

Explainable Automated Coding of Clinical Notes using Hierarchical Label-wise Attention Networks and Label Embedding Initialisation

Hang Dong^{a,d}, Víctor Suárez-Paniagua^{a,d}, William Whiteley^{b,d}, Honghan Wu^{c,d}

^aCentre for Medical Informatics, Usher Institute of Population Health Sciences and Informatics, University of Edinburgh, Edinburgh, United Kingdom

^bCentre for Clinical Brain Sciences, University of Edinburgh

^cInstitute of Health Informatics, University College London, London, United Kingdom

^dHealth Data Research UK, London, United Kingdom

Abstract

Background: Diagnostic or procedural coding of clinical notes aims to derive a coded summary of disease-related information about patients. Such coding is usually done manually in hospitals but could potentially be automated to improve the efficiency and accuracy of medical coding. Recent studies on deep learning for automated medical coding achieved promising performances. However, the explainability of these models is usually poor, preventing them to be used confidently in supporting clinical practice. Another limitation is that these models mostly assume independence among labels, ignoring the complex correlations among medical codes which can potentially be exploited to improve the performance.

Methods: To address the issues of model explainability and label correlations, we propose a Hierarchical Label-wise Attention Network (HLAN), which aimed to interpret the model by quantifying importance (as attention weights) of words and sentences related to each of the labels. Secondly, we propose to enhance the major deep learning models with a label embedding (LE) initialisation approach, which learns a dense, continuous vector representation and then injects the representation into the final layers and the label-wise attention layers in the models. We evaluated the methods using three settings on the MIMIC-III discharge summaries: full codes, top-50 codes, and the UK NHS (National Health Service) COVID-19 (Coronavirus disease 2019) shielding codes. Experiments were conducted to compare the HLAN model and label embedding initialisation to the state-of-the-art neural network based methods, including variants of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).

Results: HLAN achieved the best Micro-level AUC and F_1 on the top-50 code prediction, 91.9% and 64.1%, respectively; and comparable results on the NHS COVID-19 shielding code prediction to other models: around 97% Micro-level AUC. More importantly, in the analysis of model explanations, by highlighting the most salient words and sentences for each label, HLAN showed more meaningful and comprehensive model interpretation compared to the CNN-based models and its downgraded baselines, HAN and HA-GRU. Label embedding (LE) initialisation significantly boosted the previous state-of-the-art model, CNN with attention mechanisms, on the full code prediction to 52.5% Micro-level F_1 . The analysis of the layers initialised with label embeddings further explains the effect of this initialisation approach. The source code of the implementation and the results are openly available at <https://github.com/acadTags/Explainable-Automated-Medical-Coding>.

Conclusion: We draw the conclusion from the evaluation results and analyses. First, with hierarchical label-wise attention mechanisms, HLAN can provide better or comparable results for automated coding to the state-of-the-art, CNN-based models. Second, HLAN can provide more comprehensive explanations for each label by highlighting key words and sentences in the discharge summaries, compared to the n -grams in the CNN-based models and the downgraded baselines, HAN and HA-GRU. Third, the performance of deep learning based multi-label classification for automated coding can be consistently boosted by initialising label embeddings that captures the correlations among labels. We further discuss the advantages and drawbacks of the overall method regarding its potential to be deployed to a hospital and suggest areas for future studies.

Keywords: Automated medical coding, Deep learning, Attention Mechanisms, Explainability, Natural Language Processing, Multi-label classification, Label correlation

arXiv:2010.15728v3 [cs.CL] 25 Feb 2021

1. Introduction

Diagnostic or procedural coding of medical free-text documents (e.g. discharge summaries) aims to derive a coded summary of disease-related information about patients, for clinical care, audit, and research. In hospitals, such coding is usually done manually, requiring much cognitive human effort, but could potentially be automated. An automated program could efficiently take a clinical note as input and then output medical codes from existing classification systems, e.g. ICD (International Classification of Diseases). This could facilitate coding professionals provide more accurate results.

This clinical task is technically challenging, due to (i) the explainability required to process long documents, in average about 2000 tokens in a discharge summary in MIMIC-III [1], and thus pose a “needle-in-a-haystack” issue to locate the key words and sentences relevant to each code; (ii) the complex label correlations in the multi-label setting, in average about 16 different ICD-9 (the Ninth Revision) codes per discharge summary in the MIMIC-III dataset [2], which inherently exhibit the complex relations among codes; and (iii) a large set of codes when using all the codes as candidates for prediction, e.g. around 13k unique codes in ICD-9 and many times further in ICD-10 [3] and ICD-11 [4].

Automated medical coding has been studied for more than a decade. Early studies mostly use systems based on rules, grammar, and string matching, as reviewed in [5]. Recent studies adapt deep learning based document classification methods, which commonly formalise the task as a *multi-label classification* problem [6, 7, 1]. Typically, they use variations of Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) to derive a continuous representation of clinical notes matched to the high-dimensional coding space. However, few studies have tackled the above challenges above regarding explainability and label correlations.

Explainability (or interpretability, used interchangeably in this paper) is a key requirement for models applied to the clinical domain, particularly regarding the ethical aspect and to build medical professionals’ trust in machine learning models [8, 9]. Also, to facilitate the work of coding professionals, a desired automated coding system should be able to highlight the most essential part of a long clinical note to support the assignment of medical codes. To address this, models based on CNNs can be adapted to highlight n -gram information to support the explanation, as in [7]. Solely the n -grams, however, may not be enough to provide accurate interpretation reflecting

the document structure.

In this work, we propose to highlight the most essential words and sentences in a document for automated medical coding, inspired and adapted from Hierarchical Attention Networks (HAN) [10] and the recent model, Hierarchical Attention bi-directional Gated Re-current Units (HA-GRU) [1]. With attention mechanisms, HAN can highlight the salient words and sentences related to the overall prediction. However, HAN could not generate a specific interpretation for each label. HA-GRU [1] can provide a *sentence*-level explanation for each label, but still could not specify the most essential *words* leading to the decision of each code. We present a novel model, Hierarchical Label-wise Attention Network (HLAN), which has label-wise word-level and sentence-level attention mechanisms, so as to provide a richer explainability of the model.

We formally evaluated HLAN along with HAN, HA-GRU, and CNN-based neural network approaches for automated medical coding. With better or comparative coding performance in various data settings, HLAN can further generate more comprehensive explanations through key sentences and words for each label, as indicated from the analysis on model explainability. The analysis of the false positive predictions also shows that the explanation based on the hierarchical label-wise attention mechanisms in HLAN can serve as a reference for medical professionals and engineers to make reasonable coding decisions and system iterations even when the model seems to predict erroneously.

Apart from model interpretability, another issue not thoroughly studied in deep learning based multi-label classification is label correlation. Medical codes are related and can be predicted together, for example, the code 486 (ICD 9 for Pneumonia) commonly appeared for over 1.5k times (out of about 53k documents) with the code 518.81 (Acute respiratory failure) in the MIMIC-III dataset. Such co-occurrences are under-lying by the clinical, biomedical, and biological associations among different diseases. Deep learning for multi-label classification represents the label space with orthogonal vectors: each label as a one-hot vector and each label set as a multi-hot representation [11, 7]. This, however, assumes independence among labels.

We propose an effective label embedding initialisation approach to tackle the label correlation problem. We encode the label correlation using pre-trained label embeddings from the label sets in the training data, derived from the coding practice. Then the label embeddings are used to initialise the weights in the final hidden layer and label-wise attention layers. The idea

is that the linear projection can automatically leverage the label similarity encoded in the continuous label embedding space. This approach shows consistent and significant improvement, while not requiring hyper-parameter tuning or further computational complexity.

We evaluate our approach with three specific datasets based on the openly available, MIMIC-III database [2], containing clinical notes in the critical care sector in the US. The first two datasets, full code and top-50 code predictions, are the same as in the work [7] for comparison. The third dataset was created to simulate the task of identifying high-risk patients for shielding during the COVID-19 (Coronavirus disease 2019) pandemic by predicting the ICD-9 codes matched to the codes used in the UK NHS (National Health Service) patient shielding identification method¹.

Thus, the contribution of the paper includes:

- A novel, Hierarchical Label-wise Attention Network (HLAN) for automated medical coding. The proposed HLAN model provides an explanation in the form of attention weights on both the word level and the sentence level for the prediction of each medical code.
- An effective label embedding (LE) initialisation approach to enhance the performance of various deep learning models for multi-label classification. Analysis of the LE initialised layers shows the efficacy to leverage label correlations for medical coding.
- A formal comparison of the main deep learning based methods for automated coding. Experiments on three datasets based on the MIMIC-III discharge summaries, i.e. full code prediction, top-50 code prediction, and the NHS COVID-19 shielding-related code prediction, show the advantage of the proposed method over the state-of-the-art methods (CNNs, Bi-GRU) and downgraded baselines (HA-GRU, HAN). Label embedding initialisation significantly improved the performance of neural network models in most evaluation settings. An analysis and comparison of the model interpretability demonstrate the most comprehensive explanations from the HLAN model.

The rest of the paper is organised as follows. First, we review the related work on automated medical coding with explainability, deep learning methods for multi-label classification, and

label correlation in Section 2. Then, we present the problem formulation, followed by the proposed model, HLAN, and the idea of LE initialisation in Section 3. The experiments, including datasets, experimental settings, main and per-label results, analysis and comparison of model explainability, and analysis on the layers initialised with LE, are in Section 4. We finally discuss the advantages and drawbacks of the overall methods in Section 5 and summarise the work in Section 6.

2. Related Work

We will first present the task of automated medical coding with the methods used especially in most recent studies, then introduce in detail the mainstream breakthrough on deep learning-based multi-label classification for the task, and finally review the label correlation issue, particularly relevant to the medical and clinical domain.

2.1. Automated Medical Coding with Explainability

Automated medical coding is the task of transforming medical records, especially the natural language in the clinical notes, into a set of structured, medical codes to facilitate clinical care, audit, and research [5]. The applied alphanumeric codes in the clinical domain, such as ICD and SNOMED-CT, represent patients' diagnosis, procedures and other information with controlled clinical terminology.

One of the earliest reviews back in 2010 [5] surveyed 113 studies on coding or classification of clinical notes. Most of the studies applied tools with rule-based, grammar-based, and string matching methods, and they in overall suffered the challenges of reasoning and the lack of method generalisability. The field of automated medical coding has in more recent years been advanced with the open, benchmarking datasets like radiology reports in [12] and MIMIC-III [2] discharge summaries. With the datasets, *deep learning* based approaches have been proposed and tested, which have generally demonstrated better performance than traditional machine learning methods. The work in [6] compared the deep learning based method, CNN, with several traditional machine learning methods, support vector machine, random forests, and logistic regression, for ICD-9 code prediction (number of ICD-9 codes $|Y|=38$) from 978 radiology reports in [12]. The result showed comparable or improved results of the deep learning approach to the traditional methods, even without parameter tuning in the CNN model. The work in [7] adapted CNN with attention mechanisms and established a state-of-the-art performance in predict-

¹<https://digital.nhs.uk/coronavirus/shielded-patient-list/methodology>

ing the full set ($|Y|=8,921$) and the top-50 most frequent ICD-9 codes ($|Y|=50$) from MIMIC-III discharge summaries.

A key aspect of clinical applications is their requirement of the *explainability* of models. Users are entitled to a “right of explanation” when their data being used for AI algorithms, as potentially regulated by the General Data Protection Regulation (GDPR) [9]. For clinical applications, e.g. radiology, the Joint European and North American Multisociety Statement raises great ethical concern on AI algorithms regarding explainability, i.e. “the ability to explain what happened when the model made a decision, in terms that a person understands” [8, p. 438]. While deep learning achieves better results in general, the approach is inherently less transparent than traditional methods due to its extremely complex networks of non-linear activation.

Few studies explored the explainability of deep learning models for automated medical coding. A representative work is the study [7], which compared the ability of different models to highlight n -grams along with the models’ ICD-9 code prediction. A manual evaluation showed that the CNN model with attention mechanisms can generate more meaningful n -grams relevant to the labels [7]. The study [1] proposed a Hierarchical Attention bi-directional Gated Recurrent Unit (HA-GRU) to produce a sentence-level explanation for each code, instead of n -gram-level explanation. In this work, we propose an approach with enhanced interpretability, from both the label-wise word-level and the sentence-level attention weights, to support automated coding.

2.2. Deep Learning-based Multi-label Classification with Attention Mechanisms

Automated medical coding is mainly formulated as a multi-label classification problem [13, 14, 7, 1], where each object (e.g. clinical note) is associated with a set of labels (e.g. diagnosis or procedure ICD codes) instead of a single label in binary or multi-class classification.

Deep learning has become the main approach for multi-label document classification [11, 15] in recent years. The advantage of multi-label deep learning models lies in their straightforward problem formulation and strong approximation power on large datasets, resulting in better performance over traditional machine learning approaches, as compared in [16, 15, 6]. For automated coding, some of the notable neural network models adapted for multi-label classification are variations of CNNs [6, 7] and RNNs [1] with attention mechanisms. Pre-trained models with multi-head self-attention blocks (e.g. BERT, Bi-directional Encoder Representations from Transformers) [17],

while substantially improved many NLP tasks, so far still are under-performing for automated coding with the MIMIC-III discharge summaries [18, 19].

The idea of the above mentioned *attention mechanism* is a key, recent advancement in deep learning for NLP, originated from machine translation to align (or attend to) words in the source sentence in one language to predict each of the target words in another language [20]. This inspires to jointly learn to represent the important words and sentences while classifying a document in HAN [10], thus also enables to *explain* the inner working of deep learning models. HAN was adapted to a multi-label classification setting to classify socially shared texts in [15] and for automated medical coding [1]. Founded on the studies above, our approach provides a richer label-wise attention mechanism at both the word and the sentence level for automated medical coding.

