Accurate and Well-Calibrated ICD Code Assignment with a Chunk-Based Classifier Attending over Diverse Label Embeddings

Gonçalo Gomes
Dept. Data Science and Engineering
Instituto Superior Técnico
goncaloecgomes@tecnico.ulisboa.pt

Abstract

Although the International Classification of Diseases (ICD) has been adopted worldwide, manually assigning ICD codes to clinical text is time-consuming, error-prone, and expensive, motivating the development of automated method. This paper describes a novel deep learning approach for ICD coding, combining several ideas from previous related work. In particular, we split long clinical documents into chunks, and use a strong Transformer-based model for processing each of the chunks independently. The resulting representations are processed with a max-pooling operation, and combined with a label embedding mechanism that explores diverse ICD code synonyms. Experiments with different splits of the MIMIC-III dataset show that the proposed approach outperforms the current state-of-the-art models in ICD coding, while also leading to properly calibrated model that can effectively inform downstream tasks such as text quantification.

1 Introduction

The International Classification of Diseases (ICD[1]) coding system, proposed by the World Health Organization, stands as a universally embraced standard for precise documentation in the medical domain (O'malley et al., 2005). Still, the manual assignment of ICD codes to clinical text is a time-consuming, labor intensive, and error-prone task, which has led to the exploration of automated coding methods, e.g. using deep learning algorithms for text classification.

Despite many previous efforts, automatic ICD coding is still challenging, since clinical notes consist of long text narratives, using a specialized medical vocabulary, and that are associated to a high dimensional, sparse, and imbalanced label space.

In addition to accurately classifying individual clinical notes, estimating the prevalence of ICD codes within a dataset is also important for many practical applications. This corresponds to a text quantification problem (Schumacher et al., 2021; Moreo et al., 2022), requiring properly calibrated text classification models.

This paper describes a novel approach for ICD coding, combining several ideas from previous works. We split long clinical documents into chunks, and use a strong Transformer-based model (Yang et al., 2022a) for processing each of the text chunks independently. The resulting representations are processed with a max-pooling operation, and combined with a label embedding mechanism inspired by that of Yuan et al. (2022), which explores diverse ICD code synonyms. Additionally, taking inspiration on the MLP-based quantification approach from Coutinho and Martins (2023), we explored a training setup in which multi-label classification and text quantification are jointly addressed. This additional step was explored as an approach to potentially improve model calibration.

Following previous studies, the proposed model was evaluated on the publicly available MIMIC-III dataset (Johnson et al., 2016), specifically analyzing results on two subsets of hospital discharge summaries, namely MIMIC-III-50 (Mullenbach et al., 2018) and MIMIC-III-clean (Edin et al., 2023). Our approach surpasses common baselines and previous state-of-the-art models for ICD coding, across all evaluated metrics, while also leading to properly calibrated model that can effectively inform downstream tasks such as text quantification.

The remaining parts of this paper are organized as follows: Section 2 reviews existing literature. Section 3 introduces our novel framework for ICD coding and quantification. Section 4 presents the experimental results, establishing a direct comparison with previous studies. Section 5 summarizes our contributions and discusses future research directions. The paper ends with a discussion on limitations and ethical considerations.

[1] https://www.who.int/standards/classifications/classification-of-diseases

2 Related Work

Several previous studies have addressed the problem of automatic ICD coding. For instance, Mullenbach et al. (2018) introduced the Convolutional Attention for Multi-Label classification (CAML) approach, which is still commonly considered as a baseline. CAML employs a label-wise attention mechanism, enabling the model to learn distinct document representations for each label, through the use of attention to select relevant parts of the document for each ICD code. The authors conducted experiments on MIMIC datasets (Lee et al., 2011; Johnson et al., 2016), and the train-test splits developed for this work were latter made publicly available. This study is considered an important milestone for reproducibility.

Aiming to address CAML's limitations in capturing variable-sized text patterns, Xie et al. (2019) improved the convolutional attention model by introducing a densely connected CNN with multi-scale feature attention (MSATT-KG), which produces variable n-gram features and adaptively selects informative features based on neighborhood context. This method also incorporates a graph CNN to capture hierarchical relationships among medical codes. In turn, Li and Yu (2020) proposed MultiResCNN, i.e. a novel CNN architecture combining multi-filter convolutions and residual convolutions, capturing patterns of different lengths and achieving superior performance over CAML.

Vu et al. (2020) introduced LAAT, i.e. a model that combines an RNN-based encoder with a new label attention mechanism for ICD coding. LAAT aimed to handle the variability in text segment lengths and the interdependence among different segments related to ICD codes. Additionally, the authors introduced a hierarchical joint learning mechanism to address the class imbalance issue.

Yuan et al. (2022) put forth the Multiple Synonyms Matching Network (MSMN) as an alternative approach to ICD coding. Rather than relying on the ICD code hierarchy, the authors leveraged synonyms to enhance code representation learning and improve coding performance.

In recent years, text classification research has shifted towards the use of Transformer-based language models. Dai et al. (2022) compared Transformer-based models for long document classification, focusing on mitigating the computational overheads associated with encoding large texts. Huang et al. (2022) investigated limitations associated to the use of pre-trained Transformer-based language models, identifying challenges associated to large label spaces, long input lengths, and domain disparities. The authors proposed PLM-ICD, i.e. a framework that effectively handles these challenges and achieves superior results on the MIMIC-III dataset, surpassing previously existing methods.

In a recent study, Edin et al. (2023) argued that the proper assessment of model performance on ICD coding had often struggled with weak configurations, poorly designed train-test splits, and inadequate evaluation procedures. The authors pinpointed significant issues with the MIMIC-III splits released by Mullenbach et al. (2018), and proposed a new split using stratified sampling, to ensure a complete representation of all classes.

On what regards text quantification, a variety of different algorithms has been proposed in recent years (Schumacher et al., 2021). Still, few previous studies have specifically considered multi-label settings (Moreo et al., 2022). Coutinho and Martins (2023) explored the use of a Multi-Layer Perceptron (MLP) model, inspired on under-complete denoising auto-encoders. The MLP was trained to refine estimates provided by the probabilistic classify and count method, considering label correlations. Experiments with different MIMIC-III datasets showed that the proposed method could outperform baseline approaches such as Classify and Count and Probabilistic Classify and Count.

