# Code Synonyms Do Matter: Multiple Synonyms Matching Network for Automatic ICD Coding

**Zheng Yuan**[12*] **Chuanqi Tan**[2] **Songfang Huang**[2]
[1]Tsinghua University [2]Alibaba Group
yuanz17@mails.tsinghua.edu.cn
{chuanqi.tcq,songfang.hsf}@alibaba-inc.com

## Abstract

Automatic ICD coding is defined as assigning disease codes to electronic medical records (EMRs). Existing methods usually apply label attention with code representations to match related text snippets. Unlike these works that model the label with the code hierarchy or description, we argue that the code synonyms can provide more comprehensive knowledge based on the observation that the code expressions in EMRs vary from their descriptions in ICD. By aligning codes to concepts in UMLS, we collect synonyms of every code. Then, we propose a multiple synonyms matching network to leverage synonyms for better code representation learning, and finally help the code classification. Experiments on the MIMIC-III dataset show that our proposed method outperforms previous state-of-the-art methods.

## 1 Introduction

International Classification of Diseases (ICD) is a classification and terminology that provides diagnostic codes with descriptions for diseases[1]. The task of ICD coding refers to assigning ICD codes to electronic medical records (EMRs) which is highly related to clinical tasks or systems including patient similarity learning (Suo et al., 2018), medical billing (Sonabend et al., 2020), and clinical decision support systems (Sutton et al., 2020). Traditionally, healthcare organizations have to employ specialized coders for this task, which is expensive, time-consuming, and error-prone. As a result, many methods have been proposed for automatic ICD coding since the 1990s (de Lima et al., 1998).

Recent methods treat this task as a multi-label classification problem (Xie and Xing, 2018; Li and Yu, 2020; Zhou et al., 2021), which learn deep representations of EMRs with an RNN or CNN encoder and predict codes with a multi-label classifier. Recent state-of-the-art methods propose label attention that uses the code representations as attention queries to extract the code-related representations[2] (Mullenbach et al., 2018). Following this idea, many works further propose using code hierarchical structures (Falis et al., 2019; Xie et al., 2019; Cao et al., 2020) and descriptions (Cao et al., 2020; Song et al., 2020) for better label representations.

In this work, we argue that the synonyms of codes can provide more comprehensive information. For example, the description of code *244.9* is "Unspecified hypothyroidism" in ICD. However, this code can be described in different forms in EMRs such as "low t4" and "subthyroidism". Fortunately, these different expressions can be found in the Unified Medical Language System (Bodenreider, 2004), a repository of biomedical vocabularies that contains various synonyms for all ICD codes. Therefore, we propose to leverage synonyms of codes to help the label representation learning and further benefit its matching to the EMR texts.

To model the synonym and its matching to EMR text, we further propose a **M**ultiple **S**ynonyms **M**atching **N**etwork (**MSMN**)[3]. Specifically, we first apply a shared LSTM to encode EMR texts and each synonym. Then, we propose a novel multi-synonyms attention mechanism inspired by the multi-head attention (Vaswani et al., 2017), which considers synonyms as attention queries to extract different code-related text snippets for code-wise representations. Finally, we propose using a biaffine-based similarity of code-wise text representations and code representations for classification.

We conduct experiments on the MIMIC-III dataset with two settings: full codes and top-50 codes. Results show that our method performs better than previous state-of-the-art methods.

* Work done at Alibaba DAMO Academy.

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

[2] "Label" equals to "code" in some contexts of this paper.

[3] Our codes and model can be found at https://github.com/GanjinZero/ICD-MSMN.



*Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*
*Volume 2: Short Papers*, pages 808 - 814

## 2 Approach

Consider free text $S$ (usually discharge summaries) from EMR with words $\{w_i\}_{i=1}^N$. The task is to assign a binary label $y_l \in \{0, 1\}$ based on $S$. Figure 1 shows an overview of our method.