2.3. Label Correlation

In multi-label classification, labels are potentially correlated to each other. As the example in Section 1, the medical codes of “Pneumonia” and “Acute Respiratory Failure” tend to appear together in the MIMIC-III discharge summaries. In automated medical coding, the number of unique code $|Y|$ is large ($|Y| = 8921$ in the MIMIC-III dataset) and further the possible label relations (e.g. the number of pairwise combinations is near to $|Y|^2$). Such correlations among the labels represent additional knowledge that could be exploited to improve performance [21].

This issue of *label correlation* (or “label dependence”) remains an ongoing challenge [21] in multi-label classification, especially with deep learning models. Deep learning for multi-label classification mostly represents the label space with orthogonal vectors: each label as a one-hot vector and each label set as a multi-hot representation, in general domains [11] and clinical domains [7, 1]. Combined with the sigmoid activation and binary cross-entropy loss, this overall approach, effectively, assumes independence among labels.

One recent approach to address the problem is through weight initialisation [22, 23]: initialising higher weights for dedicated neurons (each encoding a co-occurrence relation among labels) in the final hidden layer. The approach showed performance improvement, however, it is not computationally efficient to assign each neuron in the final hidden layer to represent one of the massive (even the pairwise) patterns of label relations. An alternative method is through regularisation in [15] to enforce the output layer of the neural network to

satisfy constraints on label relations. This requires to further tune the hyper-parameters of the regularisers so that a relatively marginal improvement (0.5-1.5% example-based F_1 on scientific paper abstracts and questions in social Q&A platforms) could be achieved. In this study, we further propose a novel effective weight initialisation approach to tackle the label correlation problem, by initialising pre-trained dense label embeddings instead of the sparse co-occurrence representations.

3. Proposed Method

We formalise automated medical coding from clinical notes as a multi-label text classification problem [11]. With deep learning, multi-label classification mainly contains two, integrated parts, (i) a neural document encoder, representing documents into a continuous representation, and (ii) a prediction layer, matching the document space to the label space. We present the problem formalisation and the deep learning based multi-label classification in Section 3.1. Then, regarding the neural document encoder, we propose the hierarchical label-wise attention network in Section 3.2, followed by the idea of label embedding initialisation in the prediction layer in Section 3.3.

3.1. Problem Formulation with Deep Learning Models

Formally multi-label classification can be defined as follows. Suppose X denoting the collection of textual sequences (e.g. clinical notes), and $Y = \{y_1, y_2, \dots, y_{|Y|}\}$ denotes the full set of labels (i.e. ICD codes) of size $|Y|$. Each instance $x_d \in X$ is a word sequence of a document, where d is the document index. Each $x_d \in X$ is associated with a label set $Y_d \subseteq Y$. Each label set Y_d can be represented as a $|Y|$ -dimensional *multi-hot* vector, $\vec{Y}_d = [y_{d1}, y_{d2}, \dots, y_{d|Y|}]$ and $y_{dl} \in \{0, 1\}$, where a value of 1 indicates that the l th label y_l has been used to annotate (is relevant to) the d th instance, and 0 indicates irrelevance. The task is to learn a complex function $f : X \rightarrow Y$ based on a training set $D = \{x_d, \vec{Y}_d | d \in [1, m]\}$, where m is the number of instances in the training set.

Neural document encoders in deep learning models (e.g. CNN, RNN, and BERT, as review in Section 2.2) represent each word sequence x as a continuous vector h , with matrix projection and non-linear activation. The representation h is projected to the label space and turned into $p_{dl} \in (0, 1)$ with the sigmoid function ($\sigma(x) = \frac{1}{1+e^{-x}}$), as defined in Equation 1 below, where the weight w_l (a row vector in W) and the bias b are parameters to be learned during the training process. The obtained p_{dl} is

the probability of the label (e.g. ICD code) y_l being related to the document (e.g. discharge summary) d .

$$p_{dl} = \sigma(w_l h + b), \text{ or collectively as } p_d = \sigma(Wh + b) \quad (1)$$

The loss function is commonly the binary cross-entropy loss [11] as defined in Equation 2, which measures the sum of negative log-likelihood of the predictions p_{dl} of the actual labels. A large deviation between \vec{Y}_d and p_{dl} will cause a greater value in the L_{CE} and thus will be penalised during training.

$$L_{CE} = - \sum_d \sum_l (\vec{Y}_{dl} \log(p_{dl}) + (1 - \vec{Y}_{dl}) \log(1 - p_{dl})) \quad (2)$$

For inference, a calibration threshold Th (default as 0.5) is set to assign the label to the document when $p_{dl} > Th$.

3.2. Hierarchical Label-wise Attention Network

Following the framework above, the neural document encoder in HLAN (as illustrated in Figure 1) takes into input the word sequence $x_d = \{x_{d1}, x_{d2}, \dots, x_{dn}\}$, where x_{di} denotes the sequence of tokens in the i th of all n sentences, and output the document representation. The distinction to HAN [10] is that HLAN represents the same document differently at both the word-level and the sentence-level regarding different labels. HLAN extends the contextual vectors in HAN to the label-wise contextual matrices, V_w and V_s . The document representation also becomes a matrix, C_d , where each row (corresponding to each label) has the same dimensionality as h .

As shown in Figure 1, the model consists of an embedding layer, hidden layers (hierarchical label-wise attention layers), and a prediction (or projection) layer. First, the embedding layer transforms the one-hot input representation u_{di} of each token in the sequence of the i th sentence x_{di} into a low-dimensional continuous vector, $e_{di} = W_e u_{di}$, where we used the neural word embedding algorithm, Word2vec [24], to pre-train W_e for its efficiency.

Second, we applied the Gated Recurrent Unit (GRU) [25], a type of RNN unit, to capture long-term dependencies in the clinical narrative. An RNN unit ‘‘reads’’ each token in the sequence one by one, every time producing a new hidden state $h^{(t)}$, corresponding to the token at time t . Different from the vanilla RNN unit, GRU additionally considers the previous tokens by using a reset gate $r^{(t)}$ and an update gate $z^{(t)}$. This allows to model the dependencies among tokens in long sequences. A GRU can be formally defined as in the Equations 3 below, where \vec{h} denotes the hidden states through forward processing,

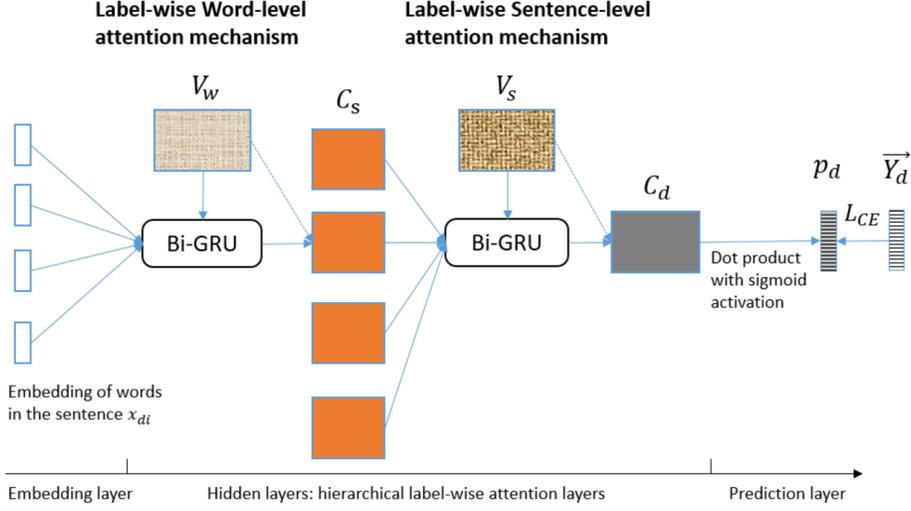


Figure 1: Hierarchical Label-wise Attention Network (HLAN)

σ is a non-linear activation function (e.g. sigmoid function), $W_{hr}, W_{hz}, W_{h\tilde{h}} \in \mathbb{R}^{d_h \times d_h}$ are weights, and $b_r, b_z \in \mathbb{R}^{d_h}$ represent bias terms. A bi-directional adaptation was applied by concatenating the hidden states at each time after read the sequence both forwardly (\rightarrow) and backwardly (\leftarrow) to form a more comprehensive representation, $h^{(t)} = [\vec{h}^{(t)}; \overleftarrow{h}^{(t)}] \in \mathbb{R}^{2d_h}$. This sub-architecture is generally known as Bi-GRU [25].

$$\begin{aligned}
 r^{(t)} &= \sigma(W_{er}e^{(t)} + W_{hr}\vec{h}^{(t-1)} + b_r) \\
 z^{(t)} &= \sigma(W_{ez}e^{(t)} + W_{hz}\vec{h}^{(t-1)} + b_z) \\
 \tilde{h}^{(t)} &= \tanh(W_{e\tilde{h}}e^{(t)} + W_{h\tilde{h}}(r^{(t)} \circ \vec{h}^{(t-1)})) \\
 \vec{h}^{(t)} &= (1 - z^{(t)}) \circ \vec{h}^{(t-1)} + z^{(t)} \circ \tilde{h}^{(t)}
 \end{aligned} \tag{3}$$

For simplicity, we use the function $h = \text{Bi-GRU}(e, \Theta)$ to denote the whole process (with bi-directional concatenation of hidden states) above. Instead of applying one single Bi-GRU layer to represent the whole document, we applied a word-level Bi-GRU to represent each sentence and then a sentence-level one to represent the whole document, as illustrated in Figure 1. This captures the hierarchical structure of the document and relieves the burden of having a too lengthy sequence for each GRU [1] (e.g. from the original sequence length 2500 in the MIMIC-III discharge summaries to only 100 on the word level and 25 on the sentence level).

A common way is to represent the whole sequence as the concatenated hidden state $h^{(t)}$ of the last time t . This representation tends to emphasise the ending elements (i.e. words or sentences) and does not discriminate between the elements in a sequence. In fact, the key information for medical cod-

ing is contained in a well selected part of the lengthy discharge summary. We therefore use an attention mechanism to learn a weighted average of the hidden states to form a final representation as in [10, 20]. The attention scores are based on an alignment (or a similarity computation) of each hidden representation in a sequence to a context vector. The context vector is usually *shared* for all labels as in [10, 15], whereas in medical coding, it is essential to interpret the amount of attention paid regarding a *specific* medical code to the clinical note.

$$\begin{aligned}
 h^{(i)} &= \text{Bi-GRU}(e, \Theta_w) \\
 v^{(i)} &= \tanh(W_w h^{(i)} + b_w) \\
 \alpha_{wl}^{(i)} &= \frac{\exp(V_{wl} \bullet v^{(i)})}{\sum_{o \in [1, n_t]} \exp(V_{wl} \bullet v^{(o)})} \\
 C_{sl} &= \sum_{i \in [1, n_t]} \alpha_{wl}^{(i)} h^{(i)}
 \end{aligned} \tag{4}$$

Thus, the adapted, *label-wise word-level attention mechanism* is defined in Equations 4 above. The context matrix for the word-level attention mechanism is denoted as $V_w \in \mathbb{R}^{|Y| \times d_w}$, where each row V_{wl} (of attention layer size d_w) is the context vector corresponding to the label y_l . The attention score $\alpha_{wl}^{(i)}$ for the label y_l is calculated as a softmax function of the dot product similarity between the vector representation $v^{(i)}$ (transformed from the i th hidden state $h^{(i)}$ with a feed-forward layer) and the context vector V_{wl} for the same label. n_t denotes the number of tokens in a sentence. The sentence representation C_{sl} , as a row vector in $C_s \in \mathbb{R}^{|Y| \times 2d_h}$, for the label y_l , is computed as the weighted average of all the hidden state vectors $h^{(i)}$.

In a similar way, we can compute the *label-wise sentence-level attention mechanism* as defined in Equations 5, which encodes each row C_{sl} in the sentence representations C_s to a label-wise sentence representation $S_l^{(r)}$, to be non-linearly transformed to $U_l^{(r)}$ and aligned to the corresponding row V_{sl} in sentence-level contextual matrix $V_s \in \mathbb{R}^{|Y| \times d_s}$, and outputs the sentence-level attention scores α_{sl} (for a label y_l) and the document representation matrix $C_{dl} \in \mathbb{R}^{|Y| \times 4d_h}$. To note that the dimensionality of S_l^r and thus C_{dl} are further doubled to $4d_h$ through the Bi-GRU process.

$$\begin{aligned} S_l^{(r)} &= \text{Bi-GRU}(C_{sl}, \Theta_S) \\ U_l^{(r)} &= \tanh(W_S S_l^{(r)} + b_S) \\ \alpha_{sl}^{(r)} &= \frac{\exp(V_{sl} \bullet U_l^{(r)})}{\sum_{q \in [1, n]} \exp(V_{sl} \bullet U_l^{(q)})} \\ C_{dl} &= \sum_{r \in [1, n]} \alpha_{sl}^{(r)} S_l^{(r)} \end{aligned} \quad (5)$$

Then, we use a label-wise, dot product projection with logistic sigmoid activation to model the probability of each label to each document, as defined in Equation 6, adapted from Equation 1. The parameters in w_l are row vectors in the projection matrix W .

$$p_{dl} = \sigma(w_l C_{dl} + b_l) \quad (6)$$

We finally optimise the binary cross-entropy loss function in Equation 2 with L_2 regularisation using the Adam optimiser [26].

3.3. Label Embedding Initialisation

For automated medical coding, the diagnostic and procedural codes (or labels) have complex semantic relations, and can potentially be leveraged to improve prediction. Clinically, these code relations represent the correlation among diseases and medical procedures from the medical coding practice.

As we reviewed in Section 2.3, previous studies on weight initialisation to address the label correlation issue mostly focus on a co-occurrence based representation of labels [22, 23]. Both studies dedicate a neuron in the final hidden layer to initialise one single co-occurrence pattern. There are, however, very limited neurons to be assigned to initialise the massive number of label relations, especially for the large label size in automated coding.

