Does the Magic of BERT Apply to Medical Code Assignment?
A Quantitative Study

Abstract
Unsupervised pretraining is an integral part of many natural language processing systems, and transfer learning with language models has achieved remarkable results in downstream tasks. In the clinical application of medical code assignment, diagnosis and procedure codes are inferred from lengthy clinical notes such as hospital discharge summaries. However, it is not clear if pretrained models are useful for medical code prediction without further architecture engineering. This paper conducts a comprehensive quantitative analysis of various contextualized language models’ performances, pretrained in different domains, for medical code assignment from clinical notes. We propose a hierarchical fine-tuning architecture to capture interactions between distant words and adopt label-wise attention to exploit label information. Contrary to current trends, we demonstrate that a carefully trained classical CNN outperforms attention-based models on a MIMIC-III subset with frequent codes. Our empirical findings suggest directions for building robust medical code assignment models.

1 Introduction
Clinical notes generated by healthcare professionals are parts of electronic health records and provide an essential source for intelligent healthcare applications (Zhang et al., 2020). Medical information management aims to assign standard medical codes to each clinical document for categorization purposes, which requires professional medical knowledge and is usually costly and error-prone (Hsia et al., 1988; Stanfill et al., 2010). The International Classification of Diseases (ICD) system, as the most used coding system, provides a global standard for reporting diseases and health conditions. The rapid development of machine learning and natural language processing (NLP) can replace manual code assignment with automatic coding systems (Farkas and Szarvas, 2008; Crammer et al., 2007; Farzandipour et al., 2010). Practical medical code assignment requires to capture semantic concepts (Falis et al., 2019) and tackle the challenges of lengthy note encoding and large-dimensional code schemes.

Pretrained language models (PTM) such as BERT (Devlin et al., 2019) learn contextualized text representation and have started a new era in NLP. NLP applications benefit from large-scale pretraining on massive corpora, and universal language representations from PTMs have been successfully utilized in downstream tasks via transfer learning. In the field of clinical NLP, incorporating pretrained contextualized language models to encode lengthy clinical notes for large-scale medical code prediction has not been well-studied. Recently, Li and Yu (2020), Ji et al. (2020b) and Dong et al. (2020) performed preliminary experiments with pretrained models; however, these three pilot studies failed to achieve satisfactory results or provide in-depth analysis.

This paper investigates language models pretrained in various domains. A language domain here defines a distribution over a topical field such
as biomedical documents and clinical notes. Specifically, we investigate the following three research questions (RQ1-3).

RQ1: What kind of BERT pretraining works best? We adopt domain-specific corpora and BERT variants pretrained with different domain adaptation illustrated in Fig. 1, and compare their performance on the MIMIC-III benchmark.

RQ2: What kind of BERT fine-tuning formulation works best for long notes? We employ classical fine-tuning, develop a hierarchical architecture for long clinical notes, and consider label-aware feature representation.

RQ3: Are BERT models better than convolution-based approaches? We reproduce convolutional neural networks (CNNs) with classical pretrained word embeddings and conduct a comparison.

Our contributions are as follows.

- This paper conducts a comprehensive quantitative study to investigate the effect of knowledge transfer via mixed-domain and task-adaptive language model pretraining in different domains, and a thorough comparative study to answer the research questions.
- We propose a hierarchical BERT architecture with a label attention mechanism to enhance contextualized representation with label awareness for long clinical notes.
- We demonstrate that the classical CNN model with appropriate training can improve the predictive performance, achieving new state-of-the-art results on frequent medical codes (MIMIC-III top-50 dataset).

Generalizable Insights about Machine Learning in the Context of Healthcare

Understanding medical text is a long-lasting research problem. We study an essential task in medical information management and diagnosis support - medical code assignment, assigning medical codes to clinical notes. Language models can either be pretrained using a large corpus of medical text or trained from scratch and fine-tuned for a specific task. Several pretrained models have been published for medical NLP, and they differ in the collections of medical texts used for pretraining, e.g., biomedical, clinical, and medical-related social domains. We study the usefulness and relative merits of these different pretrained models and suggest improvements to the neural network architecture to improve performance with long notes: a hierarchical model for longer notes and label-wise attention to leverage relevant information about medical codes. Despite our careful attempts, we nevertheless find that fine-tuning pretrained models performs worse than carefully training conventional neural architectures from scratch. Hence, our results provide practical guidance for building medical information management systems: pretrained models offer a convenient plug-and-play solution; however, training robust existing standard models offers an appealing practical alternative with good performance in practice.

