 2020) to enhance data quality with domain-specific knowledge. Our method involves retrieving relevant data from PubMed, generating augmented samples using advanced LLMs, filtering out low-quality data, and fine-tuning models to maximize detection accuracy. This comprehensive strategy enriches the dataset and improves data integrity, leading to significant

[1] https://github.com/phanben110/KAD-FoodHazard

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 812–822

gains in Macro-F1 scores, the primary evaluation metric for this task.

Our team achieved outstanding results in the competition, securing 2nd place in Subtask 1 and Subtask 2. These outcomes demonstrate the strength of our food hazard detection approach, integrating domain-specific knowledge to overcome data limitations and highlighting RAG's potential in generating high-quality training data.

2 Background

2.1 Food Hazard Detection Dataset

SemEval-2025 Task 9 (Randl et al., 2025) focuses on explainable classification of food incident reports from web sources. It aids automated crawlers in identifying food-related issues on platforms like social media. The task comprises two subtasks: (1) text classification for food hazard and product category prediction (ST1), predicting both the type of hazard and the product category; (2) food hazard and product *"vector"* detection (ST2), predicting the exact hazard and product mention.

The dataset comprises 6,644 expert-labeled recall titles *(year, month, day, country, title, full text)* from various sources, covering 1,142 products in 22 categories and 128 hazards across 10 categories. Table 1 shows the data splits. The data exhibit class imbalance and label diversity (see Appendix D).

Dataset	Samples	Classes			
		hazard-cat.	product-cat.	hazard	product
Train	5,082	10	22	128	1,022
Val	565	9	18	93	312
Test	997	10	20	110	447

Table 1: Statistics of the SemEval-2025 Task 9 dataset, including the number of samples and class distributions for training, validation, and test sets.

2.2 Related Works

Data augmentation techniques, such as back translation (Sennrich et al., 2016) and Easy Data Augmentation (EDA) (Wei and Zou, 2019), have been widely used in NLP to address data limitations, though they risk altering meaning (Feng et al., 2021). LLMs like ChatGPT (Achiam et al., 2024) and Llama (Touvron et al., 2023) further enhance augmentation, improving performance across various tasks (Ding et al., 2024). For instance, the CLaC team in SemEval-2024 Task 4 (Nayak and Kosseim, 2024) used paraphrase augmentation to mitigate data scarcity, while GPT-4 has been employed in biomedical relation extraction to enhance model performance (Phan et al., 2024).

RAG has also been explored for biomedical information retrieval, with a study by Li et al. (2025) highlighting its potential to improve answer relevance, noting challenges in grounding and contextual accuracy. Additionally, retrieval-based approaches using the PubMed API have been employed to inject external knowledge into NLP tasks. For example, Thomo (2024) used PubMed queries to extract relevant literature, integrating retrieved documents into language models to enhance medical question answering. Building on this line of work, our Knowledge-Augmented Data method leverages RAG to improve food hazard detection with higher accuracy and reliability.

3 System Overview

Our proposed method, Knowledge-Augmented Data for Food Hazard Detection, enhances the quality and diversity of training data by integrating external knowledge and advanced filtering techniques. As illustrated in Figure 2, it leverages LLMs combined with Retrieval-Augmented Generation (RAG) using external sources such as the PubMed API[2] to generate high-quality augmented data, boosting model robustness and performance.

The framework consists of four main steps: (1) simplifying complex queries using LLMs to retrieve relevant external knowledge; (2) generating augmented samples based on the retrieved context; (3) filtering low-quality data through a score-based validation process; and (4) fine-tuning multiple deep learning models on the enriched dataset merging original and high-quality augmented samples. Finally, as shown in Figure 1, an ensemble mechanism combines predictions from different models to achieve optimal results in food hazard detection.

3.1 Information Retrieval System

The Information Retrieval System is crucial in augmenting data by incorporating external knowledge. Since complex queries often fail to return results via the PubMed API, the system first simplifies the original query Q_{complex} into a more concise form Q_{simple} using LLMs with *Prompt 1* (see Appendix B). The complex query Q_{complex} typically contains detailed scientific terminology, multiple conditions, or lengthy descriptions of food safety concerns, which can be overly specific for effective

