

# MLT-DFKI at CLEF eHealth 2019: Multi-label Classification of ICD-10 Codes with BERT

Saadullah Amin, Günter Neumann, Katherine Dunfield,  
Anna Vechkaeva, Kathryn Annette Chapman, and Morgan Kelly Wixted \*

DFKI GmbH, Campus D3 2, 66123 Saarbrücken, Germany  
{saadullah.amin, guenter.neumann, katherine.dunfield,  
anna.vechkaeva, kathryn.annette.chapman,  
morgan.wixted}@dfki.de  
<https://www.dfki.de>

**Abstract.** With the adoption of electronic health record (EHR) systems, hospitals and clinical institutes have access to large amounts of heterogeneous patient data. Such data consists of structured (insurance details, billing data, lab results etc.) and unstructured (doctor notes, admission and discharge details, medication steps etc.) documents, of which, latter is of great significance to apply natural language processing (NLP) techniques. In parallel, recent advancements in transfer learning for NLP has pushed the state-of-the-art to new limits on many language understanding tasks. Therefore, in this paper, we present team DFKI-MLT's participation at CLEF eHealth 2019 Task 1 of automatically assigning ICD-10 codes to non-technical summaries (NTSs) of animal experiments where we use various architectures in multi-label classification setting and demonstrate the effectiveness of transfer learning with pre-trained language representation model BERT (Bidirectional Encoder Representations from Transformers) and its recent variant BioBERT. We first translate task documents from German to English using automatic translation system and then use BioBERT which achieves an F<sub>1</sub>-micro of 73.02% on submitted run as evaluated by the challenge.

**Keywords:** Semantic Indexing, Transfer Learning, Multi-label Classification, ICD-10 Codes

## 1 Introduction

EHR systems offer rich source of data that can be utilized to improve health care systems by applying information extraction, representation learning and

---

\*On behalf of the PRECISE4Q consortium

Copyright © 2019 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0). CLEF 2019, 9-12 September 2019, Lugano, Switzerland.

predictive modeling [32] techniques. Among many other applications, one such task is the automatic assignment of *International Statistical Classification of Diseases* (ICD) codes [27] to clinical notes, otherwise called semantic indexing of clinical documents [9]. The problem is to learn a mapping from natural language free-texts to medical concepts such that, given a new document, the system can assign one or more codes to it. Approximating the mapping in this setting can be seen as multi-label classification and is one way to solve the problem, besides hierarchical classification [33], learning to rank and unsupervised methods.

In this study, we describe our work on CLEF eHealth 2019 [19] Task 1 [26], which is about multilingual information extraction from German non-technical summaries (NTSs) of animal experiments collected from AnimalTestInfo database to classify according to ICD-10 codes, German modification version 2016<sup>1</sup>. The AnimalTestInfo database was developed in Germany to make the non-technical summaries (NTSs) of animal research studies available in a searchable and easily accessible web-based format. Each NTS was manually assigned an ICD-10 code with the goal of advancing the integrity and reporting of responsible animal research [5]. This task requires an automated approach to classify the NTSs, whereby the data exhibits challenging attributes of multilingualism, domain specificity and codes skewness with hierarchical structure.

We explore various models, starting with traditional bag-of-words support vector machines (SVM) to standard deep learning architectures of convolutional neural networks (CNN) and recurrent neural networks (RNN) with three types of attention mechanisms; namely, hierarchical attention Gated Recurrent Unit (GRU) [8], self-attention Long-Short Term Memory (LSTM) [14], and codes attentive LSTM. Finally, we show the effectiveness of fine-tuning state-of-the-art pre-trained BERT models [10, 22], which requires minimal task specific changes and works well for small datasets. However, the significant performance boost comes from translating the German NTSs to English and then applying the same models, yielding an absolute gain of 6.22% f-score on dev set, from best German model to English model. This can be attributed to the fact that each language has its own linguistic and cultural characteristics that may contain different signals to effectively classify a specific class [1]. Given translated texts, we also find that domain specific embeddings have more effect when considering static word embeddings [25], giving an avg. gain of 2.77% over contextual embeddings [34], where the gain is 0.86%.

## 2 Related Work

Automatic assignment of ICD codes [9] to health related documents has been well studied, both in previous CLEF shared tasks and in general. Traditional approaches range from rule based and dictionary look ups [6] to machine learning models [12]. However, more recently the focus has been on applying deep learning.

