CICL at SemEval-2025 Task 9: A Pilot Study on Different Machine Learning Models for Food Hazard Detection Challenge

Weiting Wang and **Wanzhao Zhang**
Computational Linguistics, University of Tübingen
{weiting.wang, wanzhao.zhang}@student.uni-tuebingen.de

Abstract

This paper describes our approaches to SemEval-2025 task 9, a multiclass classification task to detect food hazards and affected products, given food incident reports from web resources. The training data consists of the date of the incidents and the text of the incident reports, as well as the labels: "hazard-category" and "product-category" for task 1, "hazard" and "product" for task 2. We primarily focused on solving task 1 of this challenge. Our approach is in two directions: Firstly, we fine-tuned BERT-based models (BERT and ModernBERT); secondly, in addition to BERT-based models, linearSVC, random forest classifier, and LightGBM were also used to tackle the challenge. From the experiment, we have learned that BERT-based models outperformed the other models mentioned above, and applying focal loss to BERT-based models optimized their performance on imbalanced classification tasks.

1 Introduction

Food safety is one of the main concerns of consumers when making purchasing decisions. Therefore, a system that detects possible hazard-containing products and their corresponding hazards from past reports can help consumers identify certain possible hazards in food products more easily. SemEval-2025 task 9 (Randl et al., 2025) is a shared task focusing on food hazard detection, the participants are encouraged to design classification systems that detect hazards and the affected products from food safety incident reports (all texts were originally in or translated to English).[1] The challenge has two subtasks:

- Subtask 1: A text classification task to predict the category of hazards and products.
- Subtask 2: Predict the exact hazard and product.

Our group focuses primarily on subtask 1 of the challenge. To solve the task effectively, we propose our two-direction approach:[2]

Approach with pre-trained language models: Fine-tuning the pre-trained language models, namely BERT (Devlin et al., 2019) and ModernBERT (Warner et al., 2024) to adapt to pre-processed data. To minimize training cost, lightweight and free computational cost fine-tuning enhancements were used.

Approach without pre-trained language models: Training and tuning the hyperparameters of traditional machine learning models, namely linearSVC, random forest, and more recent desicion-making model, LightGBM (Shi et al., 2025); which are less time-consuming than fine-tuning BERT-based models. This method serves as a comparison to the first approach.

In addition to finding the most effective model, the way to perform data augmentation is also a challenging problem. First of all, the training data is heavily imbalanced. For example: class *food additives and flavourings* only has 24 entries while class *allergens* has 363 entries in label *hazard-category*. Therefore, for our approach with BERT-based models, we applied back-translation via MarianMT (Junczys-Dowmunt et al., 2018) framework to generate more training samples for long-tail data, improving generalization on minority classes. Besides, we replaced the standard Cross-Entropy Loss with Focal Loss (Lin et al., 2018), which dynamically reduced the influence of majority-class samples, allowing the model to learn better from under-represented categories. In addition to the imbalance of classes, the data are also non-specific and lack context. Therefore, we introduced keyword masking with contextual prompting, where key terms

[1] Datasets and the baseline models provided by the task organizers as well as the leaderboard can be found at: https://github.com/food-hazard-detection-semeval-2025/food-hazard-detection-semeval-2025.github.io

[2] Our codes are available at: https://github.com/cicl-iscl/SemEval25-Task9

were masked and replaced with the [MASK] token; as well as category-specific prompts to provide additional context, guiding the model's attention. In comparison to our system with BERT-based models, we applied simpler methods to the system without BERT-based models such as oversampling (random oversampling) to minimize the influence of the imbalanced classes in our training data, to clean up the data noise, we also applied a function to remove all punctuations and unnecessary whitespaces.