Instead of encoding the sparse co-occurrence patterns of labels, we learn low-dimensional, dense, label embeddings. For two correlated labels y_j and y_k , e.g. 486 (Pneumonia) and

518.81 (Acute respiratory failure), one would expect that the prediction of one label has an impact on the other label for some clinical notes, i.e. p_{dj} is correlated or has a similar value to p_{dk} . To achieve this, according to Equations 1 or 6, we propose to initialise their corresponding weights w_j and w_k (corresponding to the labels y_j and y_k) in W with a label representation E which reflects the actual label correlation (e.g. similarity between y_j and y_k) in a continuous space.

A straightforward idea is thus to initialise the projection matrix W using E as pre-trained label embeddings, e.g. with a neural word embedding algorithm, learned from the label sets in the training data, $\{\vec{Y}_d | d \in [1, m]\}$. For initialisation, we pre-train the label embeddings E with dimensionality the same as W . We used the Continuous Bag of Words algorithm in word2vec [24] for its efficiency and its power to represent the correlations of the labels. Figure 2 shows an intuitive visualisation, for which we used an unsupervised technique, T-SNE (t-distributed Stochastic Neighbor Embedding), to reduce the dimensionality of the learned label embeddings while preserving the local similarity and structure of the labels [27]. It can be observed that the ICD-9 code learned from the MIMIC-III training label sets can capture the semantic relations that are distinct from the ICD-9 hierarchy. For example, 486 (Pneumonia) and 518.81 (Acute respiratory failure) appear closely on the bottom while they are not under the same parent in the ICD-9 hierarchy.

Besides, the label embedding initialisation can also be applied to the context matrices V_w and V_s (see Figure 1) in the label-wise attention mechanisms. Taking the word-level attention mechanisms in Equation 4 as an example, we can initialise V_{wl} with the pre-trained label embedding E_l for the label y_l . This imposes a tendency for context vectors of the correlated labels to align the Bi-GRU encoded token representation v_i in a geometrically similar way. Similarly, we can also initialise the label-wise attention layer in CNN+att [7] and in HA-GRU [1]. While the initialised layers are dynamically updated during the training, the tendency that imposed by label embeddings remains for most neural networks; we will empirically demonstrate this in the analysis of initialised layers in Section 4.8.

For automated coding, the approach can be extended by initialising label embeddings with the clinical ontologies and description texts of ICD codes. However, due to the different nature of the knowledge (i.e. embedded label relations), the external sources may bring contradictory label correlations to the ones in the dataset, as also discussed in [28]. In this research, we focus on leveraging label relations from the label sets alone

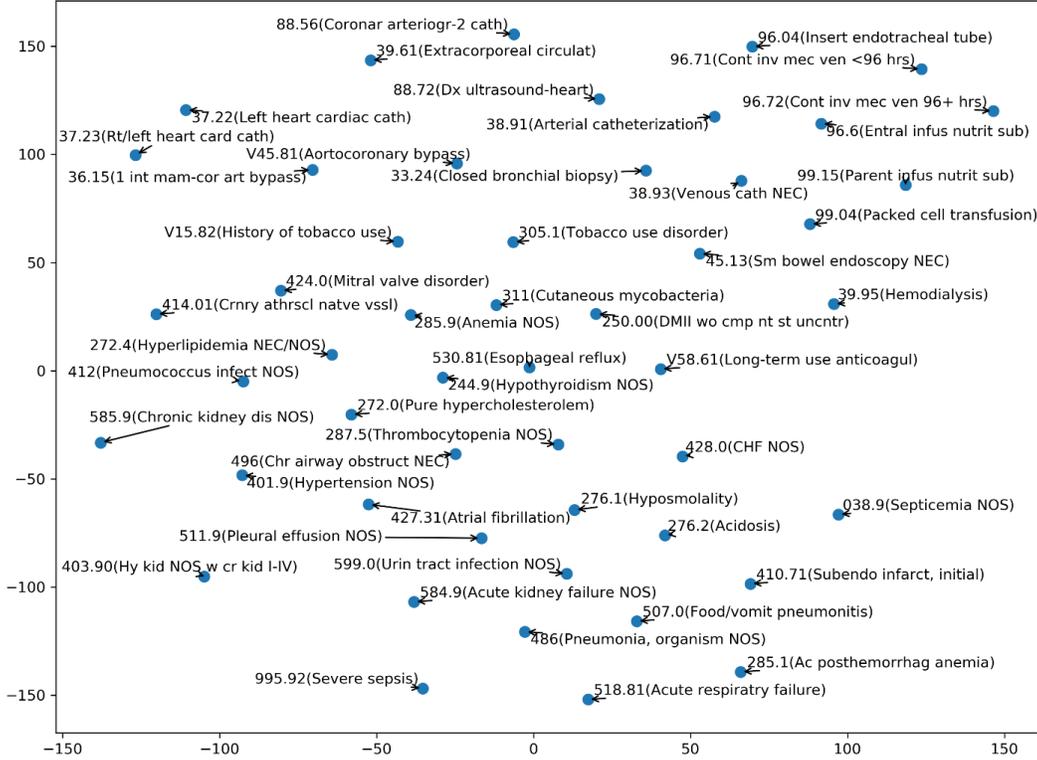


Figure 2: The 2-dimensional T-SNE plot of word2vec Continuous Bag of Words label embeddings of the 50 ICD-9 codes in MIMIC-III-50, trained on the whole training label sets, $\{\vec{Y}_d | d \in [1, m]\}$, in MIMIC-III.

as in [22, 23], as it directly reflects the label correlation of the coding practice that generated the dataset, and leave the integration of external knowledge for a future study.

4. Experiments

We tested HLAN and several strong baseline models, with label embedding initialisation, on three data sets based on the MIMIC-III database. The main results show the comparative results of HLAN to other state-of-the-art models on the datasets and the consistent improvement with label embedding initialisation. More importantly, through an analysis on model interpretability, we also show that HLAN can provide a more comprehensive explanation using the label-wise word and sentence-level attention mechanisms. Analysis of the layers initialised with label embeddings further reveals the effect of the initialisation approach. The source code of our implementation and the results are openly available at <https://github.com/acadTags/Explainable-Automated-Medical-Coding>.

4.1. Datasets

We used the benchmark dataset, MIMIC-III (“Medical Information Mart for Intensive Care”) [2], which contains clinical data from adult patients admitted to the critical care unit in the Beth Israel Deaconess Medical Center in Boston, Massachusetts between 2001 and 2012, to validate our approach. The ICD-9 codes annotated by professionals in the dataset were used as labels. We focused on discharge summaries and followed the preprocessing and data split from [7]. The preprocessed full MIMIC-III dataset has 8,922 unique codes as labels assigned to 52,724 discharge summaries, where 47,724 of them (from 36,998 patients) were used for training, 1,632 for validation, and 3,372 for testing. We also used the same top-50 setting (termed as “MIMIC-III-50”) from [7], which narrows down the labels to the top 50 by their frequencies (codes and their frequencies are available in Table S1 in the supplementary material). This has 8,067 discharge summaries for training, 1,574 for validation, and 1,730 for testing.

We further created a subset of discharge summaries annotated using the COVID-19 shielding related ICD codes. This simulates the application of identifying key patients for shield-

ing during the pandemic. We used the ICD-9 codes matched to the ICD-10 codes selected by the NHS to identify patients with medium or high risks during COVID-19. The considered patients were related to solid organ transplant recipients, people with specific cancers, with severe respiratory conditions, with rare diseases and inborn errors of metabolism, on immunosuppression therapies, or who were pregnant with significant congenital heart disease², which is still in active use and under maintenance at the time of writing this paper. While the actual EHR data and the shielded patient list from the NHS are not easy to obtain, the ICD-10 codes are openly available for reuse³. We thus used MIMIC-III to simulate the task of identifying patients for shielding during COVID-19. We selected those appeared at least 50 times in the MIMIC-III dataset, resulting in 20 ICD-9 codes (out of 79 matched codes), available in Table S2 in the supplementary material. After filtering the MIMIC-III dataset with the selected ICD-9 codes, there are 4,574 discharge summaries for training, 153 for validation, and 322 for testing. We name this dataset as “MIMIC-III-shielding”.

Statistics of the three datasets are in Table 1. Denoted by Ave , the average number of labels per document (or label cardinality) in the training set of MIMIC-III, MIMIC-III-50, and MIMIC-III-shielding are 15.88, 5.69, and 1.08, respectively. While all originated from MIMIC-III database, the three datasets represent different case scenarios in automated medical coding with various scales of data and vocabulary size (“vocab”), number of labels to predict, and the average number of labels per document. While the full MIMIC-III dataset has much more training instances, it is more complex as its number of labels $|Y|$ and vocabularies are significantly greater than MIMIC-III-50 and MIMIC-III-shielding.

Table 1: Statistics of the datasets

Dataset	Vocab	Train	Valid	Test	$ Y $	Ave
MIMIC-III-50	59,168	8,067	1,574	1,730	50	5.69
MIMIC-III-shielding	47,979	4,574	153	322	20	1.08
MIMIC-III	140,795	47,724	1,632	3,372	8,921	15.88

Figures of ICD-9 code distributions by frequency in the three datasets are available in Figure S1 in the supplementary material, along with the list of the selected codes (and their frequencies) in the MIMIC-III and MIMIC-III-shielding datasets. The

²A clearer description of the “high risk” category is in <https://digital.nhs.uk/coronavirus/shielded-patient-list/methodology/background>

³To see the annexe B in <https://digital.nhs.uk/coronavirus/shielded-patient-list/methodology/annexes>

statistics show a high imbalanced characteristics of the labels in all three data settings. Most label occurrences are from a few labels and there is a long-tail of labels having very low frequencies. This is most pronounced in the full label setting (“MIMIC-III”) and also presented in the other two datasets.

4.2. Experiment Settings

We implemented the proposed Hierarchical Label-wise Attention Network (HLAN) model and the other baselines for comparison:

1. CNN, Convolutional Neural Network, which is essentially based on [29] for text classification, and applied in [6, 30] for automated medical coding.
2. CNN+att (or CAML), CNN with a label-wise attention mechanism, proposed in [7].
3. Bi-GRU, Bi-directional Gated Recurrent Unit [25] for multi-label classification. The document representation is set as the last concatenated hidden state $h^{(l)}$.
4. HAN, Hierarchical Attention Network [10], which can be considered as a downgraded model of HLAN when the attention mechanisms are shared for all labels (see Figure 1, when V_w , V_s , and C_s , C_d become vectors, same for all labels).
5. HA-GRU, Hierarchical Attention bi-directional Gated Recurrent Unit, proposed in [1], which can be considered as a downgraded model of HLAN when the word-level attention mechanism is shared for all labels (see Figure 1, when V_w and C_s become vectors, same for all labels, while V_s and C_d are the same as in HLAN).

We applied the label embedding initialisation approach (denoted as “+LE”) to all the models above. We pre-trained the label embeddings E from the label sets in the training data with the word2vec (Continuous Bag of Words with negative sampling) algorithm [24]. The label embeddings have the dimension same as the final hidden layer or the label-wise attention layer(s) in each neural network model. We applied the Python Gensim package [31] to train embeddings, by setting the window size as 5 and minimum frequency threshold (“min_count”) as 0. Label embeddings were normalised to unit length for initialisation. Xavier initialisation [32] was used for labels not existing in the training data for faster model convergence. We used the same setting to train and initialise the 100-dimension word embeddings W_e from the documents.

The implementations of HLAN and HA-GRU were adapted from our previous implementation⁴ of HAN in [15] using the Python Tensorflow [33] framework, originated from brightmart’s implementation⁵, all under the MIT license. We adapted HA-GRU with the sigmoid activation and binary cross-entropy as described in Section 3.1, instead of the softmax activation used in the original paper [1], for a controlled comparison with other models. For CNN, CNN+att, and Bi-GRU, we adapted the implementation⁶ from [7] using the PyTorch framework [34] with the same parameters for MIMIC-III and MIMIC-50 from [7]. For MIMIC-III-shielding, we used the same hyper-parameters as in MIMIC-50. We did not get the results with HA-GRU and HLAN for the MIMIC-III dataset, due to the memory limit caused by the large label size ($|Y| = 8,921$), while for MIMIC-III-50 and MIMIC-III-shielding, we obtained the results of all models.

The input token length for the models was padded to 2,500 as in [7]. We optimised the precision@ k or micro- F_1 metrics (defined in Section 4.3) during the training⁷, according to the implementation in [7]. The batch size for CNN, CNN+att, Bi-GRU were set as 16 as in [7], for HLAN and HA-GRU as 32, and HAN as 128. For HLAN, HAN, and HA-GRU, we tried both a customised rule-based parsing of real sentences with Spacy⁸ and using text chunks of fix length as “sentences”; for both ways, we set the sentence length as 25 and padded the number of sentences to 100. The dimensions of the final document representation were 512, 500, 50, 400 for Bi-GRU, CNN, CNN+att, and HLAN (also HAN and HA-GRU), respectively. All models were trained using a single GeForce GTX TITAN X server, and the trained HLAN, HA-GRU, and HAN models were further tested using a CPU server (4-core, Intel(R) Xeon(R) Platinum 8259CL CPU @ 2.50GHz). The detailed hyper-parameter settings, containing learning rate, dropout rate, and CNN specific parameters (kernel size and filter size), with the estimated training and testing times, are in Table S3 in the supplementary material.

We also experimented with BERT as the neural document encoder. Due to the GPU memory limit, we tested the nor-

mal size of a BERT model, i.e. BioBERT-base [35], which had been further pre-trained with PubMed paper abstracts⁹ and full texts¹⁰; we used a sliding window approach to address the token limit issue (512 tokens) in BERT. Our results from the BioBERT-base model were similar to the results in [19], significantly worse than HLAN and CNN¹¹. We believe further adaptations are necessary for BERT models on automated medical coding and leave the direction for a future study.

