NNBSVR: Neural Network-Based Semantic Vector Representations of ICD-10 Codes

Monah Bou Hatoum[1*†], Jean Claude Charr[1†], Alia Ghaddar[2†], Christophe Guyeux[1†], David Laiymani[1†]

[1]Department of Computer Science, University of Bourgogne Franche-Comté, UBFC, CNRS, Belfort, 90000, France.
[2]Department of Computer Science, International University of Beirut, Beirut, 146404, Lebanon.

*Corresponding author(s). E-mail(s): monah.bou_hatoum@univ-fcomte.fr;
Contributing authors: jean-claude.charr@univ-fcomte.fr; alia.ghaddar@liu.edu.lb; christophe.guyeux@univ-fcomte.fr; david.laiymani@univ-fcomte.fr;
†These authors contributed equally to this work.

Background: Automatically predicting ICD-10 codes from clinical notes using machine learning models can reduce the burden of manual coding. However, existing methods often overlook the semantic relationships between ICD-10 codes, resulting in inaccurate evaluations when clinically similar codes are considered completely different. Traditional evaluation metrics, which rely on equality-based matching, fail to capture the clinical relevance of predicted codes.

Objective: This study introduces *NNBSVR* (Neural Network-Based Semantic Vector Representations), a novel approach for generating semantic-based vector representations of ICD-10 codes. Unlike traditional approaches that rely on exact code matching, *NNBSVR* incorporates contextual and hierarchical information to enhance both prediction accuracy and evaluation methods.

Methods: We validate *NNBSVR* using intrinsic and extrinsic evaluation methods. Intrinsic evaluation assesses the vectors' ability to reconstruct the ICD-10 hierarchy and identify clinically meaningful clusters. Extrinsic evaluation compares our relevancy-based approach, which includes customized evaluation metrics, to traditional equality-based metrics on an ICD-10 code prediction task using a 9.57 million

clinical notes corpus. **Results:** *NNBSVR* demonstrates significant improvements over equality-based metrics, achieving a 9.81% gain in micro-F1 score on the training set and a 12.73% gain on the test set. A manual review by medical experts on a sample of 10,000 predictions confirms an accuracy of 92.58%, further validating our approach.

Conclusion: This study makes two significant contributions: first, the development of semantic-based vector representations that encapsulate ICD-10 code relationships and context; second, the customization of evaluation metrics to incorporate clinical relevance. By addressing the limitations of traditional equality-based evaluation metrics, *NNBSVR* enhances the automated assignment of ICD-10 codes in clinical settings, demonstrating superior performance over existing methods.

1 Introduction

The World Health Organization's 10th revision of the International Classification of Diseases (ICD-10) provides alphanumeric codes to classify diseases, symptoms, abnormal findings, and external injury causes [17]. ICD-10 features a hierarchical structure with broadly defined diagnosis chapters containing nested subcategories for greater specificity. To address regional needs, modifications such as ICD-10-GM [22], ICD-10-AM [35], and ICD-10-CM have been developed, adapting codes and rules to support healthcare administration and analysis within specific countries and regions.

Accurate ICD-10 coding forms the cornerstone of modern healthcare administration, underpinning critical processes from billing to clinical research [11]. However, the manual coding process is fraught with challenges: it is time-intensive, susceptible to human error, and demands extensive expertise to navigate the intricate coding system [14]. These issues have catalyzed a paradigm shift towards machine learning-based automated coding, which promises to expedite the process while enhancing its consistency and reliability.

Despite the promise of automation, predicting ICD-10 codes remains challenging due to the complexity of the coding system and the inherent ambiguities in clinical documentation. Many ICD-10 codes are clinically similar yet distinct, leading to challenges when automated systems rely on traditional equality-based evaluation metrics. For instance, codes like I25.9 (Chronic ischemic heart disease, unspecified) and U82.1 (Ischemic heart disease) describe related conditions but are treated differently by conventional evaluation methods. Conversely, codes such as P74.31 (Hyperkalemia of newborn) and P74.32 (Hypokalemia of newborn) belong to the same category (P74) but require entirely different clinical interventions. P74.31 necessitates treatment to decrease potassium levels, while P74.32 requires intervention to increase potassium levels. Ignoring such specificity in automated coding could severely impact newborn care and potentially risk lives, underscoring the critical importance of accurate and nuanced ICD-10 code prediction [15].

Traditional metrics exhibit significant limitations in capturing these nuanced relationships between ICD-10 codes. They often over-penalize predictions that are semantically similar but not exact matches, while simultaneously failing to recognize critical distinctions between codes that necessitate different clinical approaches.

This rigid evaluation framework can lead to a misrepresentation of model performance, potentially impacting the efficacy of automated coding systems in healthcare settings [15, 29, 31].

To address these limitations, we propose *NNBSVR* (Neural Network-Based Semantic Vector Representations), a novel approach that generates rich vector representations for ICD-10 codes, leveraging both semantic and hierarchical information to improve the accuracy of prediction and evaluation. Unlike traditional approaches that treat each code as isolated, *NNBSVR* incorporates contextual relationships among codes, effectively capturing their intrinsic connections. This approach not only enhances the reliability of automated coding but also directly benefits healthcare processes by improving billing accuracy, supporting clinical decision-making, and enabling more nuanced public health analysis.

The primary contributions of this study are twofold:

- Development of a comprehensive vector representation method for ICD-10 codes that captures their semantics, hierarchical relationships, and clinical context, including crucial attributes such as gender and age restrictions.
- Introduction of customized evaluation metrics that assess the relevancy between predicted and true labels, moving beyond exact matching and addressing the shortcomings of traditional equality-based approaches.

These contributions offer a more clinically meaningful assessment of model performance and represent a fundamental improvement in machine learning for medical coding.

The remainder of this paper is organized as follows: Section 2 presents state-of-the-art ICD-10 vector representations and their shortcomings. Our approach is detailed in Section 3. Section 4 presents the results of a comparative study between our approach and traditional evaluation methods. Section 5 details our findings and recommendations for improving evaluation accuracy. We conclude with a summary of contributions and outline future work in Section 6.

In the following section, we review existing ICD-10 vectorization approaches, highlighting their limitations and how they motivated the development of *NNBSVR*. By understanding these shortcomings, we position *NNBSVR* as a necessary advancement for capturing the semantic, hierarchical, and contextual complexities inherent in ICD-10 coding.

2 Related Work

In this section, we review existing research on methods for representing and predicting ICD-10 codes, focusing on advancements in deep learning, natural language processing (NLP), and vectorization techniques for clinical coding. We categorize relevant literature based on early CNN and RNN approaches, later transformer-based models, and vectorization frameworks aimed at capturing semantic similarity, providing a foundation for understanding the context and limitations of current methods.

Early work in ICD coding explored convolutional and recurrent neural networks (CNNs and RNNs) to automatically assign codes from clinical text. For instance, [27]

used a multi-filter convolutional network with residual connections to capture both local and global semantic features, achieving state-of-the-art micro F1 scores of 89.5% and 94.2% on the MIMIC-III and Physionet datasets, respectively. While this architecture improved interpretability and captured essential semantic features, the model's focus on ICD-9 codes limits its applicability to the more complex, hierarchical structure of ICD-10 codes, particularly given its reliance on smaller datasets and a restricted label set [34]. Similarly, [30] used convolutional embeddings to encode semantic relationships for ICD-10 coding of death certificates, achieving an F1 score of 93.4% by incorporating features such as age and gender alongside the clinical text. Although leveraging a large dataset, the embeddings remained specialized to death certificates, reducing their generalizability across broader ICD-10 applications required in diverse healthcare contexts.

Following CNN and RNN models, transformer-based approaches emerged, demonstrating improved performance and flexibility in capturing the rich semantic and contextual relationships present in clinical text. Transformer models like BERT [19, 2], ClinicalBERT [33, 14, 5], and BioBERT [26, 9, 14] have been applied to ICD-10 prediction tasks, where their bidirectional attention mechanisms effectively capture contextual dependencies and improve model performance in multi-label classification. Transformer-based models consistently outperform CNN and RNN methods due to their capacity to handle long-term dependencies and nuanced semantics in clinical text [34]. However, they predominantly rely on equality-based evaluation, where predictions are directly compared to ground truth labels, overlooking clinically similar but non-identical codes [38, 28, 7]. This exact-matching approach often fails to align with real-world clinical needs, where semantically related codes may be more relevant than exact matches.

In addition to these deep learning models, recent vectorization approaches have aimed at generating mathematical representations of ICD codes by integrating contextual and semantic details. [39] introduced *ICD2Vec*, a framework that synthesizes information from disease definitions, clinical information, approximate synonyms, and supplementary data from authoritative sources such as NIH and CDC. Using the GatorTron-OG model, *ICD2Vec* produces 1,024-dimensional semantic embeddings of ICD-10 codes. The authors validated this approach through intrinsic evaluation (analogical reasoning) and extrinsic evaluation (disease risk prediction), achieving promising results in correlating ICD codes with biological relationships. However, *ICD2Vec* exhibits limitations as it does not explicitly encode the inherent hierarchical structure of ICD-10 codes and overlooks coding rules, such as gender restrictions for O-codes and age-specific limitations for P-codes. Additionally, *ICD2Vec*'s embeddings lack consistency in representing demographic constraints, with varying depth and coverage across codes, limiting the framework's ability to fully capture ICD-10's clinical specificity.