2 Method

Next, we outline our methods. In Section 2.1, we describe transfer learning via pretrained language models with semantic knowledge learned in different domains, specifically BERT and its variants (Fig. 1). In Section 2.2 we describe fine-tuning with different architectures, including a fully-connected classifier and a hierarchical classifier with an extra transformer atop (Fig. 2) to address the long-document challenge. Label-wise attention to learn label-aware document representations with these two fine-tuning architectures is described in Section 2.3. In Section 2.4, we introduce our experimental setup and training strategies.

2.1 Pretraining Domains

We study three types of domains: 1) general domains such as book corpora and general Wikipedia articles; 2) domains that are closely related to the target clinical domain; 3) the target clinical domain. Assigning ICD codes from clinical notes is a task in the clinical domain. We consider biomedical and health-related social domains as candidate domains closely related to the clinical domain. Inspired by domain- and task-adaptive pretraining (Gururangan et al., 2020), we investigate different ways of pretraining models for medical code assignment: 1) pretrain only on general domains and immediately transfer to the target clinical domain; 2) continue pretraining on close domains and clinical domains such as the biomedical domain, and transfer to the target clinical domain 3) pretrain on close domains from scratch and transfer to the target clinical do-
main; 4) pretrain on mixed domains and further fine-tune on the target domain. The details of methods that fall into these classes can be summarized with three categories:

**Pretraining in General Domains** does not involve specific topics or genres. We use BERT (Devlin et al., 2019) pretrained on two unsupervised prediction tasks, i.e., masked language model and next sentence prediction, using the BooksCorpus (Zhu et al., 2015) and English Wikipedia.

**Mixed-domain Pretraining** contains a mixture of domains. We consider continual training and training from scratch. Domain-adaptive pretraining has been validated for the ability to improve the predictive performance (Gururangan et al., 2020) further. We use 1) BlueBERT (Peng et al., 2019) pretrained with PubMed text and MIMIC-III clinical notes; 2) BioBERT (Lee et al., 2020) continually pretrained on domain-specific data from PubMed abstracts and PMC full-text articles; 3) BioReddiBERT (Basaldella et al., 2020) initialized from BioBERT and continually pretrained on health-related posts from health-themed forums in Reddit; 4) PubMedBERT (Gu et al., 2020) domain-specific pretrained from scratch in biomedical domain using MedPub publications; and 5) SapBERT (Liu et al., 2020) that benefits from self-aligning biomedical entities to the Unified Medical Language System (UMLS) ontology and non-parametric metric learning, leading to a better-separated embedding space.

**In-domain Continual Pretraining** continues pretraining in the target clinical domain, which is also called task-specific pretraining (Howard and Ruder, 2018; Gururangan et al., 2020). We use the ClinicalBERT (Alsentzer et al., 2019) with the second phase of pretraining in the clinical domain using clinical notes and discharge summaries.

### 2.2 Fine-tuning with Hierarchical Structure

Clinical notes, consisting of patient history and discharge summaries, are often long documents. To address this, we develop two fine-tuning strategies, one based on the straightforward truncation and the other with hierarchical architecture. In the first approach, we truncate clinical notes to 512 tokens, take the final hidden state of the first token [CLS] as the pooled representation of the truncated note (denoted as \( C \in \mathbb{R}^{d_h} \)), and apply a fully connected network (FCN) as the classifier with sigmoid activation to predict output probabilities. This straightforward fine-tuning structure is denoted as BERT-trun, and it serves as the baseline.

Besides, we propose a hierarchical fine-tuning structure to deal with long notes, shown in Fig. 2. The lengthy clinical notes (with more than 512 tokens) are first divided into several shorter subsequences to build the lower-level contextualized representation. An additional transformer network (Vaswani et al., 2017) is built atop to capture the second-level sequential dependencies between the note segments. The classifier follows the same setup of the truncated version, i.e., FCN with sigmoid. We call this hierarchical structure BERT-hier.