Through the approaches with BERT-based models, we discovered that the integration of focal loss with BERT effectively addresses class imbalance, which is consistent with the findings of previous research (Younes and Mathiak, 2022) on the handling of class imbalance in dataset mention detection. While keyword masking and contextual prompting showed the potential to improve the results. In contrast, neither back-translation for data augmentation nor our pre-processing efforts: including noise removal and utilizing SpaCy (Honnibal et al., 2020) for Named Entity Recognition yield the expected improvements. For our approach without BERT-based models, our attempts with oversampling methods did not achieve significant improvement.

2 Background

2.1 Dataset

The dataset provided by the task organizers for training consists of 6644 short texts (average length: 88 characters), including manually labeled food recall titles from official food agency websites (all texts are originally in English + translated into English). The dataset is divided into 3 subsets:

- Training set: The set consists of 5082 labeled food recall reports, each of them has 5 features (*year, month, day, country, title*), and labels *hazard-category, product-category* for subtask 1 and *hazard, product* for subtask 2 (Table 1). In addition, the full text of the recall is also provided in an additional column *text*, the participants are allowed to build their systems either on *title* or *text*.

- Validation set: 565 unlabeled food recall reports that has the same features and additional *text* column as the training set.

- Test set: 997 unlabeled food recall reports that have the same properties as the validation set.

Year	1999
Month	2
Day	24
Country	au
Title	Kooka's Country Cookies Choc Coated Assorted
Hazard-category	allergens
Product-category	cereals and bakery products
Hazard	peanuts and products thereof
Product	cookies

Table 1: A sample from the training set.

2.2 Related Works

Food hazard detection is currently underexplored, especially in its explainability (Randl et al., 2025). Despite the lack of research specifying food hazard detection and classification, previous research such as toxic spans detection (Pavlopoulos et al., 2022) and back translation (Beddiar et al., 2021) for detecting hate speech serve as inspiration for our systems. The toxic spans detection (Pavlopoulos et al., 2022) explored the possibility of fine-tuned BERT-based language model in detecting text toxicity as well as compared its performance to a BILSTM system; the results show that by fine-tuning BERT-based sequence labeling model only yields a result of F1 score 0.63; however, it still had better performance than the BILSTM classifier (F1 score 0.589). The other prior work that is mentioned above is back translation (Beddiar et al., 2021), it is a data augmentation technique where a sentence is translated into a target language and then back to the original language, lexical and syntactic variations are introduced while meaning is preserved. This approach has been shown to improve model robustness in imbalanced datasets. Therefore, we adopted the back translation method for our system with BERT-based models.[3]

3 System Overview

3.1 System with BERT-based models

We used the BERT baseline model provided by the task organizers and applied our strategies for our task due to limited cloud resources. With the release of ModernBERT during our training process, we also created our baseline and full strategies with

[3]We followed the approach outlined by DzLab: https://dzlab.github.io/dltips/en/pytorch/text-augmentation/

ModernBERT. However, due to cloud resource constraints, we were unable to leverage its 8192-token processing capability and instead limited the input length to 512 tokens. The following strategies were applied to our BERT and ModernBERT models:

Back Translation We employed back translation using the MarianMT framework to fight data imbalance with generated data. Specifically, we used the Helsinki-NLP/opus-mt-en-ROMANCE model as the encoder and Helsinki-NLP/opus-mt-ROMANCE-en[4] as the decoder to generate 68 additional samples via English → French → English and English → Spanish → English translation. An example of generating new samples with one original sample from the provided dataset using back translation is illustrated in Figure 1.

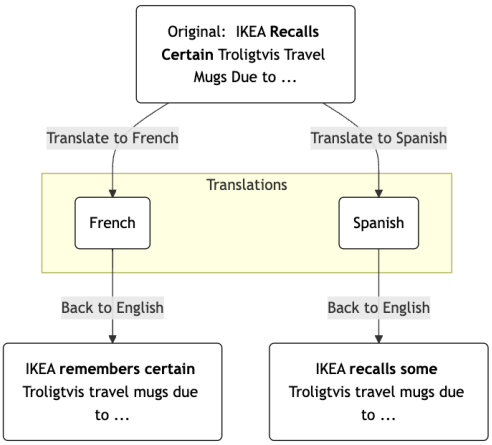

Figure 1: Back Translation Workflow

Focal Loss Focal loss is an extension of the standard cross-entropy criterion which has demonstrated strong performance in imbalanced classification tasks (Lin et al., 2018; Younes and Mathiak, 2022). It addresses class imbalance by down-weighting well-classified examples and focusing more on hard, misclassified samples. In our implementation,[5] we set $\gamma = 2$ based on the best performance in prior study (Lin et al., 2018).

