Automated Coding of Medical Diagnostics from Free-Text: the Role of Parameters Optimization and Imbalanced Classes

Luiz Virginio[1] [0000-0002-5282-7451], Julio Cesar dos Reis[1] [0000-0002-9545-2098]

[1] University of Campinas, Campinas, São Paulo, Brazil
luiz.virginio@gmail.com, jreis@ic.unicamp.br

Abstract. The extraction of codes from Electronic Health Records (EHR) data is an important task because extracted codes can be used for different purposes such as billing and reimbursement, quality control, epidemiological studies, and cohort identification for clinical trials. The codes are based on standardized vocabularies. Diagnostics, for example, are frequently coded using the International Classification of Diseases (ICD), which is a taxonomy of diagnosis codes organized in a hierarchical structure. Extracting codes from free-text medical notes in EHR such as the discharge summary requires the review of patient data searching for information that can be coded in a standardized manner. The manual human coding assignment is a complex and time-consuming process. The use of machine learning and natural language processing approaches have been receiving an increasing attention to automate the process of ICD coding. In this article, we investigate the use of Support Vector Machines (SVM) and the binary relevance method for multi-label classification in the task of automatic ICD coding from free-text discharge summaries. In particular, we explored the role of SVM parameters optimization and class weighting for addressing imbalanced class. Experiments conducted with the Medical Information Mart for Intensive Care III (MIMIC III) database reached 49.86% of f1-macro for the 100 most frequent diagnostics. Our findings indicated that optimization of SVM parameters and the use of class weighting can improve the effectiveness of the classifier.

Keywords: Automated ICD Coding, Multi-label Classification, Imbalanced Classes.

1 Introduction

The Electronic Health Records (EHRs) are becoming widely adopted in the healthcare industry [1]. EHR is a software solution used to register health information about patients, as well as to manage health organizations activities for medical billing and even population health management. The data entered in the EHR usually contain both structured data (patient demographics, laboratory results, vital signs, etc.) and unstructured data (free-text notes).

Most of the records in an EHR are textual documents such as progress notes and discharge summaries entered by health professionals who attended the patient. Discharge summary is a free-text document that is recorded in the moment of patient discharge. It describes the main health information about a patient during his/her visit to a hospital and provides final diagnosis, main exams, medication, treatments, etc.. These unstructured data inserted as free text have the advantage of giving greater autonomy to health professionals for registering clinical information, but it entails issues for automatic data analysis [2].

In this scenario, extracting codes from EHR based on terminologies and standard medical classifications is an important task because the codes can be used for different purposes such as billing and reimbursement, quality control, epidemiological studies, and cohort identification for clinical trials [3]. Diagnosis coding, for example, is used not only for reporting and reimbursement purposes (in US, for example), but for research applications such as tracking patients with sepsis [4].

Usually, several EHR records are encoded in a standardized way by terminologies such as the International Classification of Diseases (ICD)[1] which is a taxonomy of diagnostic codes organized in a hierarchical structure. ICD codes are organized in a rooted tree structure, with edges representing is-a relationships between parents and children codes. More specifically, the ICD-9 contains more than 14 thousand classification codes for diseases. Codes contain three to five digits, where the first three digits represent disease category and the remaining digits represent subdivisions. For example, the disease category "essential hypertension" has the code 401, while its subdivisions are 401.0 - Malignant essential hypertension, 401.1 - Benign essential hypertension, and 401.9 - Unspecified essential hypertension.

Extracting codes from EHR textual documents requires the review of patient data searching for information that can be coded in a standardized manner. For example, evaluate discharge summary to assign ICD codes. Trained professional coders review the information in the patient discharge summary and manually assign a set of ICD codes according to the patient conditions described in the document [5]. However, assigning diagnosis codes performed by human coders is a complex and time-consuming process. In practical settings, there are many patients and the insertion of data and coding process require software support to be further effective.

Several proposals have been conducted to attempt automating the ICD coding process (e.g., [3][6][10]). A study conducted by Dougherty *et al.* showed that an ICD coding process assisted by an auto-coding improved coder productivity by over 20% on inpatient documentation [11]. Therefore, an automated system can help medical coders in the task of ICD coding and, consequently, reduce costs. However, this task has been shown to be a very challenging problem, especially because of the large number of ICD codes and the complexity of medical free-text [12].

According to our literature review, several research challenges remain opened in this direction. Medical free-text is difficult to be handled by machine learning approaches because misspellings and not unstandardized abbreviations often compromise their quality [13]. Besides, automated ICD coding is characterized to present

[1] http://www.who.int/classifications/icd/en/

aspects that negatively affect effectiveness such as large labels set, class imbalance, inter-class correlations, and large feature sets [10]. Despite these challenges, machine learning approaches for automated coding are very promising because the model is automatically created from training data, without the need of human intervention.

