Multi-label Diagnosis Classification of Swedish Discharge Summaries – ICD-10 Code Assignment Using KB-BERT

Sonja Remmer[a,b] remmer@dsv.su.se
Anastasios Lamproudis[a] anastasios@dsv.su.se
Hercules Dalianis[a,b] hercules@dsv.su.se

[a]Department of Computer and Systems Sciences (DSV)
Stockholm University, Kista, Sweden
[b]Norwegian Centre for E-health Research, Tromsø, Norway

Abstract

The International Classification of Diseases (ICD) is a system for systematically recording patients' diagnoses. Clinicians or professional coders assign ICD codes to patients' medical records to facilitate funding, research, and administration. In most health facilities, clinical coding is a manual, time-demanding task that is prone to errors. A tool that automatically assigns ICD codes to free-text clinical notes could save time and reduce erroneous coding. While many previous studies have focused on ICD coding, research on Swedish patient records is scarce. This study explored different approaches to pairing Swedish clinical notes with ICD codes. KB-BERT, a BERT model pre-trained on Swedish text, was compared to the traditional supervised learning models Support Vector Machines, Decision Trees, and K-nearest Neighbours used as the baseline. When considering ICD codes grouped into ten blocks, the KB-BERT was superior to the baseline models, obtaining an F_1-micro of 0.80 and an F_1-macro of 0.58. When considering the 263 full ICD codes, the KB-BERT was outperformed by all baseline models at an F_1-micro and F_1-macro of zero. Wilcoxon signed-rank tests showed that the performance differences between the KB-BERT and the baseline models were statistically significant.

1 Introduction

There are both administrative and statistical purposes of ICD coding. Administrative to reimburse the clinical unit or hospital, but also to plan healthcare. The codes are assigned by both treating physicians and designated coders. The current version of the ICD system, ICD-10, contains tens of thousands of codes divided into 22 chapters (WHO, 2016).

ICD coding is time-consuming and error-prone, either missing the main diagnosis or displaying errors in the coding in up to 20 per cent of the patient records (Jacobsson and Serdén, 2013). Therefore, it would be valuable to have a supporting tool to assist the physician or coder in choosing among the codes.

In this article, Swedish patient records in the medical speciality of gastrointestinal surgery and their already assigned ICD-10 codes are used to perform supervised learning to predict ICD-10 codes. More specifically, the part of the patient records that summarises the patient's care period at the time of the discharge, the discharge summaries, and their associated ICD-10 codes are used. The assigned codes belong to the Swedish version of the ICD-10 system known as ICD-10-SE (Socialstyrelsen, 2018). The codes considered are both full ICD codes at the highest level of granularity and the full codes grouped into ten blocks. The research question is how the deep learning language model KB-BERT, compared to the traditional supervised learning models Support Vector Machines, Decision Trees, and K-Nearest Neighbours performs in pairing discharge summaries with the correct ICD codes.

2 Related Research

ICD coding has been a popular research area for decades. The interest increased with a public challenge hosted by the Computational Medical Center called the *2007 Computational Medicine Challenge*, where contestants were asked to create a system for pairing radiology reports with the correct ICD codes. Most submitted solutions used hand-crafted rules, traditional supervised learning models such as Support Vector Machines, or a combination of these two approaches (Pestian et al., 2007). One of the top-performing systems used a combination of rule-based and machine learning elements, achieving an F_1-micro score of 0.89 by utilising Decision Trees to generate rules automati-

Proceedings of Recent Advances in Natural Language Processing, pages 1158–1166
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_130

cally (Farkas and Szarvas, 2008).

Since 2007, ICD classification has shifted away from rule-based techniques, and many recent studies use traditional supervised learning methods or deep learning approaches. Examples of conventional models used in previous ICD coding papers are Support Vector Machines, Decision Trees, K-nearest Neighbours, Naïve Bayes, and ensembles of these models. In Kaur and Ginige (2018), comparing these conventional models with Multi-layer Perceptrons resulted in Decision Trees and AdaBoost using Decision Trees being the superior classifiers at F-scores of approximately 0.9. Hasan et al. (2016) compared traditional models with Convolutional Neural Networks and concluded that the results for the Convolutional Neural Networks were comparable with the results of Support Vector Machines, but that Support Vector Machines outperformed Convolutional Neural Networks as the number of classes increased. The best-achieved accuracy score of Support Vector Machines in (Hasan et al., 2016) was 0.75. For Bulgarian, Boytcheva (2011) carried out ICD classification using Support Vector Machines. She used 6,200 and 1,300 electronic patient records for training and evaluation, respectively, obtaining a precision of 0.97, a recall of 0.74, and an F-score of 0.85.

An increasingly popular deep learning approach to ICD coding tasks is using the language model BERT (**B**idirectional **E**ncoder **R**epresentations from **T**ransformers) developed in 2018 by Devlin et al. (2019). BERT was pre-trained on 3.3 billion words from two English corpora – the BooksCorpus and the English Wikipedia – making it an expert in understanding general English. However, BERT has also been adapted to domain-specific language. For example, Lee et al. (2020) developed BioBERT – a BERT model pre-trained on biomedical texts. When Amin et al. (2019) adopted BioBERT to perform ICD coding, they reached an F_1-micro score of 0.73. Moreover, a BERT model pre-trained on clinical text was developed by Alsentzer et al. (2019) and used by Biseda et al. (2020) for ICD classification, achieving an F_1-score of 0.75.

Since the original BERT and many of its domain-specific adaptations are trained on English texts, BERT has also been adapted to understand other languages. In 2020, the National Library of Sweden pre-trained a BERT model on billions of Swedish words, naming this model KB-BERT, (Malmsten et al., 2020). Malmsten et al. (2020) showed that KB-BERT outperformed the multilingual version of the BERT model.

As discussed in a review paper by Stanfill et al. (2010), it is difficult to compare previous ICD classification approaches since previous studies using the techniques apply them differently. Previous studies use different label sets, evaluate the classifiers' performance differently, and use texts written in different languages. Therefore, it is favourable to investigate the alternative ICD coding methods in the context they will be used. Moreover, while many studies explore rule-based methods, traditional supervised models, and deep learning approaches to solve ICD coding tasks, few studies use Swedish data.

Henriksson et al. (2011) attempted automatic ICD classification on Swedish data using co-occurrences of words and ICD codes. Pairing clinical notes with semantically correlated ICD codes resulted in the correct ICD code being present in the top ten suggested ICD codes in 20 per cent of the cases. When considering codes at a lower level of granularity, the correct ICD codes were found in the top 10 suggested codes in 77 per cent of the cases. Optimising the dimensionality improved these results by 18 percentage points (Henriksson and Hassel, 2013).

This study explores conventional supervised learning methods and the deep learning Swedish model KB-BERT for Swedish ICD classification. The F_1-micro is used to evaluate the different approaches. The F_1-macro is also presented. A discussion of the results follows, addressing the implications of the study.

3 Methodology

3.1 Data

3.1.1 ICD Codes

The ICD system is hierarchical, and at the highest level of granularity, the letter initiating each code is followed by three digits. In Figure 1, the anatomy of ICD codes is displayed.

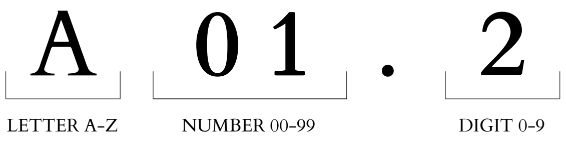

Figure 1: The anatomy of ICD codes.

Health personnel assign three-digit codes like the one exemplified in Figure 1 to the patient records, and a clinical coding tool would benefit from suggesting these full codes. However, since many codes only have a few associated patient records in the data available for this study, it might not be successful to train models to predict full codes. Therefore, to give the models a fair chance to perform and, thereby, be compared, both full codes at the highest granularity level and codes grouped at a higher level are considered in this article.

The grouped codes considered are ICD codes at a two-digit level, which are known as ICD blocks. Using full ICD codes can illustrate the implications of training models when having many classes, where many of the classes have few instances. On the other hand, using ICD codes at the block level can demonstrate the implications of solving an ICD classification task at a lower level of granularity with fewer classes and more instances per class.

This paper is delimited to ICD codes related to gastrointestinal diseases. The digestive diseases reside in ICD Chapter XI, containing codes starting with a K. Chapter XI consists of ten blocks. In Table 1, the two-digit ICD codes included in each block in ICD Chapter XI and descriptions of the diseases that the blocks cover are presented.