The learning objective function adopts the binary cross entropy loss denoted as:

\[
L = \sum_{i=1}^{m} [-y_i \log(\hat{y}_i) - (1 - y_i) \log(1 - \hat{y}_i)],
\]

where \( y_i \in \{0, 1\} \) is the ground-truth label, \( \hat{y}_i \) is the sigmoid score for prediction, and \( m \) is the number of ICD codes. We use the AdamW optimizer (Loshchilov and Hutter, 2019) to fine-tune the model with backpropagation.

#### 2.3 Label-wise Attention

To further connect the document representation with label information, we introduce label-wise attention (Mullenbach et al., 2018) in the fine-tuning procedures. The label attention network (LAN) prioritizes essential information in the hidden note representation relevant to the medical ICD codes. The LAN calculates the attention score \( A \in \mathbb{R}^{n \times m} \), which measures the importance between each pair of medical codes and words in the document. It is defined as a dot product:

\[
A = \text{Softmax}(HU),
\]

where \( H \in \mathbb{R}^{n \times h} \) is the hidden encoding of the BERT encoder’s last layer, \( U \in \mathbb{R}^{h \times m} \) is the pa-
rameter matrix of the label attention layer (also
known as the query), $n$ is the number of tokens in
the document, $h$ is the hidden representation
dimension, and $m$ is the number of ICD codes.
The attention layer's output is then calculated by
multiplying attention $A$ with the hidden repre-
sentation from residual dilated convolution layers.
The attentive representation $V \in \mathbb{R}^{m \times h}$ formal-
ized as $V = A^T H$ is further used for medical
code classification, representing sequential depen-
dency and label awareness. The BERT-trun and
BERT-hier fine-tuning architectures can both in-
tegrate the label-wise attention mechanism.

2.4 Experimental Setup
This paper focuses on assigning ICD codes to
textual discharge summaries from a hospital stay.
The Medical Information Mart for Intensive Care
III (MIMIC-III) repository (Johnson et al., 2016)
contains patients admitted to the Intensive Care
Unit (ICU) at a US medical center from 2001
to 2012. The MIMIC database is available
by request via this website https://mimic.
physionet.org. We use the “noteevents” table
in the latest release of version 1.4 with a total of
58,576 hospital admissions. Specifically, free-text
discharge summaries are extracted. Two settings
are used for experimental evaluation. One uses
the top 50 frequent labels derived from Shi et al.
(2017), while another uses the full set of ICD codes.
Table 1 shows a statistical summary of two sets of
MIMIC data.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th>Dev.</th>
<th>Test</th>
<th>Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIMIC-III top-50</td>
<td>8,066</td>
<td>1.573</td>
<td>1.729</td>
<td>50</td>
</tr>
<tr>
<td>MIMIC-III full</td>
<td>47,723</td>
<td>1.631</td>
<td>3.372</td>
<td>8,921</td>
</tr>
</tbody>
</table>

Preprocessing. We use raw notes, ICD diag-
noses, and procedures for patients from the public
clinical MIMIC-III dataset for experiments. Dis-
charge summaries labeled with a set of ICD-9 dia-
agnosis and procedure codes include descriptions of
procedures performed by a physician, diagnos-
sis notes, patient’s medical history, and discharge
instructions. Addenda of admissions are concate-
nated to a single document. The NLTK package
is utilized for tokenization, and all tokens are con-
verted into lowercase. Non-alphabetic characters,
such as numbers and punctuations, are removed.

All documents are truncated at 512 tokens and 2500
tokens for a single BERT encoder (BERT-trun)
and hierarchical BERT encoders (BERT-hier).

Training. For the prior publications and the rec-
ommendation of fine-tuning the BERT model, we
choose some common settings. For example, the dropout probability is 0.1. The Adam
optimizer (Kingma and Ba, 2014) is used to opti-
mize CNN-based models, and the AdamW opti-
mizer (Loshchilov and Hutter, 2019) to fine-tune
BERT variants. We utilize a linear learning rate
scheduler with warmup and layer-wise learning
rates when fine-tuning BERT-based models. For re-
training the CNN-based models, we use the CBOW
of word2vec (Mikolov et al., 2013) and adopt
static word embeddings. We set the batch size
for MIMIC-III top-50 and full sets at 8, and the
learning rate from $10^{-6}$ to $10^{-3}$. All the models are run on a Linux cluster with
Nvidia P100 or V100 GPUs. For the MIMIC-III
top-50 data set, the fine-tuning architecture with
truncated notes and a linear classifier has about
109M parameters. In comparison, the MIMIC-III
full set takes about 115M parameters. For the hi-
erarchical architecture, MIMIC-III top-50 and full
codes data sets have 115M and 128M parameters,
respectively. Generally speaking, fine-tuning pre-
trained models consumes the memory of a large
GPU; however, it is arguably less expensive than
training from scratch.