Another notable approach is Code2Vec, proposed by [24], one of the earliest frameworks to learn representations of ICD codes by applying the GloVe algorithm on insurance claims data to create 25-dimensional embeddings. This approach enabled the identification of disease comorbidities and related diagnoses through clustering, demonstrating effectiveness in capturing meaningful relationships between diagnoses

by using simple co-occurrence patterns in claims data. An example is the ESRD (End-Stage Renal Disease) cluster, where related conditions were successfully grouped together. However, the approach has several key limitations: it relies solely on temporal co-occurrence patterns in claims without incorporating the inherent hierarchical structure of ICD codes, treats each code as an atomic unit without leveraging code descriptions or medical knowledge, produces static embeddings that cannot capture evolving medical relationships, and shows inconsistent performance across different types of conditions, performing better for chronic diseases than for acute conditions.

[20] introduced ICD-Codex, a Python library that leverages node2vec [13] to learn vector representations of ICD codes by exploiting their inherent hierarchical structure, providing both pre-computed embeddings and functionality to generate new ones to help address ICD miscoding challenges. The approach represents ICD codes as nodes in a hierarchy using networkx graphs and learns representations through node2vec's biased random walks, allowing the model to capture both structural relationships defined by the ICD taxonomy and broader semantic similarities. Despite these strengths, ICD-Codex has several limitations: it relies solely on the hierarchical structure without incorporating clinical descriptions or external medical knowledge, treats each ICD code as an atomic unit without leveraging the compositional nature of ICD descriptions, produces static embeddings that cannot adapt to evolving medical knowledge, and has limited ability to model complex relationships beyond the strict parent-child hierarchies defined in ICD taxonomies.

Despite these advancements, existing approaches rely primarily on isolated code vectors or equality-based evaluation, restricting their capacity to capture nuanced clinical relationships within the ICD-10 coding system [40, 18, 10]. Semantic-based methods that encode ICD-10 codes within both hierarchical and context-aware frameworks remain underexplored. To address these gaps, we introduce *NNBSVR*, a novel approach that integrates hierarchical structure and clinical context to produce embeddings capturing ICD-10 semantics with greater accuracy and robustness, enhancing both prediction and evaluation.

3 Models and Materials

This section presents our comprehensive approach, *NNBSVR*, for enhancing the representation and evaluation of ICD-10 codes in healthcare. Our method comprises two main components: ICD-10 code vectorization and customization of evaluation metrics. The code vectorization process aims to generate effective vector representations for ICD-10 codes that capture their intrinsic semantics and relationships. By converting ICD-10 codes into numerical vectors, we enable comparisons based on semantic meaning rather than equality-based comparisons, allowing for a more accurate and meaningful analysis of ICD-10 codes in various applications. The second component focuses on refining the evaluation metrics to provide a more precise assessment of performance.

3.1 Code Vectorization

The code vectorization process transforms ICD-10 codes into numerical vectors that capture intrinsic semantics and relationships, enabling more meaningful comparisons than traditional equality-based methods. This process involves three key steps: dataset generation, feature selection, and vector generation.

3.1.1 Dataset Generation

We engineered a comprehensive set of features for ICD-10-AM codes, drawing from implicit code semantics and external knowledge sources. Crucially, we utilized the Australian Coding Standards (ACS) dataset [16], which provides detailed guidelines on the proper usage of ICD-10-AM codes. Working closely with medical coding experts, we extracted over 38 new features from the ACS, including gender and age restrictions, as well as constraints related to the nature of the codes, such as maternity, cancer, dental, dagger-asterisk relationships, road traffic incidents, work-related injuries, complications, and malignancy status.

These ACS-derived features were combined with features capturing hierarchical relationships and information embedded in the code structure itself. Additionally, we included the full textual description of each ICD-10-AM code and split each code into multiple components to capture hierarchical relationships. This approach resulted in a rich feature set of 68 features for each code, encapsulating both the explicit structure of ICD-10-AM and the implicit knowledge required for accurate coding.

Table 1 shows some of these features that capture the restrictions related to each ICD-10 code and its usage. Each row in Table 1 describes a feature (name and description columns), gives an example of an ICD-10 code impacted by this feature's restriction, and indicates the type of this feature (boolean, categorical, etc.). For instance, the "Child Only Usage" feature, for a given ICD-10 code, indicates if this code can only be used to describe medical conditions affecting a child. An example of such an ICD-10 code is *P90*, which represents *Convulsions of newborn* and is applicable only to children.

These features were formulated with guidance from medical coding experts to encapsulate rules and semantic criteria relevant to ICD-10-AM codes. For instance, medical coders determined that codes beginning with the letter "O" are exclusively applicable to females for conditions associated with gestation, parturition, and postpartum recovery, and thus the "Female Only Usage" attribute was annotated as *True* for these codes. Similarly, ICD-10-AM codes starting with "P" are reserved for neonates (under 30 days), resulting in the "Newborn Only Usage" feature being set to *True* for these codes.

3.1.2 Feature Selection

Our feature selection process aimed to create a comprehensive representation of ICD-10-AM codes. The selected features can be categorized as follows:

- *Usage features (such as age and gender restrictions)*: Incorporate clinical and demographic constraints on when codes apply, capturing their nuances and specificities.

Name	Description	Example of ICD-10 code	Type
Child Only usage	Used to describe diseases and medical conditions that only affect children	Z00.1 (Routine child health examination)	Boolean
Female Only usage	Used to describe diseases and medical conditions that only affect females	C52 (Malignant neoplasm of vagina)	Boolean
Male Only usage	Used to describe diseases and medical conditions that only affect males	N40 (Hyperplasia of prostate)	Boolean
Maternity Only usage	Used to describe diseases and medical conditions related to pregnancy, childbirth, and the puerperium	O00.8 (Other ectopic pregnancy)	Boolean
Adult Only usage	Used to describe diseases and medical conditions that only affect adults	H25.9 (Senile cataract, unspecified)	Boolean
NewBorn Only usage	Used to describe diseases and medical conditions that only affect newborns	P05.0 (Light for gestational age)	Boolean
Surgical Only usage	Used to describe surgical procedures performed on patients	J95.03 (Leak from tracheostomy)	Boolean
Benign Only usage	Used to describe benign neoplasms (tumors) that are not cancerous or life-threatening	D10.4 (Benign neoplasm of tonsil)	Boolean
InSitu Only usage	Used to describe in situ neoplasms (tumors), which are non-invasive and have not spread beyond their original location in the body	D01.2 (Carcinoma in situ of rectum)	Boolean
Malignant Only usage	Used to describe malignant neoplasms (tumors), which are cancerous and can spread throughout the body if left untreated	C00.8 (Overlapping malignant lesion of lip)	Boolean
Metastatic Only usage	Used to describe metastatic neoplasms (tumors), which have spread from their original location in the body to other parts of the body through the bloodstream or lymphatic system	C78.0 (Secondary malignant neoplasm of lung)	Boolean
Congenital Only usage	Used to describe conditions present at birth	A50.6 (Late congenital syphilis, latent)	Boolean
Pregnancy Only usage	Used to describe diseases and medical conditions related to pregnancy, childbirth, and the puerperium	O80 (Single spontaneous delivery)	Boolean
Psychiatric Only usage	Used to describe psychiatric disorders and mental health conditions	F01.1 (Multi-infarct dementia)	Boolean
Vaccination Only usage	Used to describe vaccinations and immunizations given to patients	Z25.1 (Need for immunisation against influenza)	Boolean
WorkRelated Only usage	Used to describe work-related injuries or conditions	U73.01 (Activity, while working for income, mining)	Boolean
Specialty	Used to describe whether this code is used in specific departments	K02.1 (Caries of dentine) (Dental)	Multiple Categorical

Table 1: Sample of the additional feature criteria used to classify the ICD-10 codes during the code vectorization process.

- *Severity features (such as malignancy and complications)*: Represent clinical severity, acuity, and impact of conditions, which is important for medical relevance.
- *Procedure-related features (such as surgical and follow-up care)*: Distinguish procedural vs diagnostic codes, capturing the relevancy required for predictive tasks.
- *Specialty features (such as dental and obstetrics)*: Specify departmental/specialty areas. It could aid clinical clustering and relevance matching.
- *Descriptor features (such as code descriptions)*: Encode textual information about the disease/diagnosis semantics, critical for capturing semantic meaning.
- *Hierarchy features (splitting the code into prefixes and suffixes)*: Capture the hierarchical taxonomic structure of ICD-10 codes. Codes with similar prefixes likely share common categories. For example, the ICD-10 code *B37.81* (candidal oesophagitis) is split into five features *[B,3,7,8,1]*.

3.1.3 Vector Generation

The process of vector generation involves transforming the dataset into a high-dimensional space that effectively represents the complex relationships and semantics of ICD-10-AM codes. This section covers how we reduced dimensionality and trained the model to create these embeddings, ensuring that our representation captures both explicit and implicit clinical knowledge in a compact form.