ICD Block	Description of diseases
K00-K14	Diseases of the oral cavity, salivary glands, jaws
K20-K31	Diseases of oesophagus, stomach, duodenum
K35-K38	Diseases of appendix
K40-K46	Hernia
K50-K52	Noninfective enteritis, colitis
K55-K64	Other diseases of intestines
K65-K67	Diseases of peritoneum
K70-K77	Diseases of the liver
K80-K87	Disorders of gallbladder, biliary tract, pancreas
K90-K93	Other diseases of the digestive system

Table 1: ICD blocks of Chapter XI.

3.1.2 Multi-label Text Classification

A hospitalised patient seldom suffers from only one disease. On the contrary, one patient can have many diagnoses, implying that one discharge summary often is paired with multiple ICD codes. Pairing one text with numerous labels is a multi-label classification task, which is different from multi-class tasks where the labels are mutually exclusive. In Figure 2, an exemplary clinical note with multiple assigned ICD codes is presented.

Discharge summary

Tidigare helt frisk kvinna med obehag i epigastrium och tilltagande smärta i arcus under 4 dagar. Konstaterat diafragmabråck. Beh för misstänkt gastroenterit utan framgång. CT visade tecken på akut kolecystit och operation genomfördes med framgång. Pat hemskickad med råd att vila i minst 2 v. Fettsnål kost och mindre portioner rekommenderas.

English translation: Previously completely healthy woman feeling discomfort in epigastrium with increasing pain in arcus for 4 days. Confirmed diaphragmatic hernia. Unsuccessfully treatm for suspected gastroenteritis. CT showed signs of acute cholecystitis. Successful operation. Pat sent home to rest for 2 w min. Low fat diet and smaller portions recommended.

Assigned ICD codes

K44.9 Diaphragmatic hernia, no obstruction or gangrene
K80.4 Acute cholecystitis

Figure 2: A partly made up and completely pseudonymised exemplary discharge summary.

3.1.3 The ICD-10 Corpus

The data used in this study is called the Stockholm EPR Gastro ICD-10 Corpus version 2 (ICD-10 Corpus)[1]. The ICD-10 Corpus resides in the research infrastructure Health Bank – the Swedish Electronic Health Record Bank[2] which is located at the Department of Computer and Systems Sciences (DSV) at Stockholm University. Health Bank contains over 2 million electronic patient records from over 500 clinical units at Karolinska University Hospital in Stockholm between 2007 and 2014. The ICD-10 Corpus was extracted from Health Bank and consists of discharge summaries from four gastrointestinal care units.

In the ICD-10 Corpus, there are only ICD codes representing digestive diseases, which is ICD Chapter XI, containing codes starting with a K (see Table 1). Moreover, the discharge summaries were filtered to those containing more than three tokens, and discharge summaries belonging to the same patient and care period assigned the same ICD codes were merged into one discharge summary.

In Table 2, the number of discharge summaries, patients, tokens, unique tokens, full ICD codes, and ICD blocks of the ICD-10 Corpus are presented. Descriptive statistics of the number of tokens per discharge summary are available in Table 3.

[1] This research has been approved by the Regional Ethical Review Board in Stockholm under permission no. 2007/1625-31/5.

[2] See the Health Bank website (`https://dsv.su.se/healthbank`) and Dalianis et al. (2015) for more information.

Number of discharge summaries	6 062
Number of unique patients	4 985
Total number of tokens	986 436
Number of unique tokens (vocabulary)	48 232
Number of unique full ICD codes	263
Number of unique ICD blocks	10

Table 2: Basic characteristics of the ICD-10 Corpus.

Min	4
Median	134
Mean	162.7
Max	1794
Std	120.5

Table 3: Number of tokens per discharge summary in the ICD-10 Corpus.

Since both full ICD codes and ICD codes grouped at the block level are considered in this paper, the descriptive statistics of the number of ICDs per discharge summary for each of the data sets are presented in Table 4. The version of the ICD-10 Corpus with full ICD codes is denoted **Full codes**, and the version of the ICD-10 Corpus with ICD codes at the block level is denoted **Blocks**.