3 Proposed Approach

This work presents a novel approach for ICD coding, aiming at strong classification performance together with well-calibrated outputs that can inform downstream tasks such as text quantification.

3.1 Chunk-Based Modeling of Clinical Text

One of the key aspects in our approach is the assumption that if an ICD code is identified in a single segment (i.e., a chunk) of the input document, then that code should clearly be assigned when classifying the document as a whole.

By carefully attending to the ICD codes in each chunk, and employing max-pooling to take into account the contribution of all chunks, we can effectively leverage the capabilities of a standard Transformer encoder, limited to a maximum of T tokens (in our case, $T = 512$), to analyze long clinical documents. To mitigate the loss of information from abruptly breaking interconnected pieces of

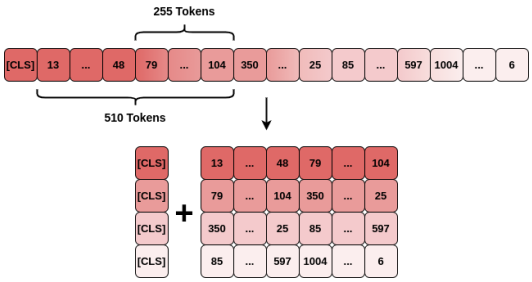

Figure 1: Smooth document segmentation with token overlaps. Note that each chunk includes, at the end, the sentence separation token [SEP] characteristic of BERT-type models, completing 512 tokens per chunk.

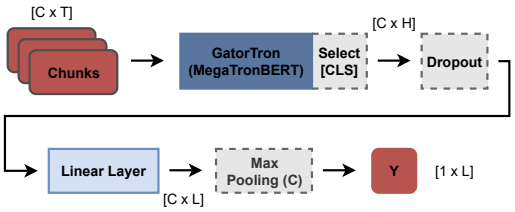

Figure 2: The chunk-based classification architecture.

text, we adopted a smooth partitioning scheme that considers large overlaps between chunks, as shown in Figure 1.

With this approach, we used a Megatron BERT model pre-trained on the healthcare domain (i.e., GatorTron, described by Yang et al. (2022a)), publicly available in the NVIDIA[2] NGC Catalog and in association with the HuggingFace[3] Transformers library. Figure 2 illustrates the chunk-based classification architecture, where C refers to the number of chunks, T corresponds to the number of tokens within each chunk, H corresponds to the dimensionality of the vectors representing each token, and L denotes the number of ICD classes.

3.2 Multi-Synonyms Attention

Inspired by Yuan et al. (2022), we enhanced our classification model through the integration of a multi-synonyms attention mechanism. The primary objective was to explore the intricate relationships between specific mentions to ICD codes, within chunks of the hospital discharge summaries, and the textual descriptions for ICD codes. This integration aimed to leverage synonyms to improve code representation learning (i.e., label embeddings), ultimately aiding in code classification.

We started by extending the ICD-9-CM code descriptions with synonyms obtained from a large medical knowledge base, specifically the UMLS metathesaurus. By aligning ICD codes with UMLS Concept Unique Identifiers (CUIs), we selected corresponding synonyms for English terms sharing the same CUIs. Additionally, we considered synonym variants by removing special characters, allowing only hyphens and brackets, and removing the coordinating conjunctions "or" and "and".

While extending the code descriptions, we observed that the lists of UMLS synonyms associated with each code were often long and repetitive, posing a risk of introducing bias in classification, and negatively impacting the meaning of code representations. To improve diversity, we gathered more synonyms from Wikidata and Wikipedia, and then selected M synonyms for each code according to a particular procedure. The synonyms were first represented as vectors through the same GatorTron model used to represent the text chunks (i.e., taking the [CLS] token representation for each synonym). Then, M vectors were selected for each ICD code through the application of the Gurobi optimizer[4], as a way to address the Maximum Diversity Problem[5], which can be formulated as follows:

$$\text{maximize} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} x_i x_j, \quad (1)$$

$$\text{subject to} \sum_{i=1}^{n} x_i = M, \quad (2)$$

$$x_i = \{0, 1\}, \quad 1 \leq i \leq n. \quad (3)$$

In the previous equations, d_{ij} is a distance metric between synonym representations i and j (i.e., the cosine distance between the vectors), and x_i takes the value 1 if element i is selected and 0 otherwise. Through this optimization problem, we selected a small subset of synonyms that effectively represents the broader embedding space for each ICD code. Here we denote by Q_l a matrix where rows correspond to the representations for the M synonyms associated to ICD code l, with each code synonym jl composed of S_jl tokens ($\{s_i^j l\}_{i=1}^{S_jl}$):

$$Q_l = \{\text{GatorEnc}(s_1^{jl}, ..., s_{S_{jl}}^{jl})\texttt{[CLS]}\}_{j=1}^{M}. \quad (4)$$

Note that the token representations within each chunk of text c are similarly produced with the GatorTron model, given by:

$$K^c = \text{GatorEnc}(x_1^c, ..., x_T^c). \quad (5)$$

[2] https://catalog.ngc.nvidia.com/
[3] https://huggingface.co/UFNLP/gatortron-base
[4] https://www.gurobi.com
[5] https://grafo.etsii.urjc.es/optsicom/mdp.html

To integrate the text representations from each chunk with the multiple synonym representations, we use an approach inspired by the multi-synonyms attention method proposed by Yuan et al. (2022), which in turn draws inspiration from the multi-head attention mechanism of the Transformer architecture (Vaswani et al., 2017).