[2] https://pubmed.ncbi.nlm.nih.gov/

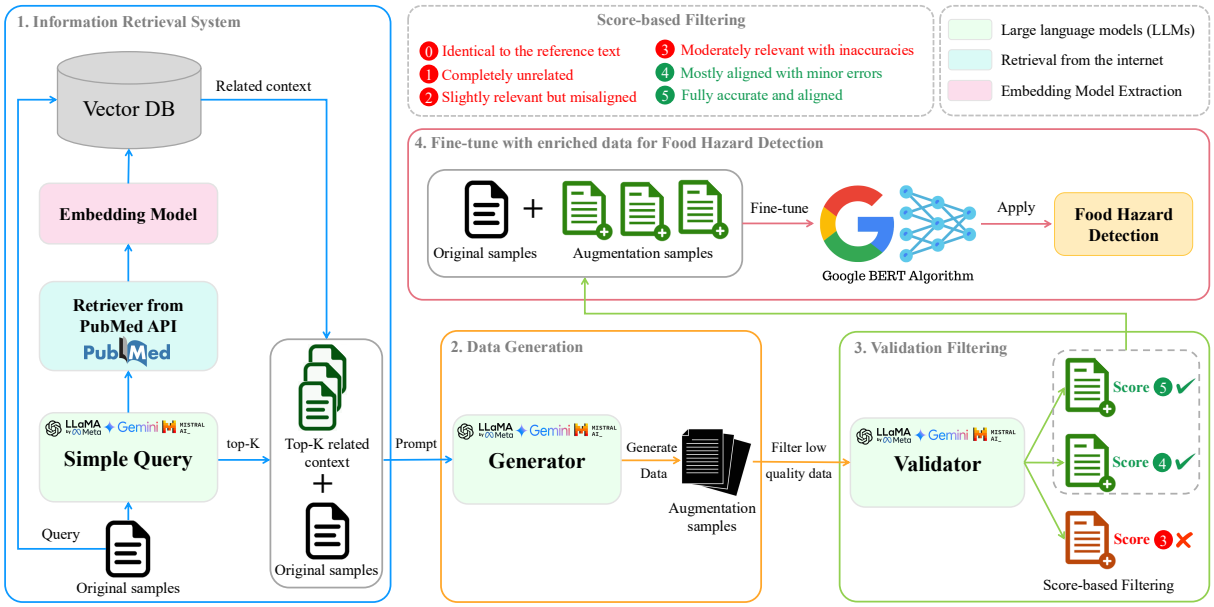

Figure 2: The Knowledge-Augmented Data method for food hazard detection comprises four components: (1) Information Retrieval, which collects relevant data from the PubMed API; (2) Data Generation, where large language models (LLMs) generate augmented samples; (3) Validation Filtering, which scores and removes low-quality data; and (4) Fine-tuning, which enhances LLMs and deep learning models to improve detection accuracy.

API retrieval. Through LLM-based transformation, Q_{simple} retains the core information needs while using more generalized terminology and focusing on essential keywords, thereby increasing the likelihood of successful matches in the database.

This simplification ensures more effective document retrieval. The refined query is then used to fetch top-K relevant documents from PubMed, which are subsequently embedded into dense vector representations for efficient storage and retrieval. To identify the most relevant documents, cosine similarity is calculated between the original query vector v_{original} and each document embedding v_d:

$$\text{sim}(v_{\text{original}}, v_d) = \frac{v_{\text{original}} \cdot v_d}{\|v_{\text{original}}\| \|v_d\|} \quad (1)$$

The top-K most relevant documents are then selected based on their similarity scores:

$$D_{\text{retrieved}} = \{d_i \mid i \in \text{argmax}_K \ \text{sim}(v_{\text{original}}, v_{d_i})\} \quad (2)$$

Only documents with a similarity score above a predefined top-K threshold are selected for augmentation, ensuring the dataset remains contextually relevant and informative. Integrating this retrieval system with RAG improves model accuracy in detecting food hazards, as demonstrated in Section 5.

3.2 Data Generation

In this step, augmented samples are generated by leveraging the retrieved context $D_{\text{retrieved}}$ and the original samples S_{original}. LLMs create new data points by integrating the retrieved context with the original content. This augmentation process enhances dataset diversity while maintaining semantic relevance and contextual consistency. The data generation process follows a structured approach using *Prompt 2* template provided in Appendix B to ensure consistency and control over the augmentation process.

3.3 Validation Filtering

After data generation, each augmented sample is scored by an LLM-based function G_{val}, evaluating its relevance and accuracy against the retrieved context $D_{\text{retrieved}}$ using *Prompt 3* (see Appendix B). The filtered set S_{filtered} is defined as:

$$S_{\text{filtered}} = \{s \mid G_{\text{val}}(s, D_{\text{retrieved}}) \geq 4\} \quad (3)$$

As shown in Figure 2, only samples scoring **4** or **5** are retained for their quality and contextual alignment. Samples scoring **0** are discarded as redundant, while those scoring **1, 2, 3** are excluded for inaccuracy or irrelevance, ensuring strong contextual relevance and reliability.

Subtask 1 (ST1)				Subtask 2 (ST2)			
Rank	**Team Name**	**Score**	**Features**	**Rank**	**Team Name**	**Score**	**Features**
1	Anastasia	0.8223	META, TITLE, TEXT	1	SRCB	0.5473	TITLE, TEXT
2	**MyMy (Our team)**	**0.8112**	**META, TITLE, TEXT**	**2**	**MyMy (Our team)**	**0.5278**	**META, TITLE, TEXT**
3	SRCB	0.8039	TITLE, TEXT	3	PATeam	0.5266	TITLE, TEXT
4	PATeam	0.8017	TITLE, TEXT	4	HU	0.5099	TITLE, TEXT
5	HU	0.7882	TITLE, TEXT	5	MINDS	0.4862	TITLE, TEXT
6	BitsAndBites	0.7873	TITLE, TEXT	6	Fossils	0.4848	TITLE, TEXT
7	CSECU-Learners	0.7863	TITLE, TEXT	7	CSECU-Learners	0.4797	TITLE, TEXT
8	ABCD	0.7860	TITLE, TEXT	8	PuerAI	0.4783	N/A
9	MINDS	0.7857	TITLE, TEXT	9	Zuifeng	0.4712	N/A
10	Zuifeng	0.7835	N/A	10	ABCD	0.4576	TITLE, TEXT
11	Fossils	0.7815	TITLE, TEXT	11	BrightCookies	0.4529	TEXT
12	PuerAI	0.7729	N/A	12	Ustnlp16	0.4512	TITLE, TEXT
13	Ustnlp16	0.7654	TITLE, TEXT	13	BitsAndBites	0.4456	TITLE, TEXT
14	FuocChu_VIP123	0.7646	N/A	14	UniBuc	0.3453	TITLE, TEXT
15	BrightCookies	0.7610	TEXT	15	OPI-DRO-HEL	0.3295	TITLE, TEXT
16	farrel_dr	0.7587	TITLE, TEXT	16	VerbaNexAI	0.3223	TITLE
17	OPI-DRO-HEL	0.7381	TITLE, TEXT	17	CICL	0.3169	TEXT
18	madhans476	0.7362	TITLE, TEXT	18	Somi	0.3048	META, TITLE, TEXT
19	Anaselka	0.6858	TITLE, TEXT	19	TechSSN3	0.2712	TEXT
20	Somi	0.6614	META, TITLE, TEXT	20	Howard University-AI4PC	0.1380	TEXT