	Full codes	**Blocks**
Min	1	1
Median	1	1
Mean	1.2	1.2
Max	6	4
Std	0.5	0.4

Table 4: Number of ICDs per discharge summary in the **Full Codes** and the **Blocks** data sets.

3.2 Models

3.2.1 Baseline

A common approach to solving ICD classification tasks is using traditional supervised learning models. The traditional supervised learning models used in this study were *Support Vector Machines*, *Decision Trees*, and *K-nearest Neighbours*. These models were chosen since they are well-established and frequently used in related studies.

The implementations of Decision Trees and K-Nearest Neighbours used were the `DecisionTreeClassifier` class from the Scikit-learn library (Pedregosa et al., 2011) and the `MLkNN` class from the Sckikit-multilearn library (Szymański and Kajdanowicz, 2018), respectively. Since the Scikit-learn implementation of Support Vector Machines (class `SVC`) is not directly suitable for multi-label data, one classifier per label was trained using the Scikit-learn implementation of one-vs-rest (class `OneVsRestClassifier`). The default hyper-parameters were used.

For the baseline models to handle text input, the text has to be represented as numerical features. For this purpose, *tf-idf* weights as they are implemented in the Scikit-learn class `TfidfVectorizer` were used. *tf-idf* is short for term frequency-inverse document frequency and represents how important a word is in a specific document, compared to the importance of that word in all documents. Basic pre-processing steps in the form of removal of punctuation and stop words and de-capitalisation were also conducted. The list of Swedish stop words was taken from the Natural Language Toolkit (NLTK) (Bird et al., 2009).

3.2.2 KB-BERT

KB-BERT was used closely following the instructions in Devlin et al. (2019) for downstream tasks and fine-tuning. More specifically, the architecture takes advantage of the presence of a special token, namely the [CLS] (classification) token representation, used initially for the NSP (**N**ext **S**entence **P**rediction) task. This representation is utilised as sentence representation and is assumed to contain information describing each instance. The architecture of the KB-BERT classifier includes at its core the KB-BERT model (bert-base-swedish-cased)[3] from which the [CLS] representation is used for each sample through a ReLU (**R**ectified **L**inear **U**nit) transformation as input to a fully connected classification layer.

In the spirit of Devlin et al. (2019), a minimal learning rate in the magnitude of $2 \cdot 10^{-5}$ was used. The number of warm-up steps was set to 155 for the model to approximately see all the data before the learning rate starts decaying. Due to memory constraints, the batch size of 32 was achieved using a batch size of 2 and a gradient accumulation of 16. The activation threshold binarising the floating numbers the KB-BERT outputs was set to the standard value of 0.5. Adam was used as the optimiser. The implementation of the KB-BERT classifier was done using the Transformers (Wolf et al., 2020), and Pytorch (Paszke et al., 2019) libraries.

[3] `https://huggingface.co/KB/bert-base-swedish-cased`

3.3 Experiment Design

To test how well the *KB-BERT, Support Vector Machines, Decision Trees*, and *K-Nearest Neighbours* perform in pairing Swedish discharge summaries with the correct ICD codes, 90 per cent of the data was used for training the models. 10-fold cross-validation was utilised to get more reliable estimates of the models' performance and to be able to test if the observed differences in classifier performance are statistically significant. The final performance of the KB-BERT and the best performing baseline model was estimated by training on all training data and testing on the 10 per cent of the data (the held-out set) not used for comparing the classifiers.

Performance was represented by the F_1-micro score. Micro averaging was chosen over macro averaging since it was considered of greater interest to train a classifier that correctly can classify as many discharge summaries as possible, rather than as many ICD codes as possible. However, macro averaged scores are presented as well.

The Wilcoxon signed-rank test (Wilcoxon, 1945) suitable for small dependent samples was used to test the statistical significance of classifier performance. The null hypotheses that the distribution of the F_1-micro scores are equal were tested against the alternative hypotheses that the distribution of the F_1-micro scores are not equal for the compared classifiers. The significance level was set to 0.01.

4 Results

4.1 Full ICD codes

The combined macro and micro averaged *Precision (P), Recall (R)*, and *F_1-score (F_1)* of the KB-BERT and the baseline models during the 10-fold cross-validation when training the models using the **Full codes** version of the ICD-10 Corpus (263 ICD-10 codes) are presented in Table 5.