### 2.1 Code Synonyms

We extend the code description $l^1$ by synonyms from the medical knowledge graph (i.e., UMLS Metathesaurus). We first align the code to the Concept Unique Identifiers (CUIs) from UMLS. Then we select corresponding synonyms of English terms from UMLS with the same CUIs and add additional synonyms by removing hyphens and the word "NOS" (Not Otherwise Specified). We denote the code synonyms as $\{l^2, ..., l^M\}$ in which each code synonym $l^j$ is composed of words $\{l_i^j\}_{i=1}^{N_j}$.

### 2.2 Encoding

Previous works (Ji et al., 2021; Pascual et al., 2021) have shown that pretrained language models like BERT (Devlin et al., 2019) cannot help the ICD coding performance, hence we use an LSTM (Hochreiter and Schmidhuber, 1997) as our encoder. We use pre-trained word embeddings to map words $w_i$ to $\mathbf{x}_i$. A $d$-layer bi-directional LSTM layer takes word embeddings as input to obtain text hidden representations $\mathbf{H} \in \mathbb{R}^h$.

$$\mathbf{H} = \mathbf{h}_1, ..., \mathbf{h}_N = \mathrm{Enc}(\mathbf{x}_1, ..., \mathbf{x}_N) \quad (1)$$

For code synonym $l^j$, we apply the same encoder with a max-pooling layer to obtain representation $\mathbf{q}^j \in \mathbb{R}^h$.

$$\mathbf{q}^j = \mathrm{MaxPool}(\mathrm{Enc}(\mathbf{x}_1^j, ..., \mathbf{x}_{N_j}^j)) \quad (2)$$

### 2.3 Multi-synonyms Attention

To interact text with multiple synonyms, we propose a multi-synonyms attention inspired by the multi-head attention (Vaswani et al., 2017). We split $\mathbf{H} \in \mathbb{R}^{N\times h}$ into $M$ heads $\mathbf{H}^j \in \mathbb{R}^{N\times \frac{h}{M}}$:

$$\mathbf{H} = \mathbf{H}^1, ..., \mathbf{H}^M \quad (3)$$

Then, we use code synonyms $\mathbf{q}^j$ to query $\mathbf{H}^j$. We take the linear transformations of $\mathbf{H}^j$ and $\mathbf{q}^j$ to calculate attention scores $\alpha_l^j \in \mathbb{R}^N$. Text related to code synonym $l^j$ can be represented by $\mathbf{H}\alpha_l^j$. We aggregate code-wise text representations $\mathbf{v}_l \in \mathbb{R}^h$ using max-pooling of $\mathbf{H}\alpha_l^j$ since the text only needs to match one of the synonyms.

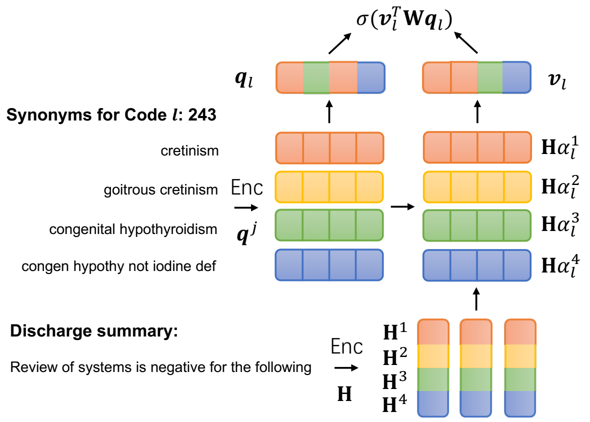


Figure 1: The architecture of our proposed MSMN. Different colors indicate different code synonyms. We also split hidden representations into different heads for multi-synonyms attention.

$$\alpha_l^j = \mathrm{softmax}(\mathbf{W}_Q\mathbf{q}^j \cdot \tanh(\mathbf{W}_H\mathbf{H}^j)) \quad (4)$$

$$\mathbf{v}_l = \mathrm{MaxPool}(\mathbf{H}\alpha_l^1, ..., \mathbf{H}\alpha_l^M) \quad (5)$$

