Comparator-based identification of food edibility from natural language description

Olga Cherednichenko[1,†], Maryna Vovk[2,†], Nataliia Sharonova[2,†], and Anzhelika Vorzhevitina[2,*,†]

[1] *Bratislava University of Economics and Management, Furdekova str. 16, Bratislava, Slovak Republic*

[2] *National Technical University "Kharkiv Polytechnic Institute", Kyrpychova str. 2, Kharkiv, 61002, Ukraine*

Abstract

Assessing the edibility of food based on consumer perception remains an underexplored yet practically significant challenge in food safety. This paper presents a novel framework for evaluating food suitability using natural language descriptions of sensory experiences, such as odor, appearance, and texture. By extracting structured features from unstructured, subjective input, our system leverages a comparator-based identification approach to infer missing attributes and assess overall edibility. The model aligns incomplete descriptions with prototypical instances from labeled data, enabling robust classification even under uncertainty. We demonstrate that this method can support nuanced, human-like judgments and serve as a foundation for intelligent decision-support tools in consumer and public health contexts. The proposed framework opens avenues for integrating qualitative perception with structured inference in critical application domains.

Keywords

natural language processing, food safety, comparator-based identification, feature modeling

1. Introduction

Ensuring the safety and suitability of food products for consumption is a persistent and globally relevant challenge. While traditional methods of food safety assessment rely on laboratory analysis, expiration labeling, and visual inspection, in everyday settings, consumers often base their decisions on informal sensory evaluations - descriptions of smell, texture, taste, and appearance. These evaluations are typically articulated in natural language and are inherently subjective, imprecise, and often incomplete. Yet, they represent a rich source of information that, if properly structured and interpreted, could inform intelligent systems capable of estimating food edibility.

This paper introduces a conceptual and technical framework for assessing food edibility based on free-form user descriptions. The approach rests on modeling these descriptions as partial observations of an underlying, structured feature space - comprising attributes such as odor type, surface texture, discoloration, moisture level, and taste anomalies. By employing techniques from natural language processing, key indicators are extracted and normalized into a set of interpretable features. To overcome the limitations of incomplete data, we propose a comparator-based identification method, which allows for the inference of missing attributes by aligning observed feature subsets with prototypical examples of known edibility status.

This method situates food products within a latent comparative space, where similarities to known spoiled or safe instances provide a probabilistic basis for prediction. Rather than relying

PhD Workshop on Artificial Intelligence in Computer Science at 9th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2025), May 15–16, 2025, Kharkiv, Ukraine

[*] Corresponding author.

[†] These authors contributed equally.

olga.cherednichenko@vsemba.sk (O. Cherednichenko); maryna.vovk@khpi.edu.ua (M. Vovk); nvsharonova@ukr.net (N. Sharonova); anzhelika.vorzhevitina.sgt.khpi@gmail.com (A. Vorzhevitina)

0000-0002-9391-5220 (O. Cherednichenko); 0000-0003-4119-5441 (M. Vovk); 0009-0004-9878-1761 (N. Sharonova) ; 0009-0001-0562-0191 (A. Vorzhevitina);

solely on absolute rules or fully observed inputs, the system can generalize from experience and provide nuanced judgments even in uncertain or borderline cases. The proposed framework not only addresses a practical consumer need but also contributes a novel perspective to the modeling of qualitative, perception-based descriptions in safety-critical domains.

The goal of this research is to introduce and evaluate a Comparator Based Identification framework that infers food edibility by analyzing free-text sensory descriptions and demonstrate its utility as an interpretable decision-support tool in food safety.

This research addresses the following key questions:

- How can logical rules for recognizing the edibility of food be formalized as predicate structures based on observable characteristics?
- How effective is the comparator model at classifying food according to their sensory characteristics compared to traditional ML models?
- How interpretable are comparator solutions?

By answering these questions, we aim to develop Comparator-Based Identification Framework including a mapping from subjective natural language inputs (odor, texture, color, etc.) to structured feature representations; designing a comparator mechanism and assessing similarity; and demonstrating the framework's utility via case studies.

The task of determining the food edibility based on external features is a typical binary classification task. Traditional machine learning methods solve it by training a classifier, but in the framework of the comparator identification method [1] the recognition process is formulated as identification by comparison. The idea is that a new or unknown food is identified not by direct determination of its species, but by comparison with already known samples of edible and dangerous ones. The method of comparator identification is based solely on the analysis of physically observable features of an object and the identification of patterns in the form of logical conditions.

In this paper, we consider mushrooms as an example of food to identify their edibility. We formalize the mushroom features as sensory descriptions, build a model based on pairwise comparisons of mushrooms, describe the comparator structure, present a logical scheme for identifying edibility without directly specifying the mushroom species, and show how the decisions of the comparator system relate to the binary classification task.

2. Review of related works

Recent advances in product classification increasingly leverage human perception and textual descriptions by integrating multi-modal learning, personalized recommendation systems, and graph-based representations. Multi-modal approaches, as demonstrated by combining textual [2] and visual features [3] through neural networks and fusion techniques, show superior performance by compensating for limitations within individual data modalities. Meanwhile, personalized recommendation systems employing models like BERT and nearest neighbor algorithms address individual preferences in e-commerce environments, enhancing user satisfaction while tackling scalability and cold-start challenges [4]. Simultaneously, text-attributed graphs and frameworks like P2TAG enable few-shot classification by fusing raw textual and structural graph information, significantly boosting classification accuracy [5]. Thus, these developments highlight the shift toward more adaptive, perception-driven classification models that capture nuanced human understanding of product categories.

Research combining food safety and natural language understanding is limited, with most NLP-for-food work focusing on structured inputs like labels and recipes to predict nutrition or categorize products [6, 7]. These approaches show high performance but assume clean, standardized data, not subjective or incomplete sensory descriptions. While food safety analytics use NLP for recall tracking or risk modeling [8], they rarely treat user-reported sensory input as core data for inference.

A closer parallel is mushroom edibility classification using structured features like odor and color [9], but these models rely on complete attribute sets and lack mechanisms for recovering missing data or aligning partial input with prototypes. Some works propose a comparator-based approach inspired by prototype/metric learning [10, 11, 12], comparative reasoning in NLP [13], and Bayesian Case Models [14], which support inference from partial, subjective text and generate interpretable, probabilistic edibility assessments.

Despite advances in large-scale language models (LLMs), their performance on Named Entity Recognition (NER) still lags behind supervised methods due to the intrinsic mismatch—NER is a sequence labeling task, while LLMs are optimized for generation. GPT-NER addresses this by reframing NER as a generation task using special entity-marking tokens, and incorporates a self-verification strategy to counter hallucinations common in LLMs [15]. Notably, GPT-NER performs comparably to supervised models across five benchmarks and excels in few-shot settings, highlighting its potential for real-world, low-resource applications. In parallel, NER plays a key role in processing domain-specific information such as aeronautical intelligence, where challenges include semantic ambiguity, data-sharing opacity, and lack of standardization. A recent survey explores how NER can support this domain, highlighting the roles of aviation-specific ontologies, knowledge systems, and thematic databases while identifying future research directions [16].

While many studies in NER focus on model architectures and training strategies, comprehensive evaluation across genres and entity types remains underexplored. One study conducts extensive testing on varied and adversarial test sets to assess the robustness of three state-of-the-art models, proposing improved reporting practices to better reflect real-world performance [17]. Another growing research area is nested NER, which addresses cases where entities overlap or are embedded within each other—issues that standard flat NER models often ignore. A review categorizes nested NER models (e.g., rule-based, hypergraph-based) and examines challenges such as error propagation and entity dependency, offering guidance for both researchers and practitioners [18]. To support multilingual NER development, the Universal NER (UNER) project presents gold-standard datasets across 12 languages with consistent annotations, facilitating cross-lingual research and providing publicly available baselines and tools [19].

While recent advancements in multi-modal learning, personalized recommendation systems, and graph-based methods have significantly improved product classification by incorporating human perception and textual descriptions, several challenges remain. First, accurately classifying products from subjective, incomplete, or noisy descriptions continues to be a critical issue, especially in domains like food safety where user sensory narratives are underutilized for inference. Second, Named Entity Recognition (NER), despite being a mature NLP task, still faces limitations in handling domain-specific data, nested structures, and cross-genre robustness, particularly when adapting large language models originally designed for generation tasks. Third, identifying significant indicators within product descriptions—especially from partial or sensory-based language—requires new methods that can infer missing attributes and align unstructured input with meaningful, interpretable prototypes. Addressing these issues will be key to developing robust, user-aware systems capable of understanding and classifying products in complex, real-world scenarios.

3. Materials and Methods

3.1. Food edibility feature extraction from natural language description

In real-world applications of food identification – particularly in unstructured settings such as foraging or home inspection – users often provide descriptions of food items in natural language, rather than structured categorical forms. To integrate such inputs into a comparator-based identification framework, it is necessary to extract structured features from textual descriptions. In our case, these features represent perceptual and contextual properties relevant to food
safety – such as color, shape, odor, texture, bruising, and presence of specific anatomical structures (e.g., gills or rings in mushrooms).

We approach this task as a rule-based information extraction problem, mapping linguistic cues to categorical variables $x_i \in X_i$, where each X_i is a finite set of permissible values for the i-th feature. For instance, the sentence "*The cap is flat and smooth with a brownish tint*" yields the feature assignments $x_{\text{cap\_shape}} = flat$, $x_{\text{cap\_surface}} = smooth$, and $x_{\text{cap\_color}} = brown$. Since user input may omit features, use synonyms, or express ambiguity, we adopt a tolerant matching procedure that:

- recognizes synonymous terms and paraphrases using a manually constructed mapping dictionary;
- allows partial filling of the feature vector $x = (x_1, x_2, \dots, x_n)$;
- defers decision-making in case of insufficient information.

Each natural-language description is thus transformed into a partial categorical vector in the comparator feature space, suitable for downstream metric-based comparison and classification. This allows flexible integration of free-form descriptions into a symbolic decision pipeline without requiring full supervision or structured data entry.

3.2. Comparator-based identification method

Comparator-based identification is a symbolic classification framework grounded in the notion of perceptual similarity between objects. Rather than relying on numerical features or statistical models, this method compares an unknown object's description with previously known instances using interpretable, feature-level match predicates. Let the perceptual description of a food item be represented as a vector $x = (x_1, x_2, \dots, x_n)$, where each $x_i \in X_i$ is a categorical value of the i-th attribute (e.g., shape, color, texture, odor). The space $X = X_1 \times X_2 \times \dots \times X_n$ forms a discrete feature space.

For any pair of objects x and y we define elementary comparators $P_i(x, y)$, which are binary predicates indicating whether the two objects agree on the i-th feature. The comparator similarity between two objects is defined as:

$$Eq(x, y) = \bigwedge_{i=1}^{n} P_i(x, y) \tag{1}$$

Classification is based on comparing an unknown object x to labeled reference objects from known classes (e.g., edible or inedible).

To improve both interpretability and decision reliability, we adopt a method for identifying significant (core) features – a subset of attributes that are most informative for distinguishing between classes. We base this step on the structural significance criterion proposed [1], which defines a feature i as significant if it contributes to class separation within the comparator framework.

Let E and P denote the sets of known safe and unsafe food items, respectively. Formally, feature i is considered structurally essential if:

$$\exists x \in E, y \in P \text{ such that } P_j(x, y) = 1 \ \forall j \neq i, \text{ but } P_i(x, y) = 0 \tag{2}$$

That is, there exist objects from opposite classes that differ only in the i -th feature, making it decisive for classification in at least one instance.

The core feature set $C \subseteq \{1, \dots, n\}$ is defined as the minimal set for which classification accuracy remains unchanged when only features from C are used for comparison:

$$\forall x \in X, \delta_C(x) = \delta(x), \tag{3}$$

where $\delta_C(x)$ is the comparator decision rule restricted to features in C. This reduction allows building simpler, explainable classifiers focused on perceptually relevant attributes.

By using such comparator-based principles and isolating core features, our method enables symbolic, transparent decision-making in safety-critical applications, such as identifying food edibility from natural language descriptions.

3.3. Comparator model of mushroom edibility

The UCI Mushroom Dataset [20] is a well-established benchmark for classification tasks involving categorical features. It contains 8,124 labeled observations of mushrooms belonging to the Agaricus and Lepiota families. Each observation is described by 22 nominal attributes that capture observable features, such as cap shape and color, gill attachment, bruising, odor, ring type, and habitat (see Figure 1). The target variable indicates whether the mushroom is edible or poisonous. Because all features are categorical and the classes are balanced, this dataset is used to evaluate models that handle symbolic data, feature selection strategies, and interpretable decision rules [21].

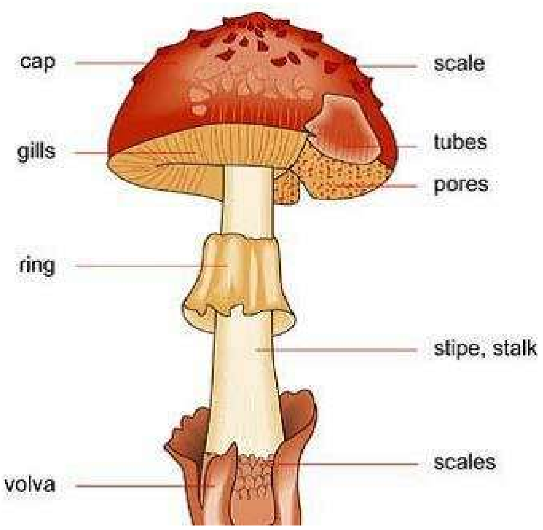

Figure 1: Diagram of a Mushroom. (Source [22]).

We have chosen this dataset for three main reasons. First, the categorical nature of the attributes aligns well with the logic-based framework of comparator identification, in which each feature is treated as a finite-valued variable and encoded via logical indicators. Second, the availability of ground truth class labels allows us to rigorously evaluate the performance of classification rules derived from comparator principles. Third, the dataset provides a natural context for demonstrating core feature extraction and decision strategies.

We will describe each mushroom by a set of observable features - a sensory description. In the UCI Mushroom dataset, each specimen is assigned such features as cap shape and color, cap surface, presence of spots ("bruises"), odor, characteristics of laminae (their attachment, distance, size, color), stem shape and structure (presence of a ring, its number and type, thickness/shape of the stem, color of the stem above and below the ring), color of spore powder, growing environment and population, etc. Formally, each mushroom x is matched with a vector of features:

$$x = (x_1, x_2, \dots, x_n), \tag{4}$$

where x_i is the value of the i-th feature. For each feature a finite set of allowed values of X_i is defined (e.g., cap color $x_1 \in X_1 = \{\text{red}, \text{brown}, \text{white}, \dots\}$. Thus, the space of mushroom descriptions can be represented as a Cartesian product of $X = X_1 \times X_2 \times \dots \times X_n$. This space is a vector feature space in terms of comparator identification [1]. The vector $x \epsilon X$ contains all available information about the mushroom obtained through the observer's "sensors": sight (color, shape), smell (odor), touch (surface texture), and others. It is essential that the sign of edibility is not a direct sensory attribute - it cannot be observed directly. It must be established indirectly, by comparing the perceptual attributes of an unknown mushroom with those of known edible or poisonous mushrooms.

Each component of the sensory description can be interpreted as a result of measurement or perception: e.g., x_{color} is the color of the cap as registered by sight; x_{smell} is the categorical value of the odor (almondy, unpleasant, absent, etc.) as perceived by the sense of smell; x_{ring} is the presence/type of ring on the stalk as determined visually or by touch, etc. Thus, the sensory description provides a necessary and sufficient set of inputs for mushroom identification using the comparator.

In the comparator identification method, the key role is played by the operation of comparing two objects by their features. Consider two mushrooms with descriptions $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_n)$. For each feature i, let us define an elementary comparator as the feature matching predicate:

$$P_i(x, y) = \begin{cases} 1, & if\ x_i = y_i, \\ 0, & if\ x_i \neq y_i, \end{cases} \tag{5}$$

where x_i and y_i are the values of the i-th feature in objects x and y. The predicate $P_i(x, y)$ indicates whether the objects are comparable in terms of the i-th feature, i.e., whether the observed feature is the same for them. For example, $P_{\text{color}}(x, y) = 1$ if two mushrooms have the same cap color; $P_{\text{smell}}(x, y) = 1$ if their smell belongs to the same category; $P_{\text{ring}}(x, y) = 1$ if either both have a ring of the same type or both do not, etc. In the special case, if $P_i(x, y) = 1$ $\forall i \in (\overline{1, n})$, then two mushrooms have identical sensory description (match on all features).

Not all features are equally informative for determining edibility, so comparisons emphasize those traits that correlate with the edible/poisonous class. Some features may be indistinguishable for the purpose of identifying edibility. For example, the veil-type attribute in the Data Set takes the same value for all mushrooms, so that a comparison based on this attribute does not provide useful information (it is always the same and does not influence the decision). At the same time, the odor or spore color attributes are extremely important: it is known that certain odor values are found only in poisonous mushrooms. Thus, in a comparator analysis of features, we can divide them into:

- diagnostic attributes that are critical and their difference directly indicates class (e.g., the presence of an acrid/chemical odor virtually guarantees poisonousness);
- minor attributes for which edible and poisonous mushrooms may overlap, and the coincidence/difference of these attributes only in combination with other attributes affects the inference;
- neutral attributes that have little influence on the inference of edibility.

Formalizing a statement of the form "*mushroom x looks like an edible mushroom*", we can introduce a similarity measure based on a set of pairwise comparisons. One approach is to count the number of matching features between x and some known edible mushroom e. Let's denote by $sim\ (x, y)$ the number of matches:

$$sim\ (x, y) = \sum_{i=1}^{n} P_i\ (x, y) \tag{6}$$

where P_i is the truth indicator of the matching condition. Then $sim\ (x, y) = n$ means that the descriptions of x and y completely match. A mushroom x can be called "similar" to y if $sim\ (x, y)$ is large, i.e. the objects coincide in most of the key features. By limiting case we can introduce a threshold τ: consider x similar to y if $sim\ (x, y) \geq \tau$. In other words, we introduce a binary similarity predicate $S_\tau(x, y)$ - "x is similar to y in at least τ features". This predicate is a composition of individual comparisons by attributes: $S_\tau(x, y)$ is true if enough individual $P_i(x, y)$ for important i are true. For example, the statement "this mushroom is similar to the edible species Agaricus" can be interpreted as: this mushroom has the same cap shape, plate color, lack of odor, and presence of a ring as some reference edible champignon, i.e., the corresponding P_i for these characteristic features are satisfied.

It should be noted that the target "edibility" itself is not part of the sensory description and is not directly involved in the comparison - it is the one we want to define. Therefore, a comparator conclusion about edibility can only be made indirectly, through comparison of the other attributes with already studied mushrooms whose edibility is known.

The space of descriptions X can be endowed with the structure of a metric space for quantifying the similarity of mushrooms. One natural variant of the metric is the Hamming metric [23, 24], defined as the number of differing features:

$$d(x,y) = \sum_{i=1}^{n} (1 - P_i\,(x,y)) = n - sim(x,y) \tag{7}$$

Such a metric $d(x,y)$ is 0 if the mushrooms x and y have an identical description, and is increased by 1 for each feature in which they differ. Proximity (similarity) can be defined through a metric: the smaller $d(x,y)$ is, the more "similar" the mushrooms are. By introducing a threshold $\varepsilon \geq 0$, we can define the binary relation:

$$R_\varepsilon \subseteq X \times X\colon (x,y) \in R_\varepsilon \Leftrightarrow d(x,y) \leq \varepsilon \tag{8}$$

Given $\varepsilon = 0$ the relation R_0 expresses exactly the identity of the descriptions. For $\varepsilon > 0$ the relation becomes a relation of ε-similarity: the mushrooms x and y differ in no more than ε features. It is clear that R_0 is an equivalence relation on the set of objects (partitioning the space X into classes of identical descriptions), while R_ε may not have transitivity at $\varepsilon > 0$, but defines a neighborhood (clusters) of similar objects.

In the language of predicates, we can define the corresponding similarity predicates. For example, the predicate:

$$S_\varepsilon(x,y) = 1 \Leftrightarrow d(x,y) \leq \varepsilon \tag{9}$$

In particular, if $\varepsilon = 0$:

$$S_0(x,y) \Leftrightarrow \bigwedge_{i=1}^{n} P_i(x,y), \tag{10}$$

i.e., complete matching of descriptions. Similarity predicates allow to formalize statements of the form "*object x belongs to the same class as object y*". In the classical theory of comparator identification, this corresponds to the notion of an equivalence predicate of object identity.

In other words, within the framework of our problem we can say that edible mushrooms form one equivalence class (by the relation "*having the same edibility status*", defined through the similarity of key properties), and poisonous mushrooms form another. The identification task boils down to determining which of these two equivalence classes the unknown mushroom belongs to.

The mechanism of decision making in the comparator structure relies on a set of predicates of similarity with standard samples (references). Suppose we have a set of known edible mushrooms $E = (e_1, e_2, \dots, e_{n_e})$ and poisonous mushrooms $P = (p_1, p_2, \dots, p_{n_p})$ (these samples can be considered as a training sample or expert knowledge). The unknown mushroom class x is decided based on analyzing $d(x, e_j)$ and $d(x, p_k)$ - distances to known benchmarks of both classes. Formally, two distance functions can be introduced:

$$D_E(x) = \min_{e \in E} d(x,e)\,, D_P(x) = \min_{p \in P} d(x,p), \tag{11}$$

i.e., the distance from x to the nearest edible and nearest poisonous specimen, respectively. The classification rule is then given as:

$$\delta(x) = \begin{cases} \delta_1, if\ D_E(x) < D_P(x) \\ \delta_2, if\ D_P(x) < D_E(x), \\ \delta_3, if\ D_E(x) = D_P(x) \end{cases} \tag{12}$$

where δ_1 means “edible”, δ_2 – “poisonous”, and δ_3 – “undefined”.

In the case of equal distances, either a finer criterion can be applied or the decision is postponed to request additional features. Thus, the decision is made in favor of the class to which benchmarks the mushroom x happens to be closer in the feature space. This mechanism is a formalization of the principle "*an unknown object belongs to the same class as the most similar known object*". The comparator identification method actually assigns objects to classes based on their similarity to representatives of these classes, dividing the set of objects into equivalence classes automatically.

It is important to emphasize that when $\varepsilon = 0$ (the requirement of complete coincidence of descriptions), the rule is reduced to an exact match with the reference: an unknown mushroom is classified as edible if at least one known edible mushroom with an identical set of features is found (otherwise, the poisonous one is checked). However, in real conditions, new combinations of features that have not been found before are possible. Then we have to rely on $\varepsilon > 0$, i.e. we have to allow partial coincidence. The metric-based decision mechanism naturally takes partial matches into account: even if there is no exact analog in memory, the mushroom will be assigned to the class whose sample is most similar (minimum distance). This approach is robust to feature variation and noise in the data, as it does not require a perfect match, but uses a proximity measure.

The above mechanism can also be described as a logical scheme based on comparison predicates. The logical model of identification represents the solution as inference based on sets of conditions. In the simplest case, for $\varepsilon = 0$, we can write the logical expression for the mushroom x belonging to the edible class as a disjunction of conjunctions reflecting the match with each edible reference:

$$\varphi_{edible}(x) = \bigvee_{e \in E} \left(\bigwedge_{i=1}^{n} G_e(x_i, e_i) \right), \tag{13}$$

where each $G_e(x_i, e_i)$ means "*mushroom x matches in feature i with known edible mushroom e*":

$$G_e(x_i, e_i) = \begin{cases} 1, if\ x_i = e_i \\ 0, if\ x_i \neq e_i \end{cases} \tag{14}$$

Similarly, we can define the formula $\varphi_{poisonous}(x)$ in terms of known poisonous mushrooms. If $\varphi_{edible}(x) = 1$, the mushroom is identified as edible; if $\varphi_{poisonous}(x) = 1$, it is identified as poisonous. In the case when none of the formulas is reach to 1 (i.e., there is no complete match with any of the benchmark), a fuzzy or stepwise logical solution is used. For example, it is possible to check conditions in descending order of their diagnostic significance:

Step 1: Check for signs clearly indicating poisonousness. If feature i is found, the values of which never occur in edible mushrooms (but do occur in poisonous ones), and x_i has just such a value - immediately classify the mushroom as poisonous (a decision without further doubt). For example, if $x_{odor} =$ "pungent" or $x_{odor} =$ "fishy", then the mushroom is definitely poisonous (in the UCI Mushroom dataset, all mushrooms with a pungent or fishy odor are poisonous).

Step 2: If no obvious poisonous features are found, check for characteristic combinations of features of edible mushrooms. For example, for some edible mushrooms, the following combination may be typical $P_{\text{cap-shape}}(x, e) \wedge P_{\text{ring-typ}}\ (x, e) \wedge P_{\text{odor}}(x, e)$ for some reference e. In other words, if a mushroom x satisfies most of the conditions characteristic of a certain edible species (or group of species), then a reasonable conclusion can be drawn about its edibility.

Step 3: If doubts remain (there are both edible traits and uncharacteristic abnormalities), a more refined analysis is performed: comparison with the closest edible and poisonous references (e.g. by the $d(x, y)$ metric as described above) and analysis of which differences prevent unambiguous identification. Additional information or an expert may need to be brought in at this step. In logical terms, step 3 corresponds to evaluating the truth of similarity predicates at some $\varepsilon > 0$ and selecting a class based on the maximum number of fulfilled predicates P_i with the benchmarks of each class.