---

<sup>1</sup> <https://www.dimdi.de/static/de/klassifikationen/icd/icd-10-gm/kode-suche/htmlgm2016/>

Many techniques have been proposed using CNNs, RNNs and hybrid systems. [11] uses shallow CNN and improves its predictions for rare labels by dictionary-based lexical matching. [4] addresses the challenges of long documents and high cardinality of label space in MIMIC-III [17] by modifying Hierarchical Attention Network [39] with labels attention. More recent focus has been on using sequence-to-sequence (seq2seq) [35] based encoder-decoder based architectures. [31] first builds a multilingual death cause extraction model using LSTMs encoder-decoder, with concatenated French, Hungarian and Italian fast-Text embeddings as inputs and causes extracted from ICD-10 dictionaries as outputs. The output representations are then passed to an attention based biLSTM classifier which predicts the codes. [15] uses character level CNN [41] encoders for French and Italian, which are genealogically related languages and similar on a character level, with a biRNN decoder. [16] enriches word embeddings with language-specific Wikipedia and creates an ensemble model from a CNN classifier and GRU encoder-decoder. Few other techniques have also been proposed to use sequence-to-sequence framework and obtained good results [2, 24].

While successful, these approaches make an auto-regressive assumption on output codes, which may hold true only when there is one distinct path from parent to child code for a given document. However, in the ICD codes assignment, a document can have multiple disjoint paths in a directed acyclic graph (DAG), formed by concepts hierarchy [33]. Also, for a smaller dataset, the decoder may suffer from low variance vocabulary and data sparsity issues. In [7], a novel Hierarchical Multi-label Classification Network (HMCN) with feed-forward and recurrent variations is proposed that jointly optimizes local and global loss functions for discovering local hierarchical class-relationships in addition to global information from the entire class hierarchy while penalizing hierarchical violations (a child node getting a higher score than parent). However, they only consider tree based hierarchies where a node strictly has one parent.

Contextualized word embeddings, such as ELMo [29] and BERT [10], derived from pre-trained bidirectional language models (biLMs) and trained on large texts have shown to substantially improve performance on many NLP tasks; question answering, entailment and sentiment classification, constituency parsing, named entity recognition, and text classification. Such transfer learning involves *fine-tuning* of these pre-trained models on a down-stream supervised task to get good results with minimal effort. In this sense, they are simple, efficient and performant. Motivated by this, and recent work of [22], we use BERT models for this task and achieve better results than CNN and RNN based methods. We also show great improvements with translated English texts.

### 3 Data

The dataset contains 8,385 training documents (including dev set) and 407 test documents, all in German. Each document has six text fields: document title, uses (goals) of the experiment, possible harms caused to animals and comments about *replacement*, *reduction* and *refinement* (in the scope of 3R principles).

| ICD-10 Code | No. of documents<br>(train + dev) |
|-------------|-----------------------------------|
| II          | 1515                              |
| C00-C97     | 1479                              |
| IX          | 930                               |
| VI          | 799                               |
| C00-C75     | 732                               |

**Table 1:** Top-5 most frequent codes

The documents are assigned one or more codes from ICD-10-GM (German Modification version 2016) which exhibits a hierarchy forming a DAG [33], where the highest-level nodes are called *chapters* and their direct child nodes are called *groups*. The depth of most chapters is one but in some cases it goes to second-level (e.g. M00-M25, T20-T32) and, in one case, up to third-level (C00-C97). Documents are assigned mixed codes such that a parent and child node can co-exists and a child node can have multiple parents. Moreover, 91 documents are missing one or more of six text fields and only 6,472 have labels (5,820 in train set and 652 in dev set), while 52 of them have only chapter level codes. Table 1 shows top-5 most frequent codes. These classes account for more than 90% of the dataset leading to a high imbalance. Due to a shallow hierarchy, we consider the problem as multi-label classification instead of hierarchical classification.

## 4 Methods

Since the documents are domain specific and in German, we argue that it might be difficult for open-domain and multilingual pre-trained models to do effective transfer learning. Furthermore, [1] suggests that each language has its own linguistic and cultural characteristics that may contain different signals to effectively classify a specific class. Based on this, and the fact that translations are always available as domain-free parallel corpora, we use them in our system and show improvements across all models. Since English has readily more accessible biomedical literature available as free texts, we use English translations for our documents. To perform a thorough case study, we tested several models and pre-trained embeddings. Below we describe each of them.

**Baseline** For baseline we use a TF-IDF weighted bag-of-words based linear SVM model.

**CNN** Convolutional Neural Network (CNN) learns local features of input representation through varying number and sizes of filters performing convolution operation. They have been very successful in many text classification tasks [18, 41]. While many advanced CNN architectures exist, we use a shallow model of [20].