3 Results
We conduct a series of experiments with different
pretrained models on various domains using two
fine-tuning architectures and reproduce classical
CNN-based models with word embeddings from
scratch. We will make our code publicly available.

Different evaluation metrics are utilized for ex-
perimental evaluation, including micro and macro
F1 scores and area under the receiver operating
characteristic curve (AUC-ROC). We evaluate the
metrics of precision at $k$, where $k = 5$ for MIMIC-
III subset with top-50 frequent codes and $k = 8, 15$
for full sets of MIMIC-III, given the observation
that most medical documents are assigned no more
than 20 codes.

3.1 Pretraining in Close Domains Improves
Prediction (RQ1)
This section studies pretraining in different do-
 mains (Sec. 2.1) to evaluate which pretraining
Table 2: Results of PLM fine-tuning with BERT-hier + LAN in various domains on MIMIC-III dataset with top-50 and full ICD codes. Clinical notes are truncated at length of 2500. Hierarchical fine-tuning structure with label-wise attention is used. **Bold** text denotes the best and *italic* text denotes the second best.

<table>
<thead>
<tr>
<th>Model</th>
<th>MIMIC-III Top-50 Codes</th>
<th>MIMIC-III Full Codes</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC-ROC</td>
<td>F1</td>
<td>P@5</td>
<td>AUC-ROC</td>
<td>F1</td>
<td>P@5</td>
<td>P@10</td>
<td>Macro</td>
<td>Micro</td>
<td>Macro</td>
<td>Micro</td>
<td>Macro</td>
<td>Micro</td>
<td>P@5</td>
<td>P@10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BERT-base</td>
<td>82.7</td>
<td>86.3</td>
<td>40.8</td>
<td>50.8</td>
<td>52.2</td>
<td>82.2</td>
<td>96.6</td>
<td>5.8</td>
<td>44.1</td>
<td>63.3</td>
<td>48.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioBERT</td>
<td>89.4</td>
<td>92.0</td>
<td>61.0</td>
<td>65.6</td>
<td>62.8</td>
<td>84.4</td>
<td>97.5</td>
<td>5.1</td>
<td>42.5</td>
<td>62.6</td>
<td>47.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioREdditBERT</td>
<td>88.8</td>
<td>91.7</td>
<td>60.4</td>
<td>66.0</td>
<td>63.1</td>
<td>85.2</td>
<td>97.4</td>
<td>6.4</td>
<td>47.0</td>
<td>65.8</td>
<td>50.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMedBERT full text</td>
<td>87.1</td>
<td>89.6</td>
<td>59.4</td>
<td>64.8</td>
<td>62.4</td>
<td>86.5</td>
<td>98.0</td>
<td>3.0</td>
<td>40.6</td>
<td>62.4</td>
<td>47.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SapBERT full text</td>
<td>88.6</td>
<td>90.8</td>
<td>63.3</td>
<td>68.1</td>
<td>64.4</td>
<td>87.4</td>
<td>98.1</td>
<td>4.3</td>
<td>44.5</td>
<td>65.2</td>
<td>50.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClinicalBERT</td>
<td>89.2</td>
<td>91.6</td>
<td>59.5</td>
<td>64.8</td>
<td>62.0</td>
<td>84.7</td>
<td>97.4</td>
<td>6.0</td>
<td>46.6</td>
<td>65.1</td>
<td>49.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Effect of more pretraining data with BERT-hier + LAN on MIMIC-III dataset with top-50 and full ICD codes. Clinical notes are truncated at length of 2500. **Bold** text denotes better performance.