	Macro			*Micro*		
Classifier	*P*	*R*	*F_1*	*P*	*R*	*F_1*
KB-BERT	0.00	0.00	0.00	0.00	0.00	0.00
SVM	0.06	0.01	0.01	0.85	0.05	0.10
DT	0.10	0.09	0.09	0.30	0.28	0.29
KNN	0.11	0.03	0.05	0.55	0.17	0.26

Table 5: Combined scores for the **Full codes** data set during the 10-fold cross-validation.

Overall, the results were poor. All models underperformed, and the best performing model for the full codes was the Decision Trees, achieving an F_1-micro of 0.29 and an F_1-macro of 0.09. The KB-BERT failed to perform at all, obtaining F_1-scores of zero.

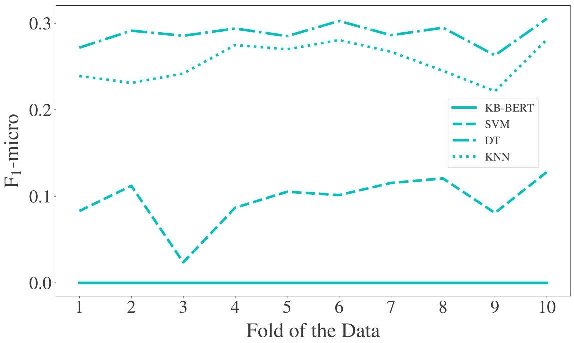

Figure 3: F_1-micro for the **Full codes** data set during each fold of the data.

As becomes evident from Figure 3, the classifier ranks remained constant during the 10-fold cross-validation, resulting in the smallest possible Wilcoxon test statistic, 0, for all pairwise comparisons of classifiers. This means that there is reason to trust that the differences in classifier performance observed in Table 5 were not only due to chance but reflect actual characteristics of the data set.

For the KB-BERT, early stopping was used during the 10-fold cross-validation, and convergence was achieved with respect to the **B**inary **C**ross-**E**ntropy (BCE) loss function at ten epochs during most runs. Therefore, when training the KB-BERT on all of the training data and testing it on the **held-out test set**, it was trained for ten epochs. Still, the **KB-BERT** failed to perform and obtained an **F_1-micro and F_1-macro of zero**. When training the best-achieving baseline classifier, the **Decision Trees**, on the full training set and testing it on the held-out test set, it achieved an **F_1-micro of 0.31** and an **F_1-macro of 0.09**.

4.2 ICD Blocks

For the **Blocks** version of the ICD-10 Corpus, comparing the F_1-micro of the KB-BERT with the baseline models during the 10-fold cross-validation, the KB-BERT was superior to the baseline models. The Support Vector Machines was the baseline model with the highest F_1-micro. Macro and micro averaged Precision (P), Recall (R), and F_1-score (F_1) of the KB-BERT and the baseline models during the ten folds are presented in Table 6.

	Macro			*Micro*		
Classifier	*P*	*R*	*F_1*	*P*	*R*	*F_1*
KB-BERT	0.67	0.55	0.60	0.87	0.77	0.82
SVM	0.76	0.33	0.41	0.90	0.61	0.72
DT	0.54	0.50	0.52	0.72	0.69	0.71
KNN	0.63	0.41	0.48	0.79	0.64	0.71

Table 6: Combined scores for the **Blocks** data set during the 10-fold cross-validation.

Looking at Figure 4, one can see that, as was the case with the **Full codes** version of the data set, the classifier ranks comparing the KB-BERT and each baseline classifier remained intact throughout the 10-fold cross-validation for the **Blocks** version of the data set. This implies that these Wilcoxon test statistics were 0 and that the observed differences between the KB-BERT and the baseline classifiers are likely to reflect that the KB-BERT and the baseline models perform differently on this data set. However, unlike in the case with full codes, the baseline classifiers are not statistically distinguishable.