We specifically split K^c into Z heads, setting this value to be equal to the maximum number of synonyms per code, i.e. $Z = M$:

$$K^c = K^c_1, ..., K^c_Z. \quad (6)$$

The code synonyms $\{Q_l\}_{l=1}^{L}$ are used to query K^c, and by calculating attention scores α_l over K^c, we identify the parts from the chunk's text that are more related to code's synonym l:

$$\alpha_l = \{\text{Softmax}(W_Q Q_l \,.\, \text{tahn}(W_K K^c))\}_{c=1}^{C}, \quad (7)$$

We then use avg-pooling of $\text{tanh}(K)\alpha_l$ assuming our intention is to create code-wise text representations R by averaging the contributions from synonyms:

$$R = \{\text{AvgPool}(\text{tahn}(K)\alpha_l)\}_{l=1}^{L}. \quad (8)$$

To assess whether the text of a chunk c contained code l, we evaluate the similarity between the code-wise text representation R_c and code's embeddings V. We aggregate the code synonym representations Q to form a code representation V through avg-pooling, resulting in a matrix with each row depicting a global representation of each code. To measure the similarity for classification, we apply a bi-affine transformation. Finally, after carefully attending to the ICD codes in each chunk using synonyms to enhance the classification, we employ max pooling to consolidate the results:

$$V = \text{AvgPool}(Q^1, Q^2, ..., Q^M), \quad (9)$$

$$Y = \sigma(\text{MaxPool}(\text{Diag}(R_1^T WV), ..., \text{Diag}(R_C^T WV))). \quad (10)$$

Unlike previous approaches that perform classification using code-dependent parameters, which can be challenging to define for rare codes, our bi-affine function uses code-independent parameters WV. This approach simplifies the learning process, at the same time making it more effective.

Figure 3 illustrates the process behind the chunk-based classification method that considers the multi-synonyms attention mechanism.

For model training, noting that we are in the presence of a multi-label classification task, we adopted

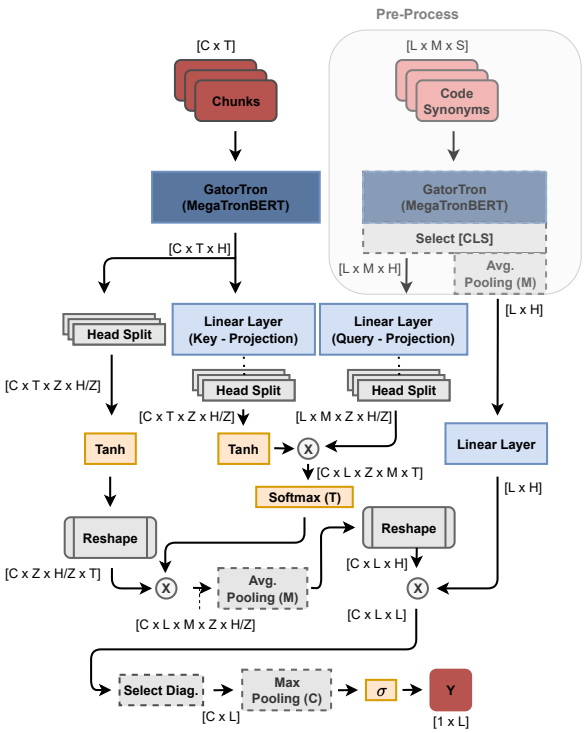

Figure 3: The chunk-based classification architecture that considers a multi-synonyms attention mechanism.

the widely-used Binary Cross-Entropy (BCE) loss, which treats each class independently and can be formally described as follows:

$$\mathcal{L}_C = \sum_{l=1}^{L} -y_l \log(\hat{y}_l) - (1 - y_l)\log(1 - \hat{y}_l). \quad (11)$$

The variable $y_l \in \{0, 1\}$ represents the ground truth for a code l, while $\hat{y}_l$ represents the probability of that code being present, as given by the classifier, and L is the number of different ICD codes.

3.3 Joint Classification and Quantification

Following previous work by Coutinho and Martins (2023), we considered the use of an under-complete denoising auto-encoder to quantify the prevalence of ICD codes within a set of documents, accounting with label associations. We integrated this quantification module, implemented as a three-layer MLP, together with the classifier, performing end-to-end training of the resulting model. We hypothesise that the classification and the quantification objectives can naturally complement each other, contributing to improved model calibration.

Notice that classification operates at the level of individual instances, while quantification operates over groups of instances. To integrate both objectives within end-to-end training, we follow the steps described next:

1. **Shuffling and setting a limit:** We shuffle the training dataset at the start of each training epoch. We also establish a limit that simulates the maximum number of instances that will be considered for quantification.

2. **Iterative data collection:** We process the instances individually as we progress through the training set. For each instance that is processed, we collect the classification results until we hit the previously defined maximum limit. This creates a new group of instances for each new instance that is processed, consisting of the ones we have processed thus far, plus the latest instance. The processing of each instance is made as follows:

   (a) **Computation of classification loss:** When processing each new instance, we apply our classification model and calculate the classification loss associated to that instance.

   (b) **Computation of quantification loss:** We take the classification output and add it to the previous classification outputs. This combination allows us to compute a probabilistic classify and count vector, denoting the estimated relative frequency of each class label within the group of instances. We then process this vector using the aforementioned MLP, which refines the probabilistic classify and count estimates. We finally calculate the quantification loss with the refined estimates.

   (c) **Aggregation of results:** The loss values computed in the previous steps are aggregated into a total loss, which is used to update model parameters for each batch of instances that is processed.

3. **Repeat and reset:** We follow the iterative process (steps (a) to (c)) until we reach the maximum number of instances designated for the quantification set. Once this limit is reached, we reset the quantification group and establish a new maximum limit for the instances to be quantified, continuing with model training until a stopping criteria is meet.

Our combined loss function can be formally described by the following equation, where λ is an hyper-parameter controlling the relative influence of the quantification loss:

$$\mathcal{L} = \mathcal{L}_C + \lambda \mathcal{L}_Q. \tag{12}$$

The classification loss ($\mathcal{L}_C$) is the BCE formally described in Equation 11, while the quantification loss ($\mathcal{L}_Q$) uses the MSE, given by:

$$\mathcal{L}_Q(\hat{p}_\epsilon^{\text{MLP}}, p_\epsilon) = \sum_{l=1}^{L} |\hat{p}_\epsilon^{\text{MLP}}(l) - p_\epsilon(l)|^2, \tag{13}$$

where p_ϵ is the ground-truth quantification result (i.e., the relative class frequency within the set of instances) for each of the L class labels.

The MSE loss was preferred over other regression-type losses, such as the MAE, because it provides a smoother optimization landscape, leading to more stable and accurate results.