<table>
<thead>
<tr>
<th>Model</th>
<th>MIMIC-III Top-50 Codes</th>
<th>MIMIC-III Full Codes</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC-ROC</td>
<td>F1</td>
<td>P@5</td>
<td>AUC-ROC</td>
<td>F1</td>
<td>P@5</td>
<td>P@10</td>
<td>Macro</td>
<td>Micro</td>
<td>Macro</td>
<td>Micro</td>
<td>P@5</td>
<td>P@10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMedBERT abstract</td>
<td>90.1</td>
<td>92.5</td>
<td>62.3</td>
<td>67.0</td>
<td>63.8</td>
<td>85.2</td>
<td>97.1</td>
<td>5.8</td>
<td>37.5</td>
<td>66.6</td>
<td>50.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMedBERT full text</td>
<td>88.6</td>
<td>90.8</td>
<td>63.3</td>
<td>68.1</td>
<td>64.0</td>
<td>87.4</td>
<td>98.1</td>
<td>4.3</td>
<td>44.5</td>
<td>65.2</td>
<td>51.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioBERT abstract</td>
<td>89.4</td>
<td>91.6</td>
<td>59.5</td>
<td>64.8</td>
<td>62.7</td>
<td>84.5</td>
<td>97.3</td>
<td>6.3</td>
<td>46.5</td>
<td>65.3</td>
<td>50.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioBERT full text</td>
<td>88.8</td>
<td>91.7</td>
<td>60.4</td>
<td>66.0</td>
<td>63.1</td>
<td>85.2</td>
<td>97.4</td>
<td>6.4</td>
<td>47.0</td>
<td>65.8</td>
<td>50.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClinicalBERT disch. sum.</td>
<td>88.7</td>
<td>91.4</td>
<td>60.6</td>
<td>65.9</td>
<td>62.6</td>
<td>82.9</td>
<td>96.6</td>
<td>6.2</td>
<td>45.1</td>
<td>63.8</td>
<td>48.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClinicalBERT all notes</td>
<td>89.2</td>
<td>91.6</td>
<td>59.5</td>
<td>64.8</td>
<td>62.0</td>
<td>84.7</td>
<td>97.4</td>
<td>6.0</td>
<td>46.6</td>
<td>65.1</td>
<td>49.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3.2 Effect of Pretraining with More Data (RQ1)

Can a larger pool of unsupervised pretraining data lead to performance gain in downstream clinical prediction? To answer this question, we conduct experiments on the effect of more pretraining data using three groups of pretrained model:

- BioBERT continually pretrained in the biomedical domain with abstracts only or with full text biomedical articles.
- PubMedBERT pretrained from scratch in the biomedical domain with abstracts only or with full text biomedical articles.
- ClinicalBERT continually pretrained in the clinical domain with discharge summaries only or with all clinical notes.

In the biomedical domain, there are 200K abstracts from PubMed and 270K full-text articles from PubMedCentral. For the clinical domain, The CATEGORY value of the MIMIC-III dataset includes ‘Discharge Summary’, ‘ECG’, ‘Radiology’, and ‘Echo’. ‘Discharge summary’ indicates that the note is a discharge summary. Besides, there are
also other free-text notes. For example, ‘Report’ indicates a full report, and ‘Addendum’ indicates an additional text added to the previous report.

The results with BERT-tier fine-tuning architecture on two sets of the MIMIC-III dataset are shown in Table 3. We also study the effect using truncated short notes with fully connected tuning architecture, with results reported in Table 4. These two tables suggest that pretraining with more data leads to better performance in most cases.


<table>
<thead>
<tr>
<th>Model</th>
<th>AUC-ROC</th>
<th>F1</th>
<th>P@5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Macro</td>
<td>Micro</td>
<td>Micro</td>
</tr>
<tr>
<td>PubMedBERT abstract</td>
<td>82.4</td>
<td>85.9</td>
<td>54.7</td>
</tr>
<tr>
<td>PubMedBERT full text</td>
<td>82.1</td>
<td>84.4</td>
<td>57.5</td>
</tr>
<tr>
<td>BioBERT abstract</td>
<td>80.9</td>
<td>83.6</td>
<td>54.0</td>
</tr>
<tr>
<td>BioBERT full text</td>
<td>81.8</td>
<td>84.3</td>
<td>55.4</td>
</tr>
<tr>
<td>ClinicalBERT disch. sum.</td>
<td>81.8</td>
<td>84.2</td>
<td>55.8</td>
</tr>
<tr>
<td>ClinicalBERT all notes</td>
<td>82.3</td>
<td>85.3</td>
<td>56.9</td>
</tr>
</tbody>
</table>

3.3 Hierarchical Fine-Tuning Improves Prediction (RQ2)

This section answers to the second research question (RQ2) by comparing the BERT-trun and BERT-tier fine-tuning architectures. Table 5 shows the results on the MIMIC-III top-50 code set using the two fine-tuning architectures when used either with a fully connected classifier or the label attention network. Our results are slightly better than the preliminary results reported by Ji et al. (2020b) with appropriate training tricks. Mixed domain pretraining models, such as PubMedBERT and ClinicalBERT, gain increases in evaluation scores. However, most BERT-trun variants’ predictive performance suffers due to the lack of information when long notes are truncated into short ones. These results show that the proposed hierarchical fine-tuning architecture effectively utilizes long sequences and boosts performance. The label-wise attention mechanism can further improve the prediction in most cases.

3.4 Reproducing CNN Outperforms the SOTA on Frequent Codes (RQ3)

Several recent advances for medical code assignment are CNN-based models. This section investigates whether improved training of CNN can improve prediction. Inspired by performance improvement owing to training tricks, we reproduce the CNN model to check if appropriate training strategies can lead to better scores.

**Baselines.** CNN (Kim, 2014) is built on pretrained word vectors with 1D convolution and max-pooling for text classification. CAML (Mullenbach et al., 2018) integrates CNNs and a label-wise attention mechanism to learn rich representations. It has a variant called DR-CAML that uses ICD code descriptions to regularize the loss function. MultiResCNN (Li and Yu, 2020) combines residual learning (He et al., 2016) and multiple channels concatenation with different convolutional filters, achieving good performance in most settings. HyperCore (Cao et al., 2020) utilizes hyperbolic embedding and co-graph representation with code hierarchy. It gains slightly better performance than the MultiResCNN.

**Reproducing CNNs.** Table 6 shows the reproduced results compared with baselines. The hyperparameters are as follows: the learning rate is 0.003, the filter size is 4, the number of filters is 500, the dropout probability is 0.2, and word2vec embeddings are static. Our retrained vanilla CNN gains a new state of the art (SOTA) compared with two recent advanced MultiResCNN (Li and Yu, 2020) and HyperCore (Cao et al., 2020) on the MIMIC-III top-50 code dataset. Moreover, the P@15 score of the retrained CAML model leads to a significant performance increase in the MIMIC-III full code dataset. These results suggest that a simple model with appropriate training could achieve decent performance in NLP applications such as this clinical application.

How did the CNN- and BERT-based models perform on the full code set? We bin ICD codes into different groups according to the frequency observed in the training set of the MIMIC-III full code dataset to study the effect of code frequency on the model’s predictive performance. We take ClinicalBERT as a clinical note encoding representative and compare three fine-tuning strategies with CNN and CAML model. Fig. 3 shows models’ predictive behavior on frequent and less frequent codes taking F1 scores as the evaluation metrics. These two figures show that all compared models’ predictive performance decreases when the ICD code groups have fewer training samples. When dealing with few-shot codes, model performance...
Table 5: Results on MIMIC-III top-50 code set fine-tuning with BERT-trun and BERT-hier with FCN or LAN.