4.3. Evaluation Metrics

For comparison, we applied the same set of label-based metrics as in [7] and according to the evaluation of multi-label classification algorithms [14, 13]. The chosen metrics include micro- and macro-averaging precision (P), recall (R), F_1 score (F_1), area under the receiver operating characteristic curve (AUC), and the precision@ k .

The micro-averaging metrics treat each document-label as a separate prediction, whereas the macro-averaging metrics are an average of the per-label results. Micro- and macro-averaging applies to all the binary evaluation metrics including precision, recall, AUC. For example, the micro- and macro-averaged precision is defined in Equation 7 below. Recall is calculated in a similar way, but divided by all the true cases ($TP_i + FN_i$), and F_1 is then the harmonic mean of the calculated precision and recall, i.e. $F_1 = \frac{2 \times P \times R}{P + R}$. The AUC is defined by two metrics, the true positive rate (or recall) on the Y axis and false positive rate on the X axis, depicting the tradeoff between the two metrics when varying the calibration threshold Th [36]. The overall performance of a classifier (with a set of varied Th) can thus be reflected by AUC.

$$\text{Micro-P} = \frac{\sum_{i=1}^L TP_i}{\sum_{i=1}^L TP_i + FP_i} \quad \text{Macro-P} = \frac{1}{|L|} \sum_{i=1}^L \frac{TP_i}{TP_i + FP_i} \quad (7)$$

In some clinical application or epidemiological studies, only one type of code (either the diagnosis or the procedure code) is favoured. Thus, for the MIMIC-III full label setting, we also report Micro- F_1 results on the diagnosis codes (F_1 -diag) and procedure codes (F_1 -proc) separately as in [7].

Furthermore, we report the example-based metric, precision@ k as in [7], averaged over all the documents,

⁴<https://github.com/acadTags/Automated-Social-Annotation/tree/master/2%20HAN>

⁵https://github.com/brightmart/text_classification

⁶<https://github.com/jamesmullenbach/caml-mimic>

⁷We optimised precision@ k for CNN, CNN+att, and Bi-GRU for MIMIC-III and MIMIC-III-50, and micro- F_1 for all other models and for the MIMIC-III-shielding dataset.

⁸We parsed sentences using the rule-based pipeline component in Spacy with adding double newlines as another rule to segment sentences, see <https://spacy.io/usage/linguistic-features#sbd>.

⁹<https://pubmed.ncbi.nlm.nih.gov/>

¹⁰<https://www.ncbi.nlm.nih.gov/pmc/>

¹¹We thus do not report the BERT results here but make the implementation details and results available on <https://github.com/acadTags/Explainable-Automated-Medical-Coding>.

where each precision score is the fraction of the true positive in the top- k labels, having highest score p_{di} , for the document d . The idea is to simulate the real-world scenario that the system recommending k predicted medical codes and to evaluate the percentage of them being correct. The number of top-ranked labels k were set as 8 for MIMIC-III, 5 for MIMIC-III-50 to be consistent to the study [7], and 1 for MIMIC-III-shielding, near to the average number of labels per document (see Table 1).

4.4. Main Results

We report the mean and the standard deviation (i.e. the square root of variance) of the testing results of 10 runs with randomly initialised parameters for each model. The results of the MIMIC-III-50, MIMIC-III-shielding, and MIMIC-III datasets are shown in Table 2, 3, and 4, respectively.

For the top 50 label dataset (MIMIC-III-50, see Table 2), HLAN performed the best among all experimental settings, achieved significantly better Micro-AUC (91.9%), Micro- F_1 (64.1%), and Precision@5 (62.5%) than the second best model, CNN. This shows the advantage of the hierarchical label-wise attention mechanisms for top-50 code prediction. With the same calibration threshold, the precision of HLAN is better than CNN absolutely by 15% (73.2% vs. 57.7%), while recall is lower with a similar absolute value, indicating that tuning the threshold to balance precision and recall could further improve the F_1 scores.

For code related to high-risk patients for shielding during the COVID-19 pandemic (MIMIC-III-shielding, see Table 3), results (of Micro-AUC) show that HLAN (96.9%) and HAN (97.6%) performed comparably to the best performed model, CNN (97.9%). HLAN obtained a high value of precision@1, slightly below CNN by 1% (81.2% vs 82.2%), while the difference was not significant ($p > 0.05$). The better performance of CNN (or HAN) may be because that smaller datasets like MIMIC-III-shielding, with much fewer documents and labels (see Table 1), tends to favour models with simpler architectures.

In both MIMIC-III-50 and MIMIC-III-shielding, HA-GRU did not perform better than HLAN, this shows that the label-wise word-level attention mechanisms in HLAN further improved the performance. Also, surprisingly, the HLAN or HAN models with the real sentence split did not perform better (up to 2.8% less Micro- F_1) than using text chunk “sentences” (of 25 continuous tokens) in all three datasets. This is probably because, with the sentence split setting, some tokens and sentences were lost during the padding procedure, which could significantly affect the performance.

For the full label setting (“MIMIC-III”), HAN has better results of Micro-AUC and precision@8 than the vanilla CNN and Bi-GRU, but worse than the CNN+att approach specifically tuned for this dataset. With label embedding initialisation, CNN+att+LE achieved significant best results on MIMIC-III (an Micro-AUC of 98.6%). It is worth to further explore to enhance the scalability of HLAN so that it can process datasets with large label sizes. Also to note that results of the Macro-level metrics (averaging over labels) were dramatically lower than the Micro-level ones (calculated from document-label pairs), showing the strong imbalance of labels in MIMIC-III (see Section 4.1).

Injecting the code relations through label embedding consistently boosted the performance of automated medical coding. It is clear that most models were improved with label embedding initialisation (“+LE”). Models were affected to different extend by label embedding: CNN+att model was mostly improved with “+LE” (an increase of 6.6% Macro-AUC on MIMIC-III-shielding), the rest models (CNN, Bi-GRU, HA-GRU) being relatively less affected, while there was no significant improvement for HLAN or HAN on the datasets. This may due to the fact that the prior layers, e.g. hierarchical layers and the label-wise attention layers, could already learn some of the label relations. We thus further analyse the LE-initialised layers in Section 4.8 to understand the effect of label embedding initialisation. Besides, most metrics with the “+LE” models also have higher stability (i.e. reduced variance); and low variance is an essential characteristic to deploy a model in the clinical setting.

4.5. Result for each label

Apart from the overall performance of the models, it is also essential to see how the models perform regarding each medical code. Figures 3 show the precision and recall of the five diagnosis codes having the highest and the lowest frequencies in MIMIC-III-50. For this analysis, we selected the three best performing models, CNN, CNN+att, and HLAN, all with label embedding initialisation (“+LE”), in terms of AUC metrics for MIMIC-III-50 (see Table 2). We provide the full per-label results of HLAN+LE with the MIMIC-III-50 and MIMIC-III-shielding datasets in Table S1-S2 in the supplementary material.

In Figures 3, we can observe that the overall trend of performance is generally consistent to, while not solely dependent on, the label frequency in the training data. For the five most frequent labels, the models achieved around 70%-90% precision and recall. For example, in terms of precision, HLAN obtained

Table 2: Results on MIMIC-III-50 dataset (50 labels)

Model	Macro				Micro				Top-k
	AUC	P	R	F_1	AUC	P	R	F_1	P@5
CNN	88.1±0.3	51.5±0.9	67.4±1.0	58.4±0.5	90.9±0.2	55.6±1.1	71.2±0.9	62.4±0.6	61.8±0.3
+LE	<u>88.3±0.3</u>	<u>53.0±1.0*</u>	66.7±1.5	59.1±0.5*	<i>91.3±0.1*</i>	<i>57.7±1.4*</i>	70.4±1.4	<u>63.4±0.5*</u>	<u>62.1±0.3</u>
Bi-GRU	80.6±1.1	47.2±3.2	36.7±2.6	41.2±2.3	85.5±1.0	58.1±3.2	45.8±2.2	51.2±1.9	51.3±1.7
+LE	80.9±0.8	47.3±2.0	39.2±2.0*	42.8±1.5	85.8±0.7	57.5±2.2	48.4±2.1*	<u>52.5±1.3*</u>	<u>52.1±1.2</u>
CNN+att	88.1±0.0	63.1±0.1	<u>48.4±0.2*</u>	<u>54.8±0.2*</u>	91.1±0.0	70.9±0.2	<u>53.1±0.2*</u>	60.7±0.1	60.8±0.1
+LE	<u>88.3±0.0*</u>	<u>64.3±0.3*</u>	46.0±0.1	53.6±0.1	<u>91.3±0.0*</u>	<u>71.6±0.1*</u>	52.5±0.1	60.6±0.1	<u>61.6±0.1*</u>
HAN	87.0±0.4	<u>61.7±2.7</u>	46.3±2.3	52.8±1.1	90.1±0.3	<u>68.2±3.1</u>	52.9±2.4	59.4±0.7	59.5±0.7
+LE	87.3±0.4	61.3±3.2	46.9±2.9	<u>53.0±1.1</u>	90.3±0.4	67.9±4.1	54.2±2.8	60.1±0.7	<u>59.9±0.8</u>
HA-GRU	85.3±1.3	59.3±2.5	43.1±4.0	49.9±3.4	89.2±0.9	69.5±0.6	48.7±4.1	57.2±2.8	57.9±1.7
+LE	86.4±0.7*	<u>62.1±1.9*</u>	44.3±2.3	51.7±1.9	<u>90.1±0.5*</u>	<u>71.1±1.2*</u>	50.7±2.3	59.1±1.4*	<u>59.5±1.0*</u>
HLAN	88.4±0.7	65.0±1.2	51.0±2.6	57.1±1.6	<i>91.9±0.4</i>	72.9±0.8	<u>57.3±2.5</u>	64.1±1.4	62.5±0.7
+LE	88.4±0.5	65.5±1.5	50.2±1.1	56.8±0.8	91.9±0.3	73.2±0.6	56.9±1.0	<u>64.0±0.7</u>	<u>62.4±0.6</u>
+sent split	86.9±0.5	63.6±1.3	47.8±2.4	54.5±1.7	90.4±0.3	71.5±1.2	53.8±2.1	61.4±1.2	60.2±0.7

The results of better metric score between the model with label embedding initialisation (“+LE”) and the model *not* using LE initialisation are underlined, and the asterisk (*) further marks the paired two-tailed t-tests with .95 significant level ($p < 0.05$) between them. The best result for each metric (column) is in **bold**. The AUC, F_1 , and P@5 scores in HLAN models with *italics* indicates their significantly improved results ($p < 0.05$) over the second best model category (i.e. HLAN vs. CNN). The model with lower variance is preferred if the average scores are the same.

Table 3: Results on MIMIC-III-shielding dataset (20 labels)

Model	Macro				Micro				Top-k
	AUC	P	R	F_1	AUC	P	R	F_1	P@1
CNN	<u>96.9±0.2*</u>	59.8±1.2	59.6±1.3	59.7±0.9	<i>97.9±0.4*</i>	80.5±1.3	76.2±0.9	78.3±1.0*	82.2±0.8
+LE	96.7±0.2	60.4±2.8	60.4±2.3	60.4±2.3	97.6±0.3	78.8±2.5	76.4±1.8	77.5±0.7	81.6±0.9
Bi-GRU	91.9±1.4	57.4±3.1	43.4±2.2	49.4±2.2	93.6±0.8	77.9±2.8	58.5±1.9	66.8±1.2	72.2±1.6
+LE	92.0±1.6	<u>58.6±1.5</u>	46.8±2.3*	<u>52.0±1.4*</u>	<u>95.1±0.7*</u>	78.1±2.1	<u>61.8±2.7*</u>	<u>68.9±1.6*</u>	<u>75.1±2.0*</u>
CNN+att	88.9±1.3	46.7±4.6	37.6±2.4	41.7±3.3	93.5±0.2	86.9±1.2*	52.9±2.8	65.7±2.0	70.0±2.6
+LE	<u>95.5±0.0*</u>	<u>62.1±2.2*</u>	48.4±1.9*	<u>54.4±2.0*</u>	<u>96.1±0.0*</u>	83.3±0.5	<u>61.4±0.6*</u>	<u>70.7±0.3*</u>	<u>77.7±0.3*</u>
HAN	96.0±1.4	66.4±2.7	<u>58.2±2.0</u>	62.0±2.0	97.4±0.3	82.9±1.8	<u>68.7±2.4</u>	<u>75.1±1.5</u>	78.1±1.7
+LE	96.4±1.3	65.2±2.1	56.5±2.9	60.5±2.3	97.6±0.3	83.4±1.2	68.2±2.0	75.0±1.2	79.2±1.7
HA-GRU	93.4±2.0	<u>60.9±3.9</u>	<u>51.6±2.8</u>	<u>55.8±3.1</u>	96.7±0.4	<u>83.0±2.1</u>	65.8±2.2	<u>73.4±1.6</u>	<u>80.3±1.5</u>
+LE	93.9±2.0	59.2±4.3	49.7±4.2	54.0±4.1	96.8±0.9	81.3±4.0	<u>66.3±4.1</u>	73.0±3.6	79.1±4.3
HLAN	93.5±2.5	59.8±2.9	<u>53.2±2.6</u>	<u>56.3±2.4</u>	<u>96.9±0.7</u>	81.4±1.8	<u>69.0±2.9*</u>	<u>74.6±1.6</u>	<u>81.2±1.2</u>
+LE	93.5±1.9	60.5±4.2	52.7±5.0	56.3±4.6	96.5±0.4	81.8±2.8	65.6±4.0	72.7±3.1	79.8±3.0
+sent split	<u>94.5±1.2</u>	<u>60.9±2.1</u>	51.7±3.1	55.8±2.3	96.3±0.2	81.4±2.2	64.4±2.5	71.9±1.7	77.8±2.6

The results of better metric score between the model with label embedding initialisation (“+LE”) and the model *not* using LE initialisation are underlined, and the asterisk (*) further marks the paired two-tailed t-tests with .95 significant level between them. The best result for each metric (column) is in **bold**. The AUC, F_1 , and P@1 scores in CNN models with *italics* indicates their significantly improved results ($p < 0.05$) over the second best model category (i.e. CNN vs. HAN or HLAN). The model with lower variance is preferred if the average scores are the same.

highest to 91.7% for 427.31 (Atrial fibrillation) and lowest to 71.4% for 584.9 (Acute kidney failure). For the five least frequent labels, the results were much worse due to the fewer training data for the labels and the imbalance issue. For precision, HLAN generally performs better than CNN and CNN+att, especially there is a significant gap for low frequent labels; while for recall, CNN outperforms the other two models. We also note that the precision and recall could be tuned in favour of only one of them through changing the calibration threshold Th (now set as the default value, 0.5), considering the need and the preference of the coding work when deploying the model to support coding professionals.

4.6. Model Explanation with Hierarchical Label-wise Attention Visualisation

A critical requirement of the clinical use of automated medical coding systems is their explainability or interpretability. We propose to use label-wise word-level and sentence-level attention mechanisms in HLAN to enhance the explainability of the model. The learned word-level and sentence-level attention scores for the label y_l are $\alpha_{wl} \in (0, 1)$ and $\alpha_{sl} \in (0, 1)$ (see Equations 4 and 5, respectively). For a more concise visualisation, we propose a *sentence-weighted* word-level attention score $\tilde{\alpha}_{wl}$ to only highlight the words from the salient sentences. This adapted word-level attention score is calculated as $\tilde{\alpha}_{wl} = \mu \alpha_{sl} \alpha_{wl}$, where α_{sl} is the attention score of the sentence where the word is belong to and μ is a hyperparameter to control the

Table 4: Results on MIMIC-III dataset (8,922 labels)

Model	Macro				Micro					Top-k	
	AUC	P	R	F_1	AUC	P	R	F_1	F_1 -diag	F_1 -proc	P@8
CNN	81.8±0.7	4.5±0.4	3.7±0.5	4.1±0.4	97.0±0.1	51.0±2.8	36.9±1.7	42.8±0.9	41.1±1.0	50.7±0.9	59.6±0.5
+LE	<u>82.4±0.4*</u>	<u>4.7±0.4</u>	<u>3.6±0.2</u>	<u>4.1±0.2</u>	<u>97.1±0.1</u>	<u>53.0±2.6</u>	<u>36.9±1.2</u>	<u>43.4±0.6</u>	<u>41.7±0.6</u>	<u>51.3±0.9</u>	<u>60.3±0.4*</u>
Bi-GRU	83.5±1.6	4.9±0.2	3.6±0.5	4.1±0.4	97.3±0.3	52.8±4.6	34.8±2.2	41.8±1.5	39.3±1.6	51.7±1.3	58.9±2.2
+LE	<u>84.9±0.7*</u>	<u>5.0±0.4</u>	<u>3.6±0.5</u>	<u>4.2±0.5</u>	<u>97.6±0.1*</u>	<u>55.4±4.1</u>	<u>34.8±2.4</u>	<u>42.6±1.5</u>	<u>40±1.6</u>	<u>52.7±1.1</u>	<u>60.3±1.8</u>
CNN+att	88.6±0.2	7.7±0.2	6.4±0.3	7.0±0.2	98.4±0.0	62.8±0.3*	43.9±0.4	51.7±0.1	50.1±0.2	59.8±0.1	69.4±0.2
+LE	<u>90.2±0.0*</u>	<u>9.3±0.1*</u>	<u>8.0±0.1*</u>	<u>8.6±0.1*</u>	<u>98.6±0.0*</u>	<u>61.8±0.4</u>	<u>45.6±0.1*</u>	<u>52.5±0.1*</u>	<u>50.7±0.1*</u>	<u>60.7±0.1*</u>	<u>69.7±0.1*</u>
HAN	88.5±0.1*	5.4±0.2*	2.7±0.2*	3.6±0.2*	98.1±0.1	63.2±3.3	30.0±1.1*	40.7±0.7*	37.0±0.7*	52.6±0.9*	61.4±1.3*
+LE	<u>88.2±0.2</u>	<u>5.1±0.2</u>	<u>2.4±0.1</u>	<u>3.3±0.2</u>	<u>98.1±0.0</u>	<u>63.2±1.0</u>	<u>27.6±1.1</u>	<u>38.4±1.0</u>	<u>34.8±1.2</u>	<u>50.6±1.0</u>	<u>59.6±0.6</u>
+sent split	87.4±0.7	4.7±0.6	2.3±0.4	3.1±0.5	97.9±0.1	60.4±2.2	25.3±2.5	35.6±2.6	31.4±2.6	49.1±2.3	56.3±2.0

The results of better metric score between the model with label embedding initialisation (“+LE”) and the model *not* using LE initialisation are underlined, and the asterisk (*) further marks the paired two-tailed t-tests with .95 significant level between them. The best result for each metric (column) is in **bold**. The model with lower variance is preferred if the average scores are the same.

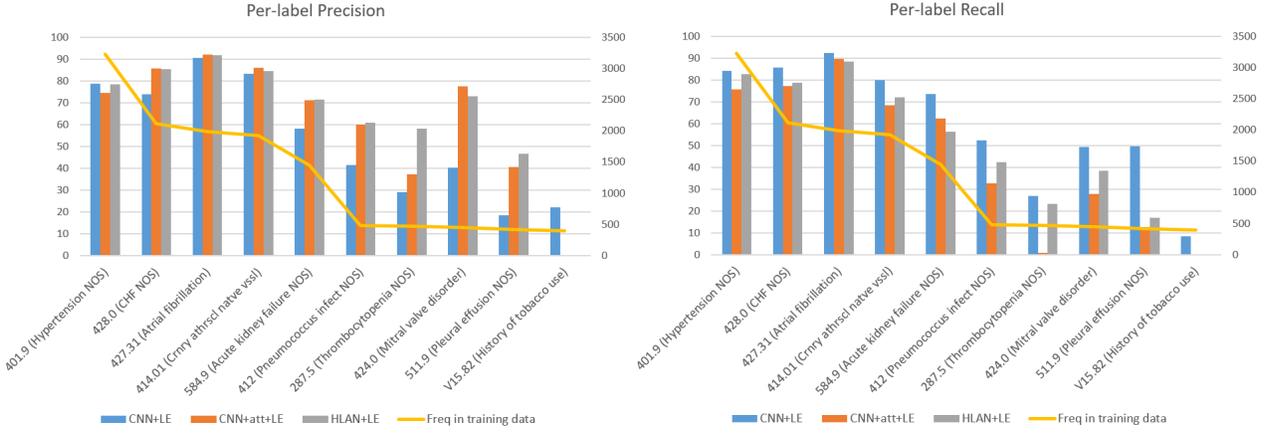


Figure 3: Precision and recall of the five most and the five least frequent ICD-9 diagnosis codes in the MIMIC-III-50 dataset. The bar chart (with the left y-axis) shows the metric score, while the line chart (with the right y-axis) shows the number of occurrences or the frequency (“Freq”) of the label in the training data.

magnitude of the final weighted attention score. A greater μ will result in highlighting more words in the clinical note and we empirically set μ as 5. We clip the value of $\tilde{\alpha}_{wl}$ to 1 if it is above 1.

An example attention visualisation for a random document (number 24) in MIMIC-III-50 using the model HLAN+LE with the parsed sentences (“+sent split”) is shown in Figure 4. The two columns on the left visualise the sentence-level attention scores α_{sl} for the two codes, 427.31 (Atrial fibrillation) and 428.0 (Congestive heart failure, unspecified), respectively. The highlighted sentences are corresponding to the sections “past medical history” and “discharge diagnosis” in the discharge summary. This is in line with our intuition that the key diagnosis information is likely to be contained in the two sections. The words are highlighted according to the adapted word-level attention score $\tilde{\alpha}_{wl}$. Words related to the code 427.31 is highlighted in yellow and for 428.0 in blue. It is clear that the most salient words are highlighted, and the model successfully recognised the abbreviations and alternative short forms com-

monly used by clinicians in the clinical note, for example “a fib” as a short form of atrial fibrillation and “chf” as the abbreviation of Congestive Heart Failure. This shows that the proposed HLAN model can learn to recognise the strongly correlated words (e.g. “a fib”) related to the label (e.g. the code 427.31) with label-wise attention mechanisms, even given the fact that the label description (i.e. the knowledge that 427.31 is “Atrial Fibrillation”) were not fed into the model during training. Other relevant words are highlighted, e.g., “ef” (short for Ejective Fraction), “pressor”, and “extremities”, which show a correlation to the code 428.0 while not indicating a causal relation to the diagnosis. We also note that the highlighted words like “age” and “drugs” were too general, which could not be directly related to the diagnosis from a clinician’s point of view. This may be related to the peaky distribution of the softmax (normalised exponential) function to form the attention scores (see Equation 4), paying the most of the attention to only a few (one or two) words in a long sentence.

For 427.31	For 428.0	Document #24 in MIMIC-III-50											
0.02	0.01	admission	date	discharge	date	service	surgery						
0.01	0.01	allergies	patient	recorded	as	having	no	known	allergies	to		drugs	attending
0.01	0.02	major	surgical	or	invasive	procedure	ex	lap	r	hemicolecotomy	mucous	fistula	
0	0	history	of	present	illness	age	over	f	presented	to	location	un	
0	0.01	admitted	to	hospital1	and	taken	directly	to	or	upon	arrival		
0.41	0.8	past	medical	history	pmhx	a	fib	aortic	stenosis	chf	last	ef	
0	0	pertinent	results	00pm	blood	wbc	rbc	hgb	hct	mcv	mch	mchc	
0	0.01	brief	hospital	course	age	over	f	transferred	from	location	un	and	
0	0.02	patient	was	taken	directly	to	the	operating	room	for	an	exploratory	
0	0	she	underwent	a	right	hemicolecotomy	mucous	fistula	ileostomy	gj	tube	placement	
0.01	0.01	fluid	balance	intraoperatively	included	units	ffp	units	plt	u	nits	prbc	
0	0	patient	was	kept	intubated	and	taken	directly	to	the	surgical	intensive	
0	0.02	she	required	maximum	pressor	support	to	maintain	sufficient	cardiac	index		
0	0.03	patient	did	show	signs	of	distal	ischemia	to	extremities	by	the	
0.01	0	family	meeting	at	latter	evening	decided	to	make	patient	cmo	patient	
0.01	0	medications	on	admission	last	name	un	amlodipine	mg	qd	benarepril	mg	
0.49	0	discharge	diagnosis	cardiopulmonary	arrest	perforated	colon	atrial	fibrillation	ventilatory	support	discharge	

Figure 4: An example of interpretation using attention visualisation from the Hierarchical Label-wise Attention Network (HLAN), the chosen example is a random document (index 24) in the MIMIC-III-50 dataset with two true positive labels, ICD-9 code 427.31 (Atrial fibrillation) and 428.0 (Congestive heart failure, unspecified). The two red columns show the sentence-level attention scores for the two codes respectively. The tokens highlighted by yellow (for code 427.31) or blue (for code 428.0) show the importance of them based on the value of sentence-weighted word-level attention scores. The deeper the colour, the higher the (sentence-weighted) attention scores, and thus the more important the highlight words or sentences contributes to the model prediction. Only the first part (11 tokens) of each sentence was shown for a clearer display.

4.7. Comparison of Model Explanations

Following the previous section, we further qualitatively analyse and compare the interpretability of the HLAN model and other baseline models. Table 5 shows how CNN, CNN+att, HAN, HA-GRU, and HLAN, all with label embedding initialisation, highlight the “important” part of a random document (number 24) to predict two different labels (427.31 and 428.0). The CNN¹² and CNN+att chose the most salient n -grams based on the max-pooling and the attention mechanism, respectively [7]. HLAN and its downgraded models, HA-GRU and HAN, alternatively, highlighted the important sentences and words. The distinction is that HAN has the same highlights of the same document for different labels (columns in Table 5), and HA-GRU has the same word-level but different sentence-level highlights across labels, while HLAN can highlight the most salient words and sentences for different labels. This gives HLAN the most comprehensive interpretability among the models.

Compared to CNN, we observe that CNN+att generated a more relevant set of n -grams. This is in accordance with the conclusion in [7]. We also found that the attention weights from CNN are unstable, i.e. the suggested n -grams from CNN were not the same among different runs. Compared to the interpretation with n -grams, highlighting the key sentences and words can produce a more comprehensive interpretation, as the latter is based on the whole hierarchical structure of a document.

¹²We further normalised the scores of n -grams in CNN based on max-pooling from [7] to probabilities, to be comparable to the attention scores in other models.

Especially with the sentence parsing (“HLAN+LE+sent split”, see the last row in Figure 5, corresponding to the visualisation in Figure 4), we can clearly see which sections of the discharge summary, along with words, contribute more to predict the label.

It is also interesting to see how the proposed model interpret when it predicted a medical code not previously assigned by the coding professionals. We selected some representative “false positive” results from the HLAN+LE model with sentence splitting in Table 6. We presented the prediction results and the highlighted explanations to an experienced clinician to validate and deduce the potential reason for the error. In Table 6, we observe that the model can explain the predictions with key sentences and words, therefore it is easier for us to know where there may have been a problem. For example, for the first two rows, “doc-68” and “doc-19” in MIMIC-III-50, the highlighted words and sentences are quite relevant to the non-coded, “false positive” ICD-9 code, indicating that there might have been missed coding or the disease was a past disease of the patient.

