TechSSN3 at SemEval-2025 Task 9: Food Hazard and Product Detection - Category Identification and Vector Prediction

Rajalakshmi S, Karpagavalli S, Karthikeyan S and **Krithika C**
Department of Computer Science and Engineering,
Sri Sivasubramaniya Nadar College of Engineering,
Chennai, Tamil Nadu, India

Abstract

Food safety is a critical global concern, and timely detection of food-related hazards is essential for public health and economic stability.The automated detection of food hazards from textual data can enhance food safety monitoring by enabling early identification of potential risks. In the Food Hazard Detection task, we address two key challenges: (ST1) food hazard-category and product-category classification and (ST2) food hazard and product vector detection. For ST1, we employ BertForSequenceClassification, leveraging its powerful contextual understanding for accurate food hazard classification. For ST2, we utilize a Random Forest Classifier, which effectively captures patterns in the extracted features for food hazard and product vector detection. This paper presents the results of the TechSSN3 team at the SemEval-2025 Food Hazard Detection Task, where we achieved a ranking of 21st in Task 1 and 19th in Task 2.

Keywords

t-SNE, TF-IDF, UHF, RFID, BERT, NLTK, Food hazards, Random Forest, NLP

1 Introduction

Food safety is a critical global concern, as contaminated food products can lead to widespread health risks, economic losses, and damage to consumer trust. Identifying food hazards early is essential for preventing outbreaks and ensuring regulatory compliance. Traditionally, food safety monitoring relies on manual inspection, regulatory reporting, and consumer complaints. However, these methods are slow, labor-intensive, and reactive rather than proactive. With the increasing availability of food-related incident reports on the web, there is an urgent need for automated systems that can detect food hazards from unstructured textual data.

The **SemEval-2025 Food Hazard Detection task** (Randl et al., 2025) aims to tackle these challenges by evaluating explainable classification models for food-incident reports. This task consists of two subtasks: (ST1) food hazard and product category classification, and (ST2) food hazard and product vector detection, which aims to identify the exact hazard-product associations. The task focuses on enhancing NLP-based hazard detection models for English-language reports, ensuring their effectiveness for regulatory and industrial applications.

We approach each subtask with methods best suited to their objectives. For ST1, we use **BertForSequenceClassification** to leverage contextual embeddings for accurate classification of food hazard and product categories. For ST2, a **Random Forest Classifier** is applied to engineered features to detect hazard-product associations, providing robustness and interpretability in relational modeling.

2 Related Work

The integration of machine learning (ML) with food hazard detection has been extensively explored, leveraging technologies such as spectroscopy, chromatography, mass spectrometry, and biosensors to identify potential contaminants. ML enhances the accuracy and efficiency of hazard detection by analyzing complex patterns in food composition, enabling real-time identification of chemical, biological, and physical hazards. Recent advancements include RFID (Radio-Frequency Identification)-based contamination sensing (Roberts, 2006), where ultra-high-frequency (UHF) RFID tags detect signal variations caused by contaminants, with ML models like XGBoost (Azmi and Baliga, 2020) achieving high accuracy. Additionally, ML-driven food hazard detection systems utilize cross-media data sources, including government reports, news, and social media, to identify emerging risks. Techniques such as semantic topic modeling and event detection further improve early warning systems. Despite these advancements, challenges persist in handling data variability, adapting models to detect novel hazards, and ensuring real-world applicability across different regulatory and geographical contexts. Moreover, integrating domain expertise with automated hazard detection remains crucial for refining model predictions and reducing false positives, ensuring the reliability of ML-based food safety monitoring systems.

3 Background

The **SemEval-2025 Food Hazard Detection task** focuses on extracting and classifying food safety incidents from textual data. The task is designed to improve the automated detection of food hazards in real-world reports, supporting early warning systems and regulatory monitoring. It is divided into two subtasks:

- (ST1) Food Hazard and Product Category Classification: Given a food-incident report, the system must classify it into one of several predefined food hazard and product categories
- (ST2) Food Hazard and Product Vector Prediction: The system must identify the exact food hazard and the associated food product from the text, providing structured outputs for fine-grained risk assessment

The SemEval-2025 Food Hazard Detection dataset provided for this task consists of 5,082 labeled samples for training, covering food hazard incidents in English. An additional 565 samples were provided as validation data, followed by 997 test samples for final evaluation. The dataset contains structured and unstructured data relevant to food safety incidents. It includes ‘year’, ‘month’, ‘day’, and ‘country’ for temporal and geographical context. The ‘title’ and ‘text’ describe incidents, serving as inputs for classification.The dataset features 10 unique hazard categories (e.g., biological, Chemical, foreign bodies) and 22 unique product categories (e.g., meat, egg and dairy products, prepared dishes and snacks, cereals and bakery products). Additionally, the hazard and product columns specify the exact contaminant (e.g., escherichia coli, listeria monocytogenes) and affected item (ground beef, hot dogs).

Figure 1 illustrates the frequency distribution of food hazard categories. Figure 2 depicts the

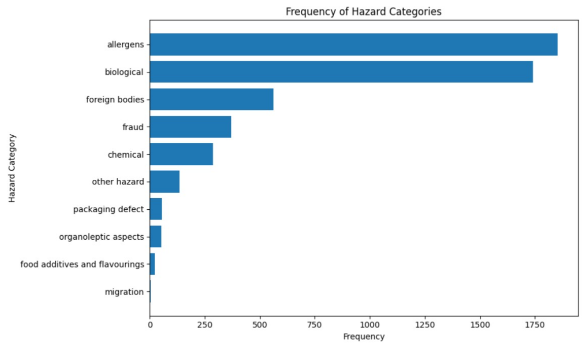

Figure 1: Frequency Distribution of Food Hazard Categories

frequency distribution of Food Product categories. Figure 3 shows a t-SNE (t-Distributed Stochastic

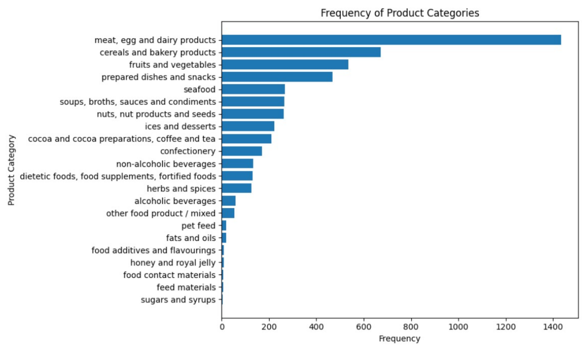

Figure 2: Frequency Distribution of Food Product Categories

Neighbor Embedding) visualization of tokenized inputs, where each point represents a data instance colored by hazard category.

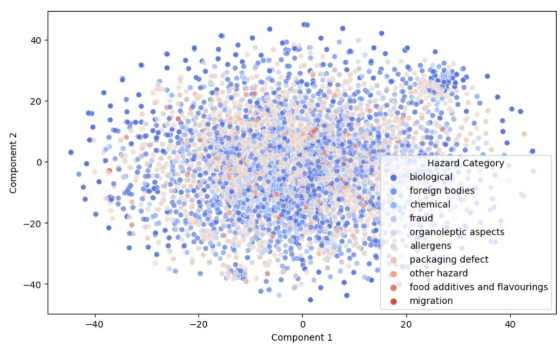

Figure 3: t-SNE Visualization of Tokenized Inputs

Our participation focused on both subtasks (ST1 and ST2) to develop a comprehensive system capable of handling hazard and product category classification and precise vector detection.

4 System Overview

This section outlines the approach taken for the SemEval-2025 Food Hazard Detection task, highlighting the key methodologies, models, and techniques used. It covers the data preprocessing, feature extraction, model selection, and training strategies used to optimize performance across both subtasks.

4.1 Subtask 1: Food hazard and product category detection, predicting the exact hazard-category and product-category.

1. **Data Preprocessing**
   Only the relevant columns were preserved, while the rest, including 'hazard', 'product', and 'title', were not used for the analysis. Categorical labels for 'hazard-category' and 'product-category' were encoded into numerical values using LabelEncoder. The 'text' data was then tokenized using a pre-trained BERT tokenizer (Koroteev, 2021), applying truncation and padding to ensure a consistent input length of 128 tokens.