Table 2: Leaderboard results for SemEval-2025 Task 9 (Top 20). Our team ***My My*** (**ST1:** 2nd, **ST2:** 2nd) delivered consistent, high-level performance across both subtasks, demonstrating the robustness and adaptability of our feature set and modeling strategy. In contrast to top teams that showed significant variance between subtasks, such as **Anastasia** (**ST1:** 1st, **ST2:** 21st) and **SRCB** (**ST1:** 3rd, **ST2:** 1st), our approach achieved both stability and accuracy throughout the competition. *META* refers to temporal and geographical features (*YEAR, MONTH, DAY, COUNTRY*).

3.4 Fine-tuning with Enriched Data

The combined dataset, consisting of the original training data D_{train} and filtered augmented samples S_{filtered}, forms the final training set:

$$D_{\text{final}} = D_{\text{train}} \cup S_{\text{filtered}} \tag{4}$$

We fine-tune multiple pre-trained models: Gemini Flash 2.0 (Team et al., 2024), PubMedBERT (Gu et al., 2021), and ModernBERT (Warner et al., 2024). This ensures that each model is adapted to the enriched data for optimal performance.

3.5 Ensemble Strategy

To enhance prediction accuracy, we employ an **ensemble strategy** that aggregates predictions from multiple models. The predicted labels for each subtask are computed using weighted sums of class probabilities:

$$\begin{cases} \hat{y}_{\text{Subtask1}} = \arg\max\limits_{y \in Y_1} \sum\limits_i w_i P_{\text{task1},i}(y) \\ \hat{y}_{\text{Subtask2}} = \arg\max\limits_{y \in Y_2} \sum\limits_i w_i P_{\text{task2},i}(y) \end{cases} \tag{5}$$

Here, w_i denotes the weight assigned to the i-th model's prediction. In our case, we use equal weighting, i.e., $w_i = \frac{1}{N}$, where N is the total number of models. $P_{\text{task1},i}(y)$ and $P_{\text{task2},i}(y)$ represent the predicted class probabilities for Subtask 1 and Subtask 2, respectively. This ensemble strategy facilitates robust, consensus-based decision-making, leading to more accurate food hazard predictions. The detailed algorithm is provided in Appendix A.

4 Experimental Setup

4.1 Model Training and Augmentation

Our experimental setup integrates knowledge-augmented data generation with fine-tuning on an enriched Food Hazard dataset. From the dataset, we utilize the following features: *YEAR*, *MONTH*, *DAY*, *COUNTRY*, *TITLE*, and *TEXT*. These fields are used for retrieval and input context for data augmentation and model training. For data augmentation, we employ GPT-3.5 Turbo[3], Gemini Flash 2.0 (Team et al., 2024), Llama 3.1 8B (Touvron et al., 2023), and Mixtral 8x7 B (Jiang et al., 2023) as LLMs, with a temperature setting of 0.7 to balance creativity and factual consistency. Knowledge retrieval is performed using vector embeddings from nomic-embed-text-v1 (Nussbaum et al., 2025) stored in a Chroma vector database. The input texts are chunked into 500-token segments with 100-token overlaps. We also incorporate ex-

[3] https://platform.openai.com/docs/models/gpt-3-5-turbo

Method	Subtask 1							Subtask 2						
	hazard-category			product-category			Macro-F1	hazard			product			Macro-F1
	P	R	F1	P	R	F1		P	R	F1	P	R	F1	
Gemini Flash	0.7395	0.7605	0.7477	0.8701	0.7803	0.8057	0.7767	0.6473	0.6694	0.647	0.3476	0.3591	0.3401	0.4936
PubMebBERT	0.7706	0.7837	0.7766	**0.8703**	0.788	0.8096	0.7931	0.6748	0.708	0.6787	0.3662	0.3867	0.3622	0.5204
ModernBERT	0.7807	0.7688	0.7734	0.8233	0.7548	0.774	0.7737	**0.6879**	0.6883	0.6768	0.3578	0.3774	0.3534	0.5151
Ensemble (MyMy)	**0.7958**	**0.8121**	**0.8032**	0.8677	**0.8083**	**0.8193**	**0.8112**	0.6866	**0.7107**	**0.6892**	**0.3705**	**0.3928**	**0.3665**	**0.5278**

Table 3: Performance comparison of methods for the Food Hazard Detection Challenge. The table reports precision (P), recall (R), and F1-score (F1) for hazard-category and product-category in Subtask 1, and hazard and product in Subtask 2. Final Macro-F1 scores highlight the ensemble as the top-performing method across both subtasks.

ternal knowledge via the PubMed API to support retrieval-augmented generation (RAG). All augmentation tasks are run on a dual NVIDIA GeForce RTX 4090 GPU setup.

For fine-tuning, we use Gemini Flash 2.0 (Team et al., 2024), PubMedBERT (Gu et al., 2021), and ModernBERT (Warner et al., 2024). Models are trained on the augmented dataset for 200 epochs using an NVIDIA A100 and RTX 4090. The training is configured with a learning rate of 5e-5, a sequence length of 512, and a batch size of 90. This pipeline effectively combines LLM-based augmentation, retrieval-augmented generation, and fine-tuning to enhance data quality and downstream model performance.