4 Experimental Evaluation

This section presents the experimental evaluation of the proposed method, establishing a comparison with previously reported results.

4.1 Datasets

Experiments were conducted using the publicly available MIMIC-III data (Johnson et al., 2016). We specifically used the same dataset splits considered in previous works, namely MIMIC-III-50 (Mullenbach et al., 2018), which only comprises the top-50 most frequent codes in the dataset, and also MIMIC-III-clean (Edin et al., 2023), which corresponds to a cleaned dataset version that contains $3,681$ unique ICD-9-CM codes. Access to the MIMIC-III data was granted through PhysioNet[6], after completing the ethical training by the Collaborative Institutional Training Initiative program. We show a more detailed analysis of the dataset splits in Appendix A.1, including a statistical overview and interval of code occurrences for relevant percentiles of code frequencies.

The quantification experiments also used MIMIC-III-50 and MIMIC-III-clean, following the general methodology from Coutinho and Martins (2023). Specifically, for assessing result quality, we sampled documents from the validation set order to form $5,000$ quantification groups of different sizes, with the size parameter varying between one and the number of documents in the set. A separate set of $1,000$ groups was also created by sampling documents from the test split. These were used for (pre-)training the MLP quantification model and test the different quantification experiments.

[6] https://physionet.org/content/mimiciii/

Parameters	MIMIC-III-50	MIMIC-III-clean
Maximum token input length	7,142	6,122
Token overlapping window	255	255
GatorTron hidden size	1,024	1,024
Synonyms per ICD code (M)	4	4
Number of heads (Z)	4	4
Maximum number of epochs	300	300
Early stopping patience	5	5
Effective batch size	16	16
Adam e	1e-8	1e-8
Starting learning rate	2e-5/2e-7	2e-5/2e-7
Ending learning rate	0	0
MLP hidden size	32	3,072
Quantification coefficient (λ)	100	100
Learning rate scheduler	linear	linear

Table 1: Hyper-parameters used for model training in the MIMIC-III-50 and MIMIC-III-clean settings. The *max number of epcochs* values are related to the classification and quantification modules.

4.2 Evaluation Metrics

To ensure a fair comparison with prior research, we assessed the proposed approach across a range of metrics also considered in previous work.

Regarding the classification task, we used micro and macro-averaged F1-scores, Area Under the Curve (AUC) scores, and precision at cutoff n. For the experiments over the MIMIC-III-50 dataset we defined $n = 5$, and for the experiments conducted on MIMIC-III-clean we considered $n = 8$ and $n = 15$, roughly aligning with the average number of codes in each split. For measuring the calibration quality of our classifier, we used the Mean Expected Calibration Error (MECE) with 20 bins.

For the quantification task, we used the Mean Absolute Error (MAE) and the Mean Relative Absolute Error (MRAE) to assess result quality.

4.3 Implementation Details

Table 1 presents the training hyper-parameters considered in our experiments.

Since the proposed model processes the input text in chunks, the maximum allowable token length is limited only by hardware constraints. During training, we had to cap the maximum input token length due to restrictions in the available GPU memory. However, we could further raise this limit in the test environment, up to 20,000 tokens.

We trained our classifiers in two stages. The first stage uses a learning rate starting at 2e-5 and proceeds until we reach the early stopping criteria. We then perform a second training stage, with a learning rate starting at 2e-7. The quantifier model (MLP) was first trained individually following the guidelines of Coutinho and Martins (2023), using a constant learning rate schedule starting at 2e-5 and proceeds until we reach the early stopping criteria.

The model that integrates the quantification objective was initialized with pre-trained classification and quantification components, obtained through the first stage of training. Thus, these components should already perform each task with reasonable competence, prior to their combination.

4.4 Experiments and Results

The experimental results present a comprehensive evaluation of the proposed approach across the different metrics, comparing it against previous methods and also against ablated model versions.

4.4.1 Classification

Tables 2 and 3 present experimental results for the proposed approach, together with results for ablated versions that do not consider the label embeddings or the joint training with the quantification objective, and with the results of previous work for both MIMIC-III dataset splits. The rows named BM correspond to our base model, while BM+MSAM refers to the addition of the multiple-synonyms attention mechanism, and BM+MSAM+CLQ refers to the joint training with classification and quantification objectives.

When it comes to the impact of the label embedding mechanism that explores multiple-synonyms, it is clear that this module played a crucial role, significantly boosting performance across all metrics. In turn, although the best results were achieved with the model variant that includes the multi-synonym attention mechanism (BM+MSAM+CLQ), jointly training the classification and quantification objetives had, in fact, a negligible impact on classification accuracy.

When compared to latter proposals, our approach outperformed the previously best-performing models reported for both splits under analysis. It is worth noting that the models reported by Edin et al. (2023) underwent an adjustment using the validation splits, as the authors reported on model performance after optimizing the decision boundary values through a grid search mechanism to maximize F1 scores in the validation splits. In contrast, our results do not involve any such adjustment, and still surpassed the best reported models to date, establishing a new state-of-the-art approach with a default decision boundary set at 0.5.

Model	Stopping Epochs	AUC Macro	AUC Micro	F1 Macro	F1 Micro	P@N P@5
CAML* (Mullenbach et al., 2018)	–	87.5	91.1	51.0	60.6	61.1
MSATT-KG† Xie et al. (2019)	–	91.4	93.6	63.8	68.4	64.4
MultiResCNN* (Li and Yu, 2020)	–	89.7	92.4	61.1	67.3	64.4
LAAT* (Vu et al., 2020)	–	90.5	92.8	59.2	66.8	64.0
PLM-ICD* (Huang et al., 2022)	–	91.7	93.8	65.4	70.5	65.7
MSMN† (Yuan et al., 2022)	–	92.8	94.7	68.3	72.5	68.0
KEPTLongformer† (Yang et al., 2022b)	–	92.6	94.8	68.9	72.9	67.3
BM	10(+0)	91.2	93.4	65.5	70.0	66.1
BM+MSAM	4(+10)	**93.7**	**95.4**	**70.4**	73.9	68.8
BM+MSAM+CLQ	4(+4)	**93.7**	**95.4**	**70.4**	**74.0**	**68.9**

Table 2: Results for the different classification methods on the MIMIC-III-50 test set. Results for methods marked with * were taken directly from Edin et al. (2023). Results for methods marked with † were taken directly from the corresponding paper.