In this paper, we aim to construct a model based on machine learning approaches for automatic ICD coding from free-text discharge summaries. In particular, we investigate the role of SVM parameters optimization and class weighting for imbalanced class addressing. In a machine learning perspective, a free-text sample could be considered as an instance in which one or more ICD codes can be assigned. It means that ICD codes (labels) are not exclusive and, therefore, a discharge summary can be labeled as belonging to multiple disease classes. That scenario is known as a multi-label classification task. In this work, we address multi-label classification problems into several multi-class, where each sample belongs to a single class. The results presented in our experimental study have shown that considering parameter values searching and the use of class weighting can bring improvements to the automatic coding task.

This article is organized as follows: Section 2 presents the related work. Section 3 introduces our experimental design. Then, Section 4 reports on our obtained results and discusses the findings. Section 5 presents the final considerations.

2 Related Work

Two approaches are usually explored in automated coding task of medical text: (i) Information Retrieval (IR) of codes from a dictionary; and (ii) machine learning or rule-based Text Classification (TC). In the first approach, an IR system is used to allow professional coders to search for a set of one or more terms in a dictionary [14]. TC approaches have been receiving an increasing attention in the task of medical text coding.

Several studies have proposed models for ICD coding and their methods ranged from manual rules to online learning. The best results for classification accuracy have been achieved by rules-based systems [15] in which hand-crafted expert rules are created. Nevertheless, these methods may be very time-consuming due to the necessity of creating hand-craft expert rules for all ICD codes.

Machine learning approaches are very promising because the model is automatically created from training data, without the need of human intervention. A literature review conducted by Stanfill et al. [16] concluded that most of studies presenting reliable results are inserted in controlled settings, often using normalized data and keeping a limited scope. For example, Zhang et al. [17] used SVMs and achieved a F1 score of 86.6%. However, they used only radiology reports with limited ICD-9 codes.

Perotte et al. [5] proposed the use of a hierarchy-based Support Vector Machines (SVM) model in the task of automated diagnosis code classification. The tests were conducted over the Medical Information Mart for Intensive Care (MIMIC II) dataset. The authors considered two different approaches for predicting ICD-9 codes: Flat SVM and hierarchy-based SVM. The flat SVM treated each ICD-9 code independent-

ly of each other whereas hierarchy-based SVM leveraged the hierarchical nature of ICD-9 codes into its modeling. The best results achieved a F1 score of 39.5% with the hierarchy-based SVM.

Several theoretical studies on multi-label classification have indicated that effectively exploiting correlations between labels can benefit the multi-label classification effectiveness [13]. Subotin et al. [18] proposed a method in which a previous model is trained to estimate the conditional probability of one code being assigned to a document, given that it is known that another code has been assigned to the same document. After, an algorithm applies this model to the output of an existing statistical auto-coder to modify the confidence scores of the codes. They tested their model for ICD-10 procedure codes.

Kavuluro et al. [10] conducted experiments to evaluate supervised learning approaches to automatically assign ICD-9 codes in three different datasets. They used different problem transformation approaches with different feature selection, training data selection, classifier chaining, and label calibration approaches. For the larger dataset, they achieved F1-score of 0.57 for codes with at least 2% of representation (diagnostics that were present in at least 2% of the records). Over all codes (1231 codes), they obtained a F1-score of 0.47, even with 80% of these codes having less than 0.5% of representation. They concluded that datasets with different characteristics and different scale (size of the texts, number of distinct codes, etc.) warrant different learning approaches.

Scheurwegs et al. explored a distributional semantic model using word2vec skip-gram model to generalize over concepts and retrieve relations between them. Their approach automatically searched concepts on Unified Medical Language System (UMLS) Metathesaurus[2], an integration of biomedical terminologies, using the MetaMap[3] tool to extract named entities and semantic predications from free text. The datasets they used are in Dutch and are derived from the clinical data warehouse at the Antwerp University Hospital. They concluded that concepts derived from raw clinical texts outperform a bag-of-words approach for ICD coding.

Berndorfer and Henriksson [19] explored various text representations and classification models for assigning ICD-9 codes to discharge summaries in Medical Information Mart for Intensive Care III (MIMIC III)[4] database. For text representation, they compared two approaches: shallow and deep. The shallow representation describes each document as a bag-of-words using Term Frequency - Inverse Document Frequency (TF-IDF), while the deep representation describes the documents as a TF-IDF-weighted sum of semantic vectors that were learned using Word2Vec. The author still tested a combination strategy, in which features from the two representations are concatenated. For classification models, Berndorfer and Henriksson explored the Flat SVM and hierarchical SVM. They concluded that the best results, with F1-score of 39.25%, was obtained by combining models built using different representations.