4.2 Evaluation

We evaluate **Subtask 1** and **Subtask 2** using the *macro-averaged F1-score*. The macro-F1 for hazards, computed over all hazard classes $\mathcal{C}_h$, is:

$$F1_{\text{hazards}} = \frac{1}{|\mathcal{C}_h|} \sum_{c \in \mathcal{C}_h} \frac{2 \cdot P_c \cdot R_c}{P_c + R_c} \tag{6}$$

where P_c and R_c are the precision and recall for hazard class c, $|\mathcal{C}_h|$ represents the number of hazard classes, and $\mathcal{C}_h$ is the set of all hazard classes. For products, we compute F1 only on the subset S where hazard predictions are correct:

$$S = \left\{ i \mid y_h^{\text{pred}}(i) = y_h^{\text{true}}(i) \right\} \tag{7}$$

where i is a sample index, $y_h^{\text{pred}}(i)$ is the predicted hazard label, and $y_h^{\text{true}}(i)$ is the ground truth hazard label for sample i. The product macro-F1 over subset S is:

$$F1_{\text{products}} = \frac{1}{|\mathcal{C}_p|} \sum_{k \in \mathcal{C}_p} \frac{2 \cdot P_k^S \cdot R_k^S}{P_k^S + R_k^S} \tag{8}$$

where P_k^S and R_k^S are the precision and recall for product class k computed over subset S, $|\mathcal{C}_p|$ is the number of product classes, and $\mathcal{C}_p$ is the set of all product classes. The final score averages both F1 scores:

$$\text{Score} = \frac{F1_{\text{hazards}} + F1_{\text{products}}}{2} \tag{9}$$

This scoring emphasizes hazard prediction. Product outputs only count when hazards are correctly predicted. A perfect system scores 1.0, correct hazards but failed products score 0.5, and incorrect hazards result in a score of 0.0, regardless of product predictions.

5 Results

5.1 Overview of the SemEval-2025 Task 9

Table 2 presents the detailed leaderboard results. Out of more than 260 participating teams worldwide, 27 system description papers were submitted for peer review. The table highlights the rankings of the top 20 systems, showcasing the diverse approaches in the shared task [4].

The SemEval-2025 Task 9: The Food Hazard Detection Challenge attracted significant global attention, emphasizing the growing research interest in automated food hazard detection. The competition was organized into two independent subtasks. Our team, ***My My*** (**ST1:** 2nd, **ST2:** 2nd), achieved substantial and consistent results, ranking second in both Subtask 1 (score: 0.8112) and Subtask 2 (score: 0.5278).

Our approach demonstrated superior overall balance across the two subtasks compared to other top-performing teams. For instance, while **Anastasia** (**ST1:** 1st, **ST2:** 21st) ranked first in Subtask 1, their performance dropped significantly to 21st place in Subtask 2. Conversely, **SRCB** (**ST1:** 3rd, **ST2:** 1st) ranked first in Subtask 2 but only third in Subtask 1. In contrast, ***My My*** maintained top-tier performance across both tasks, indicating the robustness and adaptability of our system to varying task requirements and evaluation criteria.

[4] https://food-hazard-detection-semeval-2025.github.io/

5.2 Performance of Our Ensemble Approach

Our ensemble method, which combines Gemini Flash, PubMedBERT, and ModernBERT, consistently outperformed individual models across both Food Hazard Detection Challenge subtasks. As shown in Table 3, the ensemble achieved a Macro-F1 score of 0.8112 in Subtask 1, surpassing PubMedBERT's score of 0.7931, with strong F1-scores in both hazard-category (0.8032) and product-category (0.8193) classification. In Subtask 2, the ensemble recorded a Macro-F1 score of 0.5278, exceeding ModernBERT's score of 0.5151, and also achieved the highest F1-scores for hazard (0.6892) and product (0.3665) detection. These results demonstrate that the ensemble approach effectively balances precision and recall, particularly in the context of imbalanced and diverse food hazard datasets.

The advantage of ensembling lies in leveraging the complementary strengths of each model: PubMedBERT's biomedical expertise, ModernBERT's ability to handle long contexts, and Gemini Flash's efficiency and effectiveness when fine-tuned on short-text classification tasks. By aggregating predictions, the ensemble reduces the risk of individual model biases and improves robustness, especially for rare and underrepresented classes. Our system's consistent top-2 ranking in the SemEval-2025 Challenge across both subtasks further highlights this approach's practical value and reliability for real-world food safety applications.

5.3 Analysis

Our Knowledge-Augmented Data Method stands out for its balanced performance across both subtasks, demonstrating strong robustness. Unlike Team Anastasia (fixed token chunking and ensembling) or Team SRCB (two-stage DeBERTa+LLM pipeline), our approach leverages RAG with validation filtering to ensure high-quality augmentation. This streamlined pipeline addresses class imbalance and enhances representation for rare categories, leading to consistent results across hazard and product classifications.

As shown in Appendix D, our method achieves a more balanced distribution of underrepresented classes while maintaining competitive performance. The confusion matrices in Appendix C, which provide class-wise prediction breakdowns for both subtasks, show significantly reduced false negatives in rare hazard categories, confirming our system's effectiveness. These results underscore the scalability and practicality of our method for real-world food hazard detection tasks.

6 Conclusion

Our research demonstrates that a Knowledge-Augmented Data approach significantly improves the accuracy and reliability of food hazard detection. By integrating RAG with advanced data filtering, our system achieved top rankings in SemEval-2025 Task 9, showcasing the effectiveness of domain-specific knowledge in addressing data limitations. This highlights the potential of knowledge-driven AI to enhance food safety by rapidly and accurately identifying incidents from diverse online sources. Future work will optimize model efficiency for large-scale deployment and improve recall in product classification to maximize real-world impact. Importantly, our results underscore the value of explainability and transparency, which are essential for building trust and facilitating adoption in practical food safety applications.