Model	Stopping Epochs	AUC Macro	AUC Micro	F1 Macro	F1 Micro	P@N P@8	P@N P@15
CAML* Mullenbach et al. (2018)	–	91.4	98.2	20.4	55.4	67.7	52.8
MultiResCNN* (Li and Yu, 2020)	–	93.1	98.5	22.9	56.4	68.5	53.5
LAAT* (Vu et al., 2020)	–	94.0	98.6	22.6	57.8	70.1	54.8
PLM-ICD* (Huang et al., 2022)	–	95.9	98.9	26.6	59.6	72.1	56.5
BM	68(+0)	91.7	96.1	16.9	52.1	66.1	50.6
BM+MSAM	8(+5)	96.3	98.9	30.5	60.3	**73.3**	**57.5**
BM+MSAM+CLQ	8(+6)	**96.4**	**99.0**	**31.2**	**60.5**	**73.3**	57.4

Table 3: Results for the different classification methods on the MIMIC-III-clean test set. Results for methods marked with * were taken from Edin et al. (2023).

For the MIMIC-III-50 setup, the proposed approach outperforms the best reported model to date (i.e., KEPTLongFormer) across all metrics securing leading scores of 93.7 (+1.1), 95.4 (+0.6), 70.4 (+1.6), 74.0 (+1.1), and 68.9 (+1.6) in terms of macro-AUC, micro-AUC, macro-F1, micro-F1, and P@5, respectively. For the MIMIC-III-clean setup, the proposed approach outperforms the best reported model to date (i.e., PLM-ICD) also across all metrics, securing leading scores of 96.4 (+0.5), 99.0 (+0.1), 31.2 (+4.6), 60.4 (+0.8), 73.3 (+1.2) and 57.4 (+0.9) in terms of macro-AUC, micro-AUC, macro-F1, micro-F1, P@8, and P@15.

To explore the influence of using a different number of synonyms, we considered the BM+MSAM+CLQ model and varied M between 2, 4, or 8 synonyms on a test over the MIMIC-III-50 dataset. Similarly to Yuan et al. (2022), our experiments showed that $M = 4$ lead to the best results, as can be observed in Table 4.

We also analyzed the proposed approach in terms of calibration performance. In Table 5, we explicitly examine the calibration error over different sets of ICD codes: Low percentile (Low Pth) corresponds to the average value of the calibration error calculated for the 10% of ICD codes with

	AUC Macro	AUC Micro	F1 Macro	F1 Micro	Prec@N P@5
$M = 1$	93.5	95.2	69.3	72.5	68.0
$M = 2$	93.6	95.3	69.8	73.4	68.3
$M = 4$	**93.7**	**95.4**	**70.4**	**73.9**	**68.8**
$M = 8$	93.4	95.1	69.2	72.9	68.0

Table 4: Results when considering a different number of synonyms (M) on the MIMIC-III 50 dataset.

Dataset	Classifier	Mean	Low Pth	Medium Pth	High Pth
MIMIC-III-50	BM	3.5e-2	2.1e-2	3.0e-2	5.1e-2
	BM+MSAM	**2.7e-2**	**2.0e-2**	**2.5e-2**	**3.6e-2**
	BM+MSAM+CLQ	2.9e-2	2.1e-2	2.6e-2	3.7e-2
MIMIC-III-clean	BM	2.4e-3	1.1e-4	8.4e-4	16.0e-3
	BM+MSAM	**1.6e-3**	**1.9e-4**	**8.5e-4**	**7.7e-3**
	BM+MSAM+CLQ	**1.6e-3**	2.0e-4	8.8e-4	8.0e-3

Table 5: Calibration quality according to the MECE metric, for all the proposed classification models and on different percentiles of the MIMIC-III splits.

the lowest frequency rates in the training set of the respective MIMIC-III split. In turn, medium percentile (Medium Pth) represents the average value of the calibration error for the 10% of ICD codes with medium frequency rates, falling within the 55% to 65% range in the respective MIMIC-III split training set; Finally, high percentile (High Pth) indicates the average value of the calibration error for the 10% of medical codes with the highest frequency of occurrence in the training set of the respective MIMIC-III split.

The results show that the the label embedding mechanism that explores multiple-synonyms also offers notable benefits in terms of model calibration. The joint optimization of classification and quantification objectives failed to further improve calibration performance on both MIMIC-III splits.

Besides presenting overall classification results, we also analyzed model performance for specific ICD codes, using the MIMIC-III-clean split. When considering the top-10 most frequent ICD-9-CM codes, Table 6 presents the results per code, using our best performing model. We obtained a mean precision of 75.56%, a recall of 79.34%, and an F1 score of 77.39%, i.e. results which we believe that can attest to the usefulness of our approach.

In turn, Table 7 presents performance metrics for some relevant chronic diseases, representing some of the main focuses of health care investigation. These results again attest to the usefulness of the proposed classification method.

Appendix A.2 details the classification performance across different chapters of ICD codes.

Code	Description	Precision	Recall	F1
401.9	*Unspecified essential hypertension*	75.82	86.26	80.71
38.93	*Venous Catheterization, Not Elsewhere Classified*	68.84	72.40	70.58
428.0	*Heart failure*	80.68	82.97	81.81
427.31	*Atrial fibrillation*	90.38	92.06	91.21
414.01	*Coronary atherosclerosis of native coronary artery*	81.52	86.15	83.77
96.04	*Insertion Of Endotracheal Tube*	78.36	82.13	80.20
96.6	*Enteral Infusion Of Concentrated Nutritional Substances*	69.76	78.32	73.80
99.04	*Transfusion Of Packed Cells*	65.72	59.59	62.50
584.9	*Acute kidney failure, unspecified*	72.58	69.76	71.15
250.00	*Diabetes mellitus without mention of complication type II or unspecified type, not stated as uncontrolled*	71.95	83.72	77.39
	Average	75.56	79.34	77.39

Table 6: Results for the 10 most frequent ICD-9-CM codes in the MIMIC-III-clean test dataset.