<table>
<thead>
<tr>
<th>Model</th>
<th>AUC-ROC Macro Micro</th>
<th>F1 Macro Micro</th>
<th>P@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT-base-trun + FCN</td>
<td>80.1 83.0</td>
<td>46.3 51.6</td>
<td>51.1</td>
</tr>
<tr>
<td>BERT-base-hier + FCN</td>
<td>85.4 88.4</td>
<td>53.1 59.4</td>
<td>57.6</td>
</tr>
<tr>
<td>BERT-base-hier + LAN</td>
<td>83.7 86.5</td>
<td>48.1 54.2</td>
<td>53.5</td>
</tr>
<tr>
<td>BlueBERT-trun + FCN</td>
<td>78.2 82.1</td>
<td>23.7 36.6</td>
<td>45.9</td>
</tr>
<tr>
<td>BlueBERT-hier + FCN</td>
<td>89.1 91.8</td>
<td>59.9 65.1</td>
<td>62.5</td>
</tr>
<tr>
<td>BlueBERT-hier + LAN</td>
<td>89.4 92.0</td>
<td>61.0 65.6</td>
<td>62.8</td>
</tr>
<tr>
<td>BioBERT-trun + FCN</td>
<td>81.8 84.3</td>
<td>30.3 35.4</td>
<td>34.3</td>
</tr>
<tr>
<td>BioBERT-hier + FCN</td>
<td>87.3 89.5</td>
<td>59.9 65.6</td>
<td>62.7</td>
</tr>
<tr>
<td>BioBERT-hier + LAN</td>
<td>88.8 91.7</td>
<td>60.4 66.0</td>
<td>63.1</td>
</tr>
<tr>
<td>BioRedditBERT-turn + FCN</td>
<td>80.7 83.5</td>
<td>49.3 55.5</td>
<td>53.8</td>
</tr>
<tr>
<td>BioRedditBERT-hier + FCN</td>
<td>87.0 89.4</td>
<td>59.1 64.2</td>
<td>61.9</td>
</tr>
<tr>
<td>BioRedditBERT-hier + LAN</td>
<td>87.1 89.6</td>
<td>59.4 64.8</td>
<td>62.4</td>
</tr>
<tr>
<td>PubMedBERT-trun + FCN</td>
<td>82.1 84.4</td>
<td>52.6 57.3</td>
<td>55.7</td>
</tr>
<tr>
<td>PubMedBERT-hier + FCN</td>
<td>88.1 91.4</td>
<td>57.7 64.2</td>
<td>62.2</td>
</tr>
<tr>
<td>PubMedBERT-hier + LAN</td>
<td>88.6 90.8</td>
<td>63.3 68.1</td>
<td>64.4</td>
</tr>
<tr>
<td>SapBERT-trun + FCN</td>
<td>82.3 84.8</td>
<td>51.2 56.8</td>
<td>55.2</td>
</tr>
<tr>
<td>SapBERT-hier + FCN</td>
<td>88.5 90.3</td>
<td>63.3 67.3</td>
<td>63.8</td>
</tr>
<tr>
<td>SapBERT-hier + LAN</td>
<td>88.5 90.8</td>
<td>62.2 66.7</td>
<td>63.1</td>
</tr>
<tr>
<td>ClinicalBERT-trun + FCN</td>
<td>82.3 85.3</td>
<td>50.6 56.9</td>
<td>55.7</td>
</tr>
<tr>
<td>ClinicalBERT-hier + FCN</td>
<td>87.2 90.7</td>
<td>54.6 62.5</td>
<td>60.7</td>
</tr>
<tr>
<td>ClinicalBERT-hier + LAN</td>
<td>89.2 91.6</td>
<td>59.5 64.8</td>
<td>62.0</td>
</tr>
</tbody>
</table>

drops sharply. These results again suggest that medical code prediction algorithms should focus on less frequent codes and enhance the robustness of less frequent codes.

3.5 Discussion

Pretrained models in both general and specific domains have shown effectiveness in capturing contextual information. However, they encounter limitations in this study. In the dataset with top-50 frequent codes, fine-tuning with pretrained models can achieve a good performance; however, the PTM fine-tuning does not work well for high-dimensional structured prediction with a full label set that has more than 8,000 labels. This study suggests focusing on less frequent codes.

Self attention-based models suffer from the complexity of $O(n^2d)$, where $n$ is the sequence length, and $d$ is the dimension of hidden representation, making it hard to encode extremely long documents. We investigated how to incorporate the pretrained BERT model and its variants with hierarchical fine-tuning architecture to tackle lengthy clinical document encoding. Nevertheless, CNN-based models (Mullnebach et al., 2018; Li and Yu, 2020; Ji et al., 2020a) and RNN-based models (Shi et al., 2017) perform considerably well with the relatively small model scale and remain a meaningful direction. Recently, some improved transformer-based models such as Longformer (Beltagy et al., 2020), Linformer (Wang et al., 2020) and Big Bird (Zaheer et al., 2020) aim to solve the problem of encoding long document and mitigating the quadratic complexity. We leave these emerging models as future work.