- **Dimension Calculation**: In this step, we first used the categorical to numerical conversion for some features like severity and specialty. Then, we applied the min-max normalization to be on a similar scale. Finally, we used the *TruncatedSVD* function from the *Scikit-Learn* [25] python library that helped us in dimensional reduction. *TruncatedSVD* is a matrix factorization technique that can transform the data to a lower dimensional space while preserving most of its variance. It does this by keeping only the most significant singular vectors, which are linear combinations of the original features that capture the most variance in the data. To determine the optimal number of features to keep, we used a method similar to the

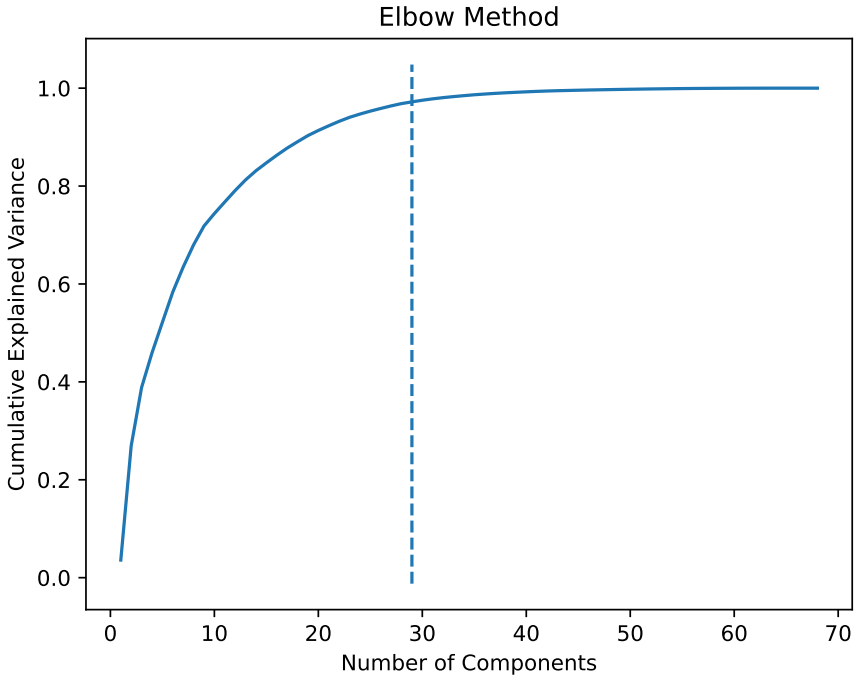


Fig. 1: Optimal Number of Components Determined by Truncated SVD and the Elbow Method.

Elbow one [36], commonly used in cluster analysis to find the optimal number of clusters. We plotted the cumulative explained variance as a function of the number of components and looked for the point where the explained variance increase was lower than a significant threshold as more components are added. This point, known as the *elbow*, represents a good trade-off between having a manageable number of features and preserving as much information as possible.
Moreover, to find the elbow point more objectively, we used the *KneeLocator* [36] algorithm from the *kneed* [3] Python library. This function fits a line from the first to the last point of the curve and finds the data point farthest from this line, corresponding to the elbow point as shown in Fig. 1. This approach allowed us to determine in a methodical and reproducible way, the optimal number of features to keep.

- **Vector Training**: We used a deep learning approach using a BERT pretrained model for clinical textual data (ClinicalBERT [19]). The ClinicalBERT model was used as an embedding layer, followed by a pooling layer and an output classifier layer. The learning rate parameter was set to $5e^{-5}$, the batch size to 32, the maximum sequence length to 150 tokens, and fine-tuned the model for five epochs. These hyperparameters were chosen based on the recommended settings for fine-tuning

BERT models and to strike a balance between model performance and computational efficiency. The output layer consisted of 29 dimensions calculated using the *TruncatedSVD* and *KneeLocator* functions as shwon in Fig. 1. Also, we used the *Adam* optimizer and the *Mean Square Error (MSE)* loss function. It is worth mentioning that MSE is commonly used as a loss function for regression tasks where the output is a continuous vector. Cross-entropy is more suited for classification where the output is class probabilities. Since we are interested in generating a n-dimensional vector representation of the ICD-10 codes, MSE is more appropriate.

To summarize, the first step in our approach encompasses dataset generation, feature selection, and vector generation, resulting in a comprehensive representation of ICD-10-AM codes. By leveraging the Australian Coding Standards, inherent code structures, and expert knowledge, we created a rich feature set that captures lexicographic, semantic, hierarchical, and clinical aspects of the codes. This multifaceted representation allows us to encode complex medical and statistical patterns crucial for improving the accuracy of downstream predictive tasks.

Through dimensionality reduction and deep learning techniques, we have transformed these features into compact vector embeddings for each ICD-10 code. These embeddings capture the intrinsic semantics and relationships between codes, enabling the application of similarity functions to measure code relevancy. This vector space representation forms the foundation for our next step: modifying evaluation metrics to provide a more nuanced and clinically relevant assessment of ICD-10 code prediction performance.

With these vector representations of ICD-10 codes in hand, we can now move beyond simple equality-based comparisons. In the following section, we describe how we leverage these embeddings to create more nuanced and clinically relevant evaluation metrics for assessing the performance of ICD-10 code prediction models.

3.2 Evaluation metric Customization

Traditional evaluation metrics for multi-label classification tasks, such as accuracy, precision, recall, and F1-score, are typically calculated using a confusion matrix based on exact matches between predicted and true labels. However, in the context of ICD-10 code prediction, this approach can be problematic due to the semantic relationships between codes and the imbalanced nature of the dataset. We propose a novel approach that leverages our vector representations to provide a more nuanced evaluation.

Consider a set of n samples $X = \{x_i\}_{i=1}^{n}$ with true-label sets $\mathcal{Y} = \{y_i\}_{i=1}^{n}$ and predicted-label sets $\hat{\mathcal{Y}} = \{\hat{y}_i\}_{i=1}^{n}$, where sets y_i and $\hat{y}_i$ represent respectively the true and predicted sets of labels for sample x_i. Let $\Lambda = \{\lambda_j\}_{j=1}^{m}$ be the set of all unique ICD-10 codes, where m is the total number of unique codes. Each ICD-10 code $\lambda_j \in \Lambda$ is mapped to a d-dimensional vector representation $v_j = f(\lambda_j)$ through a function $f : \Lambda \to \mathbb{R}^d$. We denote $|y_i|$ as the cardinality of the true-labels set y_i, and $|\hat{y}_i|$ as the cardinality of the predicted-labels set $\hat{y}_i$. Therefore, the sets y_i and $\hat{y}_i$ can be expressed as:

$$y_i = \{y_{ij}\}_{j=1}^{|y_i|}$$

$$\hat{y}_i = \{\hat{y}_{ij}\}_{j=1}^{|\hat{y}_i|}$$

where j is the j-th label in the true label set y_i and the j-th label in predicted-label set $\hat{y}_i$ for sample x_i.

Each true label $y_{ij} \in y_i$ and predicted label $\hat{y}_{ij} \in \hat{y}_i$ is mapped to their vector representations $v_{ij} = f(y_{ij})$ and $\hat{v}_{ij} = f(\hat{y}_{ij})$ respectively. The cosine similarity between these vector representations is calculated as:

$$\cos(v_{ij}, \hat{v}_{ij}) = \frac{v_{ij}^\top \hat{v}_{ij}}{\|v_{ij}\|_2 \|\hat{v}_{ij}\|_2}$$

where $\|v_{ij}\|_2$ and $\|\hat{v}_{ij}\|_2$ are the L_2 norms of v_{ij} and $\hat{v}_{ij}$ respectively. Let τ be a tunable threshold hyperparameter in the range [0,1] that controls the strictness of the matching criteria, with lower values allowing more dissimilar vectors to match and higher values requiring stronger similarity to be considered as a match.

The binary indicator δ_{ij}, which determines whether the predicted and true label vectors are considered relevant, is defined as:

$$\delta_{ij} = \begin{cases} 1, & \text{if } \cos(v_{ij}, \hat{v}_{ij}) \geq \tau \\ 0, & \text{otherwise} \end{cases}$$

Using this framework, we can redefine the components of our confusion matrix as follows:

- **True Positive (TP)**: The number of true labels for which there is a corresponding predicted label with cosine similarity above the threshold τ. Each true label is counted only once if at least one matching predicted label is found.

$$TP_i = \sum_{j=1}^{|y_i|} \max_k \delta_{ijk}$$

  Here, $\max_k \delta_{ijk}$ ensures that each true label y_{ij} is counted as a match only if at least one predicted label $\hat{y}_{ik}$ meets the similarity threshold. We use $\max_k$ because we are checking across all predicted labels (indexed by k) for each true label (indexed by j).
- **False Positive (FP)**: The number of predicted labels that do not match any true label above the threshold. This identifies predicted labels that are incorrectly identified as related.

$$FP_i = \sum_{k=1}^{|\hat{y}_i|} \left(1 - \max_j \delta_{ijk}\right)$$

  This formula ensures that each predicted label $\hat{y}_{ik}$ is counted as a false positive if it does not sufficiently match any true label. We sum over k (predicted labels) and use $\max_j$ to check across all true labels (indexed by j) for each predicted label. If no true label matches the predicted label ($\max_j \delta_{ijk} = 0$), it is counted as a false positive.

- **False Negative (FN)**: The number of true labels that do not have any matching predicted label with cosine similarity above the threshold τ. This indicates true labels that were missed.

$$FN_i = \sum_{j=1}^{|y_i|} \left(1 - \max_k \delta_{ijk}\right)$$

This calculation ensures that each true label y_{ij} is counted as a false negative if no predicted label sufficiently matches it. We sum over j (true labels) and use $\max_k$ to check across all predicted labels (indexed by k) for each true label. If no predicted label matches the true label ($\max_k \delta_{ijk} = 0$), it is counted as a false negative.

- **True Negative (TN)**: The number of cases correctly identified as unrelated. This value is generally derived based on the other components.