### 2.4 Classification

We classify whether the text $S$ contains code $l$ based on the similarity between code-wise text representation $\mathbf{v}_l$ and code representation. We aggregate code synonym representations $\{\mathbf{q}^j\}$ to code representation $\mathbf{q}_l \in \mathbb{R}^h$ by max-pooling. We then propose using a biaffine transformation to measure the similarity for classification:

$$\mathbf{q}_l = \mathrm{MaxPool}(\mathbf{q}^1, \mathbf{q}^2, ..., \mathbf{q}^M) \quad (6)$$

$$\hat{y}_l = \sigma(\mathrm{logit}_l) = \sigma(\mathbf{v}_l^T\mathbf{W}\mathbf{q}_l) \quad (7)$$

Previous works (Mullenbach et al., 2018; Vu et al., 2020) classify codes via[4]:

$$\hat{y}_l = \sigma(\mathrm{logit}_l) = \sigma(\mathbf{v}_l^T\mathbf{w}_l) \quad (8)$$

Their work need to learn code-dependent parameters $[\mathbf{w}_l]_{l\in\mathcal{C}} \in \mathbb{R}^{\|\mathcal{C}\|\times h}$ for classification, which suffers from training rare codes. On the contrary, our biaffine function that uses $\mathbf{W}\mathbf{q}_l$ instead of $\mathbf{w}_l$ only needs to learn code-independent parameters $\mathbf{W} \in \mathbb{R}^{h\times h}$.

### 2.5 Training

We optimize the model using binary cross-entropy between predicted probabilities $\hat{y}_l$ and labels $y_l$:

$$\mathcal{L} = \sum_{l\in\mathcal{C}} -y_l \log(\hat{y}_l) - (1 - y_l)\log(1 - \hat{y}_l) \quad (9)$$

[4]We omit the biases in all equations for simplification.

| | Train | Dev | Test |
|---|---|---|---|
| MIMIC-III Full | | | |
| # Doc. | 47,723 | 1,631 | 3,372 |
| Avg # words per Doc. | 1,434 | 1,724 | 1,731 |
| Avg # codes per Doc. | 15.7 | 18.0 | 17.4 |
| Total # codes | 8,692 | 3,012 | 4,085 |
| MIMIC-III 50 | | | |
| # Doc. | 8,066 | 1,573 | 1,729 |
| Avg # words per Doc. | 1,478 | 1,739 | 1,763 |
| Avg # codes per Doc. | 5.7 | 5.9 | 6.0 |
| Total # codes | 50 | 50 | 50 |

Table 1: Statistics of MIMIC-III dataset under full codes and top-50 codes settings.

| Parameters | Full | Top 50 |
|---|---|---|
| Emb. dim. | 100 | 100 |
| Emb. dropout | 0.2 | 0.2 |
| LSTM Layer ($d$) | 2 | 1 |
| LSTM hidden dim. | 256 | 512 |
| LSTM output dim. ($h$) | 512 | 512 |
| Synonyms count ($M$) | 4 | 8 |
| Rep. dropout | 0.2 | 0.2 |
| R-Drop weight | 5.0 | 5.0 |
| Epoch | 20 | 20 |
| Peak lr. | 5e-4 | 5e-4 |
| Batch size | 16 | 16 |
| Adam $\epsilon$ | 1e-8 | 1e-8 |
| Weight decay | 0.01 | 0.01 |
| Clipping grad. | 1.0 | 1.0 |

Table 2: Hyper-parameters used for training MIMIC-III full setting and MIMIC-III 50 setting.

## 3 Experiments

### 3.1 Dataset

MIMIC-III dataset (Johnson et al., 2016) contains deidentified discharge summaries with human-labeled ICD-9 codes. We list the document counts, average word counts per document, average codes counts per document, and total codes of the MIMIC-III dataset in Table 1. We use the same splits with previous works (Mullenbach et al., 2018; Vu et al., 2020) with two settings as full codes (MIMIC-III full) and top-50 frequent codes (MIMIC-III 50). We follow the preprocessing of Xie et al. (2019) and Vu et al. (2020) to truncate discharge summaries at 4,000 words. We measure the results using macro AUC, micro AUC, macro $F_1$, micro $F_1$ and precision@k ($k = 5$ for MIMIC-III 50, 8 and 15 for MIMIC-III full).