Block	Chronic Disease	Unique codes (Present)	Percentage	Performance metrics	
				Macro-F1	Micro-F1
250	*Diabetes mellitus*	33	1.943%	29.71	65.21
401-405	*Hypertensive Disease*	14	3.303%	29.38	76.78
410-414	*Ischemic Heart Disease*	32	3.279%	31.11	68.99
428	*Heart Failure*	15	2.471%	37.19	71.53
585;403-404	*Renal Failure*	16	1.600%	35.19	58.89
490-496	*Pulmonary Disease*	16	1.209%	48.16	67.32

Table 7: Results for some relevant chronic diseases. The columns named "Unique Codes" and "Percentage" refer to the number of unique codes of the respective block within the MIMIC-III-clean test dataset, and to the corresponding percentage of occurrences.

4.4.2 Quantification

Tables 8 and 9 show quantification test results, using both MIMIC-III splits. The results correspond to the standard Classify and Count (CC) and Probabilistic Classify and Count (PCC) methods, as well as to the use of an MLP separately trained for quantification, following the experimental setup from Coutinho and Martins (2023). In the case of BM+MSAM+CLQ, the MLP trained jointly with the classifier was used for quantification.

Analysing Table 8 regarding MIMIC-III-50 split, we observe that the PCC method performs less when using the model results that jointly optimize classification and quantification objectives. These results aligned with the calibration performance reported in the previous section. Additionally, we find that the joint optimization does not improve performance over the separate training of an MLP for quantification, as previously proposed by Coutinho and Martins (2023). A possible explanation relates to the fact that MIMIC-III-50 does not feature severe class imbalance issues. With a sufficient amount of data for all ICD codes, the multi-synonym attention mechanism is effective in producing well-calibrated classification outputs, leading to good quantification performance.

Regarding MIMIC-III-clean complaining a more challenge scenario (i.e more imbalanced and higher feature space), Table 9 shows that BM+MSAM+CLQ model outperforms all reported baselines.

Model	CC		PCC		MLP/CLQ	
	MAE	MRAE	MAE	MRAE	MAE	MRAE
BM	2.11e-02	1.08e-01	1.50e-02	9.67e-02	1.14e-02	6.83e-02
BM+MSAM	1.72e-02	9.31e-02	1.38e-02	9.31e-02	**1.09e-02**	**6.64e-02**
BM+MSAM+CLQ	1.91e-02	9.90e-02	1.69e-02	10.9e-02	1.14e-02	6.83e-02

Table 8: Results for different quantification methods, using the results from different classification models on the MIMIC-III-50 test dataset split.

Model	CC		PCC		MLP/CLQ	
	MAE	MRAE	MAE	MRAE	MAE	MRAE
BM	1.41e-03	3.15e-01	1.24e-03	5.59e-01	8.62e-04	5.98e-01
BM+MSAM	1.41e-03	3.32e-01	1.24e-03	5.97e-01	8.62e-4	6.43e-1
BM+MSAM+CLQ	1.41e-03	3.31e-01	1.24e-03	5.64e-01	**7.03e-04**	**4.47e-01**

Table 9: Results for different quantification methods, using the results from different classification models on the MIMIC-III-clean test dataset split.

We show a more detailed analysis of the quantification results in Appendix A.3.

5 Conclusion and Future Work

This work introduced a novel deep learning method for ICD coding, which achieves state-of-the-art results in tests with two MIMIC-III dataset splits used in previous work. The proposed method processes long clinical documents in chunks, and it uses a label embedding mechanism that explores diverse ICD code synonyms. Besides achieving highly-accurate classification results, the proposed approach also produces well-calibrated estimates, that can effectively inform downstream tasks such as text quantification.

Despite the very strong results, it should be noted that our model does not exploit the hierarchical structure inherent to the ICD coding system, which could further enhance its classification capabilities. Thus, a promising avenue for further improvement involves the use of this structural knowledge, e.g. through the implementation of dual classification heads. Regarding text quantification, we believe that a path that is worth exploring concerns the use of alternative methods to further enhance the calibration of our classifier (e.g., through the use of other classification loss functions besides the BCE), since improving calibration is beneficial for classification and essential for achieving accurate results in quantification tasks.

Limitations and Ethical Considerations

While our work does not raise new ethical issues within this domain, there are general concerns to take into account.

ICD coding is very important in the context of clinical, operational, and financial healthcare decisions. Traditionally, medical coders review documents and manually assign the appropriate ICD codes, by following specific coding guidelines. Approaches such as ours can help to significantly reduce time and costs in ICD coding. Still, there are important risks associated to over-reliance on automatic coding methods. No matter how accurate a given approach is, it is still possible to misclassify documents with erroneous ICD codes, which may for instance affect patient treatment. We therefore strongly believe that automatic coding should be used to assist, rather than replace, the judgement of trained clinical professionals.

Our experiments have also relied on MIMIC-III datasets used in previous studies. While these datasets constitute useful benchmarks for developing and evaluating new methods, they are not representative of the the enormous variety of clinical and linguistic data that may be encountered in potential deployments of the method.

References

Isabel Coutinho and Bruno Martins. 2023. Exploring label correlations for quantification of ICD codes. In *Proceedings of the International Conference on Discovery Science*.

Xiang Dai, Ilias Chalkidis, Sune Darkner, and Desmond Elliott. 2022. Revisiting transformer-based models for long document classification. *arXiv preprint arXiv:2204.06683*.

Joakim Edin, Alexander Junge, Jakob D. Havtorn, Lasse Borgholt, Maria Maistro, Tuukka Ruotsalo, and Lars Maaløe. 2023. Automated medical coding on MIMIC-III and MIMIC-IV: A critical review and replicability study. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Chao-Wei Huang, Shang-Chi Tsai, and Yun-Nung Chen. 2022. PLM-ICD: Automatic ICD coding with pretrained language models. *arXiv preprint arXiv:2207.05289*.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:1–9.

Joon Lee, Daniel J Scott, Mauricio Villarroel, Gari D Clifford, Mohammed Saeed, and Roger G Mark. 2011. Open-access MIMIC-II database for intensive care research. In *Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society*.

Fei Li and Hong Yu. 2020. ICD coding from clinical text using multi-filter residual convolutional neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Alejandro Moreo, Manuel Francisco, and Fabrizio Sebastiani. 2022. Multi-label quantification. *arXiv preprint arXiv:2211.08063*.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. *arXiv preprint arXiv:1802.05695*.

Kimberly J O'malley, Karon F Cook, Matt D Price, Kimberly Raiford Wildes, John F Hurdle, and Carol M Ashton. 2005. Measuring diagnoses: ICD code accuracy. *Health Services Research*, 40:1620–1639.

Tobias Schumacher, Markus Strohmaier, and Florian Lemmerich. 2021. A comparative evaluation of quantification methods. *arXiv preprint arXiv:2103.03223*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the Annual Conference on Advances in Neural Information Processing Systems*.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. 2020. A label attention model for ICD coding from clinical text. *arXiv preprint arXiv:2007.06351*.

Xiancheng Xie, Yun Xiong, Philip S Yu, and Yangyong Zhu. 2019. EHR coding with multi-scale feature attention and structured knowledge graph propagation. In *Proceedings of the ACM international conference on information and knowledge management*.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Mona G Flores, Ying Zhang, et al. 2022a. GatorTron: A large clinical language model to unlock patient information from unstructured electronic health records. *arXiv preprint arXiv:2203.03540*.

Zhichao Yang, Shufan Wang, Bhanu Pratap Singh Rawat, Avijit Mitra, and Hong Yu. 2022b. Knowledge injected prompt based fine-tuning for multi-label few-shot ICD coding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Zheng Yuan, Chuanqi Tan, and Songfang Huang. 2022. Code synonyms do matter: Multiple synonyms matching network for automatic ICD coding. *arXiv preprint arXiv:2203.01515*.

A Appendix

This appendix presents extra information about the dataset splits, and experimental results of classification and quantification tasks.

A.1 Dataset

Table 10 provides an overview of the statistics for the MIMIC-III splits for its training, validation, and test sets, underlining its highly imbalanced label distribution, disparity between average and maximum token lengths, and high number of ICD codes assigned to each discharge summary.

Set (Split)	Samples	Words pr. Doc.		Tokens pr. Doc.		Codes pr. Doc.		Unique	Type of Codes	
		Avg.	Max.	Avg.	Max.	Avg.	Max.	Codes	Diag.	Proc.
Train (Top-50)	8,066	1,642	7,989	2,830	20,297	5.4	18	50	33	17
Val. (Top-50)	1,573	1,932	6,658	3,410	16,566	5.9	21	50	33	17
Test (Top-50)	1,729	1,964	6,470	3,465	11,871	6.0	20	50	33	17
Train (Clean)	38,401	1,514	10,500	1,651	11,758	14.0	57	3,681	2,849	832
Val. (Clean)	5,577	1,552	6,393	1,694	6,897	15.9	60	3,676	2,844	832
Test (Clean)	8,734	1,485	7,858	1,619	8,299	14.8	56	3,681	2,849	832

Table 10: Statistics for training, validation and test sets of MIMIC-III-50 (top) and MIMIC-III-clean (bottom) datasets. "Words pr. Doc": Average and maximum number of words per hospital discharge summary. "Tokens pr. Doc": Average and maximum number of tokens per hospital discharge summary. "Unique codes": Unique number of ICD codes in the respective split. "Type of codes": Unique number of diagnosis and procedure codes.

Dataset	Split	Low Pth	Medium Pth	High Pth
MIMIC-III-50	**Train**	397-449	759-914	1615-3233
	Test	60-127	148-247	402-470
MIMIC-III-clean	**Train**	4-9	36-56	308-14,598
	Test	1-4	6-27	55-2228

Table 11: Interval of code occurrences in a specific percentile of code frequency.

Table 11 presents the information about the frequency of ICD codes, divided in three relevant percentiles for training and test sets for both MIMIC-III splits. Low Pth accounts to the 10% of medical codes with the lowest frequency rates in the training set of the respective MIMIC-III split. Medium Pth, corresponds to the 10% of codes with medium frequency rates, falling within the 55% to 65% range in the respective MIMIC-III split training set. Lastly, High Pth attains for the 10% of codes with the highest frequency rates in the training set of the respective MIMIC-III split.

A.2 Classification Results

Tables 12 and 13 provide additional insights into our model's performance, specifically considering results with the BM+MSAM+CLQ model for codes within different ICD-9-CM diagnosis and procedure chapters.

Chapter	Occurrences			Percentage	Performance metrics	
	Train	Validation	Test		Macro-F1	Micro-F1
I	152,465	21,978	35,168	26.302%	39.97	68.96
II	9,200	1,401	2,076	1.590%	36.15	57.65
III	49,135	7,356	11,008	8.470%	33.28	60.37
IV	17,882	2,657	4,106	3.092%	30.71	41.33
V	17,392	2,562	3,740	2.973%	22.10	48.24
VI	15,811	2,433	3,397	2.715%	29.69	54.62
VII	99,076	14,729	22,526	17.107%	29.43	67.38
VIII	31,613	4,703	7,113	5.449%	36.12	59.90
IX	27,061	3,967	6,022	4.649%	31.27	56.47
X	22,940	3,438	5,260	3.970%	29.96	62.08
XI	151	24	33	0.026%	33.79	43.64
XII	6,056	888	1,371	1.043%	29.07	47.67
XIII	9,098	1,360	1,944	1.556%	28.52	51.07
XIV	2,228	328	471	0.380%	51.54	62.20
XV	12,656	1,740	2,565	2.128%	31.50	60.40
XVI	20,692	3,154	4,550	3.563%	15.96	39.75
XVII	87,280	13,018	19,131	14.986%	24.36	51.11

Table 12: Number of instances and performance metrics for each of the ICD-9-CM diagnosis chapters. The column named "Percentage" corresponds to the percentage of the diagnosis codes under consideration over the MIMIC-III-clean test dataset.