**Attention Models** Attention is a mechanism that was initially proposed in sequence-to-sequence based Neural Machine Translation (NMT) [3] to allow decoder to attend to encoder states while making predictions. More generally, attention generates a probability distribution over features, allowing models to

The increasing **obesity (obesity)** in the population and the associated health problems in the form of increasing cases of type 2 **diabetes**, hypertension, lipid metabolism disorders and cancers pose a socio-economic challenge and need new therapeutic interventions. A particular problem is the fact that more and more adolescents are morbidly **overweight**, suffer from type 2 **diabetes** and suffer the known long-term consequences such as blindness and amputations. The aim of this animal experiment is to investigate the influence of the secreted protein ectodysplasin A (Eda) and its intronic microRNA miR-676 on glucose metabolism and the development of **diabetes**. Eda and miRNA-676 form a so-called "gene microRNA". Pair that is parallel and proportionally upregulated in the liver of adipose mice, causing inflammatory processes. The results obtained will contribute to a better understanding of the regulation of glucose metabolism and the development of type 2 **diabetes**. Finally, the aim is to achieve new therapeutic approaches to obesity and its consequences through the described experiments.

**Fig. 1:** An example document tagged with codes E10-E14 (*diabetes mellitus*) and E65-E68 (*obesity and other overeating*) containing related words to codes descriptions.

put more weight on relevant features. In our study, we used three attention based models.

**HAN** Hierarchical Attention Network (HAN) deals with the problem of long documents classification by modeling attention at each hierarchical level of document i.e. words and sentences [39]. This allows the model to first attend word encoder outputs, in a sentence, followed by attending the sentence encoder outputs to classify a document. Like [39], we also use bidirectional Gated Recurrent Units (GRUs) as word and sentence encoder.

**SLSTM** Self-Attention Long-Short Term Memory (SLSTM) network is a simple single layer network based on bidirectional LSTMs encoder. An input sequence is first passed through the encoder and encoded representations are self-attended to produce outputs.

**CLSTM** All ICD codes have a textual description, e.g. code A80-A89 is about *viral infections of the central nervous system* that can help a model while classifying. Fig. 1 shows a document containing words related to those found in the descriptions of their labeled codes. Such words may or may not be present but the intuition is to use this additional meta-data to enrich the encoder representation by attention. To the best of our knowledge, this is the first time that the codes' descriptions are directly used to align with input text via attention. The closest work is from [4], where author uses codes attention but they directly consider code as a unit of representation creating an embedding lookup. We also create an embedding layer for codes but using their texts where a code representation is obtained via average of word embeddings of each token. We call this network as Codes Attentive LSTM (CSLSTM) and describe it more formally.

Let  $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^{n \times d}$  be an  $n$ -length input document sequence, where  $x_i$  is a  $d$ -dimensional embedding vector for input word  $w_i$  belonging to documents vocabulary  $\mathbf{V}_D$ . Let  $T = \{t_1, t_2, \dots, t_m\} \in \mathbb{R}^{m \times l}$  be  $m$ -codes by  $l$ -length titles representation matrix, where each  $t_i = \{t_{i_1}, t_{i_2}, \dots, t_{i_l}\} \in \mathbb{R}^{l \times d}$  and  $t_{i_j}$  is  $d$ -dimensional embedding vector for code  $i$ 's title word  $j$ , belonging to titles vocabulary  $\mathbf{V}_T$ . The embedding matrices are different for documents and codes titles, this is because the title words can be missing in documents vocab.

Similarly, we used different LSTM encoders for document and code words (shared encoder under performed on dev set; not reported). The network then transforms input as  $X_{out} = \text{CLSTM}(X, T)$ , with following operations:

$$\begin{aligned}
X_{enc} &= [x_{1_{enc}}, x_{2_{enc}}, \dots, x_{n_{enc}}] \\
x_{i_{enc}} &= \text{LSTM}_W(x_i) \\
T_{enc} &= [t_{1_{enc}}, t_{2_{enc}}, \dots, t_{m_{enc}}] \\
t_{i_{enc}} &= \frac{1}{l} \sum_{j=1}^l \text{LSTM}_C(t_{ij}) \\
X_{out} &= [X_{enc}; T_{enc}] \in \mathbb{R}^{(n+m) \times h} \\
A &= \text{softmax}(X_{out} X_{out}^T) \in \mathbb{R}^{(n+m) \times (n+m)} \\
X_{out} &= X_{out} + A^T X_{out} \\
X_{out} &= \frac{1}{n} \sum_{j=1}^n X_{out_j}
\end{aligned}$$

where,  $X_{enc}$  is a sequence of word encoder  $\text{LSTM}_W$  outputs and  $T_{enc}$  is a sequence of averaged title words encoding by code encoder  $\text{LSTM}_C$ . We concatenate document words sequence with titles sequence and perform self-attention  $A$ , followed by residual connection and average over resulting sequence to get final representation.