### 3.2 Implementation Details

We sample $M = 4$ and 8 synonyms per code for MIMIC-III full and MIMIC-III 50 respectively. We sample synonyms fully randomly from the synonyms set. If some ICD codes do not have enough synonyms, we just repeat these synonyms. We use the same word embeddings as Vu et al. (2020) which are pretrained on the MIMIC-III discharge summaries using CBOW (Mikolov et al., 2013) with a hidden size of 100. We apply R-Drop with $\alpha = 5$ (Liang et al., 2021) to regularize the model to prevent over-fitting. We apply the dropout with a ratio of 0.2 after the word embedding layer and before the classification layer. For text encoding, we add a linear layer upon the LSTM layer (the output dimension of the linear layer refers to LSTM output dim. in Table 2). We train MSMN with AdamW (Loshchilov and Hutter, 2019) with a linear learning rate decay. We optimize the threshold of classification using the development set. For the MIMIC-III 50 setting, we train with one 16GB NVIDIA-V100 GPU. For the MIMIC-III full setting, we train with 8 32GB NVIDIA-V100 GPUs. We list the detailed training hyper-parameters in Table 2.

### 3.3 Baselines

**CAML** (Mullenbach et al., 2018) uses CNN to encode texts and proposes label attention for coding.
**MSATT-KG** (Xie et al., 2019) applies multi-scale attention and GCN to capture codes relations.
**MultiResCNN** (Li and Yu, 2020) encodes text using multi-filter residual CNN.
**HyperCore** (Cao et al., 2020) embeds ICD codes into the hyperbolic space to utilize code hierarchy and uses GCN to leverage the code co-occurrence.
**LAAT** & **JointLAAT** (Vu et al., 2020) propose a hierarchical joint learning mechanism to relieve the imbalanced labels, which is our main baseline since it is most similar to our work.

### 3.4 Main Results

Table 3 and 4 show the main results under the MIMIC-III full and MIMIC-III 50 settings, respectively. Under the full setting, our MSMN achieves 95.0 (+2.0), 99.2 (+0.0), 10.3 (-0.4), 58.4 (+0.9), 75.2 (+1.4), and 59.9 (+0.8) in terms of macro-AUC, micro-AUC, macro-$F_1$, micro-$F_1$, P@8, and P@15 respectively (parentheses shows the differences against previous best results), which shows that MSMN obtains state-of-the-art results in most metrics. Under the top-50 codes setting, MSMN performs better than LAAT in all metrics and achieves state-of-the-art scores of 92.8 (+0.3), 94.7 (+0.1), 68.3 (+1.7), 72.5 (+0.9), 68.0 (+0.5)

| | AUC | | $F_1$ | | Precision@N | |
|---|---|---|---|---|---|---|
| | Macro | Micro | Macro | Micro | P@8 | P@15 |
| CAML (Mullenbach et al., 2018) | 89.5 | 98.6 | 8.8 | 53.9 | 70.9 | 56.1 |
| MSATT-KG (Xie et al., 2019) | 91.0 | **99.2** | 9.0 | 55.3 | 72.8 | 58.1 |
| MultiResCNN (Li and Yu, 2020) | 91.0 | 98.6 | 8.5 | 55.2 | 73.4 | 58.4 |
| HyperCore (Cao et al., 2020) | 93.0 | 98.9 | 9.0 | 55.1 | 72.2 | 57.9 |
| LAAT (Vu et al., 2020) | 91.9 | 98.8 | 9.9 | 57.5 | 73.8 | 59.1 |
| JointLAAT (Vu et al., 2020) | 92.1 | 98.8 | **10.7** | 57.5 | 73.5 | 59.0 |
| MSMN | **95.0** | **99.2** | 10.3 | **58.4** | **75.2** | **59.9** |

Table 3: Results on the MIMIC-III full test set.