---

2 https://www.nlm.nih.gov/research/umls/about_umls.html

3 https://metamap.nlm.nih.gov/

4 https://mimic.physionet.org/

Haoran et al. [20] used deep learning approaches to automatically assign ICD-9 codes to discharge summaries from MIMIC-III database. They achieved a F1-score of 53%. Their results were obtained by not using the entire discharge summary; their experiments only considered the sections of 'discharge diagnosis' and 'final diagnosis', where the description of patient diagnoses is found. Therefore, such approach was very dependent on the specificities of the database and presents difficulties to be generalized.

To the best of our knowledge based on the literature review, most of studies did not perform optimization of machine learning parameters. The studies have chosen the parameter values of the algorithms arbitrarily according to our interpretation. In addition, most of studies did not use approaches to address the problem of imbalanced class.

3 Materials and Methods

In this section, we present the materials and methods we used in the development of this work. We present the database used for testing and the procedure performed for model construction.

3.1 Dataset

The dataset used to extract the corpus of discharge summaries and respective ICD codes was MIMIC III. The discharge summaries correspond to 53.423 hospital admissions for adult patients between 2001 and 2012. ICD-9 was used to assign diagnosis codes to discharge diagnoses.

MIMIC III repository contains 55.177 discharge summaries and 6.985 different diagnosis codes. Only the 100 most frequent diagnostics were considered in this work. Therefore, we selected discharge summaries that had at least one of the 100 most frequent codes, resulting in 53.018 discharge summaries.

The distribution of labels among the samples is strongly imbalanced. The top three ICD-9 codes are:

- Unspecified essential hypertension (401.9) – present in 37.5% of the records
- Congestive heart failure, unspecified (428.0) – present in 23.8% of the records
- Atrial fibrillation (427.31) – present in 23.4% of the records

The hundredth most frequent ICD-9 code is "personal history of malignant neoplasm of prostate" (V10.46), which is presented in only 2% of the discharge summaries.

3.2 Procedure

We defined a pipeline to perform the classification task aiming to detect the ICD codes from the discharge summaries. Figure 1 presents the involved stages: preprocessing, dataset splitting, feature extraction, parameter search, and creation of

prediction model. The following subsection describe details of the conducted procedure.

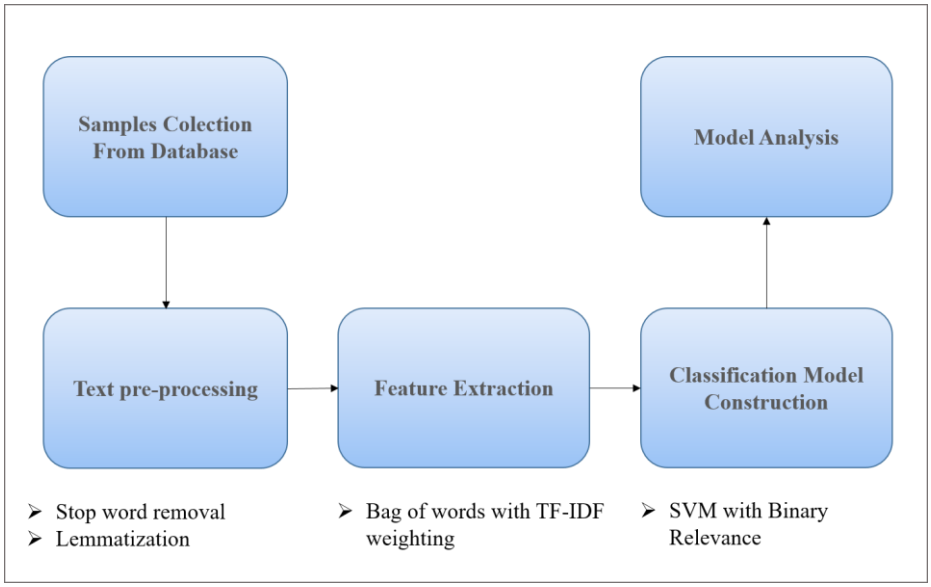

Fig. 1. Pipeline performed to construct the model for automated ICD Coding

Data Handling. We constructed our dataset extracting all discharge summaries and respective diagnosis list from MIMIC III database. Therefore, each record in the dataset consists in a discharge summary and its respective ICD-9 codes list, which is represented by a vector of 100 dimensions in which each dimension corresponds to an ICD-9 code. For a specific label in the record, if the corresponding ICD-9 code appears in the discharge summary diagnoses list, then its value in the vector is one, otherwise is zero.

For illustration purpose, Table 1 presents a sample of a record from the database. The first column represents the free-text of a discharge summary. The remaining columns represent each diagnosis code (class), where the column value is 1 or 0, depending whether the respective diagnosis was encoded for that discharge summary or not.

Table 1. Sample of a record in the dataset

Discharge Summary Text	4019 (class 1)	4280 (class 2)	…	E8782 (class 90)	V1046 (class 100)
[...] Allergies: Amlodipine Attending:[**First (LF) 898**] Chief Complaint: COPD exacerbation / Shