

FEAST: Retrieval-Augmented Multi-Hierarchical Food Classification for the FoodEx2 System

Lorenzo Molfetta^{a,1}, Alessio Cocchieri^{a,1}, Stefano Fantazzini^{a,1}, Giacomo Frisoni^{a,1}, Luca Ragazzi^{a,1} and Gianluca Moro^{a,*,1}

^aDepartment of Computer Science and Engineering, University of Bologna, Cesena Campus, Via dell'Università 50, I-47522 Cesena, Italy

Abstract. Hierarchical text classification (HTC) and extreme multi-label classification (XML) tasks face compounded challenges from complex label interdependencies, data sparsity, and extreme output dimensions. These challenges are exemplified in the European Food Safety Authority's FoodEx2 system—a standardized food classification framework essential for food consumption monitoring and contaminant exposure assessment across Europe. FoodEx2 coding transforms natural language food descriptions into a set of codes from multiple standardized hierarchies, but faces implementation barriers due to its complex structure. Given a food description (e.g., “organic yogurt”), the system identifies its base term (“yogurt”), all the applicable facet categories (e.g., “production method”), and then, every relevant facet descriptors to each category (e.g., “organic production”). While existing models perform adequately on well-balanced and semantically dense hierarchies, no work has been applied on the practical constraints imposed by the FoodEx2 system. The limited literature addressing such real-world scenarios further compounds these challenges. We propose **FEAST** (Food Embedding And Semantic Taxonomy), a novel retrieval-augmented framework that decomposes FoodEx2 classification into a three-stage approach: (1) base term identification, (2) multi-label facet prediction, and (3) facet descriptor assignment. By leveraging the system's hierarchical structure to guide training and performing deep metric learning, **FEAST** learns discriminative embeddings that mitigate data sparsity and improve generalization on rare and fine-grained labels. Evaluated on the multilingual FoodEx2 benchmark, **FEAST** outperforms the prior European's CNN baseline F1 scores by 12–38% on rare classes.

1 Introduction

Accurately mapping natural language (NL) descriptions to sparse hierarchical taxonomic structures in critical domains such as food safety systems overcomes the boundaries of a mere multi-label text classification task and transcends academic curiosity; it underpins effective public health measures, enforces regulatory rigor, and enables population-scale dietary analysis. In this context, Hierarchical Text Classification (HTC) and eXtreme Multi-Label Text Classification (XML) methods are employed to address specific, and often intertwined challenges. While HTC is designed to exploit the set of labels H organized according to a predefined hierarchical structure where the labels are connected in a parent-child relationship [3, 14, 22],

Input: Organic yoghurt, cow milk, semi-skimmed, with corn flakes, oats, and raspberries in a glass cup.

Base Term
A02NH: Yoghurt, cow milk, flavoured

(F04) Ingredients
A00DD: Processed maize-based flakes
A00DJ: Rolled oats, instant
A01EP: Raspberries (red and yellow)

(F10) Qualitative Info
A077G: Semi-skimmed

(F18) Packaging Format
A07NT: Cup/pot

(F19) Packaging Material
A07PF: Glass

(F21) Production Method
A07SE: Organic production

Output: A02NH#F04.A00DD#F04.A00DJ#F04.A01EP#F10.A077G#F18.A07NT#F19.A07PF#F21.A07SE

Figure 1: Illustration of the FoodEx2 coding system. An input food item is represented by a base term and facet descriptors detailing its key attributes, such as ingredients, processing, and packaging.

XML is designed to manage the computational challenges inherent in datasets with a vast number of labels, aiming to identify a relevant subset of labels for each instance efficiently [31, 33]. Despite their differences, both paradigms are integral to the real-world classification systems, where their combined strengths enable effective solutions.

With an increasing interest in efficient and sustainable AI [24], recent advances in HTC have primarily been driven by pre-trained transformers and graph-based techniques that integrate label semantics and hierarchical information into end-to-end models [6, 28]. Global architectures, which directly encode the entire label hierarchy in a single classifier, have demonstrated superior performance compared to local methods that deploy multiple individual classifiers for different parts of the hierarchy [35]. However, most of these models have been evaluated on well-balanced and semantically dense hierarchies. Concerning XML, end-to-end trainable models that leverage pre-trained transformers have shown superior performance [15]. Models such as XR-Transformer [33] and CascadeXML [17] create hierarchical label trees through clustering algorithms to improve scalability, while AttentionXML [31] employs multi-label attention mechanisms to incorporate contextual information and better predict rare labels.

In this paper, we examine FoodEx2, a food categorization system developed by the European Food Safety Authority (EFSA), which poses a far more realistic setting for HTC and XML research. Originally designed to support tasks in food safety, nutrition, and contaminant exposure assessment, FoodEx2 encodes food items using

* Corresponding Author. Email: gianluca.moro@unibo.it.

¹ Equal contribution (co-first authors).

a two-component structure: a unique base term and a series of facet categories with associated descriptors reporting key attributes of the food—as exemplified in Figure 1. Unlike existing HTC and XML applications, FoodEx2 features interconnected ontological structures, where NL food descriptions must be mapped to codes belonging to multiple hierarchies containing thousands of elements. Therefore, FoodEx2’s food items simultaneously belong to various categorization dimensions (e.g., qualitative information, packaging information, production methods) across different hierarchical levels. Furthermore, the sparse nature of the FoodEx2 label space raises several obstacles: (1) the uneven distribution of descriptors across facet categories requires models to handle extreme cases of class imbalance; (2) the facet codes must be derived solely from the NL input—despite the presence of extensive implicit facet information in the FoodEx2 taxonomy—accentuating the necessity for architectures that can effectively distill and exploit hierarchical, semantic, and contextual clues; (3) aligning the multi-faceted code generation with a structured taxonomy necessitates a system that not only classifies text accurately but also respects hierarchical dependencies during prediction.

To address these points, we introduce **FEAST** (Food Embedding And Semantic Taxonomy), a novel framework that redefines the FoodEx2 classification problem as an HTC XML one. By decomposing the problem into sub-tasks—base term identification, facet category determination, and facet descriptor classification—our solution effectively integrates hierarchical information across these elements while maintaining high classification accuracy. Built on state-of-the-art encoder networks and large language models (LLMs) leveraging information from a hierarchical taxonomy, to the best of our knowledge, **FEAST** represents the top-performing framework for food classification within the FoodEx2 system.

2 Related Work

Hierarchical Text Classification HTC methods generally fall into two main categories: local, which build separate classifiers at each level of the hierarchy, and global, which treat HTC as multi-label classification by integrating hierarchical structure directly into a single model [3, 35]. Modern HTC publications leverage transformer encoders such as BERT and RoBERTa to set strong baselines [4], with distributed coordination mechanisms proving effective for managing complex hierarchical structures at scale [21, 23]. Exploiting semantic features through hybrid learning frameworks has shown significant improvements in text classification tasks [10]. Extensions like HiAGM [35] embed hierarchy via Tree-LSTM or Graph Convolutional Networks (GCNs) to propagate label information, while HTCInfoMax [7] tackles label imbalance by enhancing representations of rare classes. HiMatch [6] further aligns document and label semantics using GCNs with hierarchy-aware matching losses. Encoder-decoder models recast HTC as a generation task, with Seq2Tree [32] employing depth-first linearization of label trees to capture dependencies.

Extreme Multi-Label Text Classification In XML, transformer-based methods have largely outperformed earlier bag-of-words and tree-based algorithms [4]. XR-Transformer [33] introduces a recursive architecture with increasing label resolutions, forming a hierarchical label tree through k-means clustering on label features, while CascadeXML [17] extends this by training a single transformer across multiple label resolutions, improving efficiency. These architectures scale to hundreds of thousands of labels by pruning low-confidence branches and leveraging the label hierarchy to constrain predictions. Traditional classifiers often lack flexibility when adding or modifying categories. However, procedures such as iterative refinement of

category representations [8] and transfer learning for cross-domain adaptation [9] have shown promise in addressing these limitations. Retrieval-augmented models address this issue by embedding queries and candidates into a shared semantic space [18]. Bi-encoder models generate dense embeddings for queries and candidates independently [16], while cross-encoders jointly encode query-label pairs for improved reranking at higher computational cost [18].

Applications to Food-Specific Domains Recent studies have addressed the classification of long texts in the food domain using hierarchical transformer-based models. [30] proposed a two-stage hierarchical transformer architecture to classify food safety news events. By chunking documents for BERT encoding, and then feeding embeddings into a transformer to capture inter-chunk context, this two-stage hierarchical design demonstrated improved accuracy in multi-class food safety event classification. Although this work classified news event categories rather than a predefined food ontology like FoodEx2, it illustrates a hierarchical modeling technique tailored for handling long food-related descriptions. [26] investigates multi-level grocery products classification via dynamic masking to enforce hierarchical consistency during prediction. Their technique ensures that when the model predicts the parent category, they mask out any sub-categories that do not belong to one of the parents. Their masking markedly improves hierarchical consistency and yields over 90% F1 at the deepest level, compared to flat classifiers.

In food science, [13] fine-tunes BERT to categorize packaged foods using ingredient lists and product names to assign category labels and predict nutrition quality scores. They showed that transformers surpass rule-based methods and highlight the utility of contextual embeddings for mapping free-text to structured food categories.

Concerning hierarchy-aware formulations, [25] addresses product categorization within the GS1 Global Product Classification system, which includes a food hierarchy similar to FoodEx2. By employing tree-based ensemble methods combined with a custom “Penalized Information Gain” metric that increases the weight of errors at higher hierarchy levels, they yielded a 12.7% improvement in accuracy at the deepest level (and +12.6% F1) compared to a flat classifier. Although their method used gradient boosting trees rather than neural networks, their results underscore the benefit of incorporating and respecting hierarchical structures during model training.

While no papers target FoodEx2 coding system, its requirements align with other works: NL text processing [30], shared contextual embedding spaces [13], and strict parent-child dependencies [25, 26].

3 Data Curation

The FoodEx2 dataset comprises 72,197 instances characterized by three features: the NL food description (`ENFOODNAME`), the FoodEx2 base term (`BASETERM_NAME`), and the target FoodEx2 code (`FACETS`). The dataset employs a two-component structure: *base terms* and *facets*. Since FoodEx2 does not provide the complete representation of base terms, facets, and auxiliary metadata, we used the latest available FoodEx2 Matrix catalog (`MTX_12`).

Base Term Primary food item identifier. Each description must be associated with precisely one base term (4,367 possible ones).

Facets Facets are categorized into explicit and implicit ones. Explicit facets are modifiers directly mentioned in the NL input, while implicit facets are intrinsic properties of the base term. There are a total of 28 facet categories, each one possessing a specific number of facet descriptors, ranging from as few as 2 (for facet `F26`) to as many as 19,744 (for facet `F01`).

Table 1: Coverage of base terms and facets in our dataset relative to those available in MTX_12, both before and after preprocessing.

Metric	Before Preprocessing		After Preprocessing	
	Coverage	%	Coverage	%
Base term class	1,726 / 4,367	39.52	1,650 / 4,367	37.78
Facet category	27 / 28	96.43	28 / 28	100.00
Facet descriptor	1,508 / 28,675	5.26	1,355 / 28,675	4.73

Code Format The encoding follows a specific syntax where the base term is followed (optionally) by “#” and a sequence of facet categories with related facet descriptors, each group separated by “\$”. Figure 1 shows an example used by the FoodEx2 system.

3.1 Preprocessing

We performed extensive preprocessing to ensure data consistency and reliability. To this end, we removed entries with missing values (198), duplicates (30,314), and food descriptions with anonymized information (4,536). For the remaining entries, we standardized inconsistent input formats, corrected malformed facet codes, resolved character encoding issues, and addressed cases where identical descriptions were mapped to multiple target codes by retaining only the most informative annotation. Starting from 72,197 entries, these steps reduced the dataset to 28,648.

3.2 Key Observations

The dataset’s class coverage before and after preprocessing can be seen in Table 1, while distributions for facet’s categories and descriptions are reported in Table 2. The FoodEx2 taxonomy presents several intricate characteristics that impact classification performance.

Multilingualism The dataset exhibits multilingual content, with many entries incorporating specialized culinary terms or food references from various languages (i.e., “*Rabbit, leg, cooked, preservation method undefined, intérieur cuit/à point*”).

Taxonomy Semantics We observed non-intuitive hierarchical relationships where semantically similar items often appear in distant positions within the taxonomy. For instance, “*Custard*” is positioned as the only base term at the same level as “*Starchy pudding*,” yet it is taxonomically distant from “*Rice pudding*” despite their conceptual similarities. This non-intuitive arrangement suggests successful classification models must capture deeper semantic relationships between food items rather than relying on surface-level pattern recognition.

Expert Knowledge Requirements Many target classes require mapping that is not obvious from surface-level names, demanding what we term “abstractive linkage”. For example, the input “*caramel peanut butter nougat roasted peanuts covered with milk chocolate*” is expected to be associated with the facet “*White sugar*” [A032J], a connection that requires domain-specific knowledge about food composition and ingredients. This highlights the need for solutions that can incorporate implicit knowledge and make inferential connections beyond the explicit content of food descriptions.

3.3 Data Splitting

To establish evaluation splits, we employed stratified sampling to maintain consistent class distributions across sets. Our test set contains 991 samples, while our validation set consists of 926 samples. To accelerate validation and minimize costs, we simulated a set of

Table 2: Distribution of facet categories and descriptors per instance after data preprocessing.

Split	Facet Categories			Facet Descriptors			Base Term-only
	Min	Avg	Max	Min	Avg	Max	
Train	0	1.75	9	0	2.23	17	2,796
Eval	0	1.59	6	0	2.08	16	213
Test	0	1.69	7	0	2.15	11	58

distractor candidate classes as if they were retrieved. Distractors are selected following the same technique described in Section 4.1. Their use was confined to the validation set, while the complete real pipeline, involving all components of our methodology, was limited to the test set. To assess generalization to unseen categories and reduce the risk of overfitting, we constructed an out-of-sample (OOS) evaluation set. This set was built by grouping samples based on their base term identifier and selecting a subset of groups to form the test portion, ensuring disjoint categories between training and test. Groups were added incrementally until the test set reached approximately 1,000 entries. The remaining groups were used as the training set. Unless otherwise specified, all experimental results reported in this paper were obtained using the OOS dataset.

4 Methodology

We outline the methodological foundations on which FEAST is built.

4.1 Hard Negative Mining

To improve the model’s ability to resolve fine-grained semantic distinctions, we developed hybrid data selection combining taxonomy-based and semantic similarity-based strategies, ensuring the model distinguishes fine-grained differences between closely related terms.

Taxonomy-Based Sampling Leveraging its hierarchical nature, we can represent the FoodEx2 ecosystem as a tree structure $\mathcal{T} = (V, E)$, where vertices correspond to unique terms and edges denote parent-child relationships. For each target term $t \in V$, we identified hard negatives through a two-phase process:

(1) *Candidate Selection*: we selected the set of nodes C_{pa} sharing the same parent $pa(v)$ as the target term t :

$$C_{pa}(t) = \{v \in V : pa(v) = pa(t), v \neq t\} \quad (1)$$

(2) *Hierarchical Structured Selection*: when insufficient candidates were available, we sampled additional nodes based on structural proximity and semantic overlap. Proximity was computed using a hop distance d_{hop} based on the Lowest Common Ancestor (LCA):

$$z = \text{LCA}(t, v) \\ d_{hop}(t, v) = |dp(t) - dp(z)| + |dp(v) - dp(z)| \quad (2)$$

with $dp(\cdot)$ being the node depth in the tree, with $dp(\text{ROOT}) = 0$.

Candidates are further ranked by semantic overlap based on the `ImplicitFacets` of the target node t and candidate node v . Letting $\mathcal{F}(v)$ be the set of implicit facets of the candidate node v , we define the semantic overlap as the number of shared implicit facets:

$$o_{\text{facets}}(t, v) = |\mathcal{F}(t) \cap \mathcal{F}(v)| \quad (3)$$

The resulting scoring function used to sample candidates is:

$$\mathcal{S}(t, v) = \frac{1 + \beta \cdot o(t, v)}{1 + d_{hop}(t, v)} \quad (4)$$

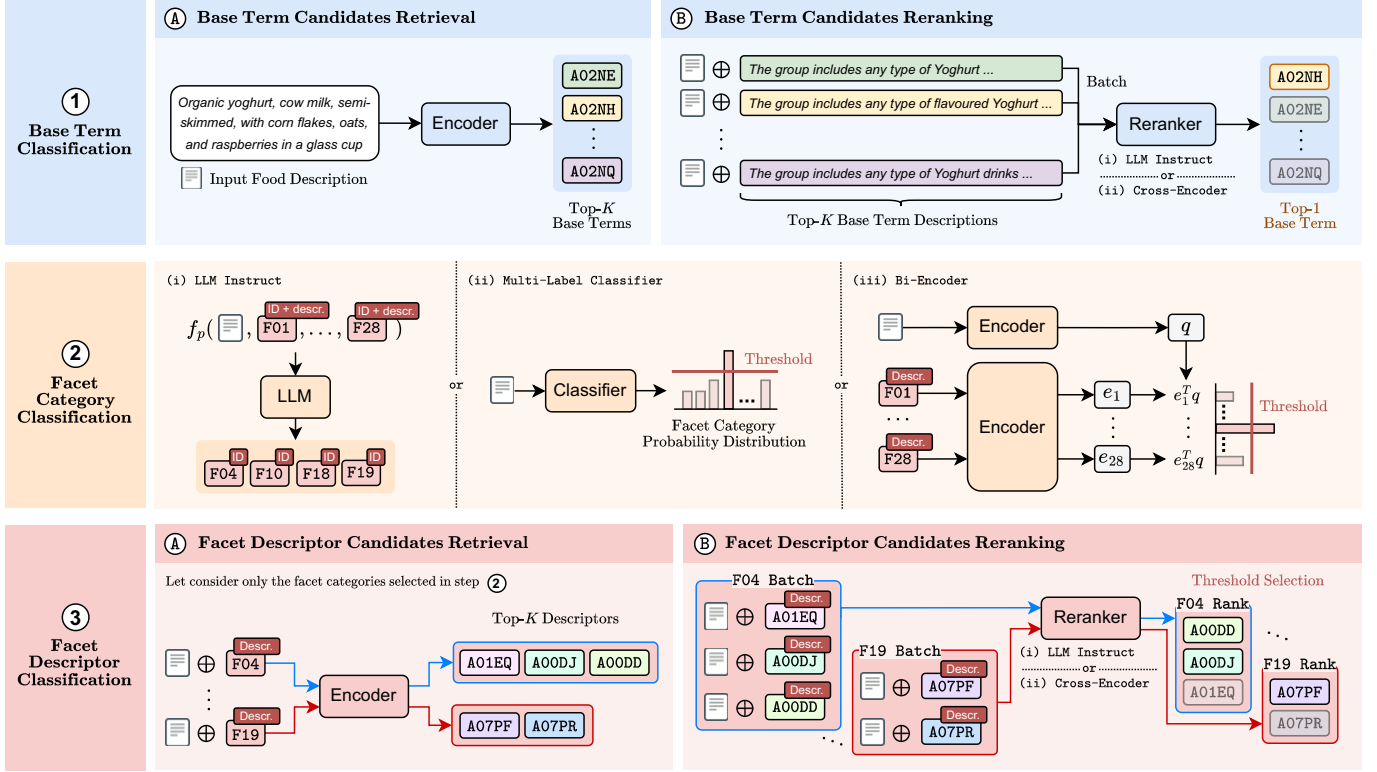


Figure 2: Overview of the proposed FoodEx2 food classification pipeline, which operates in three main stages: (1) **Base Term Classification**, where top- K candidate base terms are retrieved and reranked to produce a top-1 base term; (2) **Facet Category Classification**, which identifies relevant facet categories using either an LLM, a multi-label classifier, or a bi-encoder approach; and (3) **Facet Descriptor Classification**, which retrieves and reranks descriptor candidates for the previously selected facet categories. The system combines encoder-based retrieval with flexible reranking strategies (LLM instruct or cross-encoder) to map free-text food descriptions into structured FoodEx2 codes. \oplus denotes concatenation and $f_p(\cdot)$ represents a prompt template function.

Candidates are then sampled probabilistically according to their normalized scores as follows:

$$p(v|\mathbf{t}) = \frac{\mathcal{S}(\mathbf{t}, v)}{\sum_{u \in \mathcal{T}} \mathcal{S}(\mathbf{t}, u)} \quad (5)$$

Thus, given $\mathbf{N} = |\mathcal{H}_{\text{tax}}(\mathbf{t})|$, the set of taxonomy-based hard negatives, formally denoted as $\mathcal{H}_{\text{tax}}(\mathbf{t})$, is obtained as:

$$\mathcal{H}_{\text{tax}}(\mathbf{t}) = C_{pa}(\mathbf{t}) \cup \{v_i \sim p(v|\mathbf{t})\}_{\mathbf{N} - |C_{pa}(\mathbf{t})|} \quad (6)$$

4.2 Retrieval-Enhanced Classification

Traditional classifiers assign labels to inputs based on lexical features, often failing to capture semantic nuances such as synonymy, polysemy, or domain-specific terminology. To this end, we propose a retrieval-enhanced framework designed to map textual inputs into dense semantic representations.

Model Architecture For the retrieval process, we employ a bi-encoder architecture with shared parameters between the query and document encoding branches, following state-of-the-art practices [27]. A transformer encoder \mathcal{E} processes an input sequence x into contextualized token representations \mathbf{H} as follows:

$$\mathbf{H} = \mathcal{E}_\theta(x), \quad \mathbf{H} \in \mathbb{R}^{n \times h} \quad (7)$$

where n denotes the number of tokens in the input sequence, and h the hidden dimension of the encoder. Successively, a mean pooling operation generates a unified embedding.

Training Objective The training objective utilizes the Multiple Negative Ranking Loss (MNR) [12], which minimizes the semantic distance between query-positive document pairs while maximizing the distance to negative examples. For a batch \mathcal{B} of query-document pairs (q_i, d_i) , with the set of explicitly mined hard negatives $\mathcal{H}(q_i)$, the MNR Loss function is defined as follows:

$$\mathcal{L}_{\text{RETR}}(q, d; \theta, \psi) = -\frac{1}{\mathcal{B}} \sum_{i=1}^{\mathcal{B}} \left[\text{sim}(q_i, d_i) + \right. \\ \left. - \log \left(e^{\text{sim}(q_i, d_i)} + \sum_{d_j \in \mathcal{H}(q_i)} e^{\text{sim}(q_i, d_j)} \right) \right] \quad (8)$$

where cosine similarity is computed between the embeddings.

4.3 Entailment-Based Document Reranking

Although embedding-based retrieval reduces irrelevant candidates, subtle semantic differences can still lead to false positives among similar retrieved terms. We address this through a dedicated reranking module that reassesses the top-ranked candidates.

Given an input query q and the set of retrieved top- k document description $\mathcal{D}_k^{(q)} = \{d_1, d_2, \dots, d_k\}$, the reranking model \mathcal{R}_ϕ assigns each input-candidate pair a relevance score. Unlike the bi-encoder used for initial retrieval, we employ a cross-encoder that processes concatenated query-candidate pairs (q, d_i) through a transformer en-

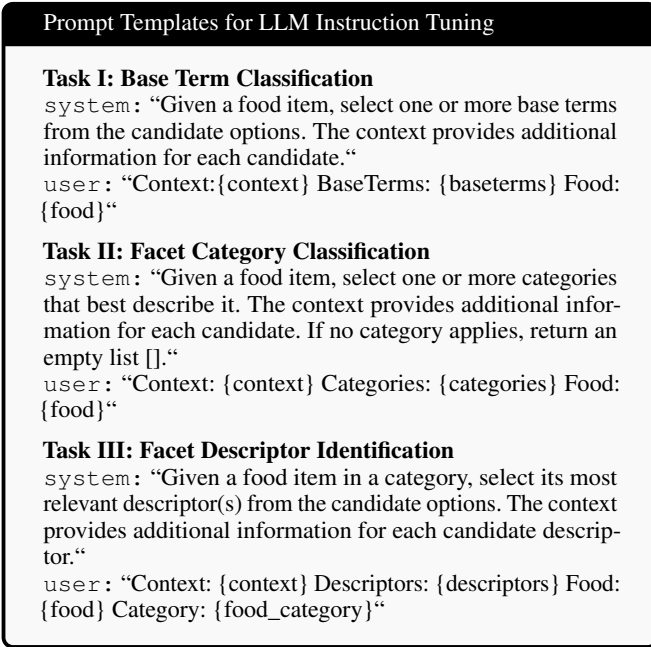


Figure 3: Prompt templates used for LLM’s multitask instruction tuning across base term, category, and descriptor classification tasks.

coder \mathcal{E} followed by a pooling layer \mathcal{P} and linear projection:

$$\rho_\phi(q, d_i) = \mathbf{w}^\top \cdot \mathcal{P} \circ \mathcal{E}([q; d_i]) + b \quad (9)$$

where \mathbf{w} and b are trainable parameters, and “;” is the concatenation operation. Training leverages hard negatives mined via the strategy in Section 4.1, using binary cross-entropy loss to distinguish positive pairs (aligned query-candidate pairs) from negatives. During inference, each candidate is independently assessed by the trained cross-encoder. We apply a confidence threshold τ to filter candidates further, retaining only those with sufficiently high relevance scores.

4.4 LLM for Classification

For the classification phase, we adopt a flexible approach that allows choosing between a lightweight encoder-only reranking model and a more expressive generative LLM. While encoder-only models are computationally efficient and fast, generative LLMs offer greater versatility through multi-instruction tuning, enabling them to handle diverse classification tasks under a unified model. This capability facilitates knowledge sharing across tasks, potentially improving generalization, though at a higher computational cost.

We fine-tune an LLM jointly on the three classification subtasks, framing classification as a sequence generation problem. Figure 3 illustrates the prompt templates used for instruction tuning, guiding the model to predict base terms, categories, or descriptors by leveraging NL context and candidate options.

5 FEAST

The proposed **FEAST** framework approaches the FoodEx2 classification task through a multi-phase prediction process. Specifically, we decompose the task into three sequential sub-tasks: base term classification, facet category classification, and facet descriptor identification. Figure 2 provides a graphical summary.

5.1 Task I - Base Term Classification

Given an input NL food description `inputText`, we identify its most relevant FoodEx2 base term through three modules.

Base Term Retriever We leverage base term descriptions as context to disambiguate relevant base terms from user input. We compute embedding representations of the food description and retrieve the top- k most similar base terms:

$$\mathcal{D}_k^{(q)} = \mathcal{M}(\text{inputText}) \quad (10)$$

Base Term Reranker To refine retrieval results and address subtle semantic differences, we implement a reranking model that assigns relevance scores to each candidate pair and filters results using a confidence threshold τ :

$$\bar{\mathcal{D}}_{\text{rank}}(\text{inputText}) = \mathcal{R}(\text{inputText}, \mathcal{D}_k^{(q)}) \quad (11)$$

LLM-Based Base Term Classifier Alternatively, we employ a fine-tuned LLM as a single-step text classifier, framing classification as a text-generation task guided by NL instructions.

5.2 Task II - Facet Category Classification

To identify applicable facet categories, we analyze three alternatives.

Multi-Label Classifier An encoder is fine-tuned for multi-label text classification. Given `inputText`, the model produces probability scores for each facet category $i \in \{1, \dots, N\}$, applying a threshold τ to determine binary assignments: $\hat{y}_i = 1$ if $\sigma(z_i) \geq \tau$ else 0. The model is trained by minimizing the Binary Cross-Entropy loss.

Bi-Encoder Classifier Two independent encoders generate embeddings for the concatenation $t_i = \text{inputText}_i \oplus \text{baseterm}_i$ and for each facet category description, respectively. We compute similarity scores between these embeddings, refine them through a multilayer perceptron, and apply a threshold to determine relevant categories.

LLM-Based Classifier Similar to the base term classifier, an LLM is prompted with the input description and the list of candidate facets. The model then generates the most relevant facet category.

5.3 Task III - Facet Descriptor Identification

We identify appropriate descriptors for each identified facet category.

Descriptor Retriever For each input text and associated facet category, we retrieve the top- k most relevant descriptors:

$$\mathcal{D}_k = \mathcal{M}(\text{inputText}, \text{descr}(c)) \quad (12)$$

Descriptor Reranker We refine the retrieved descriptors using a reranker that evaluates contextual relevance:

$$\bar{\mathcal{D}}_{\text{ranked}} = \mathcal{R}(\text{inputText}, \text{descr}(c), \mathcal{D}_k) \quad (13)$$

LLM-Based Descriptor Classifier As with previous tasks, we also implement a single-step generative approach where an LLM generates relevant descriptors for each category from a set of candidates.

Table 3: Retrieval and reranking scores (%). **Bold** and underlined values indicate the best and second-best result for each metric. **Blue** highlights the best performing retrieval model, while **orange** highlights the best performing reranker in each task averaging over all metrics.

Stage	Task	Model	Acc@1	Acc@3	Acc@5	Rec@1	Rec@3	Rec@5	NDCG@10	MRR@10	MAP@100
Retriever	Base Term (I)	GTE-Multilingual-base	95.10	99.71	99.89	94.84	99.60	99.87	98.01	97.33	97.30
		BGE-M3	95.95	<u>99.82</u>	99.93	<u>95.67</u>	<u>99.70</u>	<u>99.91</u>	<u>98.38</u>	<u>97.83</u>	<u>97.81</u>
		EuroBERT-610M	95.10	99.78	99.90	95.20	99.68	99.89	98.29	97.80	97.77
		ModernBERT	<u>96.57</u>	99.82	<u>99.92</u>	96.31	99.72	99.91	98.63	98.16	98.14
	Descriptor (III)	GTE-Multilingual-base	98.37	99.90	99.98	86.88	99.21	99.84	99.27	99.13	99.94
		BGE-M3	<u>98.76</u>	<u>99.92</u>	99.98	<u>87.21</u>	99.28	99.85	99.45	<u>99.33</u>	98.21
		EuroBERT-610M	98.60	99.91	99.97	87.05	99.30	99.84	99.45	99.31	99.28
		ModernBERT	98.90	99.93	<u>99.97</u>	87.32	99.40	99.87	99.53	99.41	<u>99.33</u>
Reranking	Base Term (I)	DeBERTa-v3-small	<u>81.01</u>	<u>89.69</u>	<u>90.00</u>	<u>89.81</u>	<u>90.00</u>	<u>99.72</u>	<u>85.12</u>	<u>86.38</u>	<u>85.12</u>
		DeBERTa-v3-base	76.01	84.69	85.00	87.06	89.93	98.80	90.12	81.38	80.12
	DeBERTa-v3-large	91.01	99.69	100.0	94.31	100.0	99.96	95.12	96.38	95.12	
	Descriptor (III)	DeBERTa-v3-small	<u>86.03</u>	<u>89.57</u>	<u>90.00</u>	<u>85.13</u>	89.74	90.82	87.80	88.17	87.35
		DeBERTa-v3-base	81.03	84.57	85.00	82.70	84.74	94.82	82.80	83.17	82.35
		DeBERTa-v3-large	96.03	99.57	100.0	96.70	99.74	99.82	97.80	98.17	97.35

6 Experimental Setup

6.1 Metrics

Our system evaluation employs two complementary metric categories. For retrieval performance, we utilize Accuracy@K, Recall@K, Precision@K, Mean Reciprocal Rank (MRR), and Mean Average Precision (MAP) to assess both the ranking quality and the system’s ability to surface relevant items effectively. For classification, we employ Precision, Recall, Support, and both Micro and Macro-averaged F1 scores to capture the system’s labeling accuracy and coverage across potentially imbalanced class distributions. These metrics are used to evaluate each sub-task component of our pipeline.

6.2 Hyperparameters

Below, we highlight the architecture and training configurations.

Task I For retrieval, we used the Sentence Transformer models with backbones including BGE-M3 [2], GTE-multilingual-base [34], ModernBERT-Large [29], and EuroBERT-610M [5], trained with a maximum sequence length of 64 and cosine similarity loss. The rerankers employed DeBERTa-v3 [11] (small/base/large) with a sequence length of 256 and binary cross-entropy loss.

Task II For classification, we used DeBERTa-v3 [11] (small/base) and RoBERTa [19] trained for 3 epochs, with 256 max sequence length, batch size 8, $2e-5$ learning rate, and linear decay.

Task III We again employed a retrieve-and-rerank approach following the same approach as in Task 1, but with sequence length increased to 96 and batch sizes appropriate to the different input characteristics.

Models We set the reproducibility random seed to 42 and used the AdamW optimizer with hyper-parameters $\beta_1=0.9$, $\beta_2=0.999$. For the LLM-based classifier, we fine-tuned LLaMA-3.1-8B [1] jointly on all three tasks using LoRA adapters with rank $r=32$ and $\alpha=32$. We set a max sequence length of 3072 and trained for 1 epoch with a batch size of 32, a learning rate of $2e-4$, and a cosine decay scheduler.

6.3 Hardware Configuration

Each run, except for the LLM fine-tuning, was performed on an internal workstation using a single Nvidia GeForce RTX3090 GPU with 24 GB of dedicated memory, 64 GB of RAM, and an Intel® Core™ i9-10900X (3.70GHz) CPU. For LLMs, we used an NVIDIA A100 with 80 GB of dedicated memory, 128 GB of RAM, and an Intel® Core™ i9-10900X (3.70GHz) CPU. The reference operating system is Ubuntu 20.04.3 LTS.

Table 4: Bi-encoder classifier results on facet category classification (Task II); b and s denote base and small size.

Model	Accuracy (%)				Recall (%)			
	@1	@3	@5	@10	@1	@3	@5	@10
RoBERTa-b	51.72	62.93	65.76	71.21	29.28	42.61	50.73	62.30
DeBERTa-v3-s	71.11	76.46	77.58	77.88	43.47	68.94	74.26	76.34
DeBERTa-v3-b	73.03	76.46	<u>77.37</u>	78.28	44.70	70.01	75.14	77.46

Table 5: Bi-encoder classifier with DeBERTa-v3-base performance metrics across different thresholds for both Macro and Micro averaging for Task II.

Threshold	Macro				Micro		
	Acc	Prec	Rec	F1	Prec	Rec	F1
0.4	72.93	57.57	50.62	53.05	89.52	86.52	88.00
0.35	73.13	57.00	52.58	54.00	88.25	87.30	87.77
0.3	71.72	55.53	53.55	53.81	86.06	88.02	87.03

7 Results

We evaluate our pipeline components across multiple tasks. Performance metrics for each module are presented below.

Retrievers Our fine-tuned ModernBERT retriever proves strong performance on both base term and descriptor retrieval tasks (see Table 3). In the Base Term task, it delivers 96.57% Acc@1, NDCG@10 of 98.63%, MRR@10 of 98.16%, and MAP@100 of 98.14%. For Descriptor retrieval, it achieves 98.90% Acc@1, NDCG@10 of 99.53%, MRR@10 of 99.41%, and MAP@100 of 99.33%. Across all intermediate thresholds (Acc@3/5 and Rec@1/3/5), ModernBERT consistently outperforms the competing models.

Rerankers As shown in Table 3, the reranker refines retrieval quality. For base terms, it achieves 91.01% accuracy@1, with NDCG@10 at 95.12% and MRR@10 at 96.38%. For facet descriptors, the reranker obtains an accuracy@1 of 96.03%, and similarly robust ranking metrics (NDCG@10 = 97.80%, MRR@10 = 98.17%). These results show the reranker’s capacity to enhance ranking precision.

Bi-Encoder Classifier Results are given in Table 4 and Table 5. The bi-encoder classifier shows moderate effectiveness with 73.03% accuracy@1. While accuracy improves at higher ranks (accuracy@3 = 76.46%, accuracy@5 = 77.37%, accuracy@10 = 78.28%), ranking quality remains a challenge. Precision@1 reaches 73.03% but declines at broader ranks (precision@10 = 16.58%), with recall@10 at 77.46%.

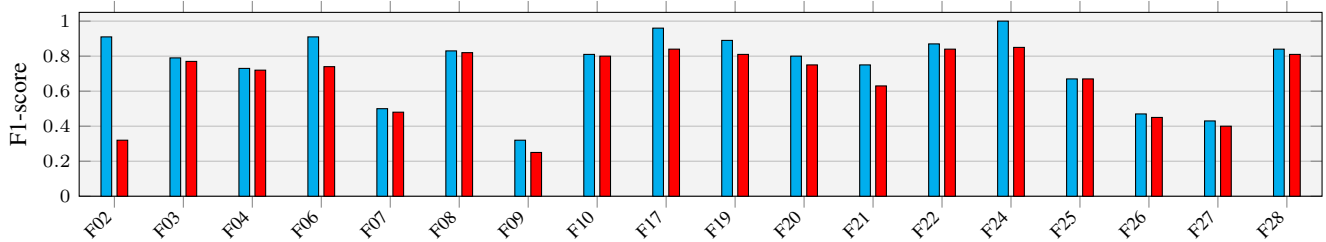


Figure 4: Comparison of F1-scores between LLaMA-3.1-8B [1] and DeBERTa-v3-small [11] across different facet categories (Task II).

Table 6: LLaMA-3.1-8B performance results across tasks on the corresponding validation sets. Exact Match is computed based on exact matches of the full set of labels in multi-label classification. Precision, recall, and F1-score are reported as micro-averages.

Task	Acc (%)	EM (%)	Prec (%)	Rec (%)	F1 (%)
Task I	99.99	98.75	99.59	98.78	99.18
Task II	96.39	55.65	83.25	74.83	78.82
Task III	99.99	99.54	100.00	99.64	99.82

Multi-Label Classifier Despite severe class imbalance and limited training data, our supervised classifier demonstrates robust performance while using only 142M parameters (see Figure 4). Even relatively rare classes (i.e., F06, F20, F21, F24) achieve high F1 scores, with only the most underrepresented classes showing lower F1 scores.

Large Language Model Table 6 reveals a performance disparity in our fine-tuned LLM, which excels at Tasks I and III but delivers suboptimal results for Task II. The model’s performance on several facet categories for Task II underperforms when compared to a simpler classifier model. This degradation occurs because facet category classification requires identifying multiple, potentially overlapping semantic dimensions simultaneously from a single food description, a challenge better addressed through the continuous semantic space of dense embeddings rather than discrete pattern matching in LLM prompts. In contrast, Tasks I and III involve more direct semantic matching that aligns well with the LLM’s contextual pattern recognition capabilities within structured prompts.

8 Discussion

Our experiments show that **FEAST** effectively tackles the FoodEx2 coding challenge by using retrieval-augmented, hierarchical modeling. The results confirm that combining deep metric learning with the FoodEx2 taxonomy helps overcome data sparsity and improves generalization to fine-grained labels. The current European standard for food classification relies on a collection of CNN networks [20]. These networks achieve average accuracy metrics of 83%, 87%, and 87% on the three tasks, respectively. Since our top-performing retriever model achieves a perfect 100% recall score in the @10 setting, we can use the reranker’s Acc@1 as our benchmark when providing the 10 retrieved elements as input. Under this assumption, **FEAST** achieves an accuracy of 91.01%, 96.39%, and 96.03% on the three tasks, respectively, thus standing as the state-of-the-art model in the food classification domain in the FoodEx2 domain.

Nevertheless, the limited coverage of the available FoodEx2 tender dataset constrains the number of available base terms and the diversity of facets. Class imbalance remains a challenge, with underrepresented facet categories exhibiting lower classification performance, even achieving a zero F1 score. Augmenting the training corpus through

targeted sampling, synthetic data generation (e.g., prompt-based descriptor synthesis), or hierarchical regularization could further enhance model robustness on low-support classes. Although our test metrics suggest strong potential, they derive from the reduced tender dataset, which may not fully reflect the real-world classification scenario of FoodEx2. The lack of standardized public benchmarks in this domain also limits comparative analysis with alternative approaches.

9 Conclusion and Future Work

We proposed **FEAST**, a retrieval-augmented, hierarchical framework for automated FoodEx2 coding that integrates dense retrieval, cross-encoder reranking, and multi-label classification. Beyond strong empirical results, **FEAST**’s modular design offers both interpretability—since retrieved candidates and reranking scores at each stage can be inspected by experts—and scalability, as the retrieval-based design allows seamless updates when new codes or facets emerge, minimizing the need for full-model retraining. Our evaluations demonstrate that **FEAST** significantly outperforms the previous European CNN-based baseline, particularly excelling in handling rare and fine-grained classes where existing models often struggle. By leveraging deep metric learning and hierarchical modeling, **FEAST** effectively addresses data sparsity and class imbalance—two major challenges inherent to the FoodEx2 taxonomy. Despite these advancements, the absence of a publicly standardized benchmark for FoodEx2 coding limits broad comparative analysis. Looking forward, we plan to develop an end-to-end pipeline that jointly optimizes base term identification, facet prediction, and descriptor assignment, improving consistency and harnessing inter-task dependencies. Further, incorporating data augmentation strategies and continual learning mechanisms could enhance **FEAST**’s robustness and adaptability to evolving taxonomies.

Ethical Statement

The raw dataset and model weights used in this research are currently not publicly available due to pending approval from the Tender’s contracting authority. We anticipate making these resources accessible to the research community once we receive the necessary clearance from the relevant authorities.

Acknowledgements

Research partially supported by: **AI-PACT** (CUP B47H22004450008, B47H22004460001); National Plan PNC-I.1 **DARE** (PNC0000002, CUP B53C22006450001); PNRR Extended Partnership **FAIR** (PE00000013, Spoke 8); 2024 Scientific Research and High Technology Program, project “**AI analysis for risk assessment of empty lymph nodes in endometrial cancer surgery**”, the Fondazione Cassa di Risparmio in Bologna; Chips JU **TRISTAN** (G.A. 101095947). LG Solution Srl for partially funding a PhD scholarship to L. Molfetta. Dinova Srl for partially funding a PhD scholarship to S. Fantazzini.