| | AUC | | $F_1$ | | |
|---|---|---|---|---|---|
| | Macro | Micro | Macro | Micro | P@5 |
| CAML | 87.5 | 90.9 | 53.2 | 61.4 | 60.9 |
| MSATT-KG | 91.4 | 93.6 | 63.8 | 68.4 | 64.4 |
| MultiResCNN | 89.9 | 92.8 | 60.6 | 67.0 | 64.1 |
| HyperCore | 89.5 | 92.9 | 60.9 | 66.3 | 63.2 |
| LAAT | 92.5 | 94.6 | 66.6 | 71.5 | 67.5 |
| JointLAAT | 92.5 | 94.6 | 66.1 | 71.6 | 67.1 |
| MSMN | **92.8** | **94.7** | **68.3** | **72.5** | **68.0** |

Table 4: Results on the MIMIC-III 50 test set.

| | AUC | | $F_1$ | | |
|---|---|---|---|---|---|
| | Macro | Micro | Macro | Micro | P@5 |
| $M = 1$ | 92.1 | 94.2 | 67.4 | 71.0 | 67.0 |
| $M = 2$ | 92.6 | 94.6 | 67.6 | 71.7 | 67.2 |
| $M = 4$ | **92.8** | **94.7** | 67.9 | 71.9 | 67.7 |
| $\underline{M = 8}$ | **92.8** | **94.7** | **68.3** | **72.5** | **68.0** |
| $M = 16$ | 92.5 | 94.6 | 66.9 | 71.5 | 67.6 |
| $\underline{\mathbf{v}_l^T \mathbf{W} \mathbf{q}_l}$ | **92.8** | **94.7** | **68.3** | **72.5** | **68.0** |
| $\mathbf{v}_l^T \mathbf{q}_l$ | 92.5 | 94.5 | 67.1 | 71.2 | 67.1 |
| $\mathbf{v}_l^T \mathbf{w}_l$ | 91.5 | 94.1 | 65.1 | 70.8 | 66.3 |

Table 5: Results of different settings including synonyms counts and scoring functions on MIMIC-III 50 dataset. Underlined setting denotes the default parameters used in MSMN.

on macro-AUC, micro-AUC, macro-$F_1$, micro-$F_1$, and P@5, respectively. We notice that the macro $F_1$ has a large variance in every epoch under the MIMIC-III full setting since it is more sensitive in a long tail problem.

### 3.5 Discussion

To explore the influence of leveraging different numbers of code synonyms, we search $M$ among $\{1, 2, 4, 8, 16\}$ on the MIMIC-III 50 dataset. Results are shown in Table 5. Compared with $M = 1$ that we only use the original ICD code descriptions, leveraging more synonyms from UMLS consistently improves the performance. Using $M = 4, 8$ achieves the best performance in terms of AUC, and $M = 8$ achieves the best performance in terms of $F_1$ and P@5. In addition, the median and mean count of UMLS synonyms are 5.0 and 5.4 respectively, which echoes why the results of $M = 4$ or 8 are better.

To evaluate the effectiveness of our proposed biaffine-based similarity function, we compare it with the baseline LAAT in Table 5. We also provide a simple function by removing $\mathbf{W}$ to $\mathbf{v}_l^T \mathbf{q}_l$ in Equation 7. Results show that the biaffine-based similarity scoring performs best among others.

To better understand what MSMN learns from the multi-synonyms attention, we plot the synonym representations $\mathbf{q}^j$ under MIMIC-III 50 setting via t-SNE (van der Maaten and Hinton, 2008) in Figure 2. We observe for some codes like *585.9* ("chronic kidney diseases"), all synonym representations cluster together, which indicates that synonyms extract similar text snippets. However, codes like *410.71* ("subendocardial infarction initial episode of care" or "subendo infarct, initial") and *403.90* ("hypertensive chronic kidney disease, unspecified, with chronic kidney disease stage i through stage iv" or "unspecified orhy kid w cr kid i iv") with very different synonyms learn different representations, which benefits to match different text snippets. Furthermore, we observe it has similar representations for sibling codes *37.22* ("left heart cardiac catheterization") and *37.23* ("rt/left heart card cath"), which indicates the model can also implicitly capture the code hierarchy.

