Czech Medical Coding Assistant based on Transformer Networks

Ladislav Lenc[a,b], Jiří Martínek[a,b], Josef Baloun[a,b], Pavel Přibáň[a,b], Martin Prantl[a,b], Stephen Eugene Taylor[a,b], Pavel Král[a,b] and Jiří Kyliš[c]

[a] *Dept. of Computer Science & Engineering, Faculty of Applied Sciences, University of West Bohemia, Univerzitni 8, Plzeň, 30100, Czech Republic*

[b] *NTIS - New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Univerzitni 8, Plzeň, 30100, Czech Republic*

[c] *ICZ Group, Na Hřebenech II 1718/10, Praha, 14000, Czech Republic*

ARTICLE INFO

Keywords:
ICD
medical
coding
diagnosis coding
text classification

ABSTRACT

The International Classification of Diseases (ICD) hierarchical taxonomy is used for so-called clinical coding of medical reports, typically presented in unstructured text. In the Czech Republic, it is currently carried out manually by a so-called clinical coder. However, due to the human factor, this process is error-prone and expensive. The coder needs to be properly trained and spends significant effort on each report, leading to occasional mistakes.

The main goal of this paper is to propose and implement a system that serves as an assistant to the coder and automatically predicts diagnosis codes. These predictions are then presented to the coder for approval or correction, aiming to enhance efficiency and accuracy. We consider two classification tasks: main (principal) diagnosis; and all diagnoses. Crucial requirements for the implementation include minimal memory consumption, generality, ease of portability, and sustainability.

The main contribution lies in the proposal and evaluation of ICD classification models for the Czech language with relatively few training parameters, allowing swift utilisation on the prevalent computer systems within Czech hospitals and enabling easy retraining or fine-tuning with newly available data. First, we introduce a small transformer-based model for each task followed by the design of a transformer-based "Four-headed" model incorporating four distinct classification heads.

This model achieves comparable, sometimes even better results, against four individual models. Moreover this novel model significantly economises memory usage and learning time. We also show that our models achieve comparable results against state-of-the-art English models on the Mimic IV dataset even though our models are significantly smaller.

1. Introduction

The objective of automatic medical coding (AMC) is to assign a set of diagnosis and procedure codes to a medical report. In the Czech Republic, as well as in many other countries, the assigned codes are used primarily for billing purposes. It is thus very important to code the reports properly, as missing codes (downcoding) lead to lost revenue for the hospital while false positives (upcoding - codes that should not have been assigned) can be considered as fraud and subject to prosecutions. Achieving precise coding in medical reports necessitates continuous training and periodic retraining of human coders due to the evolving coding rules and the financial implications associated with the codes.Any form of automation in the coding process offers a huge benefit for this costly procedure.

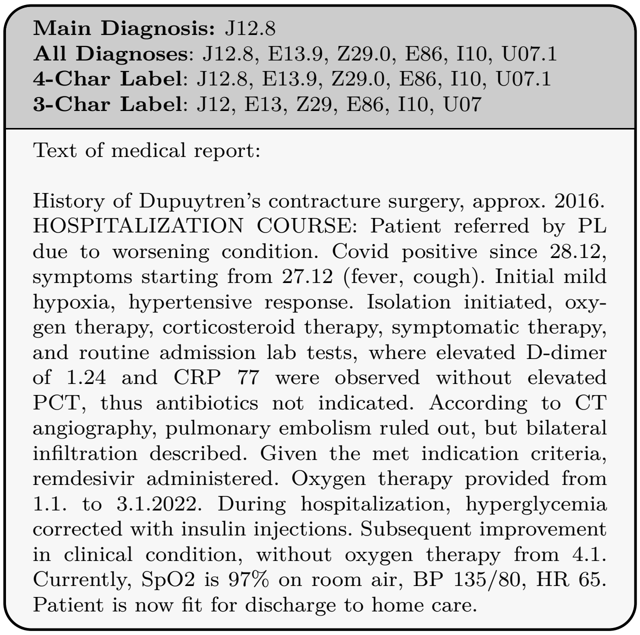
AMC has been studied already from the 1990s, experiencing renewed momentum, particularly with the boom of machine learning and neural networks application in the last several years. A great contribution for this field was the publishing of the MIMIC database (Johnson, Pollard, Shen, Lehman, Feng, Ghassemi, Moody, Szolovits, Celi and Mark, 2016). It first allowed evaluation and comparison of AMC approaches in a comparable environment.

Our task is a multi-label text classification, wherein free-text medical reports serve as input, and the desired output comprises a set of diagnoses and/or procedure codes. This task presents a significant challenge due to the extensive number of potential codes involved, placing it squarely within the domain of extreme multi-label classification (XMC) (Liu, Chang, Wu and Yang, 2017). Moreover, the labels are significantly imbalanced with a lack of annotated data for a high number of codes.

Medical reports typically consist of free-text narratives summarising a patient's hospitalisation history, encompassing details of their health condition, symptoms, descriptions of diagnoses, or the results of various laboratory tests. Each diagnosis and procedure is uniquely identified by an assigned code defined within the International Classification of Diseases (ICD) taxonomy (O'Malley, Cook, Price, Wildes, Hurdle and

[i]Corresponding author
llenc@kiv.zcu.cz ( )
https://nlp.kiv.zcu.cz/ ( )
ORCID(s):
1

Ashton, 2005), which serves as an international standard. Diagnoses specify the specific disease or injury afflicting an individual, while a procedure (codes defined in extended ICD-10-PCS system) describes a series of actions employed for diagnosing, measuring, or treating various health issues. Figure 1 shows an example medical report translated into English with associated diagnosis codes.



Main Diagnosis: J12.8
All Diagnoses: J12.8, E13.9, Z29.0, E86, I10, U07.1
4-Char Label: J12.8, E13.9, Z29.0, E86, I10, U07.1
3-Char Label: J12, E13, Z29, E86, I10, U07

Text of medical report:

History of Dupuytren's contracture surgery, approx. 2016. HOSPITALIZATION COURSE: Patient referred by PL due to worsening condition. Covid positive since 28.12, symptoms starting from 27.12 (fever, cough). Initial mild hypoxia, hypertensive response. Isolation initiated, oxygen therapy, corticosteroid therapy, symptomatic therapy, and routine admission lab tests, where elevated D-dimer of 1.24 and CRP 77 were observed without elevated PCT, thus antibiotics not indicated. According to CT angiography, pulmonary embolism ruled out, but bilateral infiltration described. Given the met indication criteria, remdesivir administered. Oxygen therapy provided from 1.1. to 3.1.2022. During hospitalization, hyperglycemia corrected with insulin injections. Subsequent improvement in clinical condition, without oxygen therapy from 4.1. Currently, SpO2 is 97% on room air, BP 135/80, HR 65. Patient is now fit for discharge to home care.

Figure 1: Example of a medical report (translated into English) and its annotations

In contrast to the multi-label classification utilised in MIMIC, Czech discharge summaries are annotated also with a principal diagnosis, which holds paramount significance within the entire array of other annotated diagnoses. Another difference is that in the Czech Republic procedures have their own code set and the coding is performed separately from the diagnoses, making it a separate task.

Therefore, the main objective of this paper is to propose and implement a medical coding assistant (MCA), an intelligent system designed to assist human clinical coders. Based on the input text in the Czech language, this system will predict diagnosis ICD codes, which will subsequently be presented to the human coders for validation or correction. The overarching goal is to enhance both efficiency (by speeding up the coding process) and improve precision (by reducing the error rate). The paper focuses on two classification tasks: detecting the principal (we will refer to it as the main diagnosis throughout the paper) diagnosis and identifying all diagnoses, each with varying levels of granularity (three or four characters). The system is intended for deployment across numerous hospitals in the Czech Republic and given the diverse landscape of hospitals in the Czech Republic, many of which may possess limited computational resources, key prerequisites for the system include minimal memory usage, versatility, portability, and sustainability. Moreover, the minimal computational requirements align with the principles of Green AI (Schwartz, Dodge, Smith and Etzioni, 2020).

The main contribution of this paper consists in proposing and evaluating several ICD classification models for Czech language reports with relatively low number of training parameters, allowing for swift utilisation on the prevalent computer systems within Czech hospitals and enabling easy retraining or fine-tuning with newly available data. First, we introduce a small transformer-based model for each task (totalling four models,) followed by the design of a versatile "Four-headed" model based on the transformer architecture (Han, Xiao, Wu, Guo, Xu and Wang, 2021), incorporating four distinct classification heads.

A crucial facet of these models is thus their ability to maintain a relatively modest number of parameters, rendering them well-suited for rapid deployment on the prevailing computer systems within Czech hospitals. Additionally, these models should offer the flexibility of straightforward retraining or fine-tuning with newly available data, ensuring their adaptability and longevity in evolving healthcare contexts.

The paper is organised as follows: In the next section, we delve into related work in the field, emphasising available datasets and ICD coding methods rooted in neural networks. Section 3 describes our proposed models, while Section 4 presents our experimental results and findings. This section also contains the description of the data and experimental setup. The final section concludes the work and proposes some future directions.

2. Related Work

Medical coding is a text classification in a specific domain, because medical text significantly differs from general text. We can also consider this task as a special case of extreme multi-label classification (XMC) and therefore we will first list several related methods for general XMC. Next, we describe MIMIC database which is mostly used for evaluation of ICD coding methods. Finally, we will summarise important methods designed specifically for the medical coding.

2.1. Text Classification

Liu et al. (2017) presented a deep learning based approach for XMC. The authors proposed an XML-CNN architecture which is a CNN similar to the one proposed by Kim (2014) and extends it with dynamic max-pooling layer. They tested the method on several benchmarks and achieved state-of-the-art results in all cases.

Bonsai (Khandagale, Xiao and Babbar, 2020) is a set of algorithms for XMC. It utilises generalised label representation and learns shallow trees. This method achieves better performance especially on rare labels from the label distribution tail.

Bhatia, Jain, Kar, Varma and Jain (2015) proposed sparse local embeddings for the XMC task. Their SLEEC method is based on label embeddings and a set of learned regressors.

2.2. MIMIC Dataset

MIMIC (The Multiparameter Intelligent Monitoring in Intensive Care) database (Goldberger, Amaral, Glass, Hausdorff, Ivanov, Mark, Mietus, Moody, Peng and Stanley, 2000) was originally published in 2000. It contains recordings of signals and periodic measurements obtained from a bedside monitor as well as clinical data obtained from the patient's medical record. The data were collected from 90 patients and in total, there are about 200 patient-days of recorded signals.

Next version, MIMIC-II, was officially released in 2006 (Moody, Mark and Goldberger, 2011). MIMIC-II has been fully de-identified in a Health Insurance Portability and Accountability Act (HIPAA) compliant manner and is available free of charge for public use after completion of an appropriate online human-subjects training course and signature of a data use agreement.

MIMIC-III (Johnson et al., 2016) was released in 2015. It has become a standard benchmark for evaluation of machine learning approaches for AMC. It contains anonymised data from years 2011 to 2012 collected in critical care units of the Beth Israel Deaconess Medical Center. It includes several types of records such as the bedside measurements and also laboratory tests, procedures, medications and discharge summaries. MIMIC-III is distributed as a relational database containing 26 tables. The data are stored as comma separated files. Important information for the AMC task are stored in the ICD_DIAGNOSES and ICD_PROCEDURES tables containing annotations for diagnoses (ICD-9 standard) and procedures respectively. The textual data are then in the NOTEEVENTS table which needs to be joined with the annotations using admission ID.

In most available works, diagnoses are classified together with procedures. The dataset contains 8,921 unique ICD-9 codes (6,918 diagnoses codes and 2,003 procedure codes). A widely adopted setup for testing of machine learning approaches on this dataset was proposed by Mullenbach, Wiegreffe, Duke, Sun and Eisenstein (2018). They used splits with 47,724 records for training, 1,632 for evaluation and 3,372 for testing. In some cases, there are multiple admissions for one patient. The authors solved this issue by placing all admission data from one patient into one set. They also prepared splits with data containing only 50 most frequent labels. The splits are denoted as *MIMIC-III full* and *MIMIC-III 50* respectively.

In 2021, MIMIC-IV release opened new possibilities for AMC testing. It is composed of admissions between years 2008 and 2019. The database contains diagnoses and procedures coded in both ICD-9 and the new ICD-10 standards. However, the texts for clinical notes were published in a separate data source as MIMIC-IV-Note only in 2023. The first published work utilizing MIMIC-IV for AMC was authored by Edin, Junge, Havtorn, Borgholt, Maistro, Ruotsalo and Maaløe (2023). The authors have prepared splits for the ICD-9 (*MIMIC-IV ICD-9*) and ICD-10 (*MIMIC-IV ICD-10*) parts of the dataset. They also analysed and revised the widely used splits for MIMIC-III and published their own split denoted as *MIMIC-III clean*.

2.3. ICD Coding

Crammer, Dredze, Ganchev, Talukdar and Carroll (2007) presented a system for radiology report ICD-9 classification. Their approach combines three classification methods and uses another classifier to combine their results. They used a dataset provided by the organisers of the International Challenge on Classifying Clinical Free Text Using Natural Language Processing (Pestian, Brew, Matykiewicz, Hovermale, Johnson, Cohen and Duch, 2007). The reported F1 achieved on the test set is 87.6%.

Farkas and Szarvas (2008) proposed a rule-based ICD-9 coding system for radiology reports on the same dataset as the paper above. As well as most other participants of the challenge, they utilised hand-crafted rules. They tried to model inter-label dependencies, collect synonyms missing in coding manuals and achieved 88.9% F1-score on the test set. This study showed that rule-based systems can be applied on tasks with limited label space (45 labels for radiology reports), however, their utilization in general medical coding with thousands of labels is limited.

Karimi, Dai, Hassanzadeh and Nguyen (2017) published a comparison of traditional and deep learning methods for the task of medical coding. They used a CNN based on Kim (2014) on the radiology reports as papers above. They used 10-fold cross-validation so the experimental setup was different than other presented methods. The authors showed that CNN outperformed SVM, logistic regression and random forests.

Xun, Jha, Sun and Zhang (2020) introduced CorNet, a Correlation network usable for various XMC tasks. The CorNet module (located at the prediction layer of a model) should learn label correlations that might be helpful due to the injection of correlation knowledge.

Kim and Ganapathi (2021) proposed Read, Attend and Code (RAC) model. RAC model solves the task as a set-to-set assignment learning problem. They used

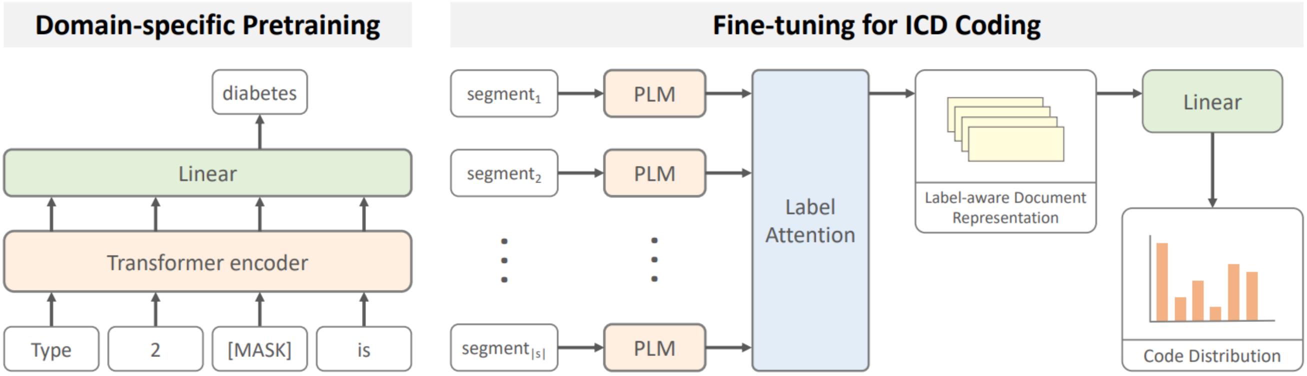


Figure 2: PLM-ICD framework Huang et al. (2022)

sentence permutation based data augmentation. They made an experiment with human annotators. The agreement was ”not so high as believed” but no exact numbers are provided. It is only reported that the system exceeds the human performance.

Most authors ignore the structure of medical messages. Zhang, Zhang, Zhang, Sang and Yang (2022) proposed Discnet using discourse structure and description of ICD codes itself. This helps with classification of rare codes. The network structure is based on Word2Vec embeddings and bidirectional GRU architecture. The presented solution achieves the best results on MIMIC-III dataset to the date of publication.

Edin et al. (2023) first performed experiments on MIMIC-IV. They analysed and criticised the widely used split created by Mullenbach et al. (2018) for MIMIC-III. The main concern was the absence of many diagnosis codes (54%) in the test set. They prepared new *MIMIC-III clean* version which was created using stratified sampling Sechidis, Tsoumakas and Vlahavas (2011) that ensures better distribution of labels across the sets. The authors also provided a thorough comparison of several state-of-the-art methods on both the original as well as the newly created split. In the same manner, they also prepared splits for MIMIC-IV (ICD-9 and ICD-10) and tested the methods on them.

Chen, Tian, Cai and Lu (2022) proposed a semi-automatic tool for medical coding and clinical data standardization. They used a dilated convolutional attention network with an N-gram Matching mechanism.

Huang et al. (2022) analyse the limited performances of pre-trained language models (PLM) while tackling large-label space classification task, such as ICD coding. As a result, they proposed a framework that consists of two phases: 1) domain-specific pretraining and 2) fine-tuning for ICD coding. Since most PLMs have input length limitation, they split the input documents to several segments, run classification for each of them and merge results.

A different approach for training neural network model was chosen by Niu, Wu, Li and Li (2023). Instead of a classic approach with cross-entropy loss, they propose to use contrastive learning with transformer-based backbone. The tests were conducted on MIMIC-III dataset.

Most experiments are carried out on English documents. The experiments with different languages rely on local datasets, that are usually not publicly available or the data have a different structure based on local customs.

Azam, Raju, Pagidimarri and Kasivajjala (2020) performed ICD 10 coding in Hindi. They used cascading hierarchical architecture with LSTM. The full ICD-10 codes with up to 7 characters are divided into 6 levels. A single classifier is then applied for each level. Each classifier takes as input the text as well as outputs of previous levels classifiers Data were obtained from private hospitals in India. Their dataset contains 135,000 records and a total number of 14,692 distinct ICD-10 codes. The reported accuracy on level 3 which corresponds to 4 character diagnosis codes is 78.3. However, it is not clear from the experiments how the multi-label nature of the AMC task is handled.

To the best of our knowledge, ICD coding in the Czech language was carried out only by Přibáň, Baloun, Martínek, Lenc, Prantl and Král (2023). The authors have realised a comparative study of several promising approaches based on deep neural nets. They experimentally showed that hierarchical GRU with attention outperformed all other models in all cases.

We would like to follow up and extend this work to provide more efficient models with a reasonable number of parameters and fast enough to be integrated into our target coding assistant system.

3. Models

In this section, we present the models that are used for our experiments. As a baseline model we chose **Small-E-Czech** (Kocián, Náplava, Štancl and Kadlec, 2021) because of its moderate amount of parameters and also the fact that it is pre-trained on a large Czech corpus. Then, following e.g. Huang et al. (2022), we employed medical domain pre-trained language models.

The limitation of many transformer-based models is the length of the input sequence. To overcome this issue we use the PLM-ICD framework which allows processing of longer input sequences.The main contribution of this section is the design and implementation of the so-called "Four-headed" model which is able to handle four classification tasks at once.

3.1. Small-E-Czech

This is an Electra-based model (variant *small*) pre-trained on general Czech text with size 250 GB. The limitation of this model is the input length, where the maximum number of input tokens is 512. For prediction, we use [CLS] token. It is a special token included in the Transformer model input and is able to aggregate the information of the whole input sequence.

3.2. Mini-transformer

To the best of our knowledge, medical reports (or documents from a medical domain in general) were excluded from the Small-E-Czech pre-training. To address this drawback, we decided to carry out the pre-training of a transformer-based model with similar size as Small-E-Czech. Hereafter it is called *mini-transformer*, see Section 3.3.1 for pre-training details.

In our preliminary experiments, we noticed that using only the [CLS] token for prediction might be insufficient in this particular task. We managed to increase the overall performance of the mini-transformer model by concatenating together vectors for the [CLS] and the [SEP] tokens.

Similarly to [CLS], the [SEP] token is another special token in the Transformer model input. The [SEP] token is meant to aggregate information of individual input sequences. In our work where the input is only one sequence the [SEP] token is at the end of the input. So in other words, we use the very first and last token where the most important pieces of information are aggregated.

The concatenation thus results in more parameters and more expressive representation available for the prediction. The bottom line is that using only the [CLS] token might be a bottleneck for the classification tasks where several thousand of classes are considered.

3.3. Mini-PLM-ICD

Based on the observation of Huang et al. (2022), we further employed their PLM-ICD framework on the basis of our mini-transformer model resulting into the mini-PLM-ICD model. As illustrated in Figure 2, training of the mini-PLM-ICD model has following phases:

1. Domain-specific pre-training (medical reports used in our case);
2. Fine-tuning for ICD coding.

The framework utilises segment pooling to tackle the long input, converting each segment of up to 256 tokens into a summary vector, all of which are combined using the label-aware attention scheme (LAAT, Vu, Nguyen and Nguyen (2020)) to address the large label set issue.

3.3.1. Medical Text Pre-training

Following the PLM-ICD framework and approach from (Huang et al., 2022), we undertake the pre-training of our BERT-like model using medical report data. Unlike Huang et al. (2022), we train our Czech model from scratch due to the absence of a pre-trained Czech biomedical or clinical BERT-like model. To be able to compare results between languages and to compare the importance of the model's size to its performance, we pre-train a model with the same parameters for English as well. Similar to Huang et al. (2022), we employ the conventional masked language modelling objective as described in the original BERT paper (Devlin, Chang, Lee and Toutanova, 2019) with word masking probability set to 15%.

The size of the new model is roughly the same as for the Small-E-Czech model. Due to its moderate number of parameters (14M) the pre-training and fine-tuning is considerably faster compared to a larger model used by Huang et al. (2022). Due to hardware constraints, we opt for this smaller model, ensuring practical deployment in real-world applications. Despite this limitation, Table 5 illustrates a marginal performance decrease of approximately 2% in Micro F1-score for English. The results in terms of Macro F1-score remain consistent, despite the model being seven times smaller than the one used in Huang et al. (2022). Given our hardware limitations, we consider this trade-off between performance and model size as acceptable.

For pre-training, we compiled all our proprietary available Czech medical texts, excluding the test data, yielding approximately 2.3 GB of plain text. For English, we utilise the entire training part of the MIMIC IV dataset, see Section 2.2.

During pre-training, we employ a batch size of 120 and a maximum input sequence length of 512. We optimise the models using the cross-entropy loss function and the AdamW (Loshchilov and Hutter, 2019) optimiser over 800K steps. A linear decay strategy is applied for the learning rate, starting at 5e-4, with 1000 warm-up steps.

3.4. Four-headed Model

The motivation behind the design of this model is two-fold. The first practical reason is that the final system's requirements include two sub-tasks: classification of the main diagnosis (MDg) and all diagnoses (AllDx) list prediction (see Section 4.1.1 below for details). The tasks are moreover solved on two precision levels (3-char and 4-char) which doubles the final number of utilised models. The computational costs and response time are important for the production version and thus using only one four-headed model that tackles all required scenarios at once is highly appreciated. The text encoding costs can be shared in this case and significantly lowered compared to separate models for each scenario.

We also assume that the single tasks can benefit from being trained simultaneously with the others. Experiments show the classification of the main diagnosis does profit from the knowledge of other diagnoses.

The proposed model is an extension of the mini-PLM-ICD (which is the core of this model) that has been described in Section 3.3. The difference is in the layers just before the output layer.

As shown in Figure 3, main diagnoses are predicted by 3- and 4-char MDg heads utilising concatenation of [CLS] token and a token averaged over all other output tokens of the Transformer model. The reason behind the concatenation is to provide more features for the classification layer since small dimensionality was shown to be a bottleneck for the prediction (see Section 3.2). All diagnoses are predicted by 3- and 4-char AllDx heads using LAAT module (see Section 3.3).

Přibáň et al. (2023) showed that there are correlations among the diagnoses, therefore we further studied the effect of CorNet (Xun et al., 2020) module that is designed for such situations. Main and all diagnoses logits are optionally fed into the CorNet module with 2048 neurons to encode the correlations between diagnoses. Finally, the main and all diagnoses' logits are decoded from CorNet representation and added to the original logits. The model is then trained in a multi-task learning manner.

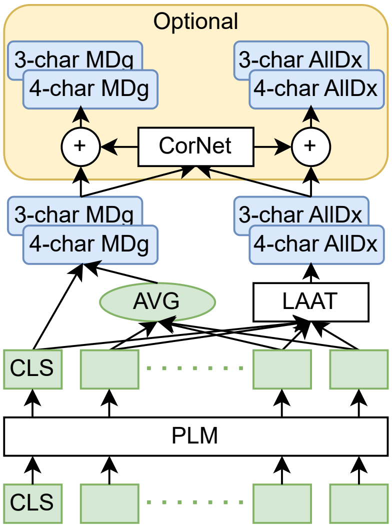


Figure 3: Four-headed model architecture

3.5. Model Comparison

In Table 1 we show the properties of the models used for our experiments. We present the model backbone, if any, number of parameters for both 3- and 4-char scenario, maximum input length and indicate if the model was pre-trained on in-domain medical data.

4. Experiments

The goal of the presented experiments is to evaluate the above described state-of-the-art models on the Czech data. Based on the comparison of the models and further analysis, the final model has been chosen for the application. We also compare our final model with SOTA models on the MIMIC IV dataset.

4.1. Czech AMC Dataset

Czech ICD-10 medical record dataset was obtained from the University hospital in Brno (UH Brno). The data were collected between years 2016 and 2022 and every record was anonymised. The dataset contains medical records describing patient's health, diagnoses, bedside measurements and also laboratory tests. In many cases, subsequent records are joined together following patient's stay in the hospital at different wards. In some cases the final outcome is an autopsy record.

Even though there are headings and logical blocks in the text, they are not consistent and may vary. Therefore, the text is considered as not explicitly structured. The dataset was annotated manually by the hospital's coders in UH Brno with main / other diagnoses labels.

The coders may include: mid-level medical staff (nurses), staff originally from other disciplines or doctors depending on the organisation of clinical coding in the hospital. They receive training through self-studies and from courses organised for coding by the Ministry of Health or commercial bodies.

For maintaining high-quality coding standards we selected a hospital having a big coding department with a long history. Furthermore, all coders have undergone extensive and recurrent training in coding. The high quality is further attested by the minimal number of cases reviewed by Czech health insurance companies. Additionally, we conducted analyses on various subsets of the data with an expert, validating this assertion.

The dataset was split into train, validation, and test parts while considering a similar distribution of labels. Exact percentage of documents in each particular split is shown in Table 2. This decision has been made to have sufficient amount of training data which is important for our final application while preserving a similar amount of data as in MIMIC IV for validation and testing.

Due to the license policy, the dataset cannot be made publicly available. We thus also provide results on MIMIC IV to allow better comparison of our methods with the state of the art.

Table 2 compares statistics of the Czech AMC and MIMIC IV datasets. The main difference lies in the

Model	Backbone	# Parameters [M]		Max Input Length	Domain Pre-training
		3-char	4-char		
HA-GRU (Přibáň et al., 2023)	–	13	–	1,250	✗
Small-E-Czech	–	13.9	15.9	512	✗
mini-transformer	–	15.1	17.0	512	✓
mini-transformer CLS+SEP	–	15.6	19.5	512	✓
mini-PLM-ICD	mini-transformer	15.6	19.5	3,072	✓
Four-headed	mini-transformer	26.0		3,072	✓
Four-headed + CorNet	mini-transformer	115.7		3,072	✓

Table 1: Comparison of utilised models

number of "Codes per instance" where the median of assigned codes are 2 and 14 for the Czech dataset and MIMIC IV, respectively. One reason is that MIMIC IV documents contain diagnosis codes together with procedure codes. Another reason for having such a low number of codes per instance might be that Czech coders are not focused on "side" and marginal diagnoses because they are not significant from the point of view of insurance payments. Furthermore, the MIMIC IV documents have two-times more words than Czech medical reports. More erudite explanation of the differences between the datasets is beyond our expertise.

4.1.1. *Test Scenarios*

The experiments on MIMIC dataset are usually performed on full codes or on a subset of 50 most frequent codes. However, the performance on full codes is relatively low and classifying only 50 codes does not make sense for our target application.

By analysing the dataset from the point of view of the chapters and their hierarchy, we have found out that the vast majority of diagnoses in the Czech dataset contain 4 or 3 characters and only a small number of diagnoses have 5 characters (see Figure 4).

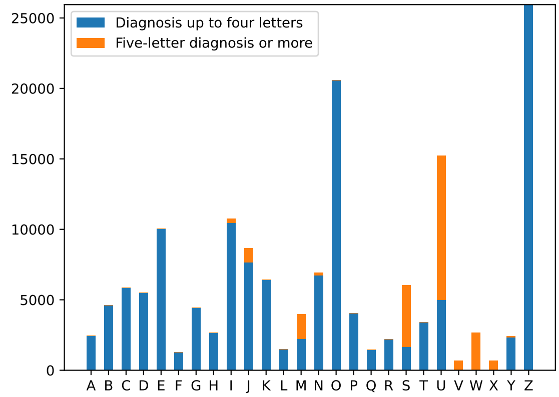


Figure 4: Number of diagnoses longer than 4 characters per chapter in Czech AMC Dataset

The decision to use only 3 and 4 character predictions was supported also by the large amount of full 5-character codes that have insufficient number of occurrences in the annotated data. This fact may lead to insufficient accuracy of the models and be misleading for the user. For the intended use-case – a coding assistant – the coding expert and potential users with whom we have consulted the scenarios suggested that it is more beneficial to have precisely coded first 4 characters than erroneous full codes. The coder can then easily complete the final code.

Based on the requirements for the final application that came out of the discussion with an expert coder and the potential users we also omit codes in the range U to Y. The range V – Y contains external causes of morbidity and mortality and U are codes for special purposes. All of these codes cannot be used as a main

	Czech AMC Dataset	MIMIC-IV ICD 10
Number of documents	432,279	122,279
Number of unique full codes	9,735 (68,000)	7,942
Number of unique 4 char codes	6,304 (9,268)	-
Number of unique 3 char codes	1,538 (1,673)	-
Codes per instance: Median (IQR)	2 (1-4)	14 (9-20)
Words per document: Median (IQR)	680 (358-1,139)	1,492 (1,147-1,931)
Documents: Train / val / test [%]	87.1 / 4.3 / 8.6	72.9 / 10.9 / 16.2

Table 2: MIMIC-IV and Czech AMC dataset statistics, MIMIC-IV ICD 10 data from Edin et al. (2023), for Czech AMC only diagnoses are considered; IQR is the interquartile range (also known as midspread or middle 50%); the statistics of codes include number of codes in data and total number of possible codes - in brackets

diagnosis and therefore they are not considered for our application.

ICD codes were therefore reduced to only first 3 or 4 characters (e.g. Z62.89 → Z62 and → Z628, respectively). The two basic required tasks are the prediction of a main diagnosis (single label classification) and all diagnoses prediction (multi-label classification). The resulting numbers of the possible labels are 1673 (3-char) and 9278 (4-char).

Based on the client application demands we have identified the following test scenarios:

- 3 character main diagnosis classification (3-char MDg);
- 3 character all diagnoses (3-char AllDx);
- 4 character main diagnosis (4-char MDg);
- 4 character all diagnoses (4-char AllDx).

As indicated above, in many cases the 4-char codes represent final ICD code. If the code is longer, it usually represents some specific sub-diagnosis that can be easily completed manually. The main reason of predicting 3 character diagnoses is that in case of low confidence results for the 4-char, the coder can check the 3-char result and edit the final set of diagnoses accordingly.

4.2. Experimental Set-up

All experiments were carried out on a computer with GPU NVidia RTX A4000 with 16 GB memory. For our fine-tuning experiments we use the following settings: AdamW(Loshchilov and Hutter, 2019) optimiser, learning rate = 2e-4, batch size 32 and early stopping resulting in the number of epochs in range 20 – 30 For mini-PLM-ICD model, the text chunk size was set to 256 tokens. The maximum input sequence is set to 3072 tokens.

4.2.1. Pre-processing

Our pre-processing pipeline is based on Mullenbach et al. (2018). All words are lowercased with removed numbers, datetimes and punctuation marks. Due to the possible bias, we also remove ICD codes and doctor names. Furthermore, we removed the header of each report containing information about the hospital and anonymised patient's records.

4.2.2. Evaluation Criteria

The AllDx test scenarios are evaluated using standard metrics for multi-label text classification - precision, recall and F1-score in both micro- and macro-averaged variants. Additionally, based on the experience of Edin et al. (2023), we present exact match ratio (EMR) metric which indicates how many reports were classified perfectly, i.e. the percentage of reports for that the classifier identified all labels correctly. This metric has been chosen to demonstrate the performance of fully automated medical coding.

In the case of MDg scenarios, we report accuracy and macro averaged precision, recall and F1-score.

4.3. Czech AMC Dataset Results

We first present the experiments on the main diagnosis classification task. Table 3 summarises the results for 3 and 4 character main diagnosis classification. First two lines are baseline results obtained in the previous study (HA-GRU) and the results of a general Czech model Small-E-Czech fine-tuned on our task. We can state that both baseline models perform comparably, reaching 78% accuracy in the 3-char task. It is crystal clear from the following lines that pre-training on the medical data brings significant improvement over the baselines. The best results were obtained by the proposed four-headed model which consistently reaches better numbers in comparison with previous models. The addition of the CorNet module does not have any impact on the results in this scenario. The best accuracies are around 85% and 77% for the 3-char and 4-char tasks, respectively.

Experiments on the AllDx scenario are reported in Table 4. It presents both 3-char and 4-char AllDx results. We show both micro and macro averaged metrics and also the exact match ratio (EMR). First, we report two baselines as in the previous case followed by five models pre-trained on medical data. Similarly as in the main diagnosis classification, domain pre-training ensures much better results. Contrary to the main diagnosis task, four-headed model brings no significant improvement. We assume that the difference results from the fact that MDg can profit from the knowledge of other diagnoses when classifying the main one. However, this does not hold for the classification of all diagnoses where there is no visible impact of the knowledge of the main diagnosis. Also the CorNet layer does not bring any improvements in this task as we expected when designing the model.

4.4. Comparison with the State of the Art

This section compares the results of our models with the state-of-the-art models on the Mimic IV dataset. To be completely fair with the comparison from Edin et al. (2023) we have also included the P@8 and P@15 metrics. We compare only the AllDx scenario because the MDg classification task is not defined in MIMIC data. Table 5 shows the results of our two best performing models in comparison with results reported by Edin et al. (2023).

4.5. Analysis & Discussion

We conducted several sets of experiments with the aim to find the best model for the prediction of ICD-10 diagnoses. At the same time, we tried to meet the requirements of the final system.

Model	**3 Char Scenario**				**4 Char Scenario**			
	Prec.	*Rec.*	*F1*	*Acc.*	*Prec.*	*Rec.*	*F1*	*Acc.*
HA-GRU (Přibáň et al., 2023)	48.4	45.6	45.1	78.2	-	-	-	-
Small-E-Czech	47.4	46.2	44.8	78.3	**52.0**	36.6	33.4	72.8
mini-transformer	56.3	55.7	54.2	83.7	37.1	38.2	35.7	75.3
mini-transformer CLS+SEP	59.7	56.9	56.3	84.4	43.8	43.9	42.3	76.7
mini-PLM-ICD	60.5	59.9	58.8	84.2	38.9	39.4	37.3	76.3
Four-headed	**63.0**	60.7	60.5	**84.9**	45.4	44.5	43.5	**77.7**
Four-headed + CorNet	62.9	**60.8**	**60.6**	84.8	45.4	**44.7**	**43.6**	77.6

Table 3: Main diagnosis classification results on the Czech AMC dataset [in %]; all metrics are macro-averaged.

Model	**3 Char Scenario**							**4 Char Scenario**						
	micro			**macro**			*EMR*	**micro**			**macro**			*EMR*
	Prec.	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*		*Prec.*	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*	
HA-GRU (Přibáň et al., 2023)	79.7	71.1	75.1	50.3	38.3	41.8	–	–	–	–	–	–	–	–
Small-E-Czech	82.8	66.2	73.6	41.6	27.7	31.4	45.7	82.0	50.1	62.2	16.3	10.0	11.4	34.6
mini-transformer	**83.3**	74.9	78.9	54.5	41.0	45.0	**53.9**	**82.2**	58.0	68.0	28.2	18.1	20.6	41.9
mini-transformer CLS+SEP	82.2	78.1	80.1	**57.1**	48.0	50.6	53.8	77.3	68.1	72.4	37.2	29.4	31.1	44.9
mini-PLM-ICD	81.3	**80.7**	**81.0**	56.6	**54.1**	**54.0**	**53.9**	77.3	**71.4**	**74.2**	39.4	**35.5**	**35.7**	46.2
Four-headed	81.0	79.4	80.2	56.7	52.0	53.0	**53.9**	76.9	70.5	73.6	**39.7**	34.3	35.2	46.3
Four-headed + CorNet	80.7	79.6	80.1	56.5	52.3	53.1	**53.9**	76.2	71.0	73.5	39.4	34.6	35.3	**46.5**

Table 4: All diagnoses classification results on the Czech AMC dataset [in %]; EMR is exact match ratio.

Model	**#Parameters**	**Micro F1**	**Macro F1**	**EMR**	**P@8**	**P@15**
PLM-ICD[a]	139M	**58.5**	21.1	0.4	**69.9**	**55.0**
mini-PLM-ICD	**19M**	56.4	**21.3**	0.3	68.6	54.1
Four-headed + CorNet	88M	55.7	15.0	**0.6**	68.2	53.6

Table 5: Results on MIMIC IV ICD-10 dataset [in %]. Result marked with [a] is from Edin et al. (2023).
For the Four-headed + CorNet model, procedures and diagnoses of MIMIC IV are treated separately in AllDx heads and then combined for the evaluation. The output of MDg heads is utilised by CorNet module and may provide additional information for AllDx heads. Without CorNet, architecture is the same as mini-PLM-ICD because the MDg heads of Four-headed model are ignored as there is no main diagnosis defined.

Overall, we evaluated various models on two corpora in two languages (Czech and English). The results are good-enough and applicable for the final system. The PLM-ICD framework seems to be working even if the pre-trained language model (PLM) is relatively small in terms of number of parameters. As shown in Table 5, we obtained very competitive results on English MIMIC IV dataset.

Overall, we can state that from the results, it turned out that MDg task has better results with prediction from concatenating the [CLS] token with the [SEP] token and/or averaging all output tokens, while LAAT is better for all diagnoses scenario.

Unfortunately, we were not able to fully employ the Four-headed model in the MIMIC IV dataset since there is no such a task as the classification of a main diagnosis. Even though we tried some modification to split label set into procedures and diagnoses (with the belief that procedures might influence the resulting diagnoses), the MIMIC's evaluation scenario considers mixed procedures and diagnosis where the model obtained similar or worse results than mini-PLM-ICD. This analysis is therefore based mainly on results based on Czech data.

As derived from the motivation and requirements for the final system, the main advantages of the proposed Four-headed model are the shared parameters and reduced computational costs (e.g for encoding the text, which has to be done multiple times in the case of separate models). Depending on the scenario, the training time of one epoch ranges from 45 to 70 minutes for *Small-E-Czech*, *mini-transformer* and *mini-transformer CLS+SEP* models with limited input length. Training time for *mini-PLM-ICD* ranges from 137 to 160 minutes while *Four-headed* and *Four-headed + CorNet* models take 196 and 219 minutes per epoch in average, respectively.

Prediction times are detailed in Table 6. We can see comparable results for short and medium inputs.

Model	Scenario	Short	Medium	Long
Small-E-Czech[a]	3-char AllDx	10.52 ± 0.51	11.12 ± 1.33	13.75 ± 0.84
	3-char MDg	10.76 ± 2.72	11.15 ± 0.36	13.84 ± 3.13
	4-char AllDx	10.60 ± 0.50	11.10 ± 0.33	13.68 ± 0.48
	4-char MDg	10.64 ± 0.70	11.06 ± 0.34	13.72 ± 0.50
mini-transformer[a]	3-char AllDx	10.71 ± 0.46	11.10 ± 0.31	13.91 ± 0.34
	3-char MDg	10.80 ± 0.43	11.21 ± 0.41	14.02 ± 0.27
	4-char AllDx	10.72 ± 0.46	11.13 ± 0.33	13.92 ± 0.34
	4-char MDg	10.84 ± 0.39	11.21 ± 0.41	14.01 ± 0.35
mini-transformer CLS+SEP[a]	3-char AllDx	10.73 ± 0.46	11.16 ± 0.36	13.91 ± 0.36
	3-char MDg	10.85 ± 0.40	11.21 ± 0.40	13.96 ± 0.33
	4-char AllDx	10.68 ± 0.47	11.09 ± 0.28	13.91 ± 0.35
	4-char MDg	10.83 ± 0.41	11.26 ± 0.43	14.05 ± 0.36
mini-PLM-ICD	3-char AllDx	10.82 ± 0.40	11.90 ± 0.37	15.82 ± 0.44
	3-char MDg	10.89 ± 0.36	12.09 ± 0.40	15.81 ± 0.45
	4-char AllDx	11.15 ± 0.35	12.14 ± 0.35	17.88 ± 0.42
	4-char MDg	11.12 ± 0.35	12.14 ± 0.38	17.89 ± 0.43
Four-headed	all	11.34 ± 0.49	12.31 ± 0.49	23.82 ± 1.48
Four-headed + CorNet	all	12.11 ± 0.40	13.09 ± 0.36	24.87 ± 0.50

Table 6: Prediction time comparison [ms]; We report mean and standard deviation of 10,000 runs for each scenario and three input texts *Short*, *Medium* and *Long* that represent 117, 508, and 2984 input tokens, respectively; Measurements were made using NVIDIA RTX A4000 and AMD Ryzen 5 5600X; [a]Long input is truncated to Max Input Length of the model

For long inputs, the prediction heads perform more computation, resulting in an increasing time difference. In summary, if we consider the necessity of all scenarios and the same input length, the time consumption of *Four-headed* model is reduced by approximately 3-4 times for both, training and prediction.

Another assumption is that combined training may positively influence the representations for final prediction of the main diagnosis. Our validation experiments have truly shown the increase of the accuracy/macro F1 for 3 char MDg and 4 char MDg scenarios and the final results (see Table 3) also indicate the better performance than single mini-PLM-ICD.

We added the CorNet module in the Four-headed model with the belief that it might bring better results since some diagnoses are related to each other. Even though the CorNet module consumes a major amount of model parameters (90 millions), it hasn't brought significant positive impact in terms of the overall performance. We suspect that the reason is due to a smaller number of codes per instance in the dataset and also the sufficient capability of the transformer model to learn the code correlation.

5. System Architecture

The entire deployed system consists of three functional units:

1. Pre-processing unit;
2. Text classification unit;
3. User interface.

These units are composed of the different interconnected individual modules. The overview of the system architecture is depicted in Figure 5. The individual units are described in the following sections.

5.1. Pre-processing

The goal of this unit is to pre-process the input text document. It contains the modules for text editing that can be adjusted based on the particular hospital (to remove different headers in the medical reports etc.). The used pre-processing steps were discussed in Section 4.2.1.

5.2. Classification Unit

This unit is the core of the MCA system. It consists of the Four-headed model implemented as a web service that provides the required output with probabilities for all classification tasks and sub-tasks. The final choice of the deployed model was made by the evaluation of our experiments.

The unit can be easily updated (e.g. the model replacement/updating or the prediction of other related tasks such as procedures classification etc.). From the point of view of the MCA system users, the regular updates of the classification model are desired. The main reason behind this is to keep up with the latest

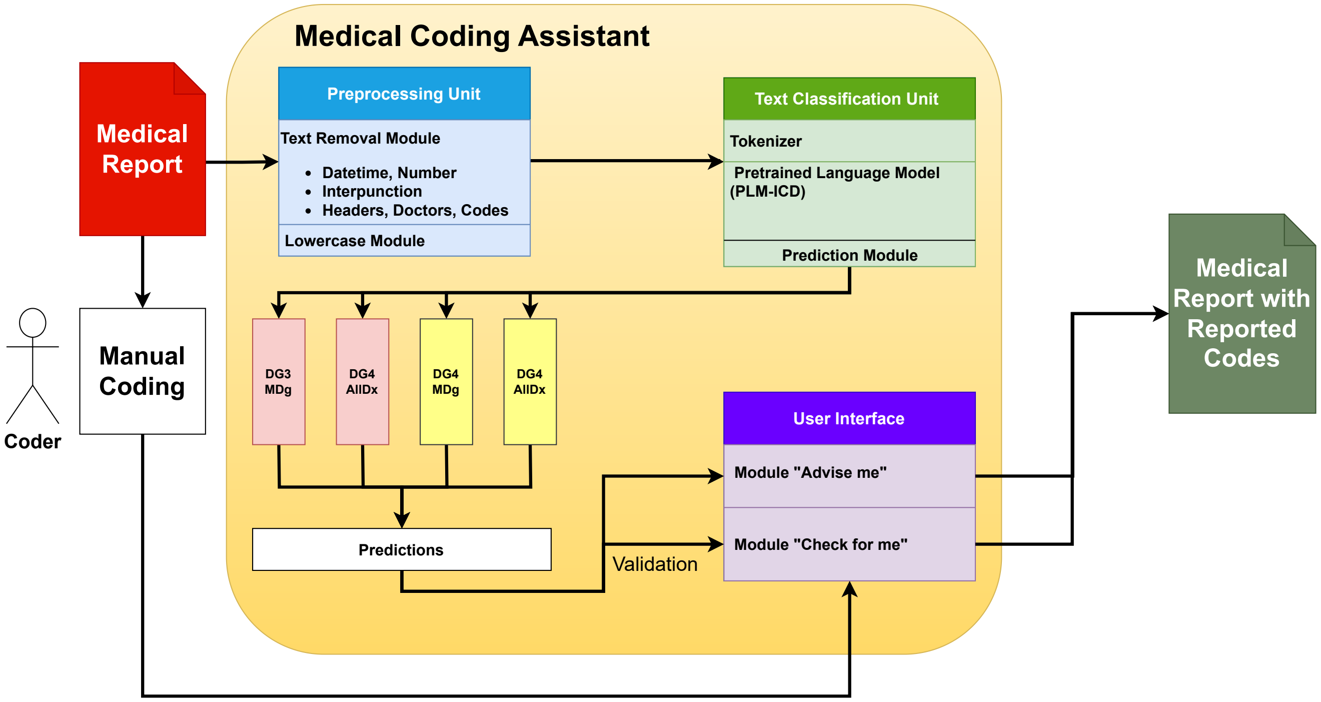


Figure 5: MCA System architecture/pipeline

ICD methodology revisions and, of course, having more and more training data (especially for rare diagnoses).

5.3. User Interface

This component serves for visualisation of the particular classification task and for communication with the user. The system operates in two use cases:

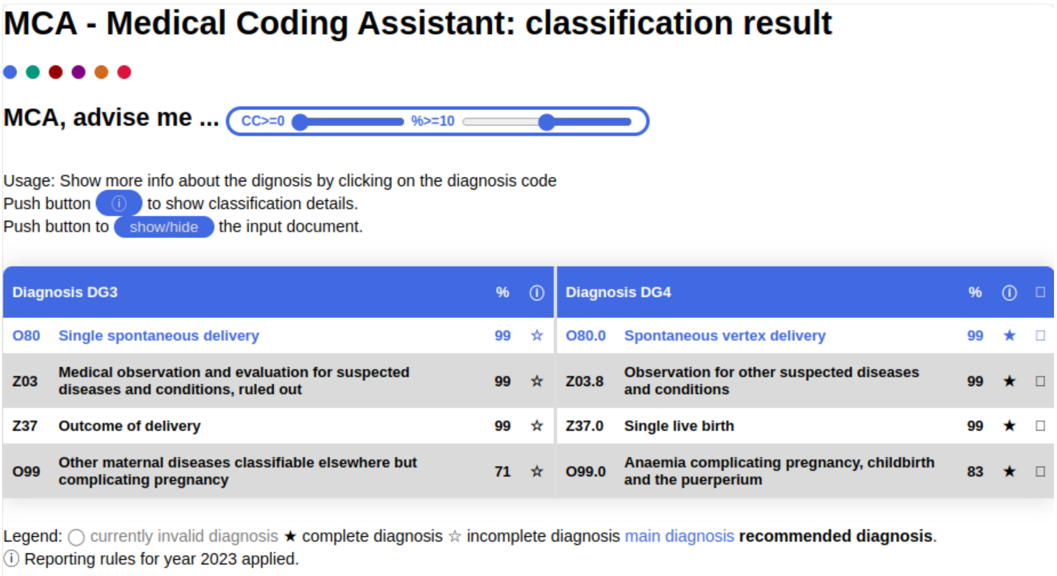
1. **"Advise me"**
   This mode is used for providing advice on coding diagnoses. User submits health record to the system that recommends diagnoses based on individual threshold for each diagnosis. The threshold was estimated experimentally as the best one in terms of usability for coders.
   The user can also view additional information about diagnoses (completeness information, potential main diagnosis, further detailed information).
2. **"Check for me"**
   This mode is used as a validator for diagnoses that are already manually coded. The user submits a health record and the list of expected diagnoses.
   The result of the check is a detailed list of both coded and classified diagnoses. Furthermore, there is additional information about diagnoses (information about completeness or other detailed information). Based on the displayed results, the user can re-evaluate and modify the coded diagnoses.

6. Conclusions & Future Work

In this paper we proposed and implemented a medical coding assistant system designed to aid human coders with diagnosis coding. This system predicts a set of diagnosis ICD codes based on input text written in Czech language. The codes are subsequently presented to a human for validation or correction. The classification is performed for 3 and 4 character codes. In some cases, it is thus necessary for the user to complete the full code. This design decision was made after discussion with coders for whom it is more beneficial to have precisely predicted first 4 characters than erroneously classified full code.

The main contribution of this research is the introduction of innovative ICD classification models tailored for the Czech language based on the transformer architecture. These models have a relatively low number of training parameters, facilitating rapid deployment on the standard computer systems found in Czech hospitals. They also allow for effortless retraining or fine-tuning as new data becomes available. Initially, we introduced a small transformer-based model for each task, totalling four models. Subsequently, we proposed a novel "Four-headed" model, which incorporates four specific classification heads.

We experimentally showed that this model achieves comparable, sometimes even better results, against four individual models. Moreover this novel model significantly reduces memory consumption and learning time, approximately four times. Therefore, the additional advantage of this model is also its reduced



MCA - Medical Coding Assistant: classification result

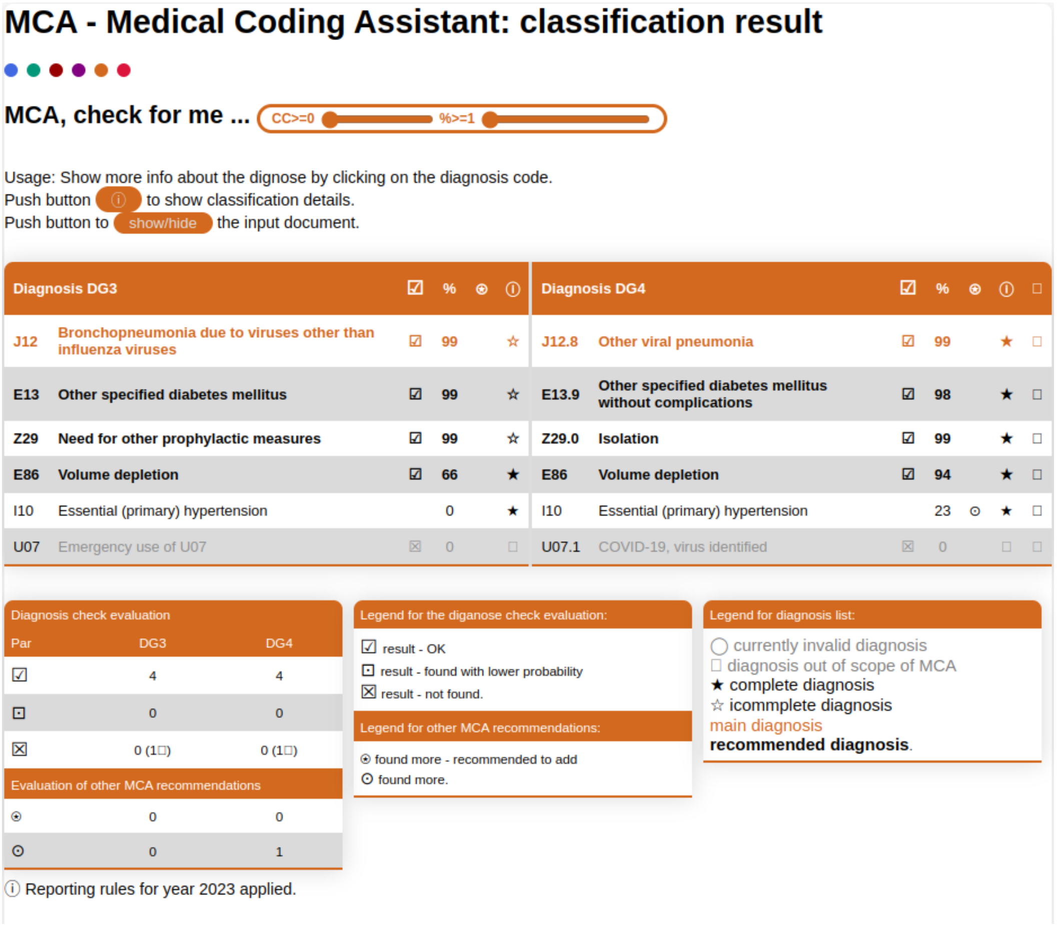
MCA, advise me ...

Usage: Show more info about the dignosis by clicking on the diagnosis code
Push button ⓘ to show classification details.
Push button to show/hide the input document.

	Diagnosis DG3	%	ⓘ		Diagnosis DG4	%	ⓘ	□
O80	Single spontaneous delivery	99	☆	O80.0	Spontaneous vertex delivery	99	★	□
Z03	Medical observation and evaluation for suspected diseases and conditions, ruled out	99	☆	Z03.8	Observation for other suspected diseases and conditions	99	★	□
Z37	Outcome of delivery	99	☆	Z37.0	Single live birth	99	★	□
O99	Other maternal diseases classifiable elsewhere but complicating pregnancy	71	☆	O99.0	Anaemia complicating pregnancy, childbirth and the puerperium	83	★	□

Legend: ○ currently invalid diagnosis ★ complete diagnosis ☆ incomplete diagnosis main diagnosis **recommended diagnosis**.
ⓘ Reporting rules for year 2023 applied.

Figure 6: MCA System "advise me" use case. This output is used for prediction of unknown report. The empty star symbol indicates that a predicted diagnosis is not "completed", so a user needs further investigation to find a full-code (either 4-char or more).



MCA - Medical Coding Assistant: classification result

MCA, check for me ...

Usage: Show more info about the dignose by clicking on the diagnosis code.
Push button ⓘ to show classification details.
Push button to show/hide the input document.

	Diagnosis DG3	☑	%	⊛	ⓘ		Diagnosis DG4	☑	%	⊛	ⓘ	□
J12	Bronchopneumonia due to viruses other than influenza viruses	☑	99		☆	J12.8	Other viral pneumonia	☑	99		★	□
E13	Other specified diabetes mellitus	☑	99		☆	E13.9	Other specified diabetes mellitus without complications	☑	98		★	□
Z29	Need for other prophylactic measures	☑	99		☆	Z29.0	Isolation	☑	99		★	□
E86	Volume depletion	☑	66		★	E86	Volume depletion	☑	94		★	□
I10	Essential (primary) hypertension		0		★	I10	Essential (primary) hypertension		23	⊙	★	□
U07	Emergency use of U07	☒	0		□	U07.1	COVID-19, virus identified	☒	0		□	□

Diagnosis check evaluation

Par	DG3	DG4
☑	4	4
⊡	0	0
☒	0 (1□)	0 (1□)
Evaluation of other MCA recommendations		
⊛	0	0
⊙	0	1

Legend for the diganose check evaluation:

☑ result - OK
⊡ result - found with lower probability
☒ result - not found.

Legend for other MCA recommendations:

⊛ found more - recommended to add
⊙ found more.

Legend for diagnosis list:

○ currently invalid diagnosis
□ diagnosis out of scope of MCA
★ complete diagnosis
☆ icommplete diagnosis
main diagnosis
recommended diagnosis.

ⓘ Reporting rules for year 2023 applied.

Figure 7: MCA System "check for me" use case. This output is used to validate manually assigned codes. The user interface is in this case much richer and it is up to a coder either to update or ignore the results.

footprint (Schwartz et al., 2020). We also showed that the proposed models achieve comparable results against state-of-the-art English models on the Mimic IV dataset even though our models are significantly smaller.

The system is currently being implemented in the University hospital in Brno with positive feedback of the users. We are planning to deploy this system across several hospitals in the Czech Republic. Given the diverse landscape of hospitals in the Czech Republic which often possess limited computational resources, the "Four-headed" model will be integrated into the final system.

For the future work, we plan to further improve the system performance. We will potentially include new data acquired from other hospitals which should lead to a more balanced training dataset. We also would like to experiment with generative models and their potential usage in this task directly as well as for their possible usage for data augmentation.

Acknowledgements

This work has been partly supported by grant no. SGS-2022-016 Advanced methods of data processing and project no. CZ.01.1.02/0.0/0.0/20_321/0024835 Analysis and Research and development (R&D) of the company ICZ corp in the use of artificial intelligence (AI) in the processing of medical documentation.

References

Azam, S.S., Raju, M., Pagidimarri, V., Kasivajjala, V.C., 2020. Cascadenet: An LSTM based deep learning model for automated icd-10 coding, in: Advances in Information and Communication: Proceedings of the 2019 Future of Information and Communication Conference (FICC), Volume 2, Springer. pp. 55–74.

Bhatia, K., Jain, H., Kar, P., Varma, M., Jain, P., 2015. Sparse local embeddings for extreme multi-label classification. Advances in neural information processing systems 28.

Chen, Y., Tian, Q., Cai, H., Lu, X., 2022. A semi-automatic data cleaning & coding tool for Chinese clinical data standardization, in: MEDINFO 2021: One World, One Health–Global Partnership for Digital Innovation. IOS Press, pp. 106–110.

Crammer, K., Dredze, M., Ganchev, K., Talukdar, P., Carroll, S., 2007. Automatic code assignment to medical text, in: Biological, translational, and clinical language processing, pp. 129–136.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 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), Association for Computational Linguistics, Minneapolis, Minnesota. pp. 4171–4186. URL: https://aclanthology.org/N19-1423, doi:10.18653/v1/N19-1423.

Edin, J., Junge, A., Havtorn, J.D., Borgholt, L., Maistro, M., Ruotsalo, T., Maaløe, L., 2023. Automated medical coding on MIMIC-III and MIMIC-IV: A critical review and replicability study. arXiv preprint arXiv:2304.10909 .

Farkas, R., Szarvas, G., 2008. Automatic construction of rule-based ICD-9-CM coding systems, in: BMC bioinformatics, Springer. pp. 1–9.

Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P., Mark, R., Mietus, J., Moody, G., Peng, C., Stanley, H., 2000. Physiobank, Physiotoolkit, and Physionet: components of a new research resource for complex physiologic signals. Circulation 101, e215–e220.

Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., Wang, Y., 2021. Transformer in transformer. Advances in Neural Information Processing Systems 34, 15908–15919.

Huang, C.W., Tsai, S.C., Chen, Y.N., 2022. PLM-ICD: Automatic ICD coding with pretrained language models, in: Proceedings of the 4th Clinical Natural Language Processing Workshop, Association for Computational Linguistics, Seattle, WA. pp. 10–20. URL: https://aclanthology.org/2022.clinicalnlp-1.2, doi:10.18653/v1/2022.clinicalnlp-1.2.

Johnson, A.E., Pollard, T.J., Shen, L., Lehman, L.H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G., 2016. MIMIC-III, a freely accessible critical care database, in: Scientific data 3.

Karimi, S., Dai, X., Hassanzadeh, H., Nguyen, A., 2017. Automatic diagnosis coding of radiology reports: a comparison of deep learning and conventional classification methods, in: BioNLP 2017, pp. 328–332.

Khandagale, S., Xiao, H., Babbar, R., 2020. Bonsai: diverse and shallow trees for extreme multi-label classification. Machine Learning 109, 2099–2119.

Kim, B.H., Ganapathi, V., 2021. Read, attend, and code: pushing the limits of medical codes prediction from clinical notes by machines, in: Machine Learning for Healthcare Conference, PMLR. pp. 196–208.

Kim, Y., 2014. Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics.

Kocián, M., Náplava, J., Štancl, D., Kadlec, V., 2021. Siamese bert-based model for web search relevance ranking evaluated on a new czech dataset. arXiv preprint arXiv:2112.01810 .

Liu, J., Chang, W.C., Wu, Y., Yang, Y., 2017. Deep learning for extreme multi-label text classification, in: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval, pp. 115–124.

Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net. URL: https://openreview.net/forum?id=Bkg6RiCqY7.

Moody, G.B., Mark, R.G., Goldberger, A.L., 2011. Physionet: Physiologic signals, time series and related open source software for basic, clinical, and applied research, in: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 8327–8330. doi:10.1109/IEMBS.2011.6092053.

Mullenbach, J., Wiegreffe, S., Duke, J., Sun, J., Eisenstein, J., 2018. Explainable prediction of medical codes from clinical text. arXiv preprint arXiv:1802.05695 .

Niu, K., Wu, Y., Li, Y., Li, M., 2023. Retrieve and rerank for automated icd coding via contrastive learning. Journal of Biomedical Informatics 143, 104396. URL: https://www.sciencedirect.com/science/article/pii/S153204642300117X, doi:https://doi.org/10.1016/j.jbi.2023.104396.

O'Malley, K.J., Cook, K.F., Price, M.D., Wildes, K.R., Hurdle, J.F., Ashton, C.M., 2005. Measuring diagnoses: Icd code accuracy. Health services research 40, 1620–1639.

Pestian, J., Brew, C., Matykiewicz, P., Hovermale, D.J., Johnson, N., Cohen, K.B., Duch, W., 2007. A shared task involving multi-label classification of clinical free text, in: Biological, translational, and clinical language processing, pp. 97–104.

Přibáň, P., Baloun, J., Martínek, J., Lenc, L., Prantl, M., Král, P., 2023. Towards automatic medical report classification in Czech, in: Proceedings of the 15th International Conference on Agents and Artificial Intelligence Volume 3, SciTePress, Lisbon, Portugal. pp. 228–233. URL: https://doi.org/10.5220/0011641900003393, doi:10.5220/0011641900003393.

Schwartz, R., Dodge, J., Smith, N.A., Etzioni, O., 2020. Green ai. Communications of the ACM 63, 54–63.

Sechidis, K., Tsoumakas, G., Vlahavas, I., 2011. On the stratification of multi-label data, in: Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part III 22, Springer. pp. 145–158.

Vu, T., Nguyen, D.Q., Nguyen, A., 2020. A label attention model for ICD coding from clinical text. arXiv preprint arXiv:2007.06351 .

Xun, G., Jha, K., Sun, J., Zhang, A., 2020. Correlation networks for extreme multi-label text classification, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA. p. 1074–1082. URL: https://doi.org/10.1145/3394486.3403151, doi:10.1145/3394486.3403151.

Zhang, S., Zhang, B., Zhang, F., Sang, B., Yang, W., 2022. Automatic ICD coding exploiting discourse structure and reconciled code embeddings, in: Proceedings of the 29th International Conference on Computational Linguistics, pp. 2883–2891.