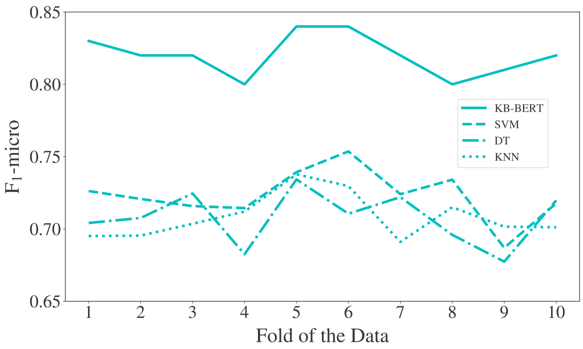

Figure 4: F_1-micro for the **Blocks** data set during each fold of the data.

Compared to the **Full codes** version of the data set, when running the KB-BERT using the **Blocks** version of the data set, KB-BERT was trained for seven epochs until the BCE loss converged. Therefore, KB-BERT was trained for seven epochs when trained on the full training set and tested on the **held-out set**. For this final evaluation, the **KB-BERT** obtained an **F_1-micro of 0.80** and an **F_1-macro of 0.58**. When training the baseline model that obtained the highest F_1-micro score during the 10-fold cross-validation, the **Support Vector Machines**, on all of the training data and testing it on the held-out data set, it reached an **F_1-micro of 0.71** and an **F_1-macro of 0.42**.

5 Discussion

5.1 KB-BERT

The results showed that at a block level, the state-of-the-art BERT model trained on Swedish text, KB-BERT, has the potential to be a successful classifier used in an ICD coding tool. It would be interesting to explore if the performance could be further improved by, for example, fine-tuning hyper-parameters such as the activation threshold. It would also be relevant to try other versions of BERT, for instance, a BERT pre-trained on Swedish clinical texts.

While the KB-BERT was the best performing classifier on the block level, it failed to classify full codes. One explanation for this could be that deep learning models are more data-hungry than traditional supervised machine learning models, meaning the KB-BERT suffers the most from moving from ICD blocks to less frequent full codes.

It should also be noted that, like other deep learning models, KB-BERT takes substantially longer to train than the baseline models, resulting in a larger carbon footprint. For example, when considering codes at the block level, it took 300 minutes for a GPU to train (fine-tune) and test the KB-BERT using 10-fold cross-validation. The corresponding number for the slowest baseline model, the Support Vector Machines, was 22 minutes on a regular laptop computer, meaning the actual training time difference probably is even greater than our estimations. While one may argue that the prediction time matters the most, training time might still matter if the idea is that the ICD coding tool should keep learning with time.

5.2 Code Frequencies, Granularity, and Combinations

One finding that stands out is the difference between the classifiers' performance when considering the 263 full ICD and ICD codes grouped into ten blocks. Comparing the results in Section 4.1 and Section 4.2, the best F_1-micro changes from 0.31 to 0.80 when going from full codes to block codes.

There are several possible explanations for the great difference between the results at a full code level and a block level. Firstly, many of the full codes have very few associated discharge summaries, meaning there are few examples to learn from. Some codes only have one associated discharge summary, leaving no instances to test on, which leads to F_1-scores of zero. In turn, having many low-frequency codes explains the discrepancy between F_1-micro and F_1-macro scores.

Secondly, as Blanco et al. (2020) suggest, the granularity itself can be a predictor of performance. This means that going from full codes to codes on the block level could have a greater impact than decreasing the number of possible label combinations. Of course, the number of possible label combinations itself also could have impacted the results. One way to address the difficulties associated with ICD coding at the full code level is to combine KB-BERT with the per-label attention mechanism proposed in the article by Blanco et al. (2021).

5.3 Generalisability

One should note that this study's classifier comparison only is valid for the specific discharge summaries used and that they might not represent Swedish gastrointestinal discharge summaries in general. For example, since the discharge summaries were written between 2007 and 2014, it may be the case that the writing style has changed since. Moreover, the four units that the discharge summaries were created at may not represent Swedish gastrointestinal care units in general.

Furthermore, the results are conditional on the specific instantiations of the classifiers used, and both the KB-BERT and the baseline models may have benefited from hyper-parameter optimisation.

The results of this study are also difficult to compare with the results from other related research since the data sets used differs, and they are often not publicly available because of privacy reasons.

5.4 ICD Coding Tool

This research's long-term goal is to develop a Swedish ICD coding tool to use in health facilities. Since health personnel assign full codes to the patient records, it would be favourable if the tool suggests full codes. Therefore, it would be suitable for future work to improve the results of this study obtained for codes at the highest level of granularity.

Moreover, it would be interesting to explore how such a tool could incorporate explainability mechanisms. Explainability could be used both as a measure to make the tool trustworthy and to help the coder decide among the suggested codes.

Furthermore, a coding tool would benefit from being developed in close contact with the end-users. Therefore, a design science study with an iterative research approach would be reasonable. In such a study, classifier requirements, such as the desired trade-off between precision and recall, could be discussed.

6 Final Remarks

6.1 Summary

To summarise, the KB-BERT outperformed the baseline classifiers when the full ICD codes of the ICD-10 Corpus were grouped into ten ICD blocks, achieving an F_1-micro of 0.80 and an F_1-macro of 0.58. These results can be compared to the baseline classifier with the highest F_1-micro, the Support Vector Machines, which reached an F_1-micro of 0.71 and an F_1-macro of 0.42. When considering the 263 full codes, the KB-BERT could not perform at all, obtaining zero F_1-micro and F_1-macro. For the data set with the full codes, the best performing classifier was the Decision Trees that reached an F_1-micro of 0.31 and an F_1-macro of 0.09.

The discrepancy between the results when considering full ICD codes and codes at a block level can partly be because many full codes had very few associated discharge summaries (low frequency). Furthermore, the granularity itself could have impacted the results since distinguishing one full code to another very similar full code is different from distinguishing one ICD block to another, not that similar, ICD block. The fact that one of the tasks had more than 20 times more labels than the other could also have influenced the results.

6.2 Conclusion

In conclusion, this paper contributed to the insufficient knowledge about performing ICD classification on a Swedish corpus by exploring how different classifiers solved a Swedish ICD classification task. One main finding is that KB-BERT showed great potential in predicting few high-frequency ICD codes grouped at the block level. Another main result is that the classifiers, especially the data-hungry KB-BERT, struggled when considering many low-frequency, finely-grained codes.

Since it is desirable for an ICD tool to suggest full codes rather than grouped codes, future work on Swedish ICD classification should focus its efforts on training models that perform well in predicting full codes. One recommendation is to get hold of data with many training examples per each full ICD code to achieve this goal, thereby avoiding the low-frequency issue. Moreover, future studies would benefit from looking into different ways for classifiers to handle a great amount of finely-grained ICD codes.

Acknowledgments

This work was partially funded by the Norwegian Research Council through the ClinCode project, number 318098.

References

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B A McDermott. 2019. Publicly Available Clinical BERT Embeddings. *arXiv:1904.03323 [cs]* http://arxiv.org/abs/1904.03323.

Saadullah Amin, Günter Neumann, Katherine Dunfield, Anna Vechkaeva, Kathryn Annette Chapman, and Morgan Kelly Wixted. 2019. MLT-DFKI at CLEF eHealth 2019: Multi-label classification of ICD-10 codes with BERT. In *CEUR Workshop Proceedings*. CEUR-WS, volume 2380.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc. https://books.google.se/books?id=KGIbfiiP1i4C.

Brent Biseda, Gaurav Desai, Haifeng Lin, and Anish Philip. 2020. Prediction of ICD Codes with Clinical BERT Embeddings and Text Augmentation with Label Balancing using MIMIC-III. *arXiv:2008.10492 [cs]* http://arxiv.org/abs/2008.10492.

Alberto Blanco, Olatz Perez-de Viñaspre, Alicia Pérez, and Arantza Casillas. 2020. Boosting ICD multi-label classification of health records with contextual embeddings and label-granularity. *Computer Methods and Programs in Biomedicine* 188:105264. https://doi.org/10.1016/j.cmpb.2019.105264.

Alberto Blanco, Sonja Remmer, Alicia Pérez, Hercules Dalianis, and Arantza Casillas. 2021. On the contribution of per-ICD attention mechanisms to classify health records in languages with fewer resources than English. *In the Proceedings of Recent Advances in Natural Language Processing, RANLP 2021, Varna, Bulgaria* .