We chose the F1-score as our primary evaluation metric for several reasons. Firstly, our dataset is imbalanced, with some ICD-10 codes appearing much more frequently than others. In such scenarios, accuracy can be misleading as a model could achieve high accuracy by simply predicting the most common classes [32]. The F1-score, being the harmonic mean of precision and recall, provides a more balanced assessment of performance [6]. Secondly, in the context of ICD-10 coding, both false positives (incorrectly assigned codes) and false negatives (missed codes) can have significant clinical implications. The F1-score equally weights precision and recall, ensuring that our evaluation considers both types of errors. Lastly, by calculating both micro and weighted F1-scores, we can assess overall performance across all classes (micro) while also accounting for class imbalance (weighted).

$$\begin{aligned}
\text{Precision}_j &= \frac{\sum_{i=1}^{n} TP_{ij}}{\sum_{i=1}^{n} TP_{ij} + \sum_{i=1}^{n} FP_{ij}} \\
\text{Recall}_j &= \frac{\sum_{i=1}^{n} TP_{ij}}{\sum_{i=1}^{n} TP_{ij} + \sum_{i=1}^{n} FN_{ij}} \\
\text{F1-Score}_j &= \frac{2 \cdot \text{Precision}_j \cdot \text{Recall}_j}{\text{Precision}_j + \text{Recall}_j}
\end{aligned}$$

$$\begin{aligned}
\text{Micro F1-score} &= \frac{2\sum_{i=1}^{n} TP_i}{2\sum_{i=1}^{n} TP_i + \sum_{i=1}^{n} FP_i + \sum_{i=1}^{n} FN_i} \\
\text{Weighted F1-score} &= \frac{\sum_{j=1}^{m} w_j \cdot F1_j}{\sum_{j=1}^{m} w_j}
\end{aligned}$$

where w_j is the weight of the j-th label, typically set to the proportion of samples with that label in the dataset.

This approach allows us to capture semantic similarities between ICD-10 codes in our evaluation, providing a more nuanced and clinically relevant assessment of model

performance. By using the F1-score, we address the challenges posed by imbalanced datasets while still capturing both precision and recall in a single metric.

4 Experiments and Results

In this section, we present the experiments conducted to compare the conventional evaluation metrics, which rely on equality matching, with our novel method that transforms labels into vectors and employs cosine similarity for predicting ICD-10 codes from clinical text. Furthermore, we present the results of the review conducted by a medical team from a private Saudi hospital, where they have verified 10,000 predictions made using our approach *NNBSVR* at three distinct thresholds.

4.1 Dataset

We acquired a dataset containing 9.57 million clinical notes for training and evaluation from a private Saudi hospital. The clinical notes were annotated with relevant ICD-10 diagnosis and procedure codes by physicians and medical coders experts, with an average of 4 codes assigned per note. The total label space consisted of 9,278 unique ICD-10 codes covering various diagnoses, procedures, and medical conditions. The data was split into training, validation, and test sets in a 70/10/20 ratio for model training and evaluation.

Prior to analysis, we preprocessed the dataset using the Unified Term Presentation (UTP) tool [15]. This tool performed several crucial transformations on the clinical textual data. It converted medical abbreviations to their expanded forms and unified medical terms to ensure consistency throughout the corpus. Additionally, *UTP* transformed dates into standardized periods and converted investigation values embedded in the clinical text into categorical representations. This preprocessing step significantly enhanced the readability and standardization of the dataset for machine learning tasks, reducing ambiguity and ensuring consistency across the corpus, and reducing the vacabulary size.

The primary aim of this study is to conduct an empirical comparison between our relevancy-based approach and the traditional equality-based metrics which are widely used in the field. This comparison will highlight the potential advantages of our method and provide a comprehensive understanding of its performance in a real-world clinical setting.

4.2 Experiments Setup

The experiments were run on the hospital's computing node which is dedicated for machine learning tasks. The clinical notes were encoded using a fine-tuned Clinical-BERT language model pre-trained on biomedical text [19]. By using the hospital's compute node, we ensured that no sensitive data was sent to an external storage space. The model architecture was implemented in Keras and TensorFlow. A 5-fold cross-validation approach was utilized to robustly evaluate the model's performance. We compared two evaluation approaches: standard equality-based matching versus cosine

similarity between prediction and true label vectors. The cosine similarity metric was assessed at 0.6, 0.7, and 0.8 thresholds.

Table 2 presents the key hyperparameters used in our classification experiments. These parameters were carefully selected based on preliminary experiments and best practices in the field of medical text classification.

Parameter	**Value**
Embedding Layer	ClinicalBERT
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Early Stopping Patience	5
Number of Folds (Cross-Validation)	5
Number of Epochs (max)	50
Cosine Similarity Threshold	0.6, 0.7, 0.8, 0.9

Table 2: Key hyperparameters used in the classification experiments.

4.3 Cosine Similarity Validation on a Real World dataset

The results presented in Table 3 demonstrate performance improvements across three approaches: traditional equality matching (EM), *ICD2Vec*, and our *NNBSVR* method. The comparison reveals a clear progression in effectiveness, with *NNBSVR* consistently outperforming both baselines across different evaluation metrics and similarity thresholds. Looking at the micro-F1 scores on the testing set, we see that while EM achieves 74.64%, *ICD2Vec* with a 0.8 threshold reaches 82.48%, and *NNBSVR* at 0.8 threshold achieves 84.14%. This pattern of improvement is consistent across different metrics and thresholds, suggesting that our approach successfully captures more nuanced relationships between ICD-10 codes.

The comparative analysis reveals interesting patterns across different evaluation settings. On the training set, *NNBSVR* achieved micro-F1 and weighted-F1 scores of 91.97% and 90.06% respectively, significantly outperforming both equality matching (83.75% and 84.31%) and *ICD2Vec*'s best performance (87.28% and 85.93% at 0.8 threshold). This pattern continued in the testing set, where *NNBSVR* maintained superior performance with 84.14% and 82.10% for micro-F1 and weighted-F1, compared to *ICD2Vec*'s 82.48% and 70.53%, and EM's 74.64% and 72.01%. The consistent superior performance of *NNBSVR* across both training and testing sets, particularly its ability to maintain high weighted-F1 scores, suggests better handling of class imbalance and more robust generalization.

Both *NNBSVR* and *ICD2Vec* showed sensitivity to similarity thresholds, but with distinct patterns. For *NNBSVR*, we observed steady improvements as the threshold increased from 0.6 to 0.8, with optimal performance at 0.8 before declining at 0.9. *ICD2Vec* showed a similar pattern but with lower overall performance and more pronounced degradation at higher thresholds. Specifically, *ICD2Vec*'s performance

Training Results					
Exp.	**Recall**	**Precision**	**F1-micro**	**F1-Weighted**	**Epochs**
EM	75.34 ± 5.66e-03	92.51 ± 6.23e-03	83.75 ± 5.81e-03	84.31 ± 6.52e-03	22
ICD2Vec 0.6	76.25 ± 5.31e-03	92.12 ± 5.54e-03	83.39 ± 5.40e-03	81.22 ± 6.38e-03	22
ICD2Vec 0.7	78.41 ± 5.58e-03	92.81 ± 5.89e-03	84.89 ± 4.67e-03	82.73 ± 6.01e-03	21
ICD2Vec 0.8	82.85 ± 5.43e-03	93.18 ± 5.25e-03	87.28 ± 5.08e-03	85.93 ± 5.28e-03	21
ICD2Vec 0.9	76.89 ± 4.87e-03	92.62 ± 4.76e-03	83.61 ± 4.42e-03	81.79 ± 4.64e-03	22
REL 0.6	81.09 ± 5.76e-03	93.98 ± 5.88e-03	87.10 ± 4.62e-03	85.93 ± 5.93e-03	19
REL 0.7	86.03 ± 5.89e-03	94.13 ± 5.80e-03	91.15 ± 4.43e-03	89.86 ± 6.17e-03	18
REL 0.8	86.62 ± 4.98e-03	94.72 ± 5.69e-03	91.97 ± 5.01e-03	90.06 ± 5.87e-03	18
REL 0.9	77.10 ± 5.07e-03	93.02 ± 5.71e-03	84.33 ± 4.89e-03	82.68 ± 5.96e-03	19
Testing Results					
EM	68.14 ± 4.52e-03	81.65 ± 3.52e-03	74.64 ± 2.28e-03	72.01 ± 2.20e-03	1
ICD2Vec 0.6	68.45 ± 3.94e-03	81.80 ± 3.43e-03	74.73 ± 3.14e-03	73.78 ± 3.49e-03	1
ICD2Vec 0.7	70.05 ± 3.83e-03	85.42 ± 4.02e-03	76.69 ± 4.08e-03	74.97 ± 4.06e-03	1
ICD2Vec 0.8	74.86 ± 4.11e-03	88.29 ± 4.29e-03	82.48 ± 4.58e-03	70.53 ± 4.32e-03	1
ICD2Vec 0.9	68.92 ± 4.80e-03	81.71 ± 4.12e-03	74.06 ± 4.34e-03	72.98 ± 4.30e-03	1
REL 0.6	73.83 ± 4.35e-03	86.43 ± 3.88e-03	79.83 ± 4.07e-03	77.64 ± 4.24e-03	1
REL 0.7	76.65 ± 5.28e-03	89.19 ± 3.79e-03	83.59 ± 3.09e-03	81.74 ± 3.15e-03	1
REL 0.8	78.14 ± 4.26e-03	90.28 ± 3.75e-03	84.14 ± 2.94e-03	82.10 ± 3.05e-03	1
REL 0.9	69.60 ± 4.39e-03	82.05 ± 3.89.e-03	75.33 ± 3.34e-03	74.81 ± 3.54e-03	1

Table 3: Comparison of ICD-10 prediction results using the *ClinicalBert* model on 9.57M samples. It outperforms traditional equality-based metrics *EM* with relevancy-based metrics *ICD2Vec* and *REL* at ratios of 0.6, 0.7,0.8, and 0.9 based on our ICD-10 vectors.

peaked at 0.8 with a micro-F1 score of 82.48% but dropped sharply to 74.06% at 0.9, while *NNBSVR* maintained better stability with a smaller decrease from 84.14% to 75.33%. This suggests that *NNBSVR*'s vector representations capture more robust and consistent semantic relationships between ICD-10 codes.

An important advantage of *NNBSVR* emerges in training efficiency. While the equality-based method required 22 epochs to converge, and *ICD2Vec* needed 2122 epochs depending on the threshold, *NNBSVR* consistently achieved convergence in fewer epochs (1819 epochs across all thresholds). This improved training efficiency is particularly noteworthy given that *NNBSVR* performs more complex vector comparisons than both baseline methods. The faster convergence suggests that *NNBSVR*'s richer feature set and more informative vector representations enable more efficient learning despite the increased computational complexity per iteration. Furthermore, both *NNBSVR* and *ICD2Vec* showed better precision than EM across all thresholds, with *NNBSVR* achieving the highest precision of 94.72% at 0.8 threshold compared to *ICD2Vec*'s 93.18%, indicating more reliable predictions.

These results demonstrate that leveraging semantic similarity through vector representations and cosine similarity provides substantial gains over exact match evaluations, achieving up to 12.73% ($\frac{84.14-74.64}{74.64}$) improvement in micro-F1 score on the testing set [14]. By representing codes as semantic vectors and using cosine similarity, we can effectively quantify the relatedness between predicted and true labels. Our relevancy-based evaluations accurately score reasonable mismatches, unlike traditional

equality-based metrics that harshly penalize valid predictions for minor label discrepancies. Notably, this performance improvement was more pronounced in the testing set compared to the training set, indicating strong generalization capabilities.

4.4 Validation on Challenging ICD-10-AM codes

Certain ICD-10-AM codes present unique challenges for machine learning models due to their broad or unspecified nature. These codes often lack specific clinical details that distinguish one condition from another, leading to potential misclassification, particularly when clinical records provide minimal contextual information. Here, we analyze some of these challenging codes and highlight examples where additional features in the *NNBSVR* approach improve model accuracy.

For instance, *J18.9* (Pneumonia, unspecified) is commonly misclassified because it encompasses a wide range of pneumonia types, including bacterial, viral, and aspiration pneumonia. This ambiguity can lead to overlaps with codes such as *J15.9* (Bacterial pneumonia, unspecified), especially in the absence of etiological information. Similarly, *F41.9* (Anxiety disorder, unspecified) is challenging to classify accurately due to symptom overlap with other mental health conditions like *F43.9* (Reaction to severe stress) and *F32.9* (Depressive episode, unspecified), which also involve symptoms such as anxiety and restlessness.

Another example is *M54.5* (Low back pain), a general term for back pain that may be confused with other similar codes like *M54.9* (Dorsalgia, unspecified) and *M51.9* (Intervertebral disc disorder, unspecified). Without specific clinical information on pain location or cause, distinguishing these codes can be difficult. Likewise, *R10.4* (Other and unspecified abdominal pain) is frequently ambiguous, as abdominal pain can indicate a range of conditions from *R10.9* (Unspecified abdominal pain) to specific digestive disorders like *K29.7* (Gastritis, unspecified). In neurology, *G93.9* (Disorder of brain, unspecified) represents a broad category that may overlap with conditions such as *R40.4* (Transient alteration of awareness) or *G30.9* (Alzheimer's disease, unspecified), where more detailed neurological symptoms or patient history would aid in precise classification. Finally, *K40.90* (Unilateral or unspecified inguinal hernia without obstruction or gangrene, not specified as recurrent) may be misclassified without information on laterality or recurrence, potentially overlapping with *K40.20* (Bilateral inguinal hernia, not specified as recurrent) or *K41.90* (Unilateral or unspecified femoral hernia, not specified as recurrent).

Table 4 presents a comparison of the accuracy rates for traditional equality-based matching *EM*, *ICD2Vec*, and *NNBSVR* on a set of ambiguous ICD-10-AM codes. The *EM* approach, which relies solely on exact code matching, struggles with these broad codes, as evidenced by relatively low accuracy percentages across all examples. *ICD2Vec*, which incorporates disease information and symptoms to generate vector representations, shows modest improvements in accuracy. For instance, it achieves a higher accuracy of 67.48% for J18.9 and 54.11% for *R10.4* compared to *EM*, demonstrating that additional clinical context can assist in refining classifications.

However, *NNBSVR*, which uses a more sophisticated feature set that includes demographic, contextual, and hierarchical data, achieves the highest accuracy across all codes. For *J18.9* (Pneumonia, unspecified), *NNBSVR* achieves an accuracy of

Code	Description	EM	ICD2Vec	NNBSVR
J18.9	Pneumonia, unspecified	63.59 %	67.48%	73.15%
F41.9	Anxiety disorder, unspecified	53.17%	53.98%	55.41%
M54.5	Low back pain	57.43%	58.87%	63.84%
R10.4	Other and unspecified abdominal pain	50.38%	54.11%	58.32%
G93.9	Disorder of brain, unspecified	49.08%	51.08%	51.16%
K40.90	Unilateral or unspecified inguinal hernia without obstruction or gangrene, not specified as recurrent	59.86%	62.23%	67.51%

Table 4: Comparison of *EM*, *ICD2Vec*, and *NNBSVR* Accuracy on Challenging ICD-10-AM Codes

73.15%, indicating an improved ability to handle this ambiguous diagnosis compared to *ICD2Vec* and *EM*. Similarly, *NNBSVR* outperforms both methods for codes like *M54.5* (Low back pain) and *K40.90* (Unilateral or unspecified inguinal hernia), where added contextual and demographic features assist in disambiguating these diagnoses.

Overall, these results illustrate that *NNBSVR*'s rich feature set enhances the model's ability to address ambiguities inherent in broad ICD-10-AM codes. *NNBSVR* provides a more nuanced approach to classification, particularly for codes with minimal clinical specificity, and demonstrates significant potential for improving automated ICD-10-AM coding in clinical practice.

4.5 Validation by Visualization

To validate our ICD-10 vector representation approach, we used two different visualization techniques: t-SNE and DenMune clustering algorithm. These methods provide both qualitative and quantitative evidence for the effectiveness of our extended feature set in capturing clinically relevant patterns and relationships.

4.5.1 t-SNE Visualization

We compared the proposed ICD-10 vector representation approach using extended features against a baseline using just the hierarchical code structure. We visualized both approaches using t-SNE [37] plots as shown in Fig 2. The extended features approach *NNBSVR* led to more clearly defined clusters with visible gaps between groups. In contrast, the hierarchical approach overlapped clusters, indicating less clear separation. This provides a useful qualitative validation that the extended feature set better captures clinically relevant patterns and relationships than the taxonomic code structure alone. The well-separated clusters imply that semantically related codes are grouped together, while irrelevant codes are appropriately distant.

4.5.2 DenMune Clustering Analysis

To further validate our approach, we employed the DenMune clustering algorithm [1] to analyze the effectiveness of different feature sets in capturing the underlying structure of ICD-10 codes. DenMune is a density-based clustering algorithm that leverages mutual nearest neighbors (MNN) to improve the detection of clusters, particularly

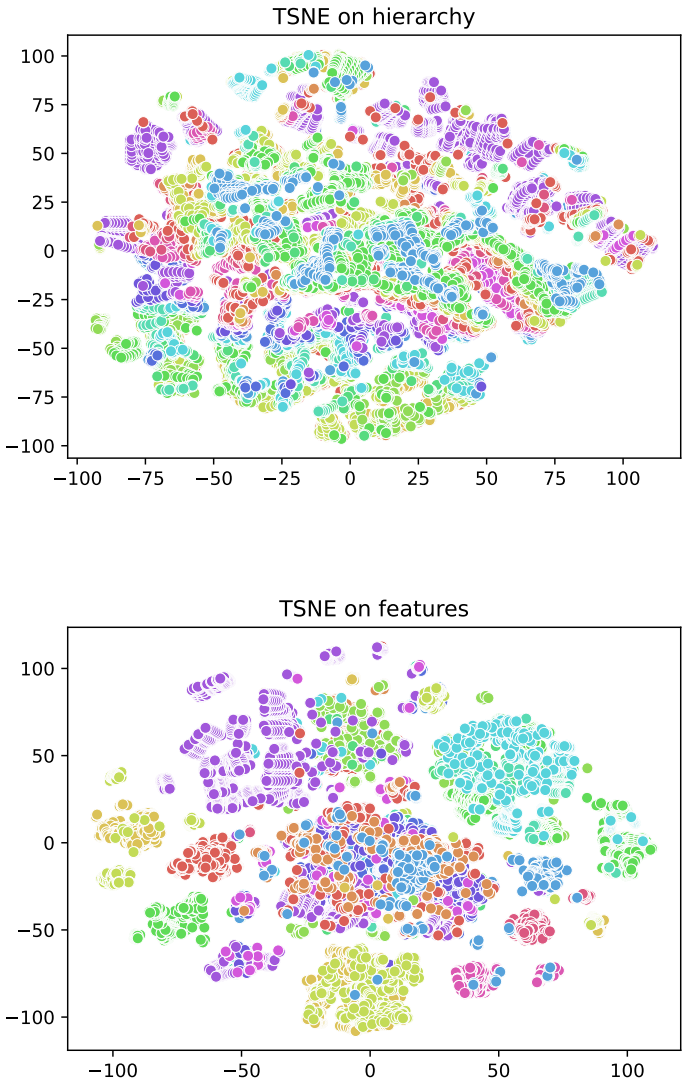
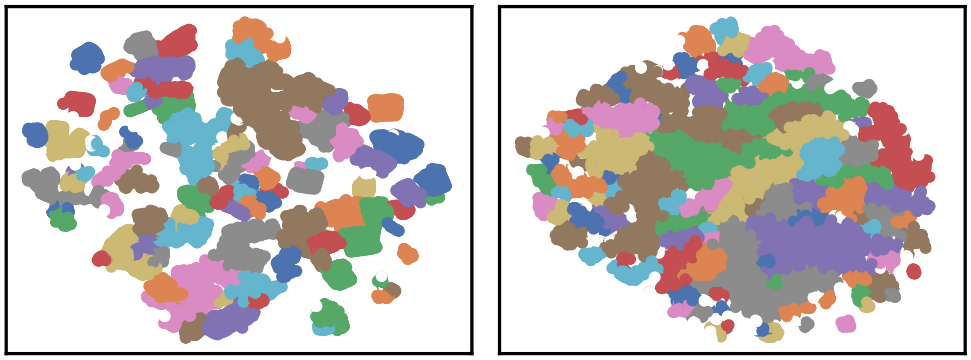


Fig. 2: Comparing Cluster Visualizations of ICD-10 Vectors: Hierarchical vs. Hierarchical with Extended Features

in high-dimensional data. Unlike conventional clustering methods, it identifies dense regions in the data and enhances the merging of weakly connected points, making it particularly suited for complex hierarchical datasets such as ICD-10 codes. By applying DenMune, we assessed the structural coherence of our vectorized ICD-10 representations, ensuring that similar diagnoses were grouped together while minimizing noise and erroneous classifications. We compared three scenarios: hierarchy only, hierarchy with additional attributes, and hierarchy with attributes and ICD-10 descriptions. The results demonstrated that incorporating extended features significantly improved cluster separation, leading to a more meaningful organization of *ICD-10* codes that better reflected their clinical relationships. These findings reinforce the validity of our approach in structuring medical codes based on their semantic and contextual similarities.



(a) Hierarchy Only (b) Hierarchical with Extended Features

Fig. 3: DenMune Clustering Visualizations for Different Feature Sets

Figure 3 presents the DenMune clustering visualizations for two scenarios. The progression from (a) to (b) demonstrates increasing cluster definition and separation, indicating that each additional set of features contributes to a more nuanced representation of the ICD-10 codes.

Table 5 provides a detailed comparison of the DenMune algorithm results across the three scenarios. Key observations include:

A higher number of detected clusters (100 to 127) as more features are incorporated, suggesting a more fine-grained representation of the ICD-10 code space. Significant increase in dimensions (5 to 68), reflecting the richer feature set provided by our approach. Elimination of type-1 noise and increase in type-2 noise, potentially indicating better identification of outliers or unique cases. Slight decrease in strong points and increase in weak points, suggesting the capture of more nuanced relationships between codes.

These results provide strong evidence for the effectiveness of our *NNBSVR* approach in capturing the complex relationships within the ICD-10 coding system. The

Metric	Hierarchy Only	Hierarchical with Extended Features
Detected Clusters	100	127
Dimensions	5	68
Noise (type-1)	33	0
Noise (type-2)	250	467
Strong Points	9016	8801
Weak Points	7936	8151
Failed to Merge	250	467
Succeeded to Merge	7686	7684

Table 5: Comparison of DenMune Clustering Results between Hierarchical relationship and the hierarchical with extended features

increase in detected clusters and dimensions, along with the changes in point classifications, demonstrate that our method captures subtle distinctions between codes that are not apparent when using only hierarchical information like *ICDCodex*. This enhanced representation is crucial for improving the accuracy of ICD-10 code prediction and related tasks in clinical settings.

4.6 Medical Team Validation

To further validate the accuracy of the ICD-10 predictions using our cosine similarity approach, a team of 8 medical coders manually reviewed a subset of the results. In total, 30,000 predictions were evaluated, 10,000 samples for each of the 0.6, 0.7, and 0.8 similarity thresholds.

The selection of these 30,000 samples (10,000 for each threshold) was an intensive process that spanned over two weeks of dedicated effort by the medical coding team. However, the medical coders deliberately included a significant proportion of complex cases that posed specific challenges related to gender restrictions, age limitations, and rules derived from the Australian Coding Standards (ACS). This rigorous selection process ensured that our validation set not only represented the breadth of clinical scenarios but also focused on the nuanced and difficult cases that truly test the capabilities of our approach.

The medical coders assessed whether the predicted ICD-10 codes accurately matched the true labels for each sample. The experts' analysis serves as an additional verification of the quality of our embeddings, besides the gain observed on the metrics.

The validation results, as presented in Table 6, were highly promising, as they demonstrated increased accuracy with higher similarity thresholds. At a threshold of 0.6, the prediction accuracy was 87.31%, according to the manual reviews. This accuracy rose to 90.37% at a threshold of 0.7 and further increased to 92.58% at a threshold of 0.8.

These findings underscore that employing a higher relevance ratio for the cosine similarity metric improves the precision of the ICD-10 predictions. The accuracy validated by the medical team aligns with and further confirms the patterns observed in the quantitative metrics during evaluation. The 92.58% accuracy achieved on the predictions with a 0.8 threshold lends additional support to the efficacy of the proposed approach.

Experiment	Accuracy
Relevancy @0.6	87.31%
Relevancy @0.7	90.37%
Relevancy @0.8	92.58%

Table 6: The outcome of the second manual validation conducted by a medical team on the predicted labels using the relevancy approach.

Overall, this manual review provides reassurance about the real-world applicability of our semantically-aware cosine similarity approach *NNBSVR*. The obtained high accuracy demonstrates that the gains observed translate to enhanced performance in practical clinical coding tasks.

4.7 Comparative Analysis of Our Code Vectorization Approach and *ICDCodex* Using Hierarchical Clustering

To assess the performance of *NNBSVR* in capturing the hierarchical structure and semantic relationships of ICD-10 codes, we conducted a comparative analysis with an existing method called *ICDCodex* [20], a node2vec-based approach focused on hierarchical embeddings [13]. Hierarchical clustering was employed to assess the ability of both methods to reconstruct the inherent hierarchical structure of the ICD-10 coding system, which consists of chapters, blocks, and categories.

We applied the k-means clustering algorithm to the vector representations generated by *NNBSVR* and *ICDCodex*, aiming to reconstruct the ICD-10 hierarchy at different levels of granularity. The number of clusters was set to match the number of chapters, blocks, and categories in the ICD-10 system. By comparing the resulting clusters with the actual ICD-10 structure, we could evaluate the effectiveness of each approach in capturing the semantic relationships and hierarchical organization of the codes.

The accuracy of the clustering results was assessed using the following formula for each level of the hierarchy:

$$\text{Accuracy} = \left(\frac{\text{Number of Correct Matches}}{\text{Total Number of Codes in the Cluster}} \right) \times 100\%$$

The results in Table 7 shed light on the comparative performance of *NNBSVR* and *ICDCodex* in capturing the hierarchical structure of the ICD-10 coding system. At the highest level of the hierarchy, both approaches demonstrate perfect accuracy in reconstructing the 22 main chapters of ICD-10, achieving a flawless 100% match. This indicates that both methods excel at grouping codes into their respective broad categories.

However, as we delve deeper into the hierarchy, *NNBSVR* begins to surpass *ICDCodex*. When clustering the codes into 280 blocks, *NNBSVR* attains an impressive accuracy of 97.15%, surpassing *ICDCodex*'s 91.48%. This notable difference suggests

that *NNBSVR* is more adept at discerning the finer-grained distinctions between related codes, effectively capturing the nuances within each chapter.

The superiority of *NNBSVR* becomes even more pronounced at the most granular level of the hierarchy for the 2,860 categories. Here, *NNBSVR* achieves a remarkable accuracy of 91.86%, considerably higher than *ICDCodex*'s 74.12%. This substantial gap underscores *NNBSVR*'s exceptional ability to identify and group closely related codes, even when the differences are subtle and intricate.

Comparison	Number of Clusters	Our Approach Accuracy	ICDCodex Accuracy
Chapters	22	100.00%	100.00%
Blocks	280	97.15%	91.48%
Categories	2860	91.86%	74.12%

Table 7: Accuracy of Hierarchical Clustering in Capturing ICD-10 Structure between *NNBSVR* and *ICDCodex*

These results provide compelling evidence for the effectiveness of *NNBSVR* in capturing the hierarchical structure and semantic relationships within the ICD-10 coding system. By incorporating a rich set of features and leveraging advanced neural network architectures, *NNBSVR* demonstrates a superior capacity to reconstruct the inherent organization of ICD-10 codes across all levels of granularity.

The significantly higher accuracy achieved by *NNBSVR* at the block and category levels highlights its potential to enhance various clinical applications, such as disease classification, patient cohort identification, and medical knowledge discovery. By more accurately capturing the semantic similarities and differences between ICD-10 codes, *NNBSVR* can facilitate more precise and meaningful analysis in healthcare research and practice.

4.8 Validation on Frequent Codes Mapping

Moreover, we extracted the 50 most frequently used ICD-10 codes in the dataset and manually identified suggested relevant codes for each based on medical expertise. Using our vector representations, we then computed the cosine similarity between these frequent codes and their related codes. At a 0.6 similarity threshold, the accuracy of mapping frequent codes to their relevant codes was 84.56%. This mapping accuracy increased to 88.05% with a 0.7 threshold and 89.18% at a 0.8 threshold. This indicates how well our vectors capture semantic relationships between codes independent of clinical note data.

4.9 Validation via Physician Labeling

Additionally, we asked physicians to provide 20 ICD-10 codes along with possible relevant codes for each. Using our vector representations, we computed the cosine similarities between their provided codes and related codes. The accuracy of mapping

physician-labeled codes to their suggested relevant codes was 89.23% at a 0.6 threshold. This further improved to 90.03% and 91.16% at thresholds of 0.7 and 0.8, respectively. This physician validation demonstrates the ability of our vectors to encode semantic clinical relevance between ICD-10 codes.

Overall, these three validation approaches (the first one based on frequent codes, second one based on the challenging ICD-10-AM codes, and the other one leveraging physician expertise), provide strong quantitative evidence for the quality of our ICD-10 vector representations in capturing semantic relationships. The high mapping accuracy to relevant codes across multiple similarity thresholds indicates the robust encoding of clinical meaning.

5 Discussion

This study introduced *NNBSVR* for generating vector representations of ICD-10 codes with improved prediction accuracy compared to existing methods. The approach differs from prior techniques by directly encoding ICD-10 code semantics rather than relying on patient data or symptoms, enabling more precise code-specific representations. While many existing approaches rely primarily on t-SNE visualizations for evaluation, our introduction of cosine similarity metrics enabled quantitative assessment of the embeddings' quality. The empirical results showed a 12.73% improvement in micro-F1 score over baseline methods, with medical expert validation achieving 90.37% accuracy on a 10,000-prediction sample.

The use of cosine similarity enabled identification of semantically related ICD-10 codes that would be penalized by traditional equality-based metrics. For example, codes J02.9 (Acute pharyngitis) and R07.0 (Pain in throat) were appropriately matched based on their shared clinical context despite having distinct labels. Similarly, related postoperative care codes like Z48.8 and Z48.9 were successfully identified as relevant matches, reducing false positives and negatives in the evaluation process.

5.1 Cost-effectiveness and Computational Complexity

The computational analysis of *NNBSVR* revealed both advantages and trade-offs. Despite increased per-iteration complexity, the model demonstrated efficient training convergence, requiring 1819 epochs compared to 22 epochs for traditional approaches across our dataset of 9.57M clinical notes. This faster convergence suggests the vector representations effectively capture key semantic structures that facilitate learning.

The primary computational overhead occurs during evaluation, where cosine similarity is calculated between predicted and true label vectors. For a multi-label classification task, this introduces complexity of $O(d)$ for each label pair comparison, where d is the vector dimensionality. The total complexity for processing n samples becomes $O(n \times |y| \times |\hat{y}| \times d)$, with $|y|$ and $|\hat{y}|$ representing true and predicted label set sizes respectively. While this exceeds the $O(|y| \times |\hat{y}|)$ complexity of equality-based metrics, the improved accuracy and faster convergence justify this additional cost for applications prioritizing clinical relevance.

The approach's efficiency extends beyond training. Once generated, the vector representations remain stable and reusable across applications unless updates are needed

(newer ICD10-AM version). This characteristic, combined with reduced false positives and negatives through semantic matching, makes *NNBSVR* particularly suitable for healthcare settings balancing accuracy requirements with resource constraints.

5.2 Limitations of the Method

Several limitations warrant consideration. While the study incorporated 3,100 unique ICD-10 codes, the model's performance on rare conditions remains untested. Codes such as Q89.9 (Congenital malformation, unspecified) and X29 (Exposure to excessive natural cold) appeared infrequently in our Saudi Arabian hospital dataset, potentially limiting generalizability to regions where these conditions are more common.

Emergency medicine presents particular challenges due to the variability in injury causes. A single diagnosis code like S52.9 (fracture of the arm) can result from numerous scenarios, each requiring different external cause codes (e.g., V43.0 for traffic accidents versus W21.0 for sports injuries). Without detailed clinical documentation specifying injury context, accurate external cause code prediction remains difficult.

Internal medicine poses similar challenges due to its broad scope and overlapping conditions. The frequent occurrence of chronic conditions like hypertension and diabetes across multiple patient visits, often documented with minimal variation, complicates the model's ability to distinguish between different clinical contexts for the same diagnosis codes.

5.3 Implementation and Future Directions

While we validated *NNBSVR* through ICD-10 code prediction, the core contribution of this work is the vector representation method itself. The prediction task, currently undergoing pilot testing, demonstrates one practical application of these vectors. To enable this application, we developed custom evaluation metrics that leverage the semantic relationships captured in our vector space, an approach that could be adapted for other use cases.

The comparative analysis with *ICD2Vec* and *ICDCodex* revealed important insights about vector representation approaches. While *ICD2Vec*'s vectors capture disease relationships through clinical information and synonyms, they miss crucial coding rules and demographic constraints. Our results showed that *NNBSVR's* more compact 29-dimensional vectors, enriched with coding standards and hierarchical information, outperformed *ICD2Vec* particularly in capturing gender-specific and age-restricted conditions. Similarly, while *ICDCodex* effectively captures hierarchical relationships through *node2vec*, its reliance solely on structural information limits its ability to represent complex clinical relationships. This was particularly evident in our hierarchical clustering analysis, where *NNBSVR* maintained high accuracy (91.86%) even at the most granular category level, compared to *ICDCodex*'s 74.12%.

Looking ahead, a promising direction for improvement lies in the loss function design. Currently, most deep learning approaches for medical coding use equality-based loss functions that treat codes as either matching or non-matching. Developing a custom loss function that incorporates our relevancy-based similarity measures could significantly improve the learning process. Such a loss function would allow the

model to learn from partially correct predictions, recognizing when predicted codes are clinically related to the true codes even if not exact matches. This could be particularly beneficial for handling rare conditions and complex co-morbidities where exact matches are less frequent.

5.4 Extending NNBSVR to Other Medical Classification Systems

The methodology developed for *NNBSVR* can be effectively adapted to other hierarchical medical classification systems, particularly the Anatomical Therapeutic Chemical (ATC) classification system. The *ATC* system's five-level hierarchical structure presents similar vectorization challenges and opportunities as *ICD-10*, making it an ideal candidate for demonstrating our approach's adaptability.

Fig. 4 demonstrates the hierarchical nature of *ATC* classification, showing how Metformin belongs to a broader family of biguanides (*A10BA*), which are blood glucose lowering drugs (*A10B*) used in diabetes (*A10*) within the alimentary tract and metabolism category (*A*). Similar to our ICD-10 implementation, *NNBSVR* would decompose this code into positional features [A, 10, B, A, 02], where each component preserves its position-specific meaning.

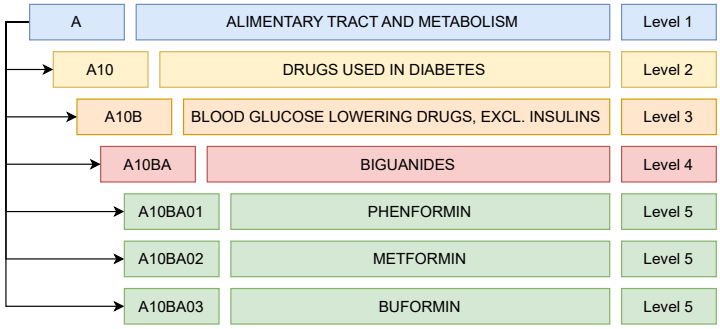


Fig. 4: Hierarchical structure of ATC classification for Metformin (A10BA02), demonstrating the five levels of classification from anatomical main group to specific chemical substance.

Beyond hierarchical features, the vector representation would incorporate usage features (administration route, dosage forms, contraindications) and semantic context (therapeutic indications, mechanism of action). For *Metformin*, this includes its oral administration route, common tablet formulations, and its role as a first-line antidiabetic agent. This rich feature set enables NNBSVR to identify semantically similar medications through cosine similarity calculations, helping healthcare providers to quickly identify therapeutic alternatives when needed.

The implementation for medical coding systems like *ATC* would maintain the same neural network architecture while adjusting dimensionality based on the specific

feature space. Beyond *ATC*, NNBSVR's methodology can be effectively adapted to other healthcare coding systems. The American Dental Association's Current Dental Terminology (*CDT*) [4] codes provide a clear example where each code consists of a letter prefix "D" followed by four digits that denote specific dental procedures and services. This structured format aligns perfectly with our hierarchical decomposition approach.

Similarly, *NNBSVR* can be applied to other standardized medical coding systems such as Current Procedural Terminology (*CPT*) [12] codes for medical procedures, Healthcare Common Procedure Coding System (*HCPCS*) [8, 23] for medical supplies and services, and Australian refind Diagnosis-Related Groups (*AR-DRG*) [21] for hospital reimbursement. Each of these systems features its own hierarchical structure and domain-specific attributes that can be effectively captured using our vector representation approach. For instance, in DRG coding, where relationships between diagnoses and procedures directly impact healthcare reimbursement, *NNBSVR*'s ability to capture nuanced relationships between codes makes it particularly valuable.

The flexibility of *NNBSVR*'s feature engineering process allows it to adapt to these varied coding systems while maintaining its core methodology. The approach can be customized to incorporate system-specific attributes while preserving the fundamental vector generation framework. This adaptability, combined with the demonstrated success in *ICD-10* coding, suggests that NNBSVR could significantly enhance automated coding processes across multiple domains of healthcare administration.

6 Conclusion

This study introduced *NNBSVR*, a method for creating vector representations of ICD-10 codes that capture their semantic relationships and clinical context. The effectiveness of these representations was demonstrated through extensive evaluation: quantitative tests showed a 12.73% improvement in micro-F1 scores over traditional approaches, while medical experts validated the practical accuracy with 92.58% agreement on real-world cases.

The primary achievement of this work is not just improved prediction accuracy, but rather the development of vectors that meaningfully represent relationships between medical codes. By incorporating features from the Australian Coding Standards and hierarchical relationships, these vectors capture clinically relevant patterns that simple structural approaches miss. For instance, the vectors successfully group related conditions while maintaining distinctions between codes that require different clinical interventions, such as hyperkalemia versus hypokalemia in newborns.

Moving forward, we see two key directions for development. First, adapting these vector representations for *DRG* classification, where understanding relationships between diagnoses and procedures directly impacts healthcare reimbursement. Second, developing custom loss functions that incorporate our relevancy-based similarity measures, potentially improving how deep learning models learn from partially correct predictions. These advances could significantly impact both the accuracy and efficiency of automated medical coding systems.

The results suggest that improving how we represent medical codes, rather than just refining prediction algorithms, can significantly advance automated medical coding. This shift in focus from prediction to representation, combined with relevancy-based learning approaches, opens new possibilities for healthcare applications that depend on understanding the complex relationships between medical concepts.

Acknowledgment

The authors would like to express their deepest gratitude to the Medical Coding Department at Specialized Medical Center Hospital in Saudi Arabia. Their unwavering support and dedication have been instrumental in the success of this research. A special note of thanks is extended to the medical coders who have shown exceptional commitment and diligence. Their tireless efforts and invaluable support have significantly enriched our work.

This work has been achieved in the frame of the EIPHI Graduate school (contract "ANR-17-EURE-0002").

Ethics Approval

This study was conducted as part of the ongoing research and quality improvement initiatives at Specialized Medical Center, Saudi Arabia. The study protocol was reviewed and approved by the hospital's Institutional Review Board (IRB).

Data Availability

The clinical notes dataset used in this study was obtained and processed within Specialized Medical Center, Saudi Arabia, in compliance with the hospital's data protection policies and patient confidentiality agreements. Due to the sensitive nature of the data and the hospital's internal regulations, the dataset cannot be made publicly available. Researchers interested in accessing the data for replication or further studies should contact the corresponding author, who will facilitate the necessary institutional approvals and data sharing agreements on a case-by-case basis, subject to the hospital's policies and guidelines.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This research received no external funding.

References

[1] Mohamed Abbas, Adel El-Zoghabi, and Amin Shoukry. Denmune: Density peak based clustering using mutual nearest neighbors. http://dx.doi.org/10.1016/j.patcog.2020.107589, January 2021.

[2] Emran Al-Bashabsheh, Ahmad Alaiad, Mahmoud Al-Ayyoub, Othman Beni-Yonis, Raed Abu Zitar, and Laith Abualigah. Improving clinical documentation: automatic inference of icd-10 codes from patient notes using bert model. *The Journal of Supercomputing*, pages 1–25, 2023.

[3] Kevin Arvai. *kneed*. Zenodo, July 2023.

[4] American Dental Association et al. *CDT 2024: Current Dental Terminology*. American Dental Association, 2023.

[5] Brent Biseda, Gaurav Desai, Haifeng Lin, and Anish Philip. Prediction of icd codes with clinical bert embeddings and text augmentation with label balancing using mimic-iii. *arXiv preprint arXiv:2008.10492*, 2020.

[6] Jasmin Bogatinovski, Ljupčo Todorovski, Sašo Džeroski, and Dragi Kocev. Comprehensive comparative study of multi-label classification methods. http://dx.doi.org/10.1016/j.eswa.2022.117215, October 2022.

[7] Silvio Domingos Cardoso, Marcos Da Silveira, Ying-Chi Lin, Victor Christen, Erhard Rahm, Chantal Reynaud-Delaître, and Cédric Pruski. Combining semantic and lexical measures to evaluate medical terms similarity, December 2018.

[8] Centers for Medicare & Medicaid Services. List of cpt/hcpcs codes, 2025. Accessed: 2025-01-27.

[9] Pei-Fu Chen, Kuan-Chih Chen, Wei-Chih Liao, Feipei Lai, Tai-Liang He, Sheng-Che Lin, Wei-Jen Chen, Chi-Yu Yang, Yu-Cheng Lin, I-Chang Tsai, et al. Automatic international classification of diseases coding system: Deep contextualized language model with rule-based approaches. *JMIR Medical Informatics*, 10(6):e37557, 2022.

[10] Yanran Chen and Steffen Eger. MENLI: Robust Evaluation Metrics from Natural Language Inference. *Transactions of the Association for Computational Linguistics*, 11:804–825, 07 2023.

[11] Peter L. Elkin and Steven H. Brown. *Diagnosis-Related Group (DRG)*, pages 379–393. Springer International Publishing, Cham, 2023.

[12] Richard A. Frank, Robert Jarrin, Jordan Pritzker, Michael D. Abramoff, Michael X. Repka, Pat D. Baird, S. Marlene Grenon, Megan Ruth Mahoney, John E. Mattison, and III Silva, Ezequiel. Developing current procedural terminology codes that describe the work performed by machines. http://dx.doi.org/10.1038/s41746-022-00723-5, December 2022.

[13] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. *arXiv*, 2016.

[14] Monah Hatoum, Jean-Claude Charr, Christophe Guyeux, David Laiymani, and Alia Ghaddar. Emte: An enhanced medical terms extractor using pattern matching rules, 2023.

[15] Monah Bou Hatoum, Jean Claude Charr, Alia Ghaddar, Christophe Guyeux, and David Laiymani. Utp: A unified term presentation tool for clinical textual data

using pattern-matching rules and dictionary-based ontologies, 2024.

[16] Independent Health and Aged Care Pricing Authority. *Australian Coding Standards for ICD-10-AM and ACHI*, 2023. Version 12.0.

[17] JA Hirsch, G Nicola, G McGinty, RW Liu, RM Barr, MD Chittle, and L Manchikanti. Icd-10: history and context. *American Journal of Neuroradiology*, 37(4):596–599, 2016.

[18] Albert Huang. 'tis but thy name: Semantic question answering evaluation with 11m names for 1m entities. *arXiv*, 2022.

[19] Jinhyuk Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*, 2019.

[20] icd-codex contributors. icd-codex: Python library for graphical and continuous representations of icd9 and icd10 codes, 2020.

[21] Independent Hospital Pricing Authority. Ar-drg version 11.0, 2025. Accessed: 2025-01-27.

[22] Nathalie Jetté, Hude Quan, Brenda Hemmelgarn, Saskia Drosler, Christina Maass, Lori Moskal, Wansa Paoin, Vijaya Sundararajan, Song Gao, Robert Jakob, et al. The development, evolution, and modifications of icd-10: challenges to the international comparability of morbidity data. *Medical care*, 48(12):1105–1110, 2010.

[23] Justin M. Johnson and Taghi M. Khoshgoftaar. Encoding high-dimensional procedure codes for healthcare fraud detection. http://dx.doi.org/10.1007/s42979-022-01252-4, July 2022.

[24] David Kartchner, Tanner Christensen, Jeffrey Humpherys, and Sean Wade. Code2vec: Embedding and clustering medical diagnosis data. http://dx.doi.org/10.1109/ICHI.2017.94, August 2017.

[25] Scikit learn developers. sklearn.decomposition.truncatedsvd — scikit-learn 1.3.2 documentation. https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html, 2023. Accessed: 2023-10-31.

[26] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[27] Fei Li, Borui Jin, Weisong Liu, Md Mahfuzur Rahman, Gang Luo, Yi Zhang, and Guoqian Jiang. Icd coding from clinical text using multi-filter residual convolutional neural network. *Journal of biomedical informatics*, 110:103500, 2020.

[28] Meijing Li, Xianhe Zhou, Keun Ho Ryu, and Nipon Theera-Umpon. An ensemble semantic textual similarity measure based on multiple evidences for biomedical documents, August 2022.

[29] Robert Long. Fairness in machine learning: Against false positive rate equality as a measure of fairness. http://dx.doi.org/10.1163/17455243-20213439, November 2021.

[30] Vincenzo Della Mea, Mihai Horia Popescu, and Kevin Roitero. Underlying cause of death identification from death certificates via categorical embeddings and

convolutional neural networks, November 2020.

[31] Brent Mittelstadt, Sandra Wachter, and Chris Russell. The unfairness of fair machine learning: Levelling down and strict egalitarianism by default. https://arxiv.org/abs/2302.02404, 2023.

[32] Hamza Haruna MOHAMMED, Erdogan DOGDU, Abdul Kadir GORUR, and Roya CHOUPANI. Multi-label classification of text documents using deep learning. http://dx.doi.org/10.1109/BigData50022.2020.9378266, December 2020.

[33] James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. Explainable prediction of medical codes from clinical text. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1:1112–1121, 2018.

[34] Pavithra Rajendran, Alexandros Zenonos, Joshua Spear, and Rebecca Pope. Embed wisely: An ensemble approach to predict icd coding, 2021.

[35] Rosemary F Roberts, Kerry C Innes, and Susan M Walker. Introducing icd-10-am in australian hospitals. *The Medical Journal of Australia*, 169(S1):S32–5, 1998.

[36] Ville Satopaa, Jeannie Albrecht, David Irwin, and Barath Raghavan. Finding a "kneedle" in a haystack: Detecting knee points in system behavior, June 2011.

[37] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[38] Genta Indra Winata, Zhaojiang Lin, Jamin Shin, Zihan Liu, and Pascale Fung. Hierarchical meta-embeddings for code-switching named entity recognition, 2019.

[39] Yichen Wu, Finale Doshi-Velez, Michael Schwartz, and Byron Zhang. Icd2vec: Mathematical representation of diseases. *arXiv preprint arXiv:2103.05148*, 2021.

[40] Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. http://dx.doi.org/10.18653/v1/D19-1053, 2019.