Chapter	Occurrences			Percentage	Performance metrics	
	Train	Validation	Test		Macro-F1	Micro-F1
I	5,508	855	1,347	3.589%	36.28	65.21
II	4,852	733	1,148	3.134%	41.74	66.70
III	91	13	17	0.056%	63.07	66.67
IV	102	15	23	0.065%	57.16	60.87
V	0	0	0	0%	0.0	0.0
VI	21	3	4	0.013%	40.00	40.00
VII	501	75	104	0.317%	26.96	39.29
VIII	9,590	1,480	2,164	6.161%	37.62	63.98
IX	47,762	6,895	10,813	30.478%	45.93	76.26
X	897	127	217	0.578%	49.96	71.83
XI	15,302	2,267	3,555	9.834%	39.15	66.59
XII	1,045	152	230	0.664%	54.48	74.77
XIII	641	102	127	0.405%	74.50	69.43
XIV	201	27	43	0.126%	64.40	68.24
XV	20	3	4	0.013%	88.89	88.89
XVI	5,990	924	1,307	3.827%	44.69	60.23
XVII	2,308	318	539	1.473%	32.01	49.90
XVIII	61,329	8,568	14,455	39.267%	26.39	66.81

Table 13: Number of instances and performance metrics for each of the ICD-9-CM procedure chapters. The column named "Percentage" corresponds to the percentage of the procedure codes under consideration over the MIMIC-III-clean test dataset.

Chapter I (i.e., infectious and parasitic diseases)

in the ICD-9-CM diagnosis codes accounts for a substantial portion of the dataset, representing 26.302% of all codes. This chapter demonstrates impressive performance metrics, achieving a macro-averaged F1 score of 39.97% and a micro-averaged F1 score of 68.96%.

Conversely, Chapter XI (i.e., complications of pregnancy, childbirth, and the puerperium) is the least frequent chapter of ICD codes, and it also corresponds to the lowest performance metrics. With a prevalence of only 0.026% in the dataset, this chapter yields macro and micro-averaged F1 scores of 33.79% and 43.64%, respectively. These scores highlight the negative impact of infrequent ICD code occurrences on the model's effectiveness.

Furthermore, we observe an interesting phenomenon in Chapter XIV (i.e., congenital anomalies). Despite representing a relatively small percentage (0.380%) of the overall dataset, the model performs performs remarkably well in this chapter. It attains macro and micro-averaged F1 scores of 51.54% and 62.20%, respectively, empirically showing the model's ability to perform few-shot learning when dealing with seldom-seen codes.

When we examine the overall distribution of procedure codes, we see that the dataset is characterized by a generally low density of procedure codes, with two notable exceptions in Chapter IX (i.e., operations on the cardiovascular system) and Chapter XVIII (i.e., miscellaneous diagnostic and therapeutic procedures), which encompass almost 70% of the dataset. However, despite the relatively low frequency of procedures in the other chapters, our model performs exceptionally well in them. For instance, Chapters VI and XV achieve performance values of 40% and 88.89% respectively in both metrics, even though these codes have a minuscule 0.013% representation within the dataset. These results underscore the model's capacity to learn even from infrequent instances, again emphasizing its few-shot learning capabilities.

Chapter XVIII in the ICD-9-CM procedure codes, which covers "miscellaneous diagnostic and therapeutic procedures," stands out as the most frequently occurring chapter in the dataset, accounting for a substantial 39.267% of the total. We achieve 26.39% for macro-averaged F1 in this chapter, and 66.81% for micro-averaged F1.

A.3 Quantification Results

Figure 4 shows Absolute Error and F1 score per class sorted by prevalence over MIMIC-III-50.

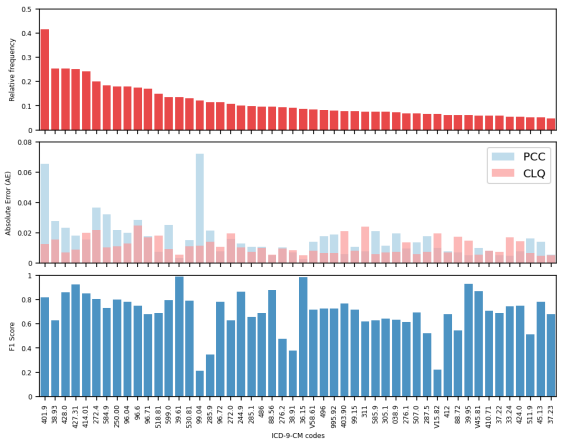

Figure 4: Relative frequency, Absolute Error, and F1-score for each ICD code over MIMIC-III-50.

Figure 4 shows that CLQ method outperforms PCC for nearly all ICD codes when it comes to accurately grasping the prevalence of each ICD code. For instance, ICD code 401.9, which is the most frequent in the test set, presents a high disparity in Absolute Error between PCC and CLQ results. Further investigation, revealed that despite having a high F1 score (81%), ICD code 401.9 has a notable difference between its precision score (75.8%) and recall score (86.3%). This suggests that the model tends to overestimate this class due to its high frequency, resulting in inaccurate posterior probabilities with the PCC approach. CLQ method appears to recognize this behavior and corrects it. Figure 5 aligns with the previous analysis.

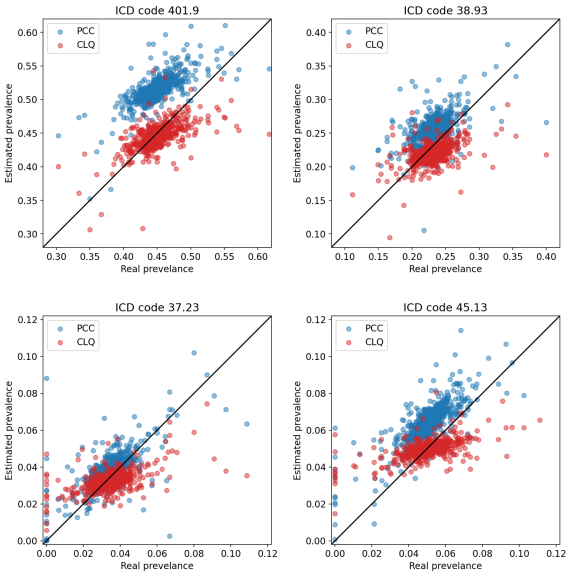

Figure 5: Estimated versus real prevalence for the two most frequent (top) and rarest (bottom) ICD codes in the MIMIC-III-50 dataset.