**BERT** Pre-training large models on unsupervised corpus with language modeling objective and then fine-tuning the same model for a downstream supervised task eliminates the need of heavily engineered task-specific architectures [10, 29, 30]. **Bidirectional Encoder Representations from Transformers (BERT)** is a recently proposed such model, following ELMo and OpenAI GPT. BERT is a multi-layer bidirectional Transformer (feed-forward multi-headed self-attention) [37] encoder that is trained with two objectives, *masked language modeling* (predicting a missing word in a sentence from the context) and *next sentence prediction* (predicting whether two sentences are consecutive sentences). BERT has improved the state-of-the-art in many language understanding tasks and recent works show that it sequentially model NLP pipeline, POS tagging, parsing, NER, semantic roles and coreference [36]. Similar works [13, 40] have been performed to understand and interpret BERT’s learning capacity. We therefore use BERT in our task and show that it achieves best results compared to other models and is nearly agnostic to domain specific pre-training (BioBERT; [22]).

## 5 Experiments

### 5.1 Pre-processing

We consider each document as one text field i.e. all six fields are joined together to form one input text. As mentioned in section 3, only 6,472 documents are

labeled, out of which 654 are in dev set form total of 840. Since there is no gold standard for these documents we cannot evaluate them, so we ignored them during training. We also abstained from adding an extra "no" class (i.e. proxy for predicting nothing) for such documents because we assume that all NTSs should be indexed (e.g. like MEDLINE auto-indexing of new PubMed articles) and therefore inherently has one or more true ICD-10 codes assigned to them. However, the official evaluation script penalizes model predictions for such documents by considering them all false positives. We will cover this in detail in results section.

To translate German documents to English we used automatic translation from Google Translate API v2<sup>2</sup>. For both, German and English, we use language specific sentence and word tokenizer offered by NLTK [23] and spaCy<sup>3</sup>, respectively. Tokens with document frequencies outside 5 and 60% of training corpus were removed and only top-10000 tokens were kept to limit the vocabulary. This applies to all models other than BERT, which uses WordPiece tokenizer [38] and builds its own vocabulary. Lastly, we remove all the classes with frequency less than 15. All the experiments were performed without any cross-validation on dev set to find best parameters.

## 5.2 Pre-trained Embeddings

We use following pre-trained models for German:

- FT<sub>de</sub>: fastText DE Common Crawl (300d)<sup>4</sup>
- BERT<sub>de</sub>: BERT-Base, Multilingual Cased (768d)<sup>5</sup>

and following for English:

- FT<sub>en</sub>: fastText EN Common Crawl (300d)
- PubMed<sub>en</sub>: PubMed word2vec (400d)<sup>6</sup>
- BERT<sub>en</sub>: BERT-Base, Cased (768d)<sup>7</sup>
- BioBERT<sub>en</sub>: BioBERT (768d)<sup>8</sup>

<sup>2</sup> <https://cloud.google.com/translate/docs/translating-text>

<sup>3</sup> <https://spacy.io/usage/models>

<sup>4</sup> <https://fasttext.cc/docs/en/crawl-vectors.html>

<sup>5</sup> [https://storage.googleapis.com/bert\\_models/2018\\_11\\_23/multi\\_cased\\_L-12\\_H-768\\_A-12.zip](https://storage.googleapis.com/bert_models/2018_11_23/multi_cased_L-12_H-768_A-12.zip)

<sup>6</sup> [https://archive.org/details/pubmed2018\\_w2v\\_400D.tar](https://archive.org/details/pubmed2018_w2v_400D.tar)

<sup>7</sup> [https://storage.googleapis.com/bert\\_models/2018\\_10\\_18/cased\\_L-12\\_H-768\\_A-12.zip](https://storage.googleapis.com/bert_models/2018_10_18/cased_L-12_H-768_A-12.zip)

<sup>8</sup> <https://github.com/naver/biobert-pretrained/releases/tag/v1.0-pubmed-pmc>

### 5.3 Models

**TF-IDF + Linear SVM** For baseline, we use scikit-learn implementation of LinearSVC with one-vs-all training [28].

For all the models, except BERT, we used a batch size of 64, max sequence length of 256, learning rate of 0.001 with Adam [21] and 50 epochs with early stopping. We used binary cross-entropy for each class as our objective function and  $F_1$ -micro score as performance metrics. All the experiments were performed on single 12 GB Nvidia TitanXp GPU. We implemented these models and our code is publicly available.<sup>9</sup>

**CNN** We configured CNN with 64 channels and filter sizes of 3, 4 and 5.

**HAN** Following [39], we also used biGRU encoders but with hidden size of 300. We set the maximum number of sentences in a documents and maximum number of words in a sentence as 40 and 10 respectively.