2. **Feature Extraction**
   BERT Tokenization was performed using the BertTokenizer from the Hugging Face Transformers library, encoding text into input_ids and attention_mask (Clark et al., 2019) for efficient processing. To optimize classification, the dataset was split into two—one for hazard-category and another for product-category—enabling independent training and specialized learning for each task.

3. **Model Selection and Training**
   For classification, the BertForSequenceClassification (Face) model was utilized, leveraging its pre-trained transformer-based architecture. Two separate instances were initialized—one for hazard-category classification and another for product-category classification. Training was conducted with specific hyperparameters, including 10 epochs for hazard classification and 12 for product classification, a batch size of 16 for training and 64 for evaluation, and an epoch-wise evaluation strategy. To prevent overfitting (Ying, 2019), an EarlyStoppingCallback (Prechelt, 2002) was applied, terminating training if no improvement was observed within two epochs. The models were trained separately for each task, with results stored in designated directories for further evaluation.

4.2 Subtask 2: Food hazard and product "vector" detection, predicting the exact hazard and product.

1. **Data Preprocessing**
   The 'text' column was utilized to process textual features. Stopwords were removed, and the text was lowercased to ensure consistency. TF-IDF (Term Frequency-Inverse Document Frequency) (Ramos et al., 2003) was then applied to convert the processed text into numerical vectors, enabling effective feature extraction for classification models.

   For categorical features, labels such as 'product-category', 'hazard- category', 'hazard', and 'product' were transformed into numerical values using Label Encoding. This conversion ensured compatibility with machine learning models while preserving the categorical relationships necessary for accurate classification.

2. **Feature Extraction**
   We employ TF-IDF vectorization to convert raw 'text' into numerical features. This technique assigns importance scores to words based on how frequently they appear in a document while reducing the weight of commonly occurring words across all documents.

   The transformed text representation is then combined with categorical encodings of structured fields such as 'hazard-category', 'product-category'. This integration allows the model to leverage both textual and structured data for improved prediction accuracy.

3. **Model Selection and Training**
   We use Random Forest Classifiers (Salman et al., 2024) for both hazard and product prediction, leveraging their ensemble learning approach to construct multiple decision trees and aggregate outputs for improved accuracy and robustness. This reduces overfitting (Ying, 2019) and enhances generalization, making it suitable for food hazard detection. Two separate classifiers were trained: the Hazard Model, which predicts specific hazard types

(e.g. Salmonella, Listeria, Metal ContaminatioN), and the Product Model, which identifies affected food products (e.g., Dairy, Seafood, Beverages).

Each model utilizes 100 decision trees, balancing performance and computational efficiency, with hyperparameter tuning to address class imbalances and ensure accurate predictions for less frequent hazard and product categories.

5 Experimental Setup

The Experimental Setup section outlines the key tools, libraries, and evaluation strategies used to develop and assess our food hazard detection models. The Results subsection presents the performance outcomes, highlighting the impact of our approach on food hazard classification and product detection.

5.1 External Libraries & Tools

Several external libraries and tools were utilized for data preprocessing, model training, and evaluation. Pandas (v1.3.5) was used for data manipulation and handling missing values. Scikit-learn (v1.0.2) (Pedregosa et al., 2011) provided essential machine learning functionalities, including Random Forest classifiers, TF-IDF vectorization, and evaluation metrics. Additionally, NLTK (Natural Language Toolkit) (v3.6.7) (Bird et al., 2009) was employed for text preprocessing, such as tokenization and stopword removal.

For NLP-based modeling, the Hugging Face Transformers library (v4.17.0) (Face) was used to implement BertForSequenceClassification, enabling efficient fine-tuning of a pre-trained BERT model for predicting the hazard and product vectors.

5.2 Evaluation Metrics

To assess model performance, multiple evaluation metrics were used. The Macro F1-score (Opitz and Burst, 2019) was the primary metric, as it ensures a balanced evaluation across all hazard and product categories, even for underrepresented classes.