Figure 3: F1 scores of different models on the MIMIC-III full code dataset (8,922 labels). (a) F1 macro; (b) F1 micro. Code frequency groups are sorted in ascending order from left to right.

4 Related Work

Rule-based and machine learning-based methods have been studied for diagnosis code assignment from clinical notes (Medori and Fairon, 2010; Perotte et al., 2014). Perotte et al. (2014) proposed an SVM-based classification algorithm with a flat and hierarchy-based classifier. Recently, the research trend turns to deep neural networks. Convolutional neural networks are one popular category with many model architecture proposed, including CAML that applies CNNs and a label-wise attention mechanism (Mullnebach et al., 2018), MultiResCNN that uses residual connection (Li and Yu, 2020) and DCAN that utilizes dilated convolutions (Ji et al., 2020a). Recurrent neural networks are also extensively studied to capture sequential dependency in clinical notes. Such recurrent models include AttentiveLSTM (Shi et al., 2017), HA-GRU (Baumel et al., 2018) and tree-of-sequences LSTM network (Xie and Xing, 2018). Attention mechanism for matching important diagnosis snippets is widely integrated into CNN- and RNN-based models (Shi et al., 2017; Dong et al., 2020). CAML (Mullnebach et al., 2018) introduced a label-wise attention mechanism to learn
Table 6: Results of retrained CNNs on MIMIC-III with top-50 and full ICD codes. Bold text denotes the new state of the art obtained by an old model.

<table>
<thead>
<tr>
<th>Model</th>
<th>MIMIC-III Top-50 Codes</th>
<th>MIMIC-III Full Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC-ROC</td>
<td>FT</td>
</tr>
<tr>
<td>CNN (Kim, 2014)</td>
<td>87.6</td>
<td>90.7</td>
</tr>
<tr>
<td>CAML (Mullenbach et al., 2018)</td>
<td>87.5</td>
<td>90.9</td>
</tr>
<tr>
<td>MultiResCNN (Li and Yu, 2020)</td>
<td>89.9</td>
<td>92.8</td>
</tr>
<tr>
<td>HyperCore (Cao et al., 2020)</td>
<td>89.5</td>
<td>92.9</td>
</tr>
<tr>
<td>CNN (retrained)</td>
<td><strong>90.8</strong></td>
<td><strong>93.1</strong></td>
</tr>
</tbody>
</table>

label-aware document representations.

Understanding clinical notes require professional medical knowledge. Many methods incorporate external knowledge sources to enhance neural architectures and facilitate clinical text understanding. Prakash et al. (2017) proposed a condensed memory network model with iterative condensation of external memory for network updating and data retrieval from Wikipedia. Bai and Vucetic (2019) used Wikipedia articles of medical codes to learn knowledge-aware embeddings jointly, and Cao et al. (2020) utilized ICD code hierarchy with hyperbolic representation. Another direction to incorporate knowledge is through language pretraining and transfer learning. Li and Yu (2020) and Ji et al. (2020b) reported preliminary results with semantic knowledge transferring. This paper conducts a comprehensive quantitative analysis.

5 Conclusion

This paper presented a comprehensive quantitative analysis of medical code assignment from clinical notes using various pretrained models with BERT. We compared the behavior of several different domain-specific BERT variants. To solve the problem of lengthy clinical note encoding, we developed two fine-tuning architectures: 1) fully connected network with simple truncation into short notes; 2) hierarchical fine-tuning architecture with long note segmentation and an additional Transformer on top. Moreover, we employed label attention to facilitate label-aware representation learning. Through intensive experiments, we found that the magic of BERT does not apply to the task of assigning ICD codes from clinical notes. In contrast, we found that a simple CNN trained from scratch can achieve superior predictive performance on frequent codes, achieving a new state of the art in the MIMIC-III top-50 dataset. This demonstrates how recent training strategies can improve old models. Our results furthermore suggest that medical code assignment algorithms should pay more attention to less frequent codes.

References


ods in Natural Language Processing. Association for Computational Linguistics.


Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don’t stop pretraining: Adapt language models to domains and tasks. In ACL.


Shaoxiong Ji, Erik Cambria, and Pekka Marttinen. 2020a. Dilated Convolutional Attention Network for Medical Code Assignment from Clinical Text. In 3rd Clinical Natural Language Processing Workshop at EMNLP.