**SLSTM** A biLSTM encoder with hidden size of 300.

**CLSTM** Similar to SLSTM, but with additional  $T$  matrix of size total number of titles (230, collected from ICD-10-GM)  $\times$  max title sequence length of 10.

**BERT** We used PyTorch’s implementation of BERT<sup>10</sup> with default parameters. To avoid memory issues, we used maximum sequence length of 256 with batch size 6.

**Ensemble** Based on dev set results, we also created an ensemble of top-2 models as weighted combination of their raw scores, where then the prediction for each example is given by:

$$\hat{y} = \mathbb{1}\{\sigma(\kappa \times S_1 + (1 - \kappa) \times S_2) > 0.5\} \in \{0, 1\}^{|C|}$$

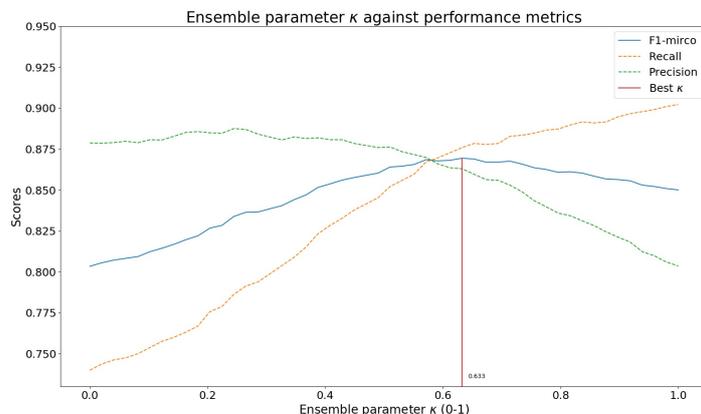
$S_1, S_2$  are raw probability scores from first and second best model respectively, while  $\sigma$  is sigmoid function and  $|C|$  is number of classes. We select best value of  $\kappa$  on dev set such that the f1 score of ensemble is higher than individual models. Fig. 2 shows  $\kappa$  variation with performance metrics.

### 5.4 Results

Table 2 summarizes the results on the dev set for all models with different pre-trained embeddings. In all of our experiments, working with translated texts

<sup>9</sup> <https://github.com/suamin/multilabel-classification-bert-icd10>

<sup>10</sup> <https://github.com/huggingface/pytorch-pretrained-BERT>



**Fig. 2:** The graph shows the effect of varying  $\kappa$  to create an ensemble of top-2 models which achieves a score of 84%, higher than its component models, on dev set at  $\kappa=0.63$ . This value was later used for test predictions.

(English) improved the score by an avg. of 4.07%. This can be attributed to the fact that there is much more English texts than other languages, but it can also be argued that English may have stronger linguistic signals to classify the classes where German models make mistakes [1].

The baseline proved to be a strong one, with the highest precision of all and outperforming HAN and CNN models, for both German and English, with common crawl embeddings. HAN performs better when documents are relatively long e.g. [4] reports strong results with HAN based models on MIMIC dataset [17], where the average document size exceeds 1900 tokens. After pre-processing, the averaged document length in our case was approximately 340. For CNN, we believe advanced variants may perform better.

SLSTM and CLSTM, both being just one layer, performed comparably and better than baseline. SLSTM is much simpler and relies purely on self-attention, which also compliments higher scores by BERT models, which are stacked multi-headed self-attention networks. For CLSTM, since many documents are missing the title words (in fact many title words never appeared in corpus), the model had weak alignment signals between documents and this additional meta-data. However, it still performed really well, getting second best score with PubMed embeddings.

BERT performed better than other models, both in German and English with an avg. score of 6% points higher. BioBERT<sub>en</sub> performed slightly (+0.86%) better than BERT<sub>en</sub>, this was also noticeable in Relation Extraction task in [22], where domain specific and general BERT performed comparably. This partly shows BERT’s ability to generalize and being robust to domain shifts (learning from only 5k training docs), however, this contradicts the recent findings of [40], where authors reflect on such issues, and catastrophic forgetting in BERT-like models. On other hand, the effect of using in-domain pre-trained models was more significant for static-embeddings; using pre-trained PubMeden vectors