The false positives in “doc-1” and “doc-65” in MIMIC-III-shielding are errors related to the wrong correlations learned from the data, particularly regarding the high granularity and subtle difference among sub-type diseases. In “doc-1”, the highlighted words “htn elev lipids” show that the patient has a certain type of hypertension, but does not necessarily mean the predicted code 416.0 for “Primary pulmonary hypertension”. In “doc-65”, the strongly highlighted word “metastasis” actu-

Table 5: Comparison of model interpretability across deep learning models of true positive predictions on a random document (index 24) in the MIMIC-III-50 dataset

Model	doc-24 to predict 427.31 (Atrial fibrillation)	doc-24 to predict 428.0 (Congestive heart failure, unspecified)
CNN+LE	<i>n</i> -gram-1 (0.105): admission date discharge date service surgery allergies patient recorded as having no known...	<i>n</i> -gram-1 (0.096): ...surgical intensive care unit she required maximum pressor support to maintain sufficient cardiac index...
	<i>n</i> -gram-2 (0.083): ...surgical or invasive procedure ex lap r hemicolectomy mucous fistula ileostomy gj tube placement history of present illness...	<i>n</i> -gram-2 (0.075): ...past medical history pmhx a fib aortic stenosis chf last ef in osteoporosis reflux...
	<i>n</i> -gram-3 (0.075): ...presented to location un with perforated viscous hd stable upon transfer to location un...	<i>n</i> -gram-3 (0.071):...on ventilation support family meeting at latter evening decided to make patient cmo patient...
CNN+att+LE	<i>n</i> -gram-1 (0.026): ...upon arrival past medical history pmhx a fib aortic stenosis chf last ef in osteoporosis reflux doctor first name hx appendectomy many years ago social history non...	<i>n</i> -gram-1 (0.017): ...pressor support to maintain sufficient cardiac index patient did show signs of distal ischemia to extremities by the afternoon urine output post...
	<i>n</i> -gram-2 (0.023): ...diagnosis cardiopulmonary arrest perforated colon atrial fibrillation ventilatory support discharge condition death discharge instructions none followup instructions none	<i>n</i> -gram-2 (0.011): ...past medical history pmhx a fib aortic stenosis chf last ef in osteoporosis reflux doctor first name hx appendectomy many years ago social history non contributory ...
HAN+LE	sent-1 (0.34): medical history pmhx a fib(0.374) aortic stenosis chf(0.206) last ef in osteoporosis reflux(0.097) doctor first name hx appendectomy many(0.28) years ago social history non contributory	
	sent-2 (0.18): arthritis fosamax q week(0.039) coumadin(0.282) qd discharge medications none discharge disposition expired discharge diagnosis cardiopulmonary(0.019) arrest(0.109) perforated(0.134) colon(0.119) atrial(0.173) fibrillation(0.023) ventilatory(0.047) support discharge condition death	
	sent-3 (0.11): admission(0.201) date(0.263) discharge(0.05) date(0.055) service(0.075) surgery(0.118) allergies(0.062) patient(0.013) recorded(0.054) as having no known allergies to drugs attending first name3(0.021) lf(0.02) chief complaint perforated bowel(0.046) major	
HA-GRU+LE	sent-1 (0.62): arthritis fosamax q week coumadin(0.06) qd discharge medications none discharge disposition expired discharge diagnosis cardiopulmonary arrest perforated colon atrial fibrillation(0.94) ventilatory support discharge condition death	Did not predict 428.0 (i.e. false negative)
HLAN+LE	sent-1 (0.54): arthritis fosamax q week coumadin qd discharge medications none discharge disposition expired discharge diagnosis cardiopulmonary arrest perforated colon atrial(1.0) fibrillation ventilatory support discharge condition death	sent-1 (0.71): medical history pmhx a fib aortic stenosis chf(1.0) last ef in osteoporosis reflux doctor first name hx appendectomy many years ago social history non contributory
	sent-2 (0.18): medical history pmhx a fib(1.0) aortic stenosis chf last ef in osteoporosis reflux doctor first name hx appendectomy many years ago social history non contributory	
+sent split	sent-1 (0.49): discharge diagnosis cardiopulmonary arrest perforated colon atrial fibrillation(1.0) ventilatory support discharge condition death discharge instructions none followup instructions none	sent-1 (0.8): past medical history pmhx a fib aortic stenosis chf(0.888) last ef(0.112) in osteoporosis reflux doctor first name hx appendectomy many years ago social history non
	sent-2 (0.41): past medical history pmhx a fib(1.0) aortic stenosis chf last ef in osteoporosis reflux doctor first name hx appendectomy many years ago social history non	

* CNN and CNN+att suggested top *n*-grams, while HAN, HA-GRU, and HLAN suggested key sentences (“sent-”) and words in the sentences. “+sent split” denotes the HLAN model using real sentence splits. The numbers in the parentheses are the attention scores (e.g. for HLAN, α_v and α_s) in the models.

** For CNN and CNN+att, some of the suggested top-3 *n*-grams were combined together if any of them overlapped; up to five tokens before and after the top *n*-grams were also displayed.

*** For HAN, HA-GRU, and HLAN, the sentences were selected by those with sentence-level attention scores above 0.1 and the words were selected by those with the word-level attention scores above 0.01. Both HAN and HA-GRU predicted 427.31 but not 428.0. The word- and sentence-level attention weights of HAN are shared for all labels, therefore the interpretation is the same for both columns.

ally is related to pancreatic cancer, rather than the more common lung cancer as predicted. This wrong correlation may be due to the imbalance of vocabularies in the training data: there are 94 (about 2% out of 4,593) discharge summaries in the training data where “pancreatic” and “metasta” appeared together in MIMIC-III-shielding, while there are significant more discharge summaries (661, about 14%) where “lung” and “metasta” appeared together.

The error in the last example was due to the subtle difference between two codes (280.00 vs 280.03). We noticed some unexpected highlights (e.g. the local oncologist’s name code) in the last example (“doc-95” in MIMIC-III-shielding). This

may be related to that “hypercalcemia” (appeared in both sent-1 and sent-2) can be caused by cancer, while neutropenia can be caused by treatments like cancer chemotherapy. The word “chemotherapy” was highlighted in another sentence with an attention weight 0.09 (not presented as below the 0.1 threshold) and the word “neutropenia” in the document was not included during the padding process. While it is very likely that the neutropenia was induced by the drug for chemotherapy (that is, 280.03, Drug induced neutropenia), we did not find direct words in the report to point the cause of the disease (thus 280.00, Neutropenia, unspecified, is also appropriate).

In general, we observe that the label-wise attention mecha-

Table 6: Examples of false positives of HLAN (with label embedding “+LE” and sentence splitting “+sent split”) on MIMIC-III-50 and MIMIC-III-shielding

Document index (dataset)	False positive ICD-9 code	Explanation with the most relevant sentences and words by attention scores	potential reason
doc-68 (MIMIC-III-50)	427.31 (Atrial fibrillation)	sent-1 (0.32): discharge diagnosis septic shock due to ascending cholangitis choledocholithiasis atrial fibrillation(1.0) with rapid ventricular response pulmonary emboli deep venous thrombosis upper gi bleed peptic ulcer sent-2 (0.31): past medical history recent pe dvt afib(1.0) htn hypotension hypothyroidism cad mild chf sent-3 (0.25): she was found to have bilateral pe s and new afib(1.0) and started on coumadin	missed coding
doc-19 (MIMIC-III-50)	401.9 (Hypertension NOS, or Unspecified essential hypertension)	sent-1 (0.84): decision made to proceed with primary right total knee arthroplasty past medical history htn(1.0) asthma allergies diabetes social history nc family history nc	past disease or missed coding
doc-1 (MIMIC-III-shielding)	416.0 (Primary pulmonary hypertension)	sent-1 (0.45): brief hospital course year old female with h o mild alzheimer s disease cea in htn(0.177) elev(0.145) lipids(0.659) bladder ca who presents as a transfer sent-2 (0.36): past medical history mild alzheimer s disease l cea in htn(0.284) elev(0.167) lipids(0.518) bladder ca no known metastasis	subtle difference in language (regarding the type of hypertension)
doc-65 (MIMIC-III-shielding)	197.0 (Secondary malignant neoplasm of lung)	sent-1 (0.31): brief hospital course yo man with history of metastatic(1.0) pancreatic cancer was admitted with dyspnea new ascites and profound hyponatremia sent-2 (0.3): history of present illness yo cantonese and spanish speaking male with metastatic(1.0) pancreatic cancer was admitted from the ed with dyspnea altered mental status and sent-3 (0.1): metastatic(1.0) pancreatic cancer evidence of progression of ct abdomen pelvis	subtle difference in language (regarding the type of secondary cancer), imbalance of vocabularies or diseases in the training data
doc-95 (MIMIC-III-shielding)	280.00 (Neutropenia, unspecified)	sent-1 (0.24): she has since been found to have a rising ldh and hypercalcemia and decided with her local(0.538) oncologist dr first name8(0.448) namepattern2 name stitle to sent-2 (0.17): at presentation on she developed hypercalcemic with a calcium(1.0) of an elevated ldh	subtle difference between the predicted label 280.00 and the ground truth label 280.03 (Drug induced neutropenia)

nisms in the HLAN model can provide a more comprehensive explanation to support the predictions. For wrong or non-coded predictions, the explanations through highlighted sentences and words can help us better understand the problem. This provides an essential reference to help coding professionals use the system and help engineers fix the problems for the next system iteration.

4.8. Analysis of Label Embedding Initialisation

We previously visualised the label embedding from the MIMIC-III dataset reduced to two dimensions using T-SNE, in Figure 2. The visualisation intuitively shows how the label embedding can capture the correlations among ICD-9 codes derived from the coding practice in the clinical setting.

It is also interesting to know, after the dynamic update during training, how the weights in the initialised layers (the final projection layer and the attention layer) preserve the semantics of the label embedding, and why, in a few cases, LE did not result in a significant improvement. We thus extracted the weights in the learned layers and measured their similarity to the original label embedding. Based on the idea of label similarity, we calculated the top-10 similar labels for every label based on the pairwise cosine similarity of the rows in the initialised layer

weights (e.g. rows such as w_j, w_k in W in Equation 1 or rows V_{wl} in V_w in Equations 4), and also the top-10 similar labels from original label embedding E , and then to see to what extent the two sets of “top-10 similar labels” overlap. We used the Jaccard Index to measure the degree of overlap between the two sets for each label, which is the size of the intersection divided by the size of the union of the two sets. We averaged the Jaccard Index over the labels. Thus the final metric reflects how the layers can retain the semantics, i.e. label similarities, of the label embedding E . We also used the models without the LE initialisation as the control group and calculated this averaged Jaccard Index from their layers for comparison.

The results are displayed in Figure 5. We selected several representative models that either were significantly improved with LE initialisation approach or did not improve with LE according to the results in Tables 2-4. The experiment shows that weights in the final projection layer (and the label-wise attention layer, if applied) with LE initialisation (“+LE”) can capture further label similarities from the label embedding. We also observed a strong correlation between the performance improvement with LE (see Tables 2-4) and the increase of averaged Jaccard Index with LE initialisation (i.e. the extent that the ini-

tialised layers captures the semantics of LE after training, as reflected in Figure 5). The models which are more enhanced by LE (for example, CNN+att, with 6.6% improvement of Macro-AUC with “+LE” in Table 3) have a greater averaged Jaccard Index compared to the models without LE (0.76 vs. 0.42-0.43) in Figure 5. On the contrary, the models which were *not* improved with LE, e.g. HLAN and HAN, for automated coding, also, was also *not* affected by LE in terms of the averaged Jaccard Index. The less effect of LE on HLAN and HAN may be because the hierarchical attention layers (especially with the label-wise attention mechanisms) could already model certain label correlations through the document-level matching process. In overall, the analysis supports the idea that LE initialisation, capturing the label correlations, is a key factor to enhance automated coding with deep learning based multi-label classification. Since LE can be visualised after dimensionality reduction (see Figure 2), this further serves as a mean to help explain the overall model.

5. Discussion

We have presented the results on the three datasets and analysed the interpretability of models and the layers initialised with label embeddings.

The main advantage of HLAN lies in its model explainability, based on the label-wise word-level and sentence-level attention mechanisms. The qualitative comparison of model explanation suggests that the highlighted key words and sentences from HLAN tend to be more comprehensive and more accurate than, those from HAN or HA-GRU and the n -gram explanation from the CNN related models. Such explainable highlights can be particularly helpful when medical coding professionals need to locate the essential part of a long clinical note. When the model suggests a code, its accompanying explanation could be served as a reference for professionals to validate whether the code should be included. This has the potential to build the users’ trust in the deep learning model and help identify missed and erroneous coding.

The label embedding initialisation approach boosted the performance and reduced the variance of most models. The method is efficient, not requiring further model parameters. It is independent of the neural encoders, and can thus be applied to various deep learning models for multi-label classification. Our analysis on the LE initialised layers show that they can preserve the semantics in the pre-trained label embeddings and therefore

can better capture the label similarity from the data. This further contributes to the explainability of the overall approach. There are a few exceptions that LE did not improve the performance, this may be due to the fact that the hierarchical layers can already model certain label correlations when optimising the document-label matching.

In terms of the performance, for MIMIC-III-50, the HLAN model with LE achieved significant better micro-level AUC (91.9%) and F_1 score (64.1%) than the previous state-of-the-art models; for MIMIC-III-shielding, HLAN and HAN performed comparably to CNN (all around 97-98% micro-level AUC); for MIMIC-III, the previous state-of-the-art model CNN+att was significantly boosted by LE initialisation, achieving best AUC and F_1 scores (Micro-level AUC of 98.6% and Micro-level F_1 of 52.5%).

It is worth noting that the higher comprehensiveness in explanation from HLAN is at the cost of further memory requirements and the training time¹³. Thus, in practice, if there are only limited computational resources (e.g. a single GPU with 12GB memory), we suggest training HLAN with a fewer number of codes, e.g. equal or less than 50, in a sub-disease domain or for specific tasks (i.e. shielding-related diseases during COVID-19) that require higher model explainability for decision making. We also notice that the vanilla CNN can be trained relatively faster with significantly less memory requirement; HAN and HA-GRU can also be applied as “downgraded” alternatives of HLAN for tasks with larger label sizes. It is also worth to explore to optimise the implementation and to distil the model of HLAN to enable its application to large label sizes.

While training deep learning models can be slow, during the testing phase, the trained models perform reasonably efficient for real-time inference. On average, it requires less than 1/3 second (330 milliseconds) to assign ICD codes with explainable highlights for a discharge summary with a CPU server using HLAN trained from MIMIC-III-50; and the CNN related models can process even faster (see Table S3 in the Supplementary Material). This allows the efficient use of the models in real-time for automated coding.

Also, the calibration threshold (default as 0.5) could be tuned to adjust the precision and recall of the system when deploying it to a coding department. While high precision is obtained when suggesting a few top-ranked predictions, a system with a higher recall can help the coding professionals to prevent

¹³The estimated training and testing time of the models are in Table S3 in the Supplementary Material.

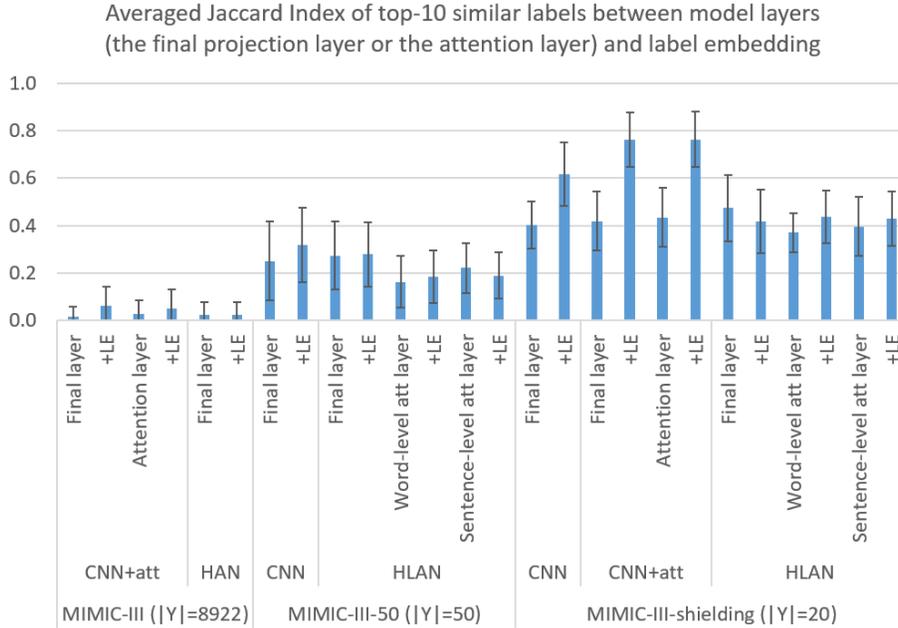


Figure 5: Averaged Jaccard Index between the sets of top-10 similar labels derived from the layers (final projection layer and label-wise attention layers with or without label embedding initialisation) and from the label embedding (LE). The higher the averaged Jaccard Index, the more similar the overall semantics between the layer weights and the pre-trained label embedding. Error bars show the standard deviation over the labels. Representative models are selected for all the three datasets. “+LE” means label embedding initialised for the layer indicated in the closest left bar.

missed coding. A higher recall can be achieved by using a lower calibration threshold, e.g. 0.3-0.4. The results are also highly varied across labels, as seen in the per-label results in Figures 3. A domain for future studies is therefore to investigate few-shot or zero-shot learning for the rare labels and we noticed one recent related work in [37], which is based on ICD-9 hierarchies and descriptions to better predict rare labels.

The results demonstrate the usefulness of label embedding to boost coding performance for most models. In this work, the label embedding was trained with the label sets in the training data using the Continuous Bag of Words algorithm in word2vec. Thus, it encodes the similarity of medical codes derived from the real-world coding practice in the critical care unit of the US hospital. This knowledge is distinct from the ICD-9 hierarchy, as visualised in Figure 2 and there may be contradictions between them. The advantage of the former is that it directly learned the label correlation from the existing hospital and does not require external knowledge. There are recently more studies on leveraging the hierarchy for medical coding as in [38, 39]. One future direction is thus to combine the local knowledge with external knowledge for the task.

The analysis of false positives in the model (see Section 4.6 and Table 6) suggests further research in the area of automated medical coding. The errors are likely due to missed coding,

past medical history rather than present diseases, nuances of language variations, imbalanced vocabularies, and high label granularity. The highlighted sentences and words helped us better determine the cause of the problems. Since missed coding is very common in real-world practice, as also pointed out recently in [40], it is worth to adapt the current algorithms to capture missing labels and emerging new labels. Information on the report template may further help the model select the relevant part of a discharge summary and differentiate a present disease from a past disease. Unable to capture the subtle variations or labels is potentially related to wrong correlations learned from the imbalance of vocabularies and labels in the dataset. This may be addressed by incorporating various external knowledge.

6. Conclusion

In this paper, we examined the existing deep learning based automated coding algorithms and introduced a new architecture, Hierarchical Label-wise Attention Network (HLAN) and a label embedding initialisation approach for automated medical coding. We tested the approaches on the benchmark datasets extracted from the MIMIC-III database, with the simulated task to predict ICD-9 codes related to the high-risk diseases selected

by the NHS for shielding during the COVID-19 pandemic. The experiment results showed that HLAN has a more comprehensive explainability and better or comparative results to the previous state-of-the-art, CNN-based approaches and the downgraded models, HAN and HA-GRU. The proposed label embedding initialisation effectively boosted the performance of the state-of-the-art deep learning models, capturing label correlations from the dataset, which reflects the coding practice.

Analyses on the experiment results of this work suggest that future studies are required in several areas: incorporating external knowledge, learning to capture missed coding, rare labels, and emerging new labels. In particular, automated medical coding work requires to be tested in real-world clinical settings and iteratively improved with inputs from relevant professionals such as coders, nurses and clinicians. Thus an open area to work on in the future is to adapt automated coding models with human corrections in real-time, which is mostly related to human-in-the-loop machine learning and active learning [41]. Inspired by these, we plan to further test and develop the approach to support the coding department in the NHS. We will consult professionals to identify and address the issues involved in deploying the system to facilitate coding staff and improve efficiency, accuracy, and overall satisfaction.

Acknowledgement

The authors would like to thank Dr Johnson Alistair in the MIMIC-III team to confirm to display the sentences of discharge summaries in this paper. The authors would also like to thank comments from Prof Cathierine Sudlow and other members in the Clinical Natural Language Processing Research Group in the University of Edinburgh. HD is supported by Health Data Research UK (HDR UK) National Phenomics Resource Project; Wellcome Institutional Translation Partnership Award (P11032). VSP is supported by HDR UK National Text Analytics Implementation Project; Wellcome Institutional Translation Partnership Award (P11029). HW is supported by HDR UK fellowship MR/S004149/1; Wellcome Institutional Translation Partnership Award (P11054); The Advanced Care Research Centre Programme at the University of Edinburgh; The Health Foundation (I-qual-PPC). This work has also made use of the resources provided by the Edinburgh Compute and Data Facility (ECDF) (<http://www.ecdf.ed.ac.uk/>).

References

- [1] T. Baumeel, J. Nassour-Kassis, R. Cohen, M. Elhadad, N. Elhadad, Multi-label classification of patient notes: case study on ICD code assignment, in: Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence, 2018, pp. 409–416.
- [2] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, R. G. Mark, MIMIC-III, a freely accessible critical care database, *Scientific Data* 3 (1) (2016) 1–9. doi:10.1038/sdata.2016.35.
- [3] D. J. Cartwright, ICD-9-CM to ICD-10-CM codes: What? why? how?, *Advances in Wound Care* 2 (10) (2013) 588–592. doi:10.1089/wound.2013.0478.
- [4] A. Stewart, ICD-11 contains nearly 4x as many codes as ICD-10: Here's what WHO has to say, <https://www.beckersasc.com/asc-coding-billing-and-collections/icd-11-contains-nearly-4x-as-many-codes-as-icd-10-here-s-what-who-has-to-say.html>, accessed 2 April, 2020 (2018).
- [5] M. H. Stanfill, M. Williams, S. H. Fenton, R. A. Jenders, W. R. Hersh, A systematic literature review of automated clinical coding and classification systems, *Journal of the American Medical Informatics Association: JAMIA* 17 (6) (2010) 646–651. doi:10.1136/jamia.2009.001024.
- [6] S. Karimi, X. Dai, H. Hassanzadeh, A. Nguyen, Automatic diagnosis coding of radiology reports: A comparison of deep learning and conventional classification methods, in: *BioNLP 2017, Association for Computational Linguistics, Vancouver, Canada*, 2017, pp. 328–332. doi:10.18653/v1/W17-2342.
- [7] J. Mullenbach, S. Wiegrefe, J. Duke, J. Sun, J. Eisenstein, Explainable prediction of medical codes from clinical text, in: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 1101–1111. doi:10.18653/v1/N18-1100.
- [8] J. R. Geis, A. P. Brady, C. C. Wu, J. Spencer, E. Ranschaert, J. L. Jaremko, S. G. Langer, A. B. Kitts, J. Birch, W. F. Shields, et al., Ethics of artificial intelligence in radiology: summary of the joint european and north american multisociety statement, *Canadian Association of Radiologists Journal* 70 (4) (2019) 329–334.
- [9] B. Goodman, S. Flaxman, European union regulations on algorithmic decision-making and a “right to explanation”, *AI magazine* 38 (3) (2017) 50–57.
- [10] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, Hierarchical attention networks for document classification, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 1480–1489.
- [11] J. Nam, J. Kim, E. Loza Mencía, I. Gurevych, J. Fürnkranz, Large-scale multi-label text classification — revisiting neural networks, in: T. Calders, F. Esposito, E. Hüllermeier, R. Meo (Eds.), *Machine Learning and Knowledge Discovery in Databases*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 437–452.
- [12] J. P. Pestian, C. Brew, P. Matykiewicz, D. J. Hovermale, N. Johnson, K. B. Cohen, W. Duch, A shared task involving multi-label classification of clinical free text, in: *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing, BioNLP '07, Association for Computational Linguistics, USA*, 2007, p. 97–104.
- [13] M.-L. Zhang, Z.-H. Zhou, A review on multi-label learning algorithms, *IEEE Transactions on Knowledge and Data Engineering* 26 (8) (2014) 1819–1837.

- [14] G. Tsoumakas, I. Katakis, I. Vlahavas, Mining multi-label data, in: O. Maimon, L. Rokach (Eds.), *Data Mining and Knowledge Discovery Handbook*, Springer US, Boston, MA, 2010, pp. 667–685. doi: 10.1007/978-0-387-09823-4_34.
- [15] H. Dong, W. Wang, K. Huang, F. Coenen, Automated social text annotation with joint multilabel attention networks, *IEEE Transactions on Neural Networks and Learning Systems* (2020) 1–15.
- [16] M.-L. Zhang, Z.-H. Zhou, Multilabel neural networks with applications to functional genomics and text categorization, *IEEE Transactions on Knowledge and Data Engineering* 18 (10) (2006) 1338–1351. doi: 10.1109/TKDE.2006.162.
- [17] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.
- [18] I. Chalkidis, M. Fergadiotis, S. Kotitsas, P. Malakasiotis, N. Aletras, I. Androutsopoulos, An empirical study on large-scale multi-label text classification including few and zero-shot labels (2020). arXiv:2010.01653.
- [19] Y. Chen, Predicting ICD-9 codes from medical notes – does the magic of BERT applies here?, <https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1204/reports/custom/report25.pdf>, Stanford CS224N Custom Project (Option 3) (2020).
- [20] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015, pp. 1–15.
- [21] E. Gibaja, S. Ventura, A tutorial on multilabel learning, *ACM Computing Survey* 47 (3) (2015) 52:1–52:38.
- [22] G. Kurata, B. Xiang, B. Zhou, Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, San Diego, California, 2016*, pp. 521–526. doi:10.18653/v1/N16-1063.
- [23] S. Baker, A. Korhonen, Initializing neural networks for hierarchical multi-label text classification, in: *BioNLP 2017, Association for Computational Linguistics, Vancouver, Canada, 2017*, pp. 307–315. doi:10.18653/v1/W17-2339.
- [24] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: *Advances in neural information processing systems*, 2013, pp. 3111–3119.
- [25] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder–decoder for statistical machine translation, in: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734.
- [26] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
- [27] L. v. d. Maaten, G. Hinton, Visualizing data using t-SNE, *Journal of machine learning research* 9 (Nov) (2008) 2579–2605.
- [28] L. Song, C. W. Cheong, K. Yin, W. K. Cheung, B. C. Fung, J. Poon, Medical concept embedding with multiple ontological representations., in: *IJCAI*, 2019, pp. 4613–4619.
- [29] Y. Kim, Convolutional neural networks for sentence classification, in: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1746–1751. doi:10.3115/v1/D14-1181.
- [30] S. Gehrmann, F. Dernoncourt, Y. Li, E. T. Carlson, J. T. Wu, J. Welt, J. Foote Jr, E. T. Moseley, D. W. Grant, P. D. Tyler, et al., Comparing deep learning and concept extraction based methods for patient phenotyping from clinical narratives, *PLoS one* 13 (2) (2018) e0192360.
- [31] R. Řehůřek, P. Sojka, Software Framework for Topic Modelling with Large Corpora, in: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, ELRA, Valletta, Malta, 2010*, pp. 45–50, <http://is.muni.cz/publication/884893/en>.
- [32] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feed-forward neural networks, in: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.
- [33] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, Tensorflow: A system for large-scale machine learning, in: *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, OSDI’16, USENIX Association, Berkeley, CA, USA, 2016*, pp. 265–283.
- [34] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32*, Curran Associates, Inc., 2019, pp. 8024–8035.
- [35] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics* 36 (4) (2019) 1234–1240. doi:10.1093/bioinformatics/btz682.
- [36] T. Fawcett, An introduction to roc analysis, *Pattern Recognition Letters* 27 (8) (2006) 861 – 874, rOC Analysis in Pattern Recognition. doi: <https://doi.org/10.1016/j.patrec.2005.10.010>.
- [37] A. Rios, R. Kavuluru, Few-shot and zero-shot multi-label learning for structured label spaces, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018*, pp. 3132–3142. doi: 10.18653/v1/D18-1352.
- [38] M. Falis, M. Pajak, A. Lisowska, P. Schrempf, L. Deckers, S. Mikhael, S. Tsaftaris, A. O’Neil, Ontological attention ensembles for capturing semantic concepts in ICD code prediction from clinical text, in: *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*, Association for Computational Linguistics, Hong Kong, 2019, pp. 168–177. doi:10.18653/v1/D19-6220.
- [39] P. Cao, Y. Chen, K. Liu, J. Zhao, S. Liu, W. Chong, HyperCore: Hyperbolic and co-graph representation for automatic ICD coding, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020*, pp. 3105–3114. doi:10.18653/v1/2020.acl-main.282.
- [40] T. Searle, Z. Ibrahim, R. Dobson, Experimental evaluation and development of a silver-standard for the MIMIC-III clinical coding dataset, in: *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language*

Processing, Association for Computational Linguistics, Online, 2020, pp. 76–85. doi:10.18653/v1/2020.bionlp-1.8.

- [41] R. M. Monarch, Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI, Shelter Island, NY: Manning Publications Company, 2021, version 11, MEAP Edition (Manning Early Access Program).

Table S1: List of ICD-9 codes in **MIMIC-III-50** (50 codes, sorted by frequency in the training data) and per-label prediction results using Hierarchical Label-wise Attention Network with label embedding initialisation (HLAN+LE).

MIMIC-III-50 ICD-9 code	Short Title*	Frequency (train, 8067 documents)	Frequency (test, 1574 documents)	Precision	Recall	F_1
401.9	Hypertension NOS	3233	195	78.5	82.8	80.5
38.93	Venous cath NEC	2139	210	60.6	47.9	53.4
428.0	CHF NOS	2115	165	85.5	78.6	81.9
427.31	Atrial fibrillation	1992	172	91.7	88.6	90.1
414.01	Crnry athrsc1 natve vssl	1921	181	84.4	72.0	77.7
96.04	Insert endotracheal tube	1581	102	59.4	60.2	59.7
96.6	Entral infus nutrit sub	1525	100	67.0	63.4	65.1
584.9	Acute kidney failure NOS	1448	135	71.4	56.4	62.9
250.00	DMII wo cmp nt st uncntr	1416	470	73.2	69.5	71.3
96.71	Cont inv mec ven <96 hrs	1395	95	64.2	49.5	55.7
99.04	Packed cell transfusion	1287	91	19.6	22.2	20.4
272.4	Hyperlipidemia NEC/NOS	1259	422	74.7	81.5	77.9
518.81	Acute respiratry failure	1186	148	65.3	56.1	60.2
39.61	Extracorporeal circulat	1096	203	94.6	96.3	95.5
599.0	Urin tract infection NOS	1067	121	73.8	59.2	65.7
96.72	Cont inv mec ven 96+ hrs	969	91	66.5	61.1	63.4
530.81	Esophageal reflux	953	138	72.2	67.5	68.3
272.0	Pure hypercholesterolem	926	435	58.0	39.6	47.0
285.9	Anemia NOS	852	266	30.3	2.0	3.6
88.56	Coronar arteriogr-2 cath	801	108	87.4	71.3	78.5
38.91	Arterial catheterization	773	215	46.4	2.8	5.2
486	Pneumonia, organism NOS	765	157	63.7	51.0	56.1
244.9	Hypothyroidism NOS	761	548	80.8	84.0	82.3
99.15	Parent infus nutrit sub	736	87	82.2	62.3	70.9
285.1	Ac posthemorrhag anemia	726	340	69.4	44.8	54.3
36.15	1 int mam-cor art bypass	719	233	94.9	92.9	93.8
276.2	Acidosis	694	362	72.6	25.4	37.6
496	Chr airway obstruct NEC	646	155	65.4	49.3	55.9
995.92	Severe sepsis	613	84	70.6	49.2	57.8
V58.61	Long-term use anticoagul	604	51	66.3	66.9	66.1
507.0	Food/vomit pneumonitis	569	155	59.6	54.4	56.7
038.9	Septicemia NOS	567	778	52.5	28.1	36.1
39.95	Hemodialysis	549	196	84.4	80.4	82.3
585.9	Chronic kidney dis NOS	544	127	56.8	42.3	48.4
88.72	Dx ultrasound-heart	530	104	51.8	22.5	31.1
410.71	Subendo infarct, initial	520	187	66.4	41.4	50.3
403.90	Hy kid NOS w cr kid I-IV	513	187	77.5	62.9	69.2
305.1	Tobacco use disorder	504	255	28.3	4.6	7.2
276.1	Hyposmolality	494	402	52.7	26.6	34.9
311	Cutaneous mycobacteria	493	251	40.0	20.9	26.9
37.22	Left heart cardiac cath	482	228	62.4	44.7	51.9
V45.81	Aortocoronary bypass	479	60	76.3	67.3	71.5
412	Pneumococcus infect NOS	477	181	60.8	42.3	48.8
287.5	Thrombocytopenia NOS	471	258	58.3	23.3	33.0
424.0	Mitral valve disorder	451	174	72.9	38.3	49.8
37.23	Rt/left heart card cath	438	226	47.3	19.2	27.0
511.9	Pleural effusion NOS	421	149	46.5	16.8	24.4
45.13	Sm bowel endoscopy NEC	415	160	60.2	70.7	64.6
33.24	Closed bronchial biopsy	407	247	75.8	50.1	60.0
V15.82	History of tobacco use	397	72	0.0	0.0	0.0

* Short titles of the ICD-9 codes are from https://mimic.physionet.org/mimictables/d_icd_diagnoses/ and https://mimic.physionet.org/mimictables/d_icd_procedures/.

Table S2: List of ICD-9 codes in **MIMIC-III-shielding** (20 codes, sorted by frequency in the training data) and per-label prediction results using Hierarchical Label-wise Attention Network with label embedding initialisation (HLAN+LE).

MIMIC-III-shielding ICD-9 code	Short Title	Frequency (train, 4574 documents)	Frequency (test, 322 documents)	Precision	Recall	F_1
197.0	Secondary malig neo lung	656	42	82.1	84.8	83.3
745.5	Secundum atrial sept def	592	41	95.3	83.7	89.0
996.81	Compl kidney transplant	480	14	94.1	97.1	95.4
042	Shigella boydii	470	30	97.7	99.3	98.5
441.2	Thoracic aortic aneurysm	430	36	86.3	73.9	79.5
416.0	Prim pulm hypertension	375	10	54.0	60.0	56.7
746.4	Cong aorta valv insuffic	263	35	94.5	76.3	84.0
288.00	Neutropenia NOS	196	39	51.2	18.5	26.5
238.75	Myelodysplastic synd NOS	170	21	87.4	50.5	63.5
996.82	Compl liver transplant	169	4	49.5	77.5	60.1
238.71	Essntial thrombocythemia	164	28	74.6	46.8	56.4
494.0	Bronchiectas w/o ac exac	152	27	87.6	71.1	76.5
288.0	Neutropenia	136	0	0.0	0.0	0.0
996.85	Compl marrow transplant	133	8	71.9	77.5	74.3
238.7	Neoplasm of uncertain behavior of other lymphatic and hematopoietic tissues	116	0	0.0	0.0	0.0
770.2	NB interstit emphysema	108	0	0.0	0.0	0.0
501	Alastrim	103	5	44.4	56.0	48.8
288.03	Drug induced neutropenia	96	17	47.0	16.5	23.0
289.59	Spleen disease NEC	95	14	71.9	45.0	55.0
446.4	Wegener's granulomatosis	49	1	20.0	20.0	20.0

* Short titles of the ICD-9 codes are from https://mimic.physionet.org/mimictables/d_icd_diagnoses/ and https://mimic.physionet.org/mimictables/d_icd_procedures/.

Figure S1: Distribution of label frequency in the training data for the datasets, MIMIC-III, MIMIC-III-50, and MIMIC-III-shielding.

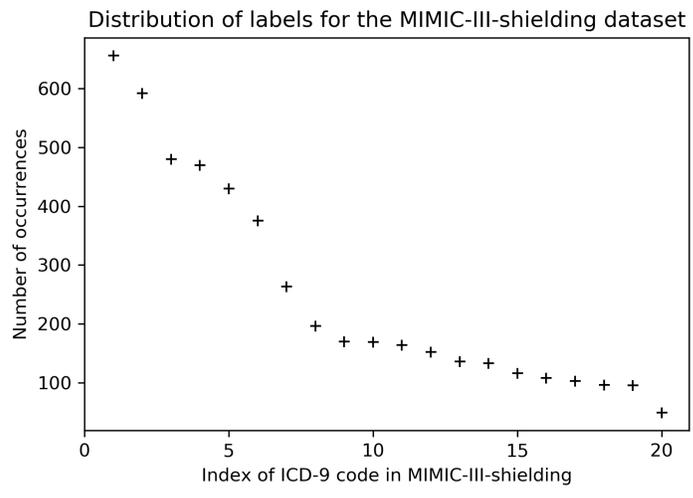
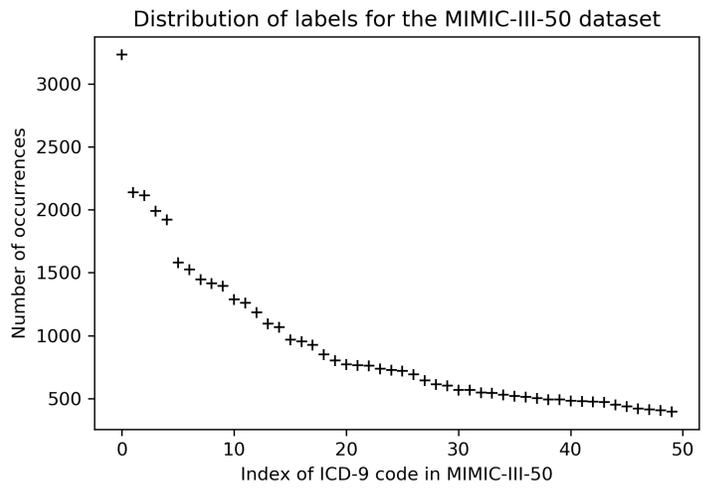
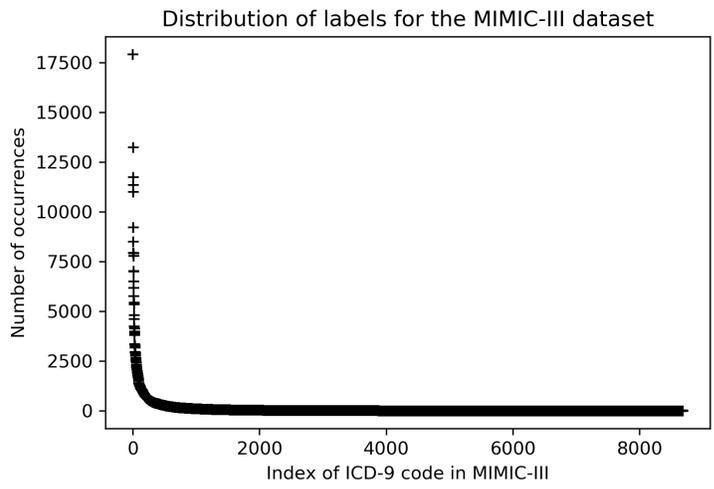


Table S3: Model parameters, training time, and testing time from the datasets.

	CNN*	CNN+att*	Bi-GRU*	HAN	HA-GRU	HLAN
Parameter settings						
Calibration threshold Th	0.5	0.5	0.5	0.5	0.5	0.5
Learning rate	0.003	0.0001	0.003	0.01	0.01	0.01
Batch size (training and testing)	16	16	16	128	32	32
Kernel size (or filter size)	4	10	-	-	-	-
# of words per document	2500	2500	2500	2500	2500	2500
# of words per sentence n_t	-	-	-	25	25	25
# of sentences per document n	-	-	-	100	100	100
# of filters	500	50	512	-	-	-
Hidden size d_h	-	-	-	100	100	100
Attention layer size (e.g. d_w , d_s in HLAN)	500	50	-	200	200	200
Final hidden layer size	500	50	512	400	400	400
Dropout rate	0.2	0.2	0	0.5	0.5	0.5
L_2 penalty	0	0	0	0.0001	0.0001	0.0001
Training time, estimated in minutes**						
From MIMIC-III-50	5	50	40-50	10	30	80
From MIMIC-III-shielding	2.5	8	20-40	10	10-15	25-30
From MIMIC-III	250	1700	100-140	100	-	-
Testing time per document, estimated in milliseconds, GPU time / CPU time***						
From MIMIC-III-50	2	5	50	34 / 40	61 / 160	141 / 330
From MIMIC-III-shielding	3	3	43	32 / 40	17 / 30	14 / 50
From MIMIC-III	2	3	50	42 / 40	-	-

“-” denotes that the parameter is inapplicable to the model or the estimated time was not obtained.

* Parameter settings for CNN, CNN+att, and Bi-GRU are the same as in Mullenbach et al., 2018.

** All models were trained and tested using a single GeForce GTX TITAN X server.

*** For HAN, HA-GRU, and HLAN, testing times on a CPU server (4-core, Intel(R) Xeon(R) Platinum 8259CL CPU @ 2.50GHz) were further reported (displayed after the GPU time).