ABCD at SemEval-2025 Task 9: BERT-based and Generation-based models combine with advanced weighted majority soft voting strategy

Le Duc Tai[1,2], Dang Van Thin[1,2],
[1]University of Information Technology-VNUHCM, Ho Chi Minh City, Vietnam
[2]Vietnam National University, Ho Chi Minh City, Vietnam
23521374@gm.uit.edu.vn, thindv@uit.edu.vn

Abstract

This paper illustrates our ABCD team system approach in ACL 2025 - SemEval-2025 Task 9: The Food Hazard Detection Challenge, aim to solving both Task 1: Text classification for food hazard prediction, predicting the type of hazard and product, and Task 2: Food hazard and product "vector" detection, predicting the exact hazard and product. Precisely, we received a food report and our system needed to automatically detect which category of hazard and product the food belonged to. However, in Task 2, we must classify the food report into the exact name of the food hazard and category. To tackle Task 1, we implement and investigate various solutions, including (1) experimenting with a large battery of BERT-based models; and (2) utilizing generation-based models, and (3) taking advantage of a custom ensemble learning method. In addition, to address Task 2, we make use of different data augmentation techniques like synonym replacement and back-translation. To enhance the context of input, we cleaned some special characters that bring more clarity into text input. Our best official results on Task 1 and Task 2 are 0.786 and 0.458 in terms of F1-score, respectively—finally, our team solution achieved top 8th in task 1 and top 10th in task 2.

1 Introduction

SemEval-2025 Task 9: The Food Hazard Detection Challenge (Randl et al., 2025) The Food Hazard Detection task evaluates explainable classification systems for titles of food-incident reports collected from the web. These algorithms may help automated crawlers find and extract food issues from web sources like social media in the future. Due to the potentially high economic impact, transparency is crucial for this task. Two sub-tasks were proposed for participants in this shared task. The first challenge is called "Text classification for food hazard prediction, predicting the type of hazard and product", in the first task the participants are required to develop a system that can classify food reports into 10 hazard categories and 22 product categories. Task 2, this task bears some resemblance to the first task, yet participants need to classify the text input into the exact vector of 128 hazards and 1068 products.

In today's interconnected world, where information flows ceaselessly across digital platforms, ensuring food safety remains a paramount concern. The ability to quickly and accurately detect food hazards from textual data is not only advantageous but imperative. Natural Language Processing (NLP), with its ability to parse through large amounts of text, plays a pivotal role in this endeavor. Using computational linguistics and machine learning techniques, NLP equips us with the tools to sift through diverse sources of textual information from social media posts to product reviews to identify potential food hazards efficiently. As a result, in this paper, we present our solutions for both Task 1 and Task 2 in SemEval-2025 Task 9: The Food Hazard Detection Challenge (Randl et al., 2025). Specifically, we employ three different approaches to address this task: (1) experimenting with a large battery of BERT-based models, (2) utilizing generation-based models, (3) taking advantage of a custom ensemble learning method.

2 Related Works

Food risk has been a major issue that poses a wide range of dangers for human health throughout history. In recent years, many researchers have taken action to tackle food danger by taking advantage of machine learning and computational force in order to predict early sight of food risk. for example, (Ma and Zheng, 2025) propose an integrated framework for classifying and analyzing food hazards by leveraging social media data from Sina Weibo. Ma and Zheng's (Ma and Zheng, 2025)

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 785–790

framework not only provides a robust method for classifying food hazard-related sentiments but also offers valuable insights for crisis management and policy formulation by mapping how online public opinion evolves during food safety emergencies. Specifically, they have shown the BERT-TextCNN model demonstrates exceptional performance in distinguishing between positive and negative sentiments, effectively capturing subtle emotional nuances in the context of the food hazard incident, and the BERTopic model successfully uncovers stage-specific topics and shows how public discourse evolves over time, offering insights into the thematic shifts during the incident. Besides that, network analysis highlights the pivotal role of certain nodes in information dissemination, confirming that both official media and influential individual users significantly impact public sentiment.

The work by (van den Bulk et al., 2022) explores the use of machine learning models to automate the classification of literature in systematic reviews on food hazards. The aim is to reduce the expert's workload while maintaining high accuracy in selecting relevant studies. Best-Performing Model: An ensemble of Naive Bayes and the Support Vector Machine (NB + SVM) achieved the highest overall performance. The study demonstrates that machine learning, particularly ensemble models (NB + SVM), can effectively support experts in systematic reviews of food hazards. The approach significantly reduces the manual screening effort without compromising quality, making it a valuable tool for food safety research.