6 Results

Table 1 presents the results of individual runs for the Conception phase of Subtask 1, where the final test result achieved a score of 0.6442, demonstrating its effectiveness in classifying food hazards. We conducted multiple submissions to evaluate different modeling approaches for food hazard and product category classification.
Submission 1 utilized a Support Vector Machine (SVM) (Salcedo-Sanz et al., 2014) model. Submission 2 employed BertForSequenceClassification with both 'text' and 'title' (BERT-TT) as input features. Submission 3 also used BertForSequenceClassification but considered only 'text' (BERT1-T) as the input feature. Submission 4 experimented with BertForSequenceClassification while increasing the learning rate (BERT2-T), using 'text' as the sole input feature.

Table 1: Subtask 1 : Food hazard and product category detection

Submission	Macro F1-Score
Submission 1 (SVM)	0.6375
Submission 2 (BERT-TT)	0.7391
Submission 3 (BERT1-T)	0.7069
Submission 4 (BERT2-T)	0.6943

Table 2 presents the results of individual runs for the Conception phase of Subtask 2, where the final test result achieved a score of 0.2712, demonstrating its effectiveness in predicting food hazard and product vectors. Submission 1 utilized a Logistic Regression model (LR) (Maalouf, 2011). Submission 2 employed a Random Forest Classifier with 'title' as the input feature (RF-T). Submission 3 also used a Random Forest Classifier but incorporated multiple input features, including 'title,' 'product-category,' and 'hazard-category,' (RF-TPH) to enhance prediction performance.

Table 2: Subtask 2 : Food hazard and product vector prediction

Submission	Macro F1-Score
Submission 1 (LR)	0.0040
Submission 2 (RF-T)	0.0116
Submission 3 (RF-TPH)	0.0991

7 Conclusion

Our system effectively addressed the SemEval-2025 Food Hazard Detection task, achieving competitive results in both food hazard and product category classification and hazard-product vector detection. The model demonstrated strong performance in classifying food hazards, while the challenge of accurately associating hazards with products highlighted areas for improvement.

8 Future Work

Future work will focus on making the model more reliable and adaptable to different datasets for better food hazard detection. Incorporating semi-supervised or self-supervised learning techniques to leverage unlabeled data could improve performance in real-world scenarios with limited annotated samples. Additionally, expanding training data through synthetic data generation or data augmentation may help address class imbalances and enhance model adaptability. Integrating contextual embeddings from domain-specific corpora, such as food safety reports and scientific literature, can provide richer feature representations, enabling the model to capture intricate relationships between hazards and products.

9 Limitations

Our approach utilizes BERT-based models for classification but may face challenges in capturing intricate contextual relationships, particularly when dealing with implicit references or specialized terminology. The computational demands of fine-tuning BERT and training Random Forest on large datasets can be significant, making real-time deployment in low-resource settings difficult. Additionally, while the dataset is well-structured for this task, real-world food safety reports often contain noisy, ambiguous, or incomplete information. The model's ability to generalize to such unstructured or multilingual data remains an area for future exploration.

References

Syeda Sarah Azmi and Shwetha Baliga. 2020. An overview of boosting decision tree algorithms utilizing adaboost and xgboost boosting strategies. *Int. Res. J. Eng. Technol*, 7(5):6867–6870.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Hugging Face. Hugging face transformers documentation (nd).

Mikhail V Koroteev. 2021. Bert: a review of applications in natural language processing and understanding. *arXiv preprint arXiv:2103.11943*.

Maher Maalouf. 2011. Logistic regression in data analysis: an overview. *International Journal of Data Analysis Techniques and Strategies*, 3(3):281–299.

Juri Opitz and Sebastian Burst. 2019. Macro f1 and macro f1. *arXiv preprint arXiv:1911.03347*.

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

Lutz Prechelt. 2002. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer.

Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

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.

Chris M Roberts. 2006. Radio frequency identification (rfid). *Computers & security*, 25(1):18–26.

Sancho Salcedo-Sanz, José Luis Rojo-Álvarez, Manel Martínez-Ramón, and Gustavo Camps-Valls. 2014. Support vector machines in engineering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 4(3):234–267.

Hasan Ahmed Salman, Ali Kalakech, and Amani Steiti. 2024. Random forest algorithm overview. *Babylonian Journal of Machine Learning*, 2024:69–79.

Xue Ying. 2019. An overview of overfitting and its solutions. In *Journal of physics: Conference series*, volume 1168, page 022022. IOP Publishing.