| Models          |                                    | P            | R            | F <sub>1</sub> |
|-----------------|------------------------------------|--------------|--------------|----------------|
| Baseline        | TF-IDF <sub>de</sub>               | <b>90.72</b> | 58.73        | 71.30          |
|                 | TF-IDF <sub>en</sub>               | 90.69        | 65.45        | 76.03          |
| CNN             | FT <sub>de</sub>                   | 86.08        | <b>57.37</b> | 68.85          |
|                 | FT <sub>en</sub>                   | 85.76        | 61.59        | 71.69          |
|                 | PubMed <sub>en</sub>               | 87.95        | 65.10        | 74.82          |
| HAN             | FT <sub>de</sub>                   | 78.86        | 58.79        | <b>67.37</b>   |
|                 | FT <sub>en</sub>                   | 83.52        | 64.50        | 72.79          |
|                 | PubMed <sub>en</sub>               | 85.10        | 69.61        | 76.58          |
| SLSTM           | FT <sub>de</sub>                   | 85.55        | 64.86        | 73.76          |
|                 | FT <sub>en</sub>                   | 87.53        | 67.65        | 76.32          |
|                 | PubMed <sub>en</sub>               | 87.33        | 70.09        | 77.77          |
| CLSTM           | FT <sub>de</sub>                   | 83.60        | 63.97        | 72.48          |
|                 | FT <sub>en</sub>                   | 84.39        | 69.14        | 76.01          |
|                 | PubMed <sub>en</sub> <sup>†</sup>  | 87.87        | 70.21        | 78.05          |
| BERT            | Multi <sub>de</sub>                | <b>70.96</b> | 83.41        | 76.68          |
|                 | BERT <sub>en</sub>                 | 79.63        | 84.60        | 82.04          |
|                 | BioBERT <sub>en</sub> <sup>‡</sup> | 80.35        | <b>85.61</b> | <b>82.90</b>   |
| Ensemble (†, ‡) |                                    | 86.29        | 83.11        | <b>84.67</b>   |

**Table 2:** Results on development set (where **blue** and **red** are best and worst score for each column and overall best is **boldfaced**)

out-performed open-domain FT<sub>en</sub> by an avg. of 2.77%. Such analysis was not performed for German due to lack of medical domain German vectors. BERT models had highest recall but relatively poor precision. This is preferable in real-world medical applications, where the recall is of much more importance.

We also combined our top-2 models, BioBERT<sub>en</sub> and CLSTM-PubMed<sub>en</sub>, to get an ensemble which performed better than both and got highest score of 84.67% on dev set. The intuition was to improve on BERT’s precision without losing too much of recall. At  $\kappa = 0.63$  we got the highest score. This increased BioBERT<sub>en</sub> precision by 7.24% at loss of 2.5% recall. Since our focus was mainly on single model systems therefore we used best single model for submission.

## 5.5 Submission and test scores

The test set contains 407 documents, which we first translate to English and then run predictions with BioBERT<sub>en</sub> as our submitted model. We obtained a test f1-micro of 73% with 86% recall and 64% precision as posted by official results. Our system ranked second but there was significant difference between test and dev set performances, especially, low precision. After the gold set was released, we probed it and realized that the official script provided by the challenge considers all predictions on test examples for which there is *no gold label* (93 of them) as *false positives*. We think that it is intrinsically impossible to compare examples

| Models          |                                    | Original |       |                | Modified |       |                |
|-----------------|------------------------------------|----------|-------|----------------|----------|-------|----------------|
|                 |                                    | P        | R     | F <sub>1</sub> | P        | R     | F <sub>1</sub> |
| Baseline        | TF-IDF <sub>de</sub>               | 89.58    | 52.74 | 66.39          | 93.01    | 52.74 | 67.31          |
|                 | TF-IDF <sub>en</sub>               | 88.31    | 60.79 | 72.01          | 91.53    | 60.79 | 73.06          |
| CNN             | FT <sub>de</sub>                   | 80.30    | 54.66 | 65.04          | 86.99    | 54.66 | 67.13          |
|                 | FT <sub>en</sub>                   | 78.09    | 58.74 | 67.05          | 83.33    | 58.74 | 68.91          |
|                 | PubMed <sub>en</sub>               | 80.89    | 64.36 | 71.69          | 86.74    | 64.36 | 73.90          |
| HAN             | FT <sub>de</sub>                   | 71.45    | 54.66 | 61.93          | 80.60    | 54.66 | 65.14          |
|                 | FT <sub>en</sub>                   | 75.88    | 62.70 | 68.67          | 82.10    | 62.70 | 71.10          |
|                 | PubMed <sub>en</sub>               | 79.51    | 66.41 | 72.37          | 84.82    | 66.41 | 74.49          |
| SLSTM           | FT <sub>de</sub>                   | 79.17    | 64.11 | 70.85          | 85.37    | 64.11 | 73.23          |
|                 | FT <sub>en</sub>                   | 82.53    | 65.77 | 73.20          | 86.26    | 65.77 | 74.63          |
|                 | PubMed <sub>en</sub>               | 77.13    | 68.07 | 72.32          | 83.15    | 68.07 | 74.85          |
| CLSTM           | FT <sub>de</sub>                   | 83.60    | 63.97 | 72.48          | 87.52    | 63.97 | 73.91          |
|                 | FT <sub>en</sub>                   | 75.74    | 65.00 | 69.96          | 82.62    | 65.00 | 72.76          |
|                 | PubMed <sub>en</sub> <sup>†</sup>  | 82.15    | 68.19 | 74.52          | 86.82    | 68.19 | 76.39          |
| BERT            | Multi <sub>de</sub>                | 54.10    | 83.39 | 65.62          | 68.23    | 83.39 | 75.05          |
|                 | BERT <sub>en</sub>                 | 62.09    | 83.26 | 71.11          | 75.20    | 83.26 | 79.03          |
|                 | BioBERT <sub>en</sub> <sup>‡</sup> | 63.68    | 85.56 | 73.02          | 76.57    | 85.56 | 80.82          |
| Ensemble (†, ‡) |                                    | 74.44    | 81.86 | <b>77.98</b>   | 83.13    | 81.86 | <b>82.49</b>   |

**Table 3:** Results on test set (where blue and red are best and worst score for each column and overall best is boldfaced). **Original** column refers to official evaluation setup and **Modified** refers to the case where we ignore test documents without gold labels for evaluation.

with predictions where gold standard is not available. To emphasize, we give an example, if we take test document with id=20486 where the gold labels are {C00-C97, C76-C80, II} and our best model predicted {C00-C97, C76-C80, II} i.e. a perfect match with maximum score. Given official evaluation, if this example did not had gold standard available then our model predictions would all had been considered as false positives, which severely degrades precision of a model which may have generalized well to predict on future examples. Table 3 shows this comparison on test set, where in "Original" column we use the same evaluation as provided by the task and in "Modified" we remove all documents from evaluation for which gold labels are not available. As can be seen, recall column is just the same as original with only precision column changes which changes f1-score as well. With the modification, all the models have similar performance as it was on dev set, as we also evaluated trained and evaluated on dev set by removing unlabeled examples. With modification, the submitted system achieves a test score of 80.82% now compared to that of 82.90% on dev set. Finally, ensemble model gets highest scores of 77.98% and 82.49% with original and modified evaluation respectively.

## 6 Discussion

Biomedical text mining is generally a challenging field but recent progresses of transfer learning in NLP can significantly reduce the engineering required to come up with domain sensitive models. Unsupervised data is cheap, and can be obtained in abundance to learn general language patterns [40], however, such data may not be readily available when dealing with in-domain and low-resource languages (e.g. Estonian medical documents). Such deficiencies encourage research for better cross-lingual and cross-domain embedding alignment methods that can be transferred effectively.

## Acknowledgements

We thank Stalin Varanasi for helpful discussions. This work was partially funded by European Union's Horizon 2020 research and innovation programme under grant agreement No. 777107.

## References

1. Reinald Kim Amplayo, Kyungjae Lee, Jinyeong Yeo, and Seung-won Hwang. Translations as additional contexts for sentence classification. *arXiv preprint arXiv:1806.05516*, 2018.
2. A Atutxa, A Casillas, N Ezeiza, I Goenaga, V Fresno, K Gojenola, R Martinez, M Oronoz, and O Perez-de Vinaspre. Ixamed at clef ehealth 2018 task 1: Icd10 coding with a sequence-to-sequence approach. CLEF, 2018.
3. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
4. Tal Baumel, Jumana Nassour-Kassis, Raphael Cohen, Michael Elhadad, and Noémie 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.
5. Bettina Bert, Antje Dörendahl, Nora Leich, Julia Vietze, Matthias Steinfath, Justyna Chmielewska, Andreas Hensel, Barbara Grune, and Gilbert Schönfelder. Rethinking 3r strategies: Digging deeper into animaltestinfo promotes transparency in in vivo biomedical research. *PLoS biology*, 15(12):e2003217, 2017.
6. Rabia Bounaama and M El Amine Abderrahim. Tlemcen university at clef ehealth 2018 team techno: Multilingual information extraction-icd10 coding. CLEF, 2018.
7. Ricardo Cerri, Rodrigo C Barros, and André CPLF De Carvalho. Hierarchical multi-label classification using local neural networks. *Journal of Computer and System Sciences*, 80(1):39–56, 2014.
8. Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

9. Koby Crammer, Mark Dredze, Kuzman Ganchev, Partha Pratim Talukdar, and Steven Carroll. Automatic code assignment to medical text. In *Proceedings of the workshop on bionlp 2007: Biological, translational, and clinical language processing*, pages 129–136. Association for Computational Linguistics, 2007.
10. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
11. Rémi Flicoteaux. Ecstra-aphp@ clef ehealth2018-task 1: Icd10 code extraction from death certificates. CLEF, 2018.
12. Julien Gobeill and Patrick Ruch. Instance-based learning for icd10 categorization. CLEF, 2018.
13. Yoav Goldberg. Assessing bert’s syntactic abilities. *arXiv preprint arXiv:1901.05287*, 2016.
14. Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
15. Julia Ive, Natalia Viani, David Chandran, André Bittar, and Sumithra Velupillai. Kcl-health-nlp@ clef ehealth 2018 task 1: Icd-10 coding of french and italian death certificates with character-level convolutional neural networks. In *19th Working Notes of CLEF Conference and Labs of the Evaluation Forum, CLEF 2018, Avignon, France, 10 September 2018 through 14 September 2018*, volume 2125. CEUR-WS, 2018.
16. Serena Jeblee, Akshay Budhkar, Saša Milic, Jeff Pinto, Chloé Pou-Prom, Krishnapriya Vishnubhotla, Graeme Hirst, and Frank Rudzicz. Toronto cl at clef 2018 ehealth task 1: Multi-lingual icd-10 coding using an ensemble of recurrent and convolutional neural networks.
17. 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. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.
18. Rie Johnson and Tong Zhang. Deep pyramid convolutional neural networks for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 562–570, 2017.
19. Liadh Kelly, Hanna Suominen, Lorraine Goeriot, Mariana Neves, Evangelos Kanoulas, Dan Li, Leif Azzopardi, Rene Spijker, Guido Zuccon, Harrison Scells, and João Palotti. Overview of the CLEF eHealth evaluation lab 2019. In Fabio Crestani, Martin Braschler, Jacques Savoy, Andreas Rauber, et al., editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Tenth International Conference of the CLEF Association (CLEF 2019). Lecture Notes in Computer Science*, Berlin Heidelberg, Germany, 2019. Springer.
20. Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.
21. Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
22. Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: pre-trained biomedical language representation model for biomedical text mining. *arXiv preprint arXiv:1901.08746*, 2019.
23. Edward Loper and Steven Bird. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*, 2002.

24. Zulfat Miftahutdinov and Elena Tutubalina. Kfı at clef ehealth 2017 task 1: Icd-10 coding of english death certificates with recurrent neural networks. In *CLEF (Working Notes)*, 2017.
25. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
26. Mariana Neves, Daniel Butzke, Antje Dörendahl, Nora Leich, Benedikt Hummel, Gilbert Schönfelder, and Barbara Grune. Overview of the CLEF eHealth 2019 Multilingual Information Extraction. In Fabio Crestani, Martin Braschler, Jacques Savoy, Andreas Rauber, et al., editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Tenth International Conference of the CLEF Association (CLEF 2019). Lecture Notes in Computer Science*, Berlin Heidelberg, Germany, 2019. Springer.
27. World Health Organization. *International statistical classification of diseases and related health problems*, volume 1. World Health Organization, 2004.
28. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
29. Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.
30. Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding with unsupervised learning. Technical report, Technical report, OpenAI, 2018.
31. Jurica Ševa, Mario Sängler, and Ulf Leser. Wbi at clef ehealth 2018 task 1: Language-independent icd-10 coding using multi-lingual embeddings and recurrent neural networks. CLEF, 2018.
32. Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. Deep ehr: a survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. *IEEE journal of biomedical and health informatics*, 22(5):1589–1604, 2018.
33. Carlos N Silla and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1-2):31–72, 2011.
34. Noah A Smith. Contextual word representations: A contextual introduction. *arXiv preprint arXiv:1902.06006*, 2019.
35. Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.
36. Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2018.
37. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.
38. Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google’s neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.
39. Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard 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*, pages 1480–1489, 2016.
40. Dani Yogatama, Cyprien de Masson d’Autume, Jerome Connor, Tomas Kocisky, Mike Chrzanowski, Lingpeng Kong, Angeliki Lazaridou, Wang Ling, Lei Yu, Chris Dyer, et al. Learning and evaluating general linguistic intelligence. *arXiv preprint arXiv:1901.11373*, 2019.
  41. Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657, 2015.