7 Limitations

Although effective, our knowledge-augmented data approach has limitations. The retrieval process from external sources, such as PubMed, can occasionally return irrelevant or incorrect results, compromising data quality. Dependency on external knowledge also introduces latency, limiting real-time applicability. The validation filtering process may also exclude valuable samples below the scoring threshold, reducing dataset diversity. Finally, the ensemble method increases computational costs, making it less suitable for resource-constrained environments. Future work will improve retrieval accuracy, refine validation techniques, and optimize computational efficiency.

Acknowledgments

We sincerely thank the SemEval-2025 Task 9 Food Hazard Detection Challenge organizers for their valuable contributions and efforts. We also extend our gratitude to the broader SemEval community for fostering innovation in semantic evaluation and NLP research. Furthermore, we gratefully acknowledge the support of the Intelligent Information Retrieval Laboratory (IIR Lab) at National Cheng Kung University (NCKU).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2024. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Bosheng Ding, Chengwei Qin, Ruochen Zhao, Tianze Luo, Xinze Li, Guizhen Chen, Wenhan Xia, Junjie Hu, Anh Tuan Luu, and Shafiq Joty. 2024. Data augmentation using LLMs: Data perspectives, learning paradigms and challenges. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1679–1705, Bangkok, Thailand. Association for Computational Linguistics.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Mingchen Li, Halil Kilicoglu, Hua Xu, and Rui Zhang. 2025. Biomedrag: A retrieval augmented large language model for biomedicine. *Journal of Biomedical Informatics*, 162:104769.

Kota Shamanth Ramanath Nayak and Leila Kosseim. 2024. CLaC at SemEval-2024 task 4: Decoding persuasion in memes – an ensemble of language models with paraphrase augmentation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 175–180, Mexico City, Mexico. Association for Computational Linguistics.

Zach Nussbaum, John X. Morris, Brandon Duderstadt, and Andriy Mulyar. 2025. Nomic embed: Training a reproducible long context text embedder.

Cong-Phuoc Phan, Ben Phan, and Jung-Hsien Chiang. 2024. Optimized biomedical entity relation extraction method with data augmentation and classification using gpt-4 and gemini. *Database*, 2024:baae104.

Korbinian Randl, John Pavlopoulos, Aron Henriksson, and Tony Lindgren. 2025. SemEval-2025 task 9: The food hazard detection challenge. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, Vienna, Austria. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2024. Gemini: A family of highly capable multimodal models.

Alex Thomo. 2024. Pubmed retrieval with rag techniques. In *Digital Health and Informatics Innovations for Sustainable Health Care Systems - Proceedings of MIE 2024, Athens, Greece, 25-29 August 2024*, volume 316 of *Studies in Health Technology and Informatics*, pages 652–653. IOS Press.

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

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

A Knowledge-Augmented Data Algorithm

Algorithm 1 outlines the Knowledge-Augmented Data Method, enhancing food hazard detection

via document retrieval, LLM-based augmentation, quality filtering, and ensemble fine-tuning.

Algorithm 1 Knowledge-Augmented Data Method

Require: Food Hazard Dataset D
Ensure: Predictions $\hat{y}_{\text{Subtask1}}, \hat{y}_{\text{Subtask2}}$
▷ Step 1: Data Splitting
Split D into $D_{\text{train}}, D_{\text{val}}, D_{\text{test}}$
▷ Step 2: Knowledge-Augmented Data Generation
▷ *2.1 Information Retrieval System*
$Q_{\text{simple}} \leftarrow G_{\text{query}}(Q_{\text{complex}})$
$D_{\text{pubmed}} \leftarrow \text{RETRIEVEPUBMED}(Q_{\text{simple}}, K)$
$V_{\text{pubmed}} \leftarrow \text{Embed}(D_{\text{pubmed}})$ ▷ Convert to embeddings
VectorDB $\leftarrow \text{Store}(V_{\text{pubmed}})$ ▷ Store into vector database
$\text{sim}(v_{\text{original}}, v_d) \leftarrow \frac{v_{\text{original}} \cdot v_d}{\|v_{\text{original}}\| \|v_d\|}$ ▷ Compute similarity
$D_{\text{retrieved}} \leftarrow \{d_i \mid i \in \text{argmax}_K(\text{sim}(v_{\text{original}}, v_d))\}$ ▷ Retrieve top-K documents
▷ *2.2 Data Generation*
$S_{\text{augmented}} \leftarrow G_{\text{gen}}(D_{\text{retrieved}}, S_{\text{original}})$
▷ *2.3 Validation Filtering*
$S_{\text{filtered}} \leftarrow \{s \mid G_{\text{val}}(s, D_{\text{retrieved}}) \geq \tau\}$
▷ *2.4 Fine-tune with Enriched Data*
$D_{\text{final}} \leftarrow D_{\text{train}} \cup S_{\text{filtered}}$
▷ Step 3: Fine-tune Models
for $M \in$ {Gemini Flash, PubMedBERT, ModernBERT} **do**
  $M \leftarrow \text{FINETUNE}(D_{\text{final}}, M)$
end for
▷ Step 4: Inference
for $M \in$ {Gemini Flash, PubMedBERT, ModernBERT} **do**
  $P_{\text{task1},M} \leftarrow \text{INFERENCE}(D_{\text{test}}, M)$
  $P_{\text{task2},M} \leftarrow \text{INFERENCE}(D_{\text{test}}, M)$
end for
▷ Step 5: Ensemble Approach
$\hat{y}_{\text{Subtask1}} \leftarrow \arg\max_{y \in Y_1} \sum_i w_i P_{\text{task1},i}(y)$
$\hat{y}_{\text{Subtask2}} \leftarrow \arg\max_{y \in Y_2} \sum_i w_i P_{\text{task2},i}(y)$
return $\hat{y}_{\text{Subtask1}}, \hat{y}_{\text{Subtask2}}$

B Prompt Template

Prompt 1: Simple Query

Simplify verbose queries from the internet into concise ones while retaining essential terms.