The logic scheme is thus reduced to a set of rules of the form "*IF <conditions of comparison>, THEN <resolution>*". These rules can be extracted from comparisons with the benchmarks and knowledge about the diagnostic value of the features. The advantage of comparator identification is that such a scheme is human verifiable: the decision is justified by explicitly stating with which known mushrooms and on which features a given specimen matches or diverges. In fact, the method fixes the course of reasoning of an expert mushroom grower: for example, "*if the mushroom has white plates and a ring on the stalk, and there is no unpleasant odor, then it looks like a champignon (edible) and does not look like a pale grebe (which has green plates and volva)*". All of this reasoning can be precisely expressed through the predicates P_i and their logical combinations.

4. Results and discussion

This section looks at an example of how to apply the proposed approach based on the comparator model.
equation

Let $x = (x_{11}, x_{12}, \dots, x_{ij}, \dots, x_{22,m_{22}})$ be the binary indicator vector that encodes every categorical attribute of a mushroom specimen in the UCI data set (Table1). For each attribute $i = (1,2, \dots, 22)$ and for each of its m_i admissible categories, a component:

$$x_{ij} = \begin{cases} 1, if \text{ the } specimen \text{ exhibits the } j - \text{th category of the } i - \text{th attribute,} \\ 0, \text{otherwise,} \end{cases}, \quad (15)$$

$j = (1,, \dots, m_i)$,

is introduced. Hence, a specimen is mapped to a 111-dimensional binary vector (the sum of all m_i). The notation x_i without a second index will be reserved for the whole block of components that belong to attribute i: $x_i = (x_{i1}, x_{i2}, \dots, x_{i,m_i})$.

Table 1
Attribute values encoding (based on data set description in [20])

Attribute i	Attribute name (according to UCI data set)	Attribute values and their notations
1	cap-shape	$x_{1,1}$: *bell*, $x_{1,2}$: *conical*, $x_{1,3}$: *convex*, $x_{1,4}$: *flat*, $x_{1,5}$: *knobbed*, $x_{1,6}$: *sunken*
2	cap-surface	$x_{2,1}$: *fibrous*, $x_{2,2}$: *grooves*, $x_{2,3}$: *scaly*, $x_{2,4}$: *smooth*
3	cap-color	$x_{3,1}$: *brown*, $x_{3,2}$: *buff*, $x_{3,3}$: *cinnamon*, $x_{3,4}$: *gray*, $x_{3,5}$: *green*, $x_{3,6}$: *pink*, $x_{3,7}$: *purple*, $x_{3,8}$: *red*, $x_{3,9}$: *white*, $x_{3,10}$: *yellow*
4	bruises	$x_{4,1}$: *true*, $x_{4,2}$: *false*
5	odor	$x_{5,1}$: *almond*, $x_{5,2}$: *anise*, $x_{5,3}$: *creosote*, $x_{5,4}$: *fishy*, $x_{5,5}$: *foul*, $x_{5,6}$: *musty*, $x_{5,7}$: *none*, $x_{5,8}$: *pungent*, $x_{5,9}$: *spicy*
6	gill-attachment	$x_{6,1}$: *attached*, $x_{6,2}$: *descending*, $x_{6,3}$: *free*, $x_{6,4}$: *notched*
7	gill-spacing	$x_{7,1}$: *close*, $x_{7,2}$: *crowded*, $x_{7,3}$: *distant*
8	gill-size	$x_{8,1}$: *broad*, $x_{8,2}$: *narrow*
9	gill-color	$x_{9,1}$: *black*, $x_{9,2}$: *brown*, $x_{9,3}$: *buff*, $x_{9,4}$: *chocolate*, $x_{9,5}$: *gray*, $x_{9,6}$: *green*, $x_{9,7}$: *orange*, $x_{9,8}$: *pink*, $x_{9,9}$: *purple*, $x_{9,10}$: *red*, $x_{9,11}$: *white*, $x_{9,12}$: *yellow*

10	stalk-shape	$x_{10,1}$: *enlarging*, $x_{10,2}$: *tapering*
11	stalk-root	$x_{11,1}$: *bulbous*, $x_{11,2}$: *club*, $x_{11,3}$: *cup*, $x_{11,4}$: *equal*, $x_{11,5}$: *rhizomorphs*, $x_{11,6}$: *rooted*, $x_{11,7}$: *missing*
12	stalk-surface-above-ring	$x_{12,1}$: *fibrous*, $x_{12,2}$: *scaly*, $x_{12,3}$: *silky*, $x_{12,4}$: *smooth*
13	stalk-surface-below-ring	$x_{13,1}$: *fibrous*, $x_{13,2}$: *scaly*, $x_{13,3}$: *silky*, $x_{13,4}$: *smooth*
14	stalk-color-above-ring	$x_{14,1}$: *brown*, $x_{14,2}$: *buff*, $x_{14,3}$: *cinnamon*, $x_{14,4}$: *gray*, $x_{14,5}$: *orange*, $x_{14,6}$: *pink*, $x_{14,7}$: *red*, $x_{14,8}$: *white*, $x_{14,9}$: *yellow*
15	stalk-color-below-ring	$x_{15,1}$: *brown*, $x_{15,2}$: *buff*, $x_{15,3}$: *cinnamon*, $x_{15,4}$: *gray*, $x_{15,5}$: *orange*, $x_{15,6}$: *pink*, $x_{15,7}$: *red*, $x_{15,8}$: *white*, $x_{15,9}$: *yellow*
16	veil-type	$x_{16,1}$: *partial*, $x_{16,2}$: *universal* (note: in the data set only *partial* occurs)
17	veil-color	$x_{17,1}$: *brown*, $x_{17,2}$: *orange*, $x_{17,3}$: *white*, $x_{17,4}$: *yellow*
18	ring-number	$x_{18,1}$: *none*, $x_{18,2}$: *one*, $x_{18,3}$: *two*
19	ring-type	$x_{19,1}$: *cobwebby*, $x_{19,2}$: *evanescent*, $x_{19,3}$: *flaring*, $x_{19,4}$: *large*, $x_{19,5}$: *none*, $x_{19,6}$: *pendant*, $x_{19,7}$: *sheathing*, $x_{19,8}$: *zone*
20	spore-print-color	$x_{20,1}$: *black*, $x_{20,2}$: *brown*, $x_{20,3}$: *buff*, $x_{20,4}$: *chocolate*, $x_{20,5}$: *green*, $x_{20,6}$: *orange*, $x_{20,7}$: *purple*, $x_{20,8}$: *white*, $x_{20,9}$: *yellow*
21	population	$x_{21,1}$: *abundant*, $x_{21,2}$: *clustered*, $x_{21,3}$: *numerous*, $x_{21,4}$: *scattered*, $x_{21,5}$: *several*, $x_{21,6}$: *solitary*
22	habitat	$x_{22,1}$: *grasses*, $x_{22,2}$: *leaves*, $x_{22,3}$: *meadows*, $x_{22,4}$: *paths*, x$x_{22,5}$: *urban*, $x_{22,6}$: *waste*, $x_{22,7}$: *woods*

Let's consider the typical process of analyzing product descriptions using a comparator model. As mentioned above, we will use the UCI Mushroom dataset and determine the edibility of a mushroom based on its description. We assume the user is describing the mushrooms directly in front of them and is indicating all their sensory perceptions (appearance, color, smell, etc.). The comparator then returns one of the 12 defined classes as the answer. We interpret the comparator's response as "mushroom X is similar to edible" (if the delta response is 1) based on the coincidence of the following features, for which (5) is true. First, we generate various descriptions that a user could provide and demonstrate which features can be extracted from them.

Below are seven sample descriptions. They are all phrased differently, with some attributes named explicitly, some hinted at, and some left unspecified – exactly the variety you would face in practice.

Example 1: "The cap is flat and smooth, light-brown in color; the gills underneath are crowded and white. I don't notice any smell at all. There's one thin pendant ring on the stalk, which tapers slightly and is white both above and below the ring. No blue bruising when I press it."

There are core attributes extracted: cap-shape – "*flat",* cap-surface – "*smooth*", cap-color – "*brown*", bruises – "*false*", odor – "*none*", gill-spacing – "*crowded*", gill-color – "*white*", stalk-shape – "*tapering*", stalk-color-above-ring – "*white*", stalk-color-below-ring – "*white*", ring-number – "*one*", ring-type – "*pendant*". Other values are undefined.

Example 2: "Tiny purple-red buttons pushing up through grassy soil - the caps look convex and a bit scaly. When I scratch the flesh it bruises blue-green, and there's a strong, almost chemical odor. Can't see any skirt or ring yet."

There are core attributes extracted: cap-shape – "*convex",* cap-surface – "*scaly*", cap-color – "*purple*" or "*red*", bruises – "*true*", odor-"*creosote*", ring-number – "*none observed*", ring-type – "*none*", and habitat – "*grasses*". Other values are undefined.

Example 3: “It grows alone on a fallen log in the woods. The top is bell-shaped, kind of cinnamon-colored, and the stalk widens at the base. The air around it smells spicy – like cloves. I didn’t notice any spore dust.”

There are core attributes extracted: cap-shape – “*bell-shaped*”, cap-color – “*cinnamon*”, stalk-shape – “*enlarging*”, odor – “*spicy*”, population – “*solitary*”, and habitat – “*woods*”. Other values are undefined.

Example 4: “These mushrooms form tight clusters on leaf litter. Caps are sunken in the middle, with a yellow surface that feels fibrous. Gills seem distant and pale gray. The stalk is silky above a single ring and orangey below. When sliced, nothing turns blue.”

There are core attributes extracted: population – “*clustered*”, habitat – “*leaves*”, cap-shape – “*sunken*”, cap-surface – “*fibrous*”, cap-color – “*yellow*”, gill-spacing – “*distant*”, gill-color – “*gray*”, bruises – “*none*”, ring-number – “*one*”, stalk-surface-above-ring – “*silky*”, and stalk-color-below-ring – “*orange*”. Other values are undefined.

Example 5: “Cap surface is smooth, pale pink; no scales or grooves. There’s definitely no ring, and the stem stays the same thickness top to bottom. I get a musty cellar smell but can’t decide if it bruises— pressing didn’t change the color. Not sure about spore-print yet.”

There are core attributes extracted: cap-surface – “*smooth*”, cap-color – “*pink*”, ring-number – “*none*”, ring-type – “*none*”, odor – “*musty*”, and bruises – “*none*”. Other values are undefined.

Example 6: “*Mature mushrooms with broad pink gills and a pleasant almond scent. The overnight spore print is deep brown. A thin pendant ring encircles the stalk, and the tissue below the ring is perfectly smooth.*”

There are core attributes extracted: odor – “*almond*”, spore-print-color – “*brown*”, gill-color – “*pink*”, ring-type – “*pendant*”, and stalk-surface-below-ring – “*smooth*”. Other values are undefined.

Example 7: “*A cluster of slick white caps gives off a distinctly fishy smell. The gills are white and the spore print is also white. A skirt-like pendant ring hangs from the stalk, which feels smooth below the ring*”.

The core attributes extracted are: odor – “*fishy*”, spore-print-color – “*white*”, gill-color – “*white*”, ring-type – “*pendant*”, and stalk-surface-below-ring – “*smooth*”. Other values are undefined.

Let $x_{ij} = 1$ denote that the mushroom exhibits the j-th value of the i-th attribute (as numbered above), and $x_{ij} = 0$ otherwise. For every description we write a conjunctive clause $\Phi_k(x)$ that fixes only the attribute–value pairs explicitly inferable from the text; all other indicators remain free.

$\Phi_1(x) = x_{1,4} \land x_{2,4} \land x_{3,1} \land x_{4,2} \land x_{5,7} \land x_{7,2} \land x_{9,11} \land x_{10,2} \land x_{14,8} \land x_{15,8} \land x_{18,2} \land x_{19,6},$

$\Phi_2(x) = x_{1,3} \land x_{2,3} \land (x_{3,7} \lor x_{3,8}) \land x_{4,1} \land x_{5,3} \land x_{18,1} \land x_{19,5} \land x_{22,1},$

$\Phi_3(x) = x_{1,1} \land x_{3,3} \land x_{5,9} \land x_{10,1} \land x_{21,6} \land x_{22,7},$

$\Phi_4(x) = x_{1,6} \land x_{2,1} \land x_{3,10} \land x_{4,2} \land x_{7,3} \land x_{9,5} \land x_{12,3} \land x_{15,5} \land x_{18,2} \land x_{21,2} \land x_{22,2},$

$\Phi_5(x) = x_{2,4} \land x_{3,6} \land x_{4,2} \land x_{5,6} \land x_{18,1} \land x_{19,5},$

$\Phi_6(x) = x_{5,1} \land x_{9,8} \land x_{13,4} \land x_{19,6} \land x_{20,2},$

$\Phi_7(x) = x_{5,4} \land x_{9,11} \land x_{13,4} \land x_{19,6} \land x_{20,8}.$

We will complete the identification task in two stages. First, we will evaluate the edibility of the mushroom based on a core set of characteristics. Then, we will refine the results using a full set of characteristics. The core set of characteristics will be based on the values available in the training dataset. To do this, we take the conjunction of all true predicates for each reference class of edible mushrooms.

$$\Phi_+ = \bigvee_{e \in E} \bigwedge_{i,j \in e} e_{ij}, \tag{16}$$

where e_{ij} represents the attributes of each edible specimen.

Similarly, we determine the class of poisonous mushrooms.

$$\Phi_- = \bigvee_{p \in P} \bigwedge_{i,j \in p} p_{ij} \quad (17)$$

Then the entire training set can be described by the formula

$$\Phi = \Phi_+ \vee \neg\Phi_- \quad (18)$$

Next, we sequentially apply the method of extracting essential features, as proposed in [1]. The resulting core, based on the solution of equation (18), contains only the features whose values allow us to distinguish between edible and poisonous mushrooms. This core ensures the equivalence of the original and reduced comparator classifiers. For this dataset, the following kernel was obtained: spore print color, gill color, ring type, stalk-surface-below-ring, i.e.

$C = \{x_9, x_{13}, x_{19}, x_{20}\}$

Let's simplify the equations that describe unknown instances of mushrooms. We obtain:

$\Phi_1(x) = x_{9,11} \wedge x_{19,6}$,

$\Phi_2(x) = x_{19,5}$,

$\Phi_4(x) = x_{9,5}$,

$\Phi_5(x) = x_{19,5}$,

$\Phi_6(x) = x_{9,8} \wedge x_{13,4} \wedge x_{19,6} \wedge x_{20,2}$,

$\Phi_7(x) = x_{9,11} \wedge x_{13,4} \wedge x_{19,6} \wedge x_{20,8}$.

It is clear that for examples 1-5, the information obtained is insufficient to assign these samples to one of the classes. However, for examples 6 and 7, it is possible to determine their proximity to one of the classes based on the core features. For these examples, we calculate the distances (11) and apply the classification rule (12). We obtain:

$D_E(x_6) = 1, D_P(x_6) = 3, D_E(x_7) = 3, D_P(x_8) = 1$,

i.e. sample 6 is closed to edible prototype and sample 7 is closed to poisonous prototype. Then, analyze those examples using the full attribute space. For the example 1 we have

$\Phi_1(x) = x_{1,4} \wedge x_{2,4} \wedge x_{3,1} \wedge x_{4,2} \wedge x_{5,7} \wedge x_{7,2} \wedge x_{9,11} \wedge x_{10,2} \wedge x_{14,8} \wedge x_{15,8} \wedge x_{18,2} \wedge x_{19,6}$,

and the closest edible and poisonous prototypes are described as

$\mathrm{E}_1(x) = x_{1,3} \wedge x_{2,4} \wedge x_{3,1} \wedge x_{4,1} \wedge x_{5,7} \wedge x_{7,1} \wedge x_{9,8} \wedge x_{10,2} \wedge x_{14,8} \wedge x_{15,8} \wedge x_{18,2} \wedge x_{19,6}$,

$\mathrm{P}_1(x) = x_{1,3} \wedge x_{2,4} \wedge x_{3,1} \wedge x_{4,2} \wedge x_{5,7} \wedge x_{7,1} \wedge x_{9,11} \wedge x_{10,2} \wedge x_{14,8} \wedge x_{15,8} \wedge x_{18,2} \wedge x_{19,6}$,

and $D_E(x_1) = 4, D_P(x_1) = 2$. Because $D_P(x_1) < D_E(x_1)$ the specimen is closer to the poisonous prototype when every available attribute is considered.

Thus, the combination "no bruises + white gills + crowded gills + pendant ring + odor none" matches key traits of *Amanita phalloides* more than of common edible *Agaricus*; without spore-print color or chemical tests, the safer classification is poisonous, i.e. avoid consumption. Hence, from a comparator perspective, Example 1 should be flagged unsafe unless further evidence pushes it toward the edible region.

Therefore, the comparator identification method applied to the edibility/poisoning task, in fact, implements binary classification, albeit in a different paradigm. That is to say, it establishes equivalence or similarity with samples. The classification of a set of objects is determined through a comparative analysis with representatives of these established classes. From a formal point of view, the result of the comparator scheme is a function, which takes on two values (e.g., 1 for edible and 0 for poisonous mushrooms). That is to say, it corresponds to the target variable in the classification task. However, the internal logic of decision-making processes differs from, for example, the decision-making processes of a tree or a neural network. The comparator model does not directly derive the formula for this function through the features. Rather, it computes a value through comparisons with reference objects.

It is evident that the rule, based on (11), is a variation of the nearest neighbor method in feature space. That is to say, the class of a new object is determined by the class of the nearest neighbor among the training data. The distinction in emphasis is as follows: comparator identification accentuates the explicability and logical structure of such a solution. While the K-nearest neighbor algorithm does not explicitly provide the class label, the comparator scheme can provide a rationale for the classification. For instance, it can be explained that the closest was a mushroom classified as edible with a distance of 2; however, the closest poisonous one had a distance of 4. Therefore, the classification of this instance as edible is supported by these data. Furthermore, the method enables the incorporation of a priori rules, thereby aligning classification with the characteristics of an expert system. Consequently, comparator solutions are directly associated with the outcomes of binary classification, while concurrently offering an interpretation through resemblance to recognized patterns. The veracity of classification is contingent upon the assumption that the feature space adequately differentiates edible species from poisonous ones. This is a prerequisite for the applicability of comparator identification.

In summary, the application of the comparator identification method to the task of mushroom edibility demonstrates an alternative, human-understandable approach to binary classification.

5. Conclusion

In this work, we have demonstrated a framework for identifying food edibility grounded in comparator-based predicate logic. To answer the first research question, we have shown how logical rules for edibility can be formalized as predicate structures based on observable sensory characteristics. Each perceptual feature – such as cap shape, odor, texture, and color – is encoded as a finite-valued predicate or an indicator variable. We have also developed a core-extraction procedure to isolate the minimal subset of features. This yields a compact, human-readable core feature set that drives the comparator decision rule. To evaluate the second research question, which concerns the effectiveness of the comparator model versus traditional machine learning approaches, we conducted experiments on the canonical UCI Mushroom dataset.

Thus, we have formally described the process by which sensory attributes are converted into a system of comparisons, the manner in which a metric and logical structure for decision making is built on this basis, and the manner in which the final verdict ("edible" or "poisonous") is obtained as a consequence of comparison with already known samples. This approach establishes a connection between a rigorous mathematical model, characterized by predicates and metrics, and practical interpretability. This interpretability is particularly valuable in the critical domain of poisonous mushroom identification.

A key contribution of this study is the interpretability of comparator solutions. Unlike "black-box" models, comparator rules provide logical explanations – for each edibility verdict, one can trace exactly which features matched which reference specimens and which predicate failures tipped the decision. The core-extraction method ensures that only features with genuine discriminative power appear in the final rule, which simplifies the explanation further. In user-facing scenarios (e.g., mobile identification apps), this transparency allows users to understand and trust the model's verdicts and supply targeted follow-up descriptions when the model is uncertain.

Acknowledgements

The EU NextGenerationEU partially funds the research study depicted in this paper through the Recovery and Resilience Plan for Slovakia under project No. 09I03-03-V01-00078.

Declaration on Generative AI

During the preparation of this paper, the authors used Grammarly in order to grammar and spelling check and DeepL in order to text translation. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

[1] O. Karataiev, D. Sitnikov, and N. Sharonova, A method for investigating links between discrete data features in knowledge bases in the form of predicate equations, in: Proc. CEUR Workshop, vol. 3387, pp. 224–235, 2023. https://ceur-ws.org/Vol-3387

[2] T. M. Tashu, S. Fattouh, P. Kiss and T. Horváth, "Multimodal E-Commerce Product Classification Using Hierarchical Fusion," 2022 IEEE 2nd Conference on Information Technology and Data Science (CITDS), Debrecen, Hungary, 2022, pp. 279-284, doi: 10.1109/CITDS54976.2022.9914136.

[3] C. Chen et al., „This looks like that: Deep learning for interpretable image recognition,„ in Proc. NeurIPS, 2019.

[4] Xu, K., Zhou, H., Zheng, H., Zhu, M., & Xin, Q. (2024). Intelligent classification and personalized recommendation of e-commerce products based on machine learning. arXiv preprint arXiv:2403.19345.

[5] Huanjing Zhao, Beining Yang, Yukuo Cen, Junyu Ren, Chenhui Zhang, Yuxiao Dong, Evgeny Kharlamov, Shu Zhao, and Jie Tang. 2024. Pre-Training and Prompting for Few-Shot Node Classification on Text-Attributed Graphs. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24). Association for Computing Machinery, New York, NY, USA, 4467–4478. https://doi.org/10.1145/3637528.3671952

[6] Y. Hu et al., „Natural language processing and machine learning approaches for food categorization and nutrition quality prediction compared to traditional methods,„ American Journal of Clinical Nutrition, 2022.

[7] M. van Erp, J. van der Sande, and C. van Son, „Using AI to analyze nutrition and sustainability of recipes,„ Frontiers in Artificial Intelligence, 2021.

[8] K. Makridis, A. Gkillas, and G. Sermpinis, „Deep learning with NLP and time-series modeling for enhanced food safety,„ Machine Learning, 2023.

[9] B. Wagner, A. Rowe, and M. Gillam, „Mushroom data creation, curation, and simulation to support binary classification,„ Scientific Reports, 2021.

[10] J. Snell, K. Swersky, and R. Zemel, „Prototypical networks for few-shot learning, „in Proc. NeurIPS, 2017.

[11] T. Kohonen, Self-Organizing Maps, 2nd ed. Berlin, Germany: Springer, 1995 (chap. on Learning Vector Quantization).

[12] Cherednichenko, O., Nebesky, L., Kováč, M. Gathering and Matching Data from the Web: The Bibliographic Data Collection Case Study. International Conference on Smart Business Technologies , 2024, pp 139–146. DOI: 10.5220/0012863500003764

[13] M. Yu et al., „Pre-training Language Models for Comparative Reasoning, „in Proc. EMNLP, 2023.

[14] B. Kim, C. Rudin, and J. A. Shah, „The Bayesian Case Model: A generative approach for case-based reasoning and prototype inference,„ in Proc. ICML, 2014.

[15] Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang GPT-NER: Named Entity Recognition via Large Language Models (2023) doi.org/10.48550/arXiv.2304.10428

[16] Baigang, M., Yi, F. A review: development of named entity recognition (NER) technology for aeronautical information intelligence. Artif Intell Rev 56, 1515–1542 (2023). https://doi.org/10.1007/s10462-022-10197-2

[17] Sowmya Vajjala, Ramya Balasubramaniam What do we Really Know about State of the Art NER? (2022) doi.org/10.48550/arXiv.2205.00034

[18] Yu Wang, Hanghang Tong, Ziye Zhu, Yun Li Nested Named Entity Recognition: A Survey ACM Transactions on Knowledge Discovery from Data (TKDD), Volume 16, Issue 6 Article No.: 108, Pages 1 – 29. https://doi.org/10.1145/3522593

[19] Mayhew, S., Blevins, T., Liu, S., Šuppa, M., Gonen, H., Imperial, J. M., ... & Pinter, Y. (2023). Universal NER: A gold-standard multilingual named entity recognition benchmark. arXiv preprint arXiv:2311.09122.

[20] Mushroom," UCI Machine Learning Repository, 1981. [Online]. Available: https://doi.org/10.24432/C5959T

[21] Nusrat Jahan Pinky, S.M. Mohidul Islam, Rafia Sharmin Alice, " Edibility Detection of Mushroom Using Ensemble Methods", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.11, No.4, pp. 55-62, 2019. DOI: 10.5815/ijigsp.2019.04.05

[22] Gupta, Aaditya. (2022). Classification Of Mushroom Using Artificial Neural Network. 10.1101/2022.08.31.505980

[23] Sun, Y., & Ge, G. (2025). Bounds and constructions of ℓ-read codes under the Hamming metric. IEEE Transactions on Information Theory.

[24] M. Shi, H. Zhu and T. Helleseth, "The Connections Among Hamming Metric, b-Symbol Metric, and r-th Generalized Hamming Metric," in *IEEE Transactions on Information Theory*, vol. 69, no. 4, pp. 2485-2493, April 2023, doi: 10.1109/TIT.2023.3239543.