3 Task Description

The Food Hazard Detection task focuses on developing interpretable classification models for categorizing titles of food-incident reports sourced from the web. These models have the potential to enhance automated web crawlers in identifying and extracting food-related risks from online platforms, including social media. Given the significant economic implications, ensuring model transparency is a key priority in this task. SemEval-2025 includes 2 sub-tasks, which are Task 1: Text classification for food hazard prediction, predicting the type of hazard and product, and Task 2: Food hazard and product "vector" detection, predicting the exact hazard and product.

3.1 Task 1: Text classification for food hazard prediction, predicting the type of hazard and product

The objective of the task is to classify food incident reports by predicting two categorical labels, "product-category" and "hazard-category" along with their corresponding entity vectors, "product" and "hazard." The dataset exhibits a significant class imbalance, with 22 product categories (e.g., meat, egg, and dairy products, cereals and bakery products, fruits and vegetables) and 10 hazard categories defining different types of food-related risks.

3.2 Task 2: Food hazard and product "vector" detection, predicting the exact hazard and product

The task focuses on the prediction of two key entity vectors: "product" and "hazard", which are extracted from food incident reports. The dataset presents a high level of granularity, encompassing 1,142 distinct product types, such as ice cream, chicken-based products, and cakes. Similarly, the hazard vector consists of 128 unique hazard types, including microbiological contaminants like "Salmonella" and "Listeria monocytogenes" as well as allergenic substances such as milk and products therefore.

3.3 Dataset Description

The dataset for this task comprises 6,644 short texts, with character lengths ranging from a minimum of 5 to a maximum of 277 and an average length of 88 characters. These texts are manually labeled food recall titles collected from official food regulatory agencies, such as the FDA. Each entry has been annotated by two domain experts specializing in food science or food technology to ensure high-quality labeling.

4 Methodology

In this section, we present our approaches for Task 1 and Task 2 in SemEval-2025 shared tasks in detail.

4.1 Data Processing

4.1.1 Data Cleaning

Before making our first approach to this task, we investigated the dataset text input, and we saw that the data contained a moderate number of noises, for instance, special characters, unnecessary white

spaces, and HTML tags. Therefore, our team decided to take some pre-processing stages:

- **Removing special characters**: the data text input contained some special characters such as $%#&ˆ and especially hyphens character which may cause some drawbacks when developing our system.
- **Removing HTML tags**: after observing the dataset, our team recognized that a great deal of HTML tags exist in text input, and this could be significant noise that can decrease the efficiency of our solutions to tackle this task.
- **Removing line break**: we consider removing line break or newline characters as noises because this appears too much in the dataset and has no positive effect on the data.
- **Text Expansion:** we also perform text expansion in English, for example: "*I'll*" into "*I will*" or "*he'd*" into "*he would*". Text expansion was utilized for data consistency, and this can help the model to generalize better.

4.1.2 Data Augmentation

The data distribution in the training dataset witnessed a significant imbalance between hazard and product labels. To be more precise, we can take hazard category labels as an example. "allergens", and "biological" labels have 1854 and 1741 records, respectively. While "food additives and flavorings", and "migration" only have 24 and 3 samples, which can be considered as very small in comparison to "allergens", and "biological" labels. Consequently, our team attempted to address unbalanced data by utilizing two data augmentation techniques, which are Back-Translation and Synonyms Replacement.

Back-translation involves translating a given text (typically from a high-resource source language) into a pivot language (often a different language with high-quality translation models) and then translating it back into the original language. This process introduces natural linguistic variations while preserving the semantic integrity of the original text. The goal is to generate revised versions of the original sentences, which can serve as additional training data to improve the robustness of the model. Our team takes advantage of the Google translator framework to perform the Back-translation method. To be more precise, we first take the whole text input, then translate it into French, and finally, the input is translated back into English.

Synonym Replacement is a data augmentation technique in Natural Language Processing that involves substituting words in a given text with their synonyms while preserving the overall semantic meaning. The primary objective is to introduce lexical variations in the training data, thereby enhancing the robustness and generalization of the model. Our process typically starts with the tokenization step, which tokenises input into individual words or subwords. After that, words suitable for replacement are identified. Typically, stop words, named entities, or domain-specific terms are excluded to avoid loss of meaning. Next, synonyms for selected words are retrieved from the lexical databases, which is WordNet from the NLTK corpus. Finally, A subset of the identified words is randomly replaced with their synonyms.

4.2 BERT-based Models Approach

Instead of experimenting with a classic machine learning method like the baseline code provided by the organizer, our team decided to take advantage of the deep learning power of BERT-based models. BERT-based models offer substantial advantages for food risk classification due to their ability to comprehend nuanced language semantics and context. Unlike traditional machine learning approaches that rely on keyword matching or shallow syntactic features, BERT excels in capturing intricate relationships within textual data. This capability is crucial in the domain of food risk classification, where understanding the subtleties of risk-related language is paramount. BERT-based models represent a significant improvement in food risk classification by leveraging their deep contextual understanding, bidirectional processing, and comprehensive language representation. These capabilities enable them to outperform traditional methods, offering more accurate and reliable assessments of food safety risks based on textual data. Our team has fine-tuned four models with different sizes.

- FacebookAI/roberta (Liu et al., 2019)
- FacebookAI/xlm-roberta (Conneau et al., 2019)
- answerdotai/ModernBERT (Warner et al., 2024)

Model	Token length	F1-score
roberta-large	512	**0.821**
deberta-v3-large	512	0.802
ModernBERT-large	512	0.789
xlm-roberta-large	512	0.785
deberta-v3-large	256	0.782
roberta-large	256	0.724
ModernBERT-large	256	0.663
xlm-roberta-large	256	0.657

Table 1: The experimental results of BERT-based classification approach on the validation set Task 1.

- microsoft/deberta-v3 (He et al., 2021)

We experiment on each model in different hyperparameter settings, and our team witnessed a significant improvement in results, which surpassed the baseline approach. Moreover, we only utilize "text" and "title" columns for input. As observed in Table 1, we can see that all our BERT results are much better than the baseline result (0.4965) in terms of the F1 score. Moreover, 2 models register more than 0.8 F1-score, which are *roberta-large* and *deberta-v3-large* in 512 token length. This is a great sign of improvement in our method. We first experimented on 256 token length due tothe limitation of GPU hardware resources, and after seeing a promising result, we only fine-tune models with 512 token length. Beside that, just after *modernBERT* was released, our team immediately utilised its new advantages in food risk classification tasks like this.

4.3 Generative-based Model Approach

In this approach, using a generative-based model, our team opted to experiment with the BART model (Lewis et al., 2020) by adapting it for a classification task through fine-tuning. BART functions as a denoising auto-encoder designed for pretraining sequence-to-sequence models. It is trained by intentionally introducing noise into text and then learning to reconstruct the original content.

Similar to the BERT-based approach, we used a tokenizer to tokenize the text inputs, which were then fed into BART. Moreover, we utilized the pretrained *facebook/bart-large* (Lewis et al., 2019). More specifically, we experiment with BART in both 512 and 1024 token lengths. As a result, the generative-based model achieved remarkable results compared to the BERT-based model, as shown in Table 2. Despite the fact that BART have a better performance than most of the BERT-based models and it has a longer token length, it did not surpass *roberta-large* result.

Model	Token length	F1-score
Weighted Voting	512-1024	**0.827**
roberta-large	512	0.823
bart-large	1024	0.821
bart-large	512	0.819
deberta-v3-large	512	0.802

Table 2: The experimental results of BART vs BERT-based vs Class weighted majority soft voting approach on the validation set Task 1.

4.4 Class weighted majority voting

Our last experiment is about an ensemble learning method, which is soft voting, yet we make some changes to make better performance. We can see the result in Table 2, Class-weighted voting techniques have a slight improvement in F1-score result which is **0.827**.

4.4.1 Step 1: Model Prediction Generation

Given an input sample, multiple independently trained classification models (e.g., Roberta, DeBERTa-V3, and BART) generate discrete class predictions. Each model assigns a single class label to the input based on its learned decision boundaries. Mathematically, for a given sample x_i, each model m produces a predicted label:

$$y_i^m \in C \tag{1}$$

where C represents the set of possible classes.

4.4.2 Step 2: Defining Class-Specific Weights

To account for differences in model reliability across categories, a set of predefined class-specific weights is introduced. These weights can be derived from various sources, such as the F1-score of each class from model evaluation, expert knowledge, or application-specific priorities. The weight function $w(c)$ assigns a weight to each class c, ensuring that classes of greater importance or higher reliability exert a stronger influence on the final decision.

$$w = \{c_1 : w_1, c_2 : w_2, \ldots, c_C : w_C\} \tag{2}$$

where w_c represents the assigned weight for class c.

4.4.3 Step 3: Weighted Vote Computation

For each sample, the class predictions from all models are collected, and a weighted voting mechanism is applied. Instead of counting votes equally, each vote is weighted by the corresponding class-specific weight. The weighted vote count for each class c is computed as follows:

$$V(c) = \sum_{m=1}^{M} 1(y_i^m = c) \cdot w(c) \tag{3}$$

where:

- M is the total number of models,
- $1(y_i^m = c)$ is an indicator function that returns 1 if model m predicts class c, and 0 otherwise,
- $w(c)$ is the predefined weight for class c.

4.4.4 Step 4: Final Prediction Selection

The final class prediction for the sample is determined by selecting the class with the highest weighted vote count:

$$\hat{y}_i = \arg\max_{c \in C} V(c) \tag{4}$$

This ensures that models' votes are not only considered in a majority rule fashion but are also adjusted based on class-specific importance.

5 Experimental Setup

We conducted our training process using HuggingFace (Wolf et al., 2020), and all BERT-based models were trained for 8 epochs. The AdamW optimizer was utilized to optimize the models. We selected a learning rate of 5e-5,4e-5 for BERT-based models. The batch sizes were set to 16 and 32, the random seed was set to 221, and the maximum token length was 512.

Cross-validation is a statistical resampling technique used to evaluate the generalization performance of models. Given the high dimensionality and complex structures of textual data, effective Cross-validation strategies are crucial to prevent overfitting, ensure robustness, and improve model reliability across unseen data. Given the imbalanced nature of the dataset, we employed the stratified K-fold cross-validation technique (Bates et al., 2023) with $K = 10$ to mitigate the effects of data imbalance on the models. Stratified cross-validation ensures that the class distribution remains consistent across folds, thereby reducing bias in performance estimation caused by unequal class distributions in random splits. This approach enables a more reliable evaluation of model performance across diverse subsets of the data.

Due to computational resource limitations, we had to adjust system settings for fine-tuning the BART model. Specifically, we reduced the batch size to 8 and employed gradient accumulation to effectively train on larger effective batch sizes. This technique allows us to accumulate gradients over multiple smaller batches before updating the optimizer, mitigating memory constraints. Furthermore, we utilized mixed precision training (FP16) and gradient checkpointing to accelerate training and reduce memory usage. Mixed precision training combines 16-bit and 32-bit floating-point operations, enabling efficient training of large-scale models like transformers. Dynamic loss scaling was employed to maintain numerical stability. Given GPU limitations, we trained BART for only 6 epochs and opted for the AdaFactor optimizer, known for its efficiency in training large models, instead of AdamW. All models were evaluated using the metric provided by the task organizers. Our team leveraged a P100 GPU, available for up to 30 free hours per week on Kaggle, for computational resources.

6 Main results

In the official final result released by the organizer, our team results in Task 1 and Task 2 are 0.786 and 0.458 in terms of F1-score, respectively. This result was achieved by using the class-weighted majority voting strategy, which combines BART, Roberta, and DeBERTa-V3 models. Moreover, in both tasks, our team also applied Back-translation and Synonyms replacement to augment the specific classes with fewer records to ease the negative effect of data imbalance. However, in Task 2, our team only leveraged a generative-based model classification approach, which is BART, to achieve a 0.458 F1-score. Task 2 has worse results since the imbalance between classes was too tremendous. Our team solution achieved top 8th in task 1 and top 10th in task 2.

7 Limitations

We think our greatest limitation is that our team can only leverage the "text" and "title" text features, but using other numerical or categorical features such as the date or country columns. This is also reduce the diversity and specificity for mod-

els to generalize the data better. In addition, our generative-based approach takes a great deal of time to train since the size of generative models is mostly larger than that of BERT-based models. The class weighted majority voting also needs as a much longer inference time, so we think it can not be used in a real-time application.

8 Conclusion and Future works

In this paper, we presented our approach for SemEval-2025 Task 9: The Food Hazard Detection Challenge. Our system leveraged BERT-based models, generative-based models, and an advanced class-weighted majority voting strategy to enhance classification performance. Through extensive experimentation, we demonstrated that combining multiple models with a weighted ensemble technique improves predictive accuracy. Our best results achieved F1-scores of 0.786 for Task 1 and 0.458 for Task 2, highlighting the effectiveness of our approach. For future work, we aim to explore additional features beyond textual data, such as metadata from food reports, to improve classification accuracy. We also plan to experiment with prompt-based learning using large language models (LLMs) and investigate efficient fine-tuning techniques to reduce computational costs.

Acknowledgements

This research was supported by The VNUHCM-University of Information Technology's Scientific Research Support Fund.

References

Stephen Bates, Trevor Hastie, and Robert Tibshirani. 2023. Cross-validation: What does it estimate and how well does it do it? *Journal of the American Statistical Association*, pages 1–12.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Biao Ma and Ruihan Zheng. 2025. Exploring food safety emergency incidents on sina weibo: Using text mining and sentiment evolution. *Journal of Food Protection*, 88(1):100418.

Korbinian Randl, John Pavlopoulos, Aron Henriksson, Tony Lindgren, and Juli Bakagianni. 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.

Leonieke M. van den Bulk, Yamine Bouzembrak, Anand Gavai, Ningjing Liu, Lukas J. van den Heuvel, and Hans J.P. Marvin. 2022. Automatic classification of literature in systematic reviews on food safety using machine learning. *Current Research in Food Science*, 5:84–95.

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. *Preprint*, arXiv:2412.13663.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.