Svetla Boytcheva. 2011. Automatic Matching of ICD-10 codes to Diagnoses in Discharge Letters. In *Proceedings of the Second Workshop on Biomedical Natural Language Processing*. Association for Computational Linguistics, Hissar, Bulgaria, pages 11–18. https://www.aclweb.org/anthology/W11-4203.

Hercules Dalianis, Aron Henriksson, Maria Kvist, Sumithra Velupillai, and Rebecka Weegar. 2015. HEALTH BANK - A Workbench for Data Science Applications in Healthcare. *CEUR Workshop Proceedings Industry Track Workshop* pages 1–18. http://ceur-ws.org/Vol-1381/paper1.pdf.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]* http://arxiv.org/abs/1810.04805.

Richárd Farkas and György Szarvas. 2008. Automatic construction of rule-based ICD-9-CM coding systems. *BMC Bioinformatics* 9(3):S10. https://doi.org/10.1186/1471-2105-9-S3-S10.

Mehedi Hasan, Alexander Kotov, April Idalski Carcone, Ming Dong, Sylvie Naar, and Kathryn Brogan Hartlieb. 2016. A study of the effectiveness of machine learning methods for classification of clinical interview fragments into a large number of categories. *Journal of Biomedical Informatics* 62:21–31. https://doi.org/10.1016/j.jbi.2016.05.004.

Aron Henriksson and Martin Hassel. 2013. Optimizing the Dimensionality of Clinical Term Spaces for Improved Diagnosis Coding Support. *Proceedings of the 4th International Louhi Workshop on Health Document Text Mining and Information Analysis (Louhi 2013)* .

Aron Henriksson, Martin Hassel, and Maria Kvist. 2011. Diagnosis Code Assignment Support Using Random Indexing of Patient Records – A Qualitative Feasibility Study. In Mor Peleg, Nada Lavrač, and Carlo Combi, editors, *Artificial Intelligence in Medicine*. Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, pages 348–352. https://doi.org/10.1007/978-3-642-22218-4_45.

Anders Jacobsson and Lisbeth Serdén. 2013. Kodningskvalitet i patientregistret (In Swedish). https://www.socialstyrelsen.se/globalassets/sharepoint-dokument/artikelkatalog/statistik/2013-3-10.pdf.

Rajvir Kaur and Jeewani Anupama Ginige. 2018. Comparative Analysis of Algorithmic Approaches for Auto-Coding with ICD-10-AM and ACHI. *Studies in health technology and informatics* 252:73–79.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36(4):1234–1240. https://doi.org/10.1093/bioinformatics/btz682.

Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. Playing with Words at the National Library of Sweden – Making a Swedish BERT. *arXiv:2007.01658 [cs]* http://arxiv.org/abs/2007.01658.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward

Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *arXiv:1912.01703 [cs, stat]* http://arxiv.org/abs/1912.01703.

Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12(85):2825–2830. http://jmlr.org/papers/v12/pedregosa11a.html.

John P Pestian, Chris Brew, Pawel Matykiewicz, D J Hovermale, Neil Johnson, K Bretonnel Cohen, and Wlodzislaw Duch. 2007. A shared task involving multi-label classification of clinical free text. In *Biological, translational, and clinical language processing*. Association for Computational Linguistics, Prague, Czech Republic, pages 97–104. https://www.aclweb.org/anthology/W07-1013.

Socialstyrelsen. 2018. Klassifikationen ICD-10 (In Swedish). Accessed 2021-08-18. https://www.socialstyrelsen.se/utveckla-verksamhet/e-halsa/klassificering-och-koder/icd-10/.

Mary H Stanfill, Margaret Williams, Susan H Fenton, Robert A Jenders, and William R Hersh. 2010. A systematic literature review of automated clinical coding and classification systems. *Journal of the American Medical Informatics Association* 17(6):646–651. https://doi.org/10.1136/jamia.2009.001024.

Piotr Szymański and Tomasz Kajdanowicz. 2018. A scikit-based Python environment for performing multi-label classification. *arXiv:1702.01460 [cs]* http://arxiv.org/abs/1702.01460.

WHO. 2016. International Classification of Diseases (ICD). Accessed 2021-04-14. http://www.who.int/classifications/icd/en/.

Frank Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1(6):80. https://doi.org/10.2307/3001968.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv:1910.03771 [cs]* http://arxiv.org/abs/1910.03771.