### 3.6 Memory Complexity

The memory usage of our proposed MSMN is dominated by Equation 4 and Equation 5. We suppose batch size as $B$, word count as $N$, label count as $C$ and synonyms count as $M$. Calculating Equation 4 for all $j$ simultaneously requires calculating Einstein summation (Daniel et al., 2018) among tensors with shape $B \times N \times h$ and shape

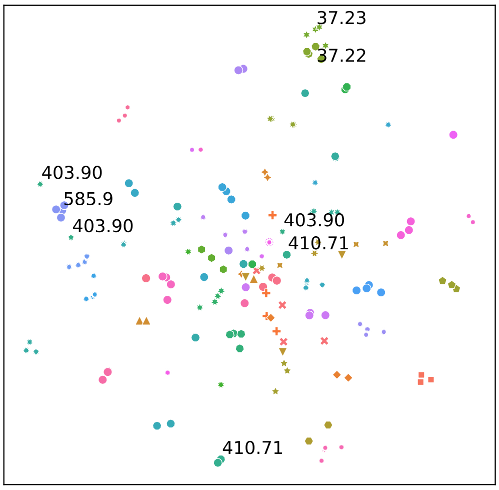


Figure 2: T-SNE visualization of code synonym representations learned from MIMIC-III 50.

$C \times M \times h$ to shape $B \times C \times N \times M$. Calculating Equation 5 requires calculating Einstein summation among tensors with shape $B \times N \times h$ and shape $B \times C \times N \times M$ to shape $B \times C \times h \times M$. The memory complexities of these two equations are linearly proportional to $M$.

## 4 Related Work

Automatic ICD coding is an important task in the medical NLP community. Earlier works use machine learning methods for coding (Larkey and Croft, 1996; Pestian et al., 2007; Perotte et al., 2014). With the development of neural networks, many recent works consider ICD coding as a multi-label text classification task. They usually apply RNN or CNN to encode texts and use the label attention mechanism to extract and match the most relevant parts for classification. The label attention relies on the label representations as attention queries. Li and Yu (2020); Vu et al. (2020) randomly initialize the label representations which ignore the code semantic information. Cao et al. (2020) use the average of word embeddings as label representations to leverage the code semantic information. Xie et al. (2019); Cao et al. (2020) use GCN to fuse hierarchical structures of ICD codes for label representations. Compared with previous works, we use synonyms instead of a single description to represent the code, which can provide more comprehensive expressions of codes.

Biomedical entity linking is a related task to automatic ICD coding. The task requires standardizing given terms to a pre-defined concept dictionary. There are two differences between biomedical entity linking and automatic ICD coding: (1) They have different target concepts. ICD coding map EMRs to ICD codes, while biomedical entity linking usually map terms to a larger dictionary like SNOMED-CT or UMLS. (2) They have different input formats. Entity linking task has labeled entities in texts, while ICD coding only provides texts. Synonyms have also been used in biomedical entity linking (Sung et al., 2020; Yuan et al., 2022). BioSYN (Sung et al., 2020) uses marginalization to sum the probabilities of all synonyms as the similarity between a term and a concept. However, we consider multi-synonyms attention to extracting different parts of clinical texts to interact with synonyms.

## 5 Conclusions

In this paper, we propose MSMN to leverage code synonyms from UMLS to improve the automatic ICD coding. Multi-synonyms attention is proposed for extracting different related text snippets for code-wise text representations. We also propose a biaffine transformation to calculate similarities among texts and codes for classification. Experiments show that MSMN outperforms previous methods with label attention and achieves state-of-the-art results in the MIMIC-III dataset. Ablation studies show the effectiveness of multi-synonyms attention and biaffine-based similarity.

## Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments and suggestions. We thank Fuli Luo, Shengxuan Luo, Hongyi Yuan, Xu Chen, and Jiayu Li for their help. This work was supported by Alibaba Group through Alibaba Research Intern Program.

## References

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Shengping Liu, and Weifeng Chong. 2020. Hypercore: Hyperbolic and co-graph representation for automatic icd coding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3105–3114.