Keyword Masking and Contextual Prompting Inspired by Masked Language Modeling (MLM) (Devlin et al., 2019), we adopted a masking strategy to replace task-relevant keywords in our training data. In our approach, words retaled to hazard (For example *hazard, risk*) and words related to product (For example *fruit, vegetables*) were replaced with the [MASK] token.[6] This guides the model to focus more on the context of 'hazard' or 'product' in order to improve prediction accuracy. Furthermore, when texts lack relevant keywords, we used contextual prompting by prepending task-specific prompts (For example, *Please pay attention to hazard-related content.*) to provide background information and improve classification performance.

3.2 System with Random Forest, LinearSVC and LightGBM

Compared to our approach with BERT-based models, we explored the potential of traditional machine learning models for complicated multiclass classification, namely RandomForestClassifier and LinearSVC from Scikit-learn (Pedregosa et al., 2011). In addition, we used LightGBM Classifier, which is an advanced decision tree-based system with superior performance and efficiency in multiclass classification tasks (Ke et al., 2017). Moreover, to tackle data imbalance, we applied the oversampling technique (random oversampling from imbalanced-learn (Lemaître et al., 2017)) to oversample minority classes.

4 Experimental Setup

Data Split During our training process, we split the training set into 80% training vs. 20% testing in cross-validation for all of our systems.

Data Preprocessing For training BERT-based models, we defined labels with fewer than 10 samples as minority classes, resulting in 34 underrepresented entries in total. Back translation was applied to the underrepresented entries and 2 new entries were generated from each category and added to the original training data. For training the other models, we applied a function to eliminate punctuation and multiple whitespaces and all training data was weighted by tf-idf.

Training Strategies Firstly, we used the BERT baseline provided by the task organizer as our baseline. Then the three strategies: back translation, focal loss, as well as keyword masking and contextual prompting were applied separately to the BERT model. Moreover, we also tested the performance of BERT with all three strategies. However, due to resource constraints, we were only able to create a baseline and an experiment with full strategies

[4]Model manuals are available at: https://huggingface.co/docs/transformers/model_doc/marian

[5]We adapted the PyTorch implementation of Focal Loss from Adeel Hassan's repository: https://github.com/AdeelH/pytorch-multi-class-focal-loss

[6]See full list in Appendix.

with ModernBERT. In addition, we created baselines with RandomForestClassifier, LinearSVC and LightBGM, oversampling method (random sampling) was applied to them. All models were trained only with the feature *title*, which consists of the titles of food safety incident reports.

5 Results

In this section, we present the performances of our systems with different settings in Macro F1 score during our validation process. Unfortunately, we were only able to upload our results from BERT with focal loss (0.6006) and LinearSVC (0.6079) to the organization leaderboard (rank #23).

5.1 Results of BERT-based models with different strategies

As shown in Table 2, the performance of BERT improved by changing the loss function from cross-entropy loss (0.669, baseline) to focal loss (0.751). However, back translation, as well as keyword masking and contextual prompting did not yield significant improvement. A possible reason is our restriction in resources. To identify minority classes, we originally suggested a dynamic system that identifies a category as a minority class if it contains fewer than $\max(2, 0.01 \times N)$ samples, where N is the total number of instances, which can define minority classes for our dataset in a more robust way. Nonetheless, applying this threshold resulted in a dataset that was too large for efficient storage. Furthermore, we are restricted to simple prompts because the large number of labels in our task makes direct task descriptions impractical due to length and complexity. The results of ModernBERT with and without full strategies are shown in Table 3,

Loss	Other Strategies	Macro F1
CE	None (Baseline)	0.669
CE	Back Translation	0.698
CE	Prompting & Masking	0.715
CE	Back Translation + Prompting & Masking	0.686
Focal	None	0.751
Focal	Back Translation	0.696
Focal	Prompting & Masking	0.717
Focal	Back Translation + Prompting & Masking	0.722

Table 2: Macro F1 scores for different experimental settings on BERT.