Follow these rules:

1. Identify and retain all critical keywords, names, and technical terms.
2. Simplify the query to be concise and under 10 words.
3. Ensure the simplified query preserves the original meaning.

Example:

- **Verbose:** The Canadian Food Inspection Agency warns consumers about undeclared pecans in Originale Augustin Ice Cream in Quebec.
- **Simplified:** Undeclared pecans in Originale Augustin Ice Cream recall.

Task: Simplify the input query: {passage}, and output only the simplified query.

Prompt 2: Data Generation

You are tasked with paraphrasing the given passage to generate data for food hazard detection.

Follow these rules:

1. Retain critical information such as food product names, batch numbers, contamination types, and affected regions.
2. Ensure contextual accuracy: the paraphrase must be precise and align with the original context. Do not alter the meaning or factual content.
3. Highlight key hazards (e.g., contamination, undeclared allergens) and their potential risks to public health.

<context> {context} </context>

Here is your task: Given the input: {passage}

- Paraphrase it according to the rules above, ensuring the augmented text highlights key food hazards and is consistent with the context.
- Output only the paraphrased result, with no additional comments.

Prompt 3: Validation Filtering

You are tasked with evaluating paraphrased text as a form of data augmentation, using the following scoring system:

Scoring System:

- **0: The data augmentation result is the same as the reference text.**
- **1: The data augmentation result is completely unrelated to the reference text.**
- **2: The data augmentation result has minor relevance but does not align with the reference text.**
- **3: The data augmentation result has moderate relevance but contains inaccuracies.**
- **4: The data augmentation result aligns with the reference text but has minor errors or omissions.**
- **5: The data augmentation result is accurate and aligns perfectly with the reference text.**

Task:

- Given the original text: {original_text}
- Given the augmented text: {augmented_text}
- Evaluate the paraphrased text and assign a score from 0 to 5.

We designed three prompt templates to enhance food hazard detection: Simple Query (Prompt 1), Data Generation (Prompt 2), and Validation Filtering (Prompt 3).

Prompt 1: Simple Query condenses verbose queries while preserving key terms for effective

knowledge retrieval.

Prompt 2: Data Generation paraphrases data while maintaining critical details, improving dataset diversity, and addressing class imbalance.

Prompt 3: Validation Filtering evaluates augmented samples, retaining only high-quality data to ensure dataset integrity.

These prompts optimize retrieval, augmentation, and quality control, strengthening food hazard detection.

C Confusion Matrices

The confusion matrices in Figures 3 and 4 show that my method gives good results. Most predictions are along the diagonal, indicating high accuracy for both tasks. Misclassifications are limited and mainly between similar classes, demonstrating the approach's effectiveness. This pattern also suggests that the model can be generalized well even for underrepresented categories.

D Data Augmentation Pipeline Analysis

D.1 Addressing Class Imbalance

The Food Hazard Detection Dataset shows significant class imbalance across several categories, as illustrated in Figures 5, 6, 7, and 8. For example, Figure 8 reveals a highly skewed *product* distribution, with a few dominant classes and over 1,142 unique product types, most of which are underrepresented. Similarly, Figure 5 shows that certain hazard types disproportionately dominate the dataset.

The data augmentation pipeline effectively mitigates these imbalances, as shown by the more balanced distributions (in red). Underrepresented classes are significantly boosted in both *hazard-category* and *product-category*. In particular, Figure 8 highlights the improved balance post-augmentation, reducing the dominance of frequent classes and ensuring fairer representation. These adjustments are essential for enhancing model performance on rare but important categories.

D.2 Model Success Rates

Table 4 provides insights into the success rates of different models used in the augmentation pipeline. The success rate is calculated using the formula:

$$\text{Success Rate (\%)} = \frac{\text{Filtered}}{\text{Augmentation}} \times 100 \quad (10)$$

Method	Augmentation	Filtered	Success Rate (%)
Llama 3.1 8B	30,000	22,659	75.53
GPT 3.5 Turbo	8,000	6,500	81.25
Gemini Flash 2.0	6,800	5,625	82.72
Mixtral 8x7B	32,171	25,187	78.27

Table 4: Comparison of different language models in the data augmentation pipeline. The table presents the total number of augmented samples, the number of filtered samples that passed quality checks, and the overall success rate (%) for each model.

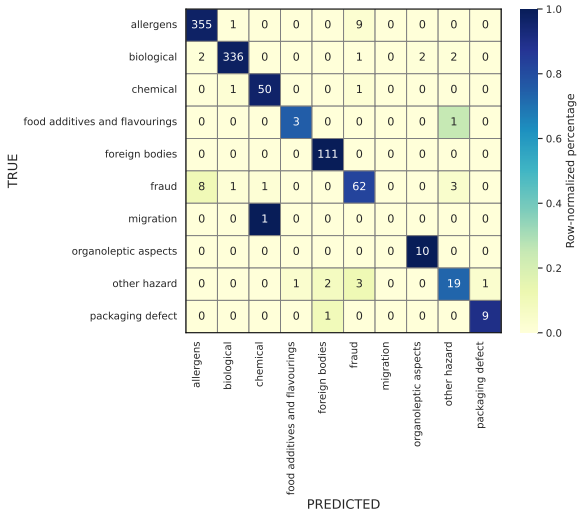

Figure 3: Confusion matrix for **hazard-category**. Each cell indicates the instances where the predicted label (columns) matches the true label (rows), with color intensity representing the row-normalized percentage.