G Daniel, Johnnie Gray, et al. 2018. Opt_einsum-a python package for optimizing contraction order for einsum-like expressions. *Journal of Open Source Software*, 3(26):753.

Luciano RS de Lima, Alberto HF Laender, and Berthier A Ribeiro-Neto. 1998. A hierarchical approach to the automatic categorization of medical documents. In *Proceedings of the seventh international conference on Information and knowledge management*, pages 132–139.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matus Falis, Maciej Pajak, Aneta Lisowska, Patrick Schrempf, Lucas Deckers, Shadia Mikhael, Sotirios Tsaftaris, and Alison O'Neil. 2019. 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)*, pages 168–177.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Shaoxiong Ji, Matti Hölttä, and Pekka Marttinen. 2021. Does the magic of bert apply to medical code assignment? a quantitative study. *Computers in Biology and Medicine*, 139:104998.

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

Leah S Larkey and W Bruce Croft. 1996. Combining classifiers in text categorization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 289–297.

Fei Li and Hong Yu. 2020. Icd coding from clinical text using multi-filter residual convolutional neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8180–8187.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-drop: Regularized dropout for neural networks. In *NeurIPS*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. 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)*, pages 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

Damian Pascual, Sandro Luck, and Roger Wattenhofer. 2021. Towards BERT-based automatic ICD coding: Limitations and opportunities. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pages 54–63, Online. Association for Computational Linguistics.

Adler Perotte, Rimma Pivovarov, Karthik Natarajan, Nicole Weiskopf, Frank Wood, and Noémie Elhadad. 2014. Diagnosis code assignment: models and evaluation metrics. *Journal of the American Medical Informatics Association*, 21(2):231–237.

John P. Pestian, Chris Brew, Pawel Matykiewicz, DJ Hovermale, Neil Johnson, K. Bretonnel Cohen, and Wlodzislaw Duch. 2007. A shared task involving multi-label classification of clinical free text. In *Biological, translational, and clinical language processing*, pages 97–104, Prague, Czech Republic. Association for Computational Linguistics.

Aaron Sonabend, Winston Cai, Yuri Ahuja, Ashwin Ananthakrishnan, Zongqi Xia, Sheng Yu, and Chuan Hong. 2020. Automated icd coding via unsupervised knowledge integration (unite). *International journal of medical informatics*, 139:104135.

Congzheng Song, Shanghang Zhang, Najmeh Sadoughi, Pengtao Xie, and Eric Xing. 2020. Generalized zero-shot text classification for icd coding. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4018–4024. International Joint Conferences on Artificial Intelligence Organization. Main track.

Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. 2020. Biomedical entity representations with synonym marginalization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3641–3650, Online. Association for Computational Linguistics.

Qiuling Suo, Fenglong Ma, Ye Yuan, Mengdi Huai, Weida Zhong, Jing Gao, and Aidong Zhang. 2018. Deep patient similarity learning for personalized healthcare. *IEEE transactions on nanobioscience*, 17(3):219–227.

Reed T Sutton, David Pincock, Daniel C Baumgart, Daniel C Sadowski, Richard N Fedorak, and Karen I Kroeker. 2020. An overview of clinical decision support systems: benefits, risks, and strategies for success. *NPJ digital medicine*, 3(1):1–10.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. 2020. A label attention model for icd coding from clinical text. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3335–3341. Main track.

Pengtao Xie and Eric Xing. 2018. A neural architecture for automated icd coding. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1066–1076.

Xiancheng Xie, Yun Xiong, Philip S Yu, and Yangyong Zhu. 2019. Ehr coding with multi-scale feature attention and structured knowledge graph propagation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 649–658.

Zheng Yuan, Zhengyun Zhao, Haixia Sun, Jiao Li, Fei Wang, and Sheng Yu. 2022. Coder: Knowledge-infused cross-lingual medical term embedding for term normalization. *Journal of Biomedical Informatics*, page 103983.

Tong Zhou, Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Kun Niu, Weifeng Chong, and Shengping Liu. 2021. Automatic icd coding via interactive shared representation networks with self-distillation mechanism. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5948–5957.