References

- [1] M. AI. Llama 3.1 - 8b. Hugging Face, July 2024. URL <https://huggingface.co/meta-llama/Llama-3.1-8B>. Released July 23, 2024.
- [2] BAAI. bge-m3. <https://huggingface.co/BAAI/bge-m3>, 2023. Sentence-transformer model designed for semantic similarity and related tasks.
- [3] S. Banerjee, C. Akkaya, et al. Hierarchical transfer learning for multi-label text classification. pages 6295–6300. ACL, 2019. doi: 10.18653/V1/P19-1633. URL <https://doi.org/10.18653/v1/p19-1633>.
- [4] N. Bertalis, P. Granse, et al. Hierarchical text classification (HTC) vs. extreme multilabel classification (XML): two sides of the same medal. *CoRR*, abs/2411.13687, 2024. doi: 10.48550/ARXIV.2411.13687. URL <https://doi.org/10.48550/arXiv.2411.13687>.
- [5] N. Boizard, H. Gisserot-Boukhlef, et al. Eurobert: Scaling multilingual encoders for european languages, 2025. URL <https://arxiv.org/abs/2503.05500>.
- [6] H. Chen, Q. Ma, et al. Hierarchy-aware label semantics matching network for hierarchical text classification. pages 4370–4379. ACL, 2021. doi: 10.18653/V1/2021.ACL-LONG.337. URL <https://doi.org/10.18653/v1/2021.acl-long.337>.
- [7] Z. Deng, H. Peng, et al. Htcinfomax: A global model for hierarchical text classification via information maximization. In K. Toutanova et al., editors, *NAACL-HLT 2021*, pages 3259–3265. ACL, 2021. doi: 10.18653/V1/2021.NAACL-MAIN.260. URL <https://doi.org/10.18653/v1/2021.naacl-main.260>.
- [8] G. Domeniconi, G. Moro, R. Pasolini, and C. Sartori. Cross-domain text classification through iterative refining of target categories representations. In A. L. N. Fred and J. Filipe, editors, *KDIR 2014 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, Rome, Italy, 21 - 24 October, 2014*, pages 31–42. SciTePress, 2014. doi: 10.5220/0005069400310042. URL <https://doi.org/10.5220/0005069400310042>.
- [9] G. Domeniconi, G. Moro, A. Pagliarini, and R. Pasolini. Markov chain based method for in-domain and cross-domain sentiment classification. In A. L. N. Fred, J. L. G. Dietz, D. Aveiro, K. Liu, and J. Filipe, editors, *KDIR 2015 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, part of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015), Volume 1, Lisbon, Portugal, November 12-14, 2015*, pages 127–137. SciTePress, 2015. doi: 10.5220/0005636001270137. URL <https://doi.org/10.5220/0005636001270137>.
- [10] G. Domeniconi, K. Semertzidis, V. López, E. M. Daly, S. Kotoulas, and G. Moro. A novel method for unsupervised and supervised conversational message thread detection. In C. Francalanci and M. Helfert, editors, *DATA 2016 - Proceedings of 5th International Conference on Data Management Technologies and Applications, Lisbon, Portugal, 24-26 July, 2016*, pages 43–54. SciTePress, 2016. doi: 10.5220/0006001100430054. URL <https://doi.org/10.5220/0006001100430054>.
- [11] P. He, X. Liu, et al. DeBERTa: decoding-enhanced bert with disentangled attention. OpenReview.net, 2021. URL <https://openreview.net/forum?id=XPZlaotutsD>.
- [12] M. L. Henderson, R. Al-Rfou, et al. Efficient natural language response suggestion for smart reply. *CoRR*, abs/1705.00652, 2017. URL <http://arxiv.org/abs/1705.00652>.
- [13] G. Hu, M. Ahmed, and M. R. L'Abbé. Natural language processing and machine learning approaches for food categorization and nutrition quality prediction compared with traditional methods. *The American Journal of Clinical Nutrition*, 117(3):553–563, 2023. ISSN 0002-9165. doi: <https://doi.org/10.1016/j.ajcnut.2022.11.022>. URL <https://www.sciencedirect.com/science/article/pii/S0002916522105526>.
- [14] V. Jain, M. Rungta, et al. Higen: Hierarchy-aware sequence generation for hierarchical text classification. pages 1354–1368. ACL, 2024. URL <https://aclanthology.org/2024.eacl-long.82>.
- [15] T. Jiang, D. Wang, et al. Lightxml: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification. In *AAAI 2021, 2021*, pages 7987–7994. AAAI Press, 2021. doi: 10.1609/AAAI.V35I9.16974. URL <https://doi.org/10.1609/aaai.v35i9.16974>.
- [16] V. Karpukhin, B. Oguz, et al. Dense passage retrieval for open-domain question answering. pages 6769–6781. ACL, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.550. URL <https://doi.org/10.18653/v1/2020.emnlp-main.550>.
- [17] S. Kharbanda, A. Banerjee, et al. Cascadexml: Rethinking transformers for end-to-end multi-resolution training in extreme multi-label classification. 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/0e0157ce5ea15831072be4744cbd5334-Abstract-Conference.html.
- [18] D. Li, Z. L. Zhu, et al. Enhancing extreme multi-label text classification: Addressing challenges in model, data, and evaluation. pages 313–321. ACL, 2023. doi: 10.18653/V1/2023.EMNLP-INDUSTRY.30. URL <https://doi.org/10.18653/v1/2023.emnlp-industry.30>.
- [19] Y. Liu, M. Ott, et al. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL <http://arxiv.org/abs/1907.11692>.
- [20] A. Livaniou. EFSA FoodEx2: Food Classification System & Smart Coding App (SCA). <https://www.bfr-akademie.de/media/wysiwyg/2024/gwk2024/efsa-foodex2-food-classification-system-and-smart-coding-app-sca.pdf>, May 2024. Backend Performance at Slide 20.
- [21] S. Lodi, G. Moro, and C. Sartori. Distributed data clustering in multi-dimensional peer-to-peer networks. In H. T. Shen and A. Bouguet-taya, editors, *Database Technologies 2010, Twenty-First Australasian Database Conference (ADC 2010), Brisbane, Australia, 18-22 January, 2010, Proceedings*, volume 104 of *CRPIT*, pages 171–178. Australian Computer Society, 2010. URL <http://portal.acm.org/citation.cfm?id=1862264&CFID=17470975&CFTOKEN=71845406>.
- [22] Y. Mao, J. Tian, et al. Hierarchical text classification with reinforced label assignment. pages 445–455. ACL, 2019. doi: 10.18653/V1/D19-1042. URL <https://doi.org/10.18653/v1/D19-1042>.
- [23] G. Moro and G. Monti. W-grid: A scalable and efficient self-organizing infrastructure for multi-dimensional data management, querying and routing in wireless data-centric sensor networks. *J. Netw. Comput. Appl.*, 35(4):1218–1234, 2012. doi: 10.1016/J.JNCA.2011.05.002. URL <https://doi.org/10.1016/j.jnca.2011.05.002>.
- [24] G. Moro, L. Ragazzi, and L. Valgimigli. Carburacy: Summarization models tuning and comparison in eco-sustainable regimes with a novel carbon-aware accuracy. In B. Williams, Y. Chen, and J. Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 14417–14425. AAAI Press, 2023. doi: 10.1609/AAAI.V37I12.26686. URL <https://doi.org/10.1609/aaai.v37i12.26686>.
- [25] O. Narushynska, A. Doroshenko, et al. Enhancing hierarchical classification in tree-based models using level-wise entropy adjustment. *Big Data and Cognitive Computing*, 9(3):65, 2025.
- [26] O. Ozyegen, H. Jahanshahi, et al. Classifying multi-level product categories using dynamic masking and transformer models. *Journal of Data, Information and Management*, 4:71–85, 04 2022. doi: 10.1007/s42488-022-00066-6.
- [27] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. pages 3980–3990. ACL, 2019. doi: 10.18653/V1/D19-1410. URL <https://doi.org/10.18653/v1/D19-1410>.
- [28] Z. Wang, P. Wang, et al. Incorporating hierarchy into text encoder: a contrastive learning approach for hierarchical text classification. pages 7109–7119. ACL, 2022. doi: 10.18653/V1/2022.ACL-LONG.491. URL <https://doi.org/10.18653/v1/2022.acl-long.491>.
- [29] B. Warner, A. Chaffin, et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *CoRR*, abs/2412.13663, 2024. doi: 10.48550/ARXIV.2412.13663. URL <https://doi.org/10.48550/arXiv.2412.13663>.
- [30] S. Xiong, W. Tian, et al. Food safety news events classification via a hierarchical transformer model. *Heliyon*, 9(7):e17806, 2023. ISSN 2405-8440. doi: <https://doi.org/10.1016/j.heliyon.2023.e17806>. URL <https://www.sciencedirect.com/science/article/pii/S2405844023050144>.
- [31] R. You, Z. Zhang, et al. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. pages 5812–5822, 2019. URL <https://proceedings.neurips.cc/paper/2019/hash/9e6a921fbc428b5638b3986c365d4f21-Abstract.html>.
- [32] C. Yu, Y. Shen, and Y. Mao. Constrained sequence-to-tree generation for hierarchical text classification. In E. Amigó et al., editors, *SIGIR '22, Madrid, Spain, 2022*, pages 1865–1869. ACM, 2022. doi: 10.1145/3477495.3531765. URL <https://doi.org/10.1145/3477495.3531765>.
- [33] J. Zhang, W. Chang, et al. Fast multi-resolution transformer finetuning for extreme multi-label text classification. pages 7267–7280, 2021. URL <https://proceedings.neurips.cc/paper/2021/hash/3bbca1d243b01b47c2bf42b29a8b265c-Abstract.html>.
- [34] X. Zhang, Y. Zhang, et al. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. pages 1393–1412, 2024.
- [35] J. Zhou, C. Ma, et al. Hierarchy-aware global model for hierarchical text classification. In D. Jurafsky et al., editors, *ACL 2020, Online, 2020*, pages 1106–1117. ACL, 2020. doi: 10.18653/V1/2020.ACL-MAIN.104. URL <https://doi.org/10.18653/v1/2020.acl-main.104>.