For example, Llama 3.1 8B generated 30,000 samples with 22,659 passing quality checks, achieving a success rate of 75.53%. Similarly, Mixtral 8x7B produced 32,171 augmented samples with 25,187 filtered samples, resulting in a success rate of 78.27%. Smaller models like GPT 3.5 Turbo and Gemini Flash 2.0 generated fewer samples but achieved higher success rates of 81.25% and 82.72%, respectively.

These results highlight a trade-off between scale and filtering efficiency in augmentation. Larger models like Llama and Mixtral excel at generating high-volume data but have slightly lower success rates due to their broader scope. On the other hand, smaller models such as GPT and Gemini produce fewer samples but maintain higher precision during filtering. This balance between quantity and quality is crucial for optimizing data augmentation pipelines.

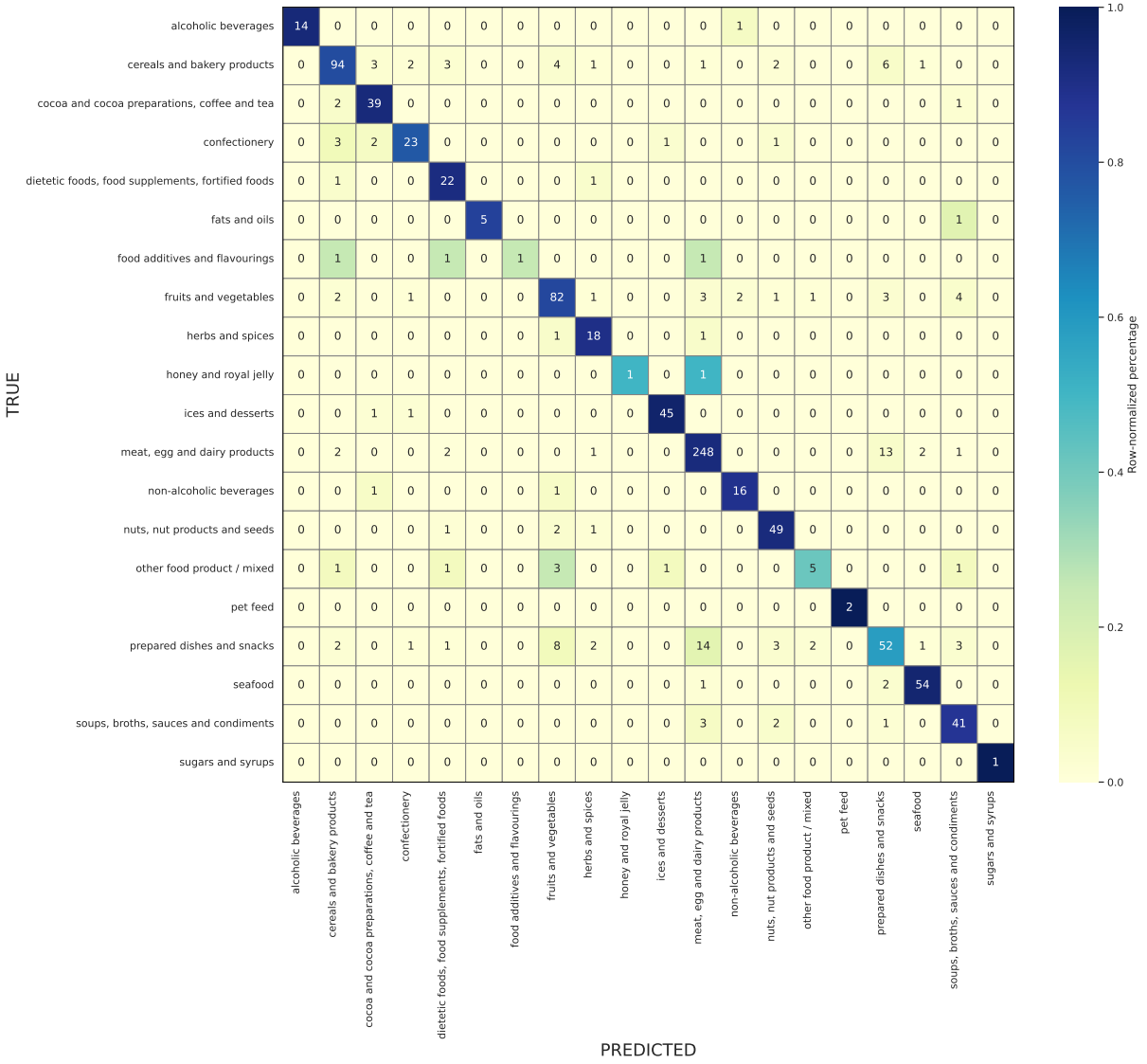

Figure 4: Confusion matrix for **product-category**. Each cell indicates the instances where the predicted label (columns) matches the true label (rows), with color intensity representing the row-normalized percentage.

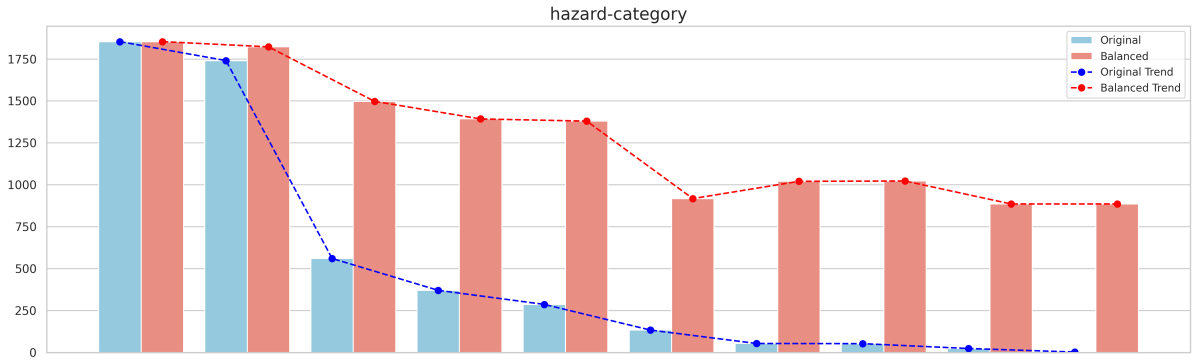

Figure 5: Comparison of **hazard-category** distributions before and after balancing.

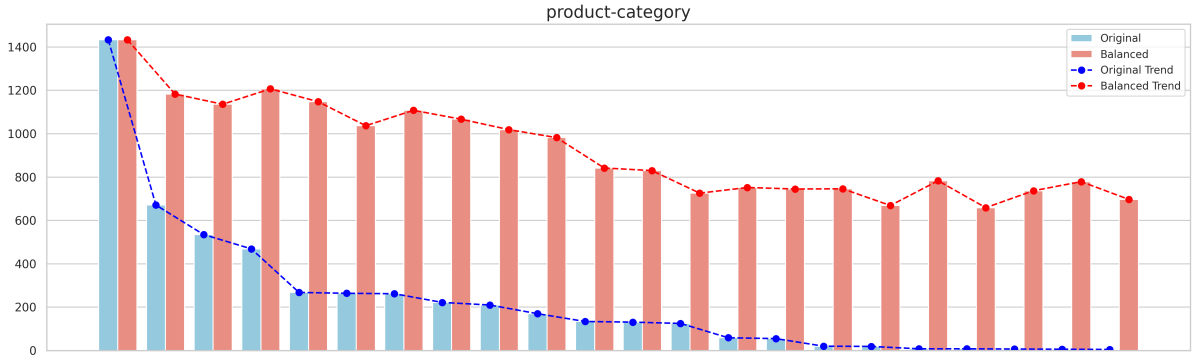

Figure 6: Comparison of **product-category** distributions before and after balancing.

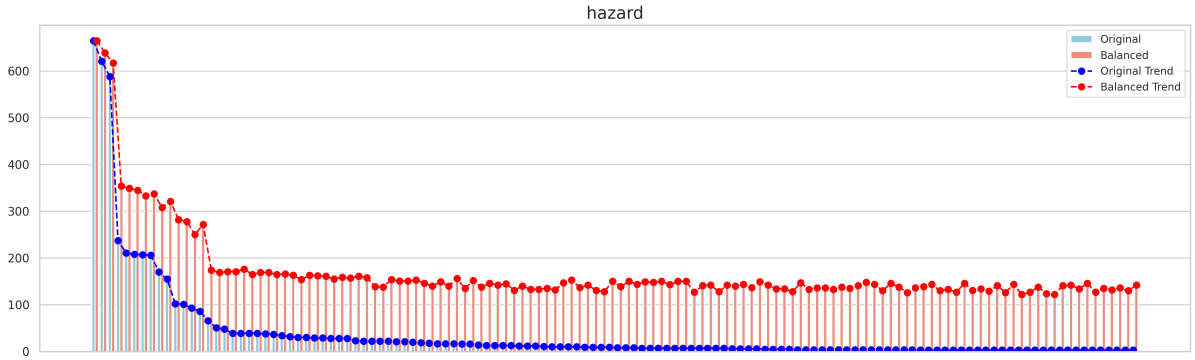

Figure 7: Comparison of **hazard** distributions before and after balancing.

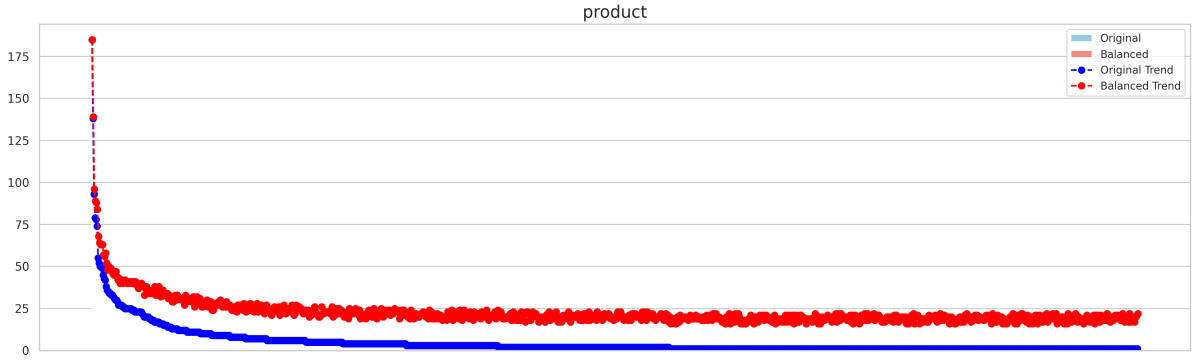

Figure 8: Comparison of **product** distributions before and after balancing.