ModernBERT has a better baseline performance (0.702) than BERT (0.669), and the ModernBERT with full strategy produces the best result among all (0.808).

Settings	Macro F1
Baseline	0.702
Full Strategy	0.808

Table 3: Macro F1 scores for ModernBERT experiments.

5.2 Results of other models with oversampling

As shown in Table 4, LinearSVC classifier without oversampling has the best result (0.639) among all non-BERT-based models. Also none of these models had outperformed BERT-based models in validation process.

Model	Oversampling	Macro F1
RF	No (Baseline)	0.507
	Yes	0.566
SVC	No	0.639
	Yes	0.630
LGBM	No	0.498
	Yes	0.515

Table 4: Macro F1 scores of RandomForestClassifier, LinearSVC and LightGBMClassifier.

6 Conclusion

Our results suggest that the BERT-based models have better performance than other models, and we discovered that applying focal loss optimized the performance of BERT-based models on imbalanced classification task. However, the combination of BERT + focal loss has a lower score than LinearSVC in the final evaluation. A possible reason is that our BERT-based models lack generalizing ability while the test set may have a different class distribution and/or degree of noises than the training data. Besides, according to our result, back translation and keyword masking/prompting also showed some benefits but rather limited. Looking ahead, we see several promising directions for further research. One key improvement for model robustness could be the generation of higher-quality augmented data, ensuring that synthetic samples closely resemble real-world instances; in addition, if sufficient resources are provided, the ensemble method could be used to optimize the performance

of multiple BERT-based models. Another potential avenue is the transition from a single-task learning framework to multi-task learning, which could help the model generalize better across related tasks.

Limitations

Due to computational and methodological limitations, our models have not reached their full potential. First of all, training BERT-based models can be computationally demanding; however, our project fully relied on public computing resources, which limited the processing capability of our models. Besides, our synthetic samples are insufficient to significantly increase the robustness of our models. Last but not least, to produce results effectively with limited computing resources, we abandoned the approach of ensembling multiple BERT-based models, which could potentially improve model performance.

References

Djamila Romaissa Beddiar, Md Saroar Jahan, and Mourad Oussalah. 2021. Data expansion using back translation and paraphrasing for hate speech detection.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in c++.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3149–3157, Red Hook, NY, USA. Curran Associates Inc.

Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. 2017. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2018. Focal loss for dense object detection.

John Pavlopoulos, Leo Laugier, Alexandros Xenos, Jeffrey Sorensen, and Ion Androutsopoulos. 2022. From the detection of toxic spans in online discussions to the analysis of toxic-to-civil transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3721–3734, Dublin, Ireland. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

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.

Yu Shi, Guolin Ke, Damien Soukhavong, James Lamb, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu, Nikita Titov, and David Cortes. 2025. *lightgbm: Light Gradient Boosting Machine*. R package version 4.6.0.99.

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.

Yousef Younes and Brigitte Mathiak. 2022. Handling class imbalance when detecting dataset mentions with pre-trained language models. In *Proceedings of the 5th International Conference on Natural Language and Speech Processing (ICNLSP 2022)*, pages 79–88, Trento, Italy. Association for Computational Linguistics.

A Appendix: Masked Words

Category	Keywords
Hazard	hazard, risk, danger, safety, damage, issue, defect
Product	product, meat, fruit, vegetables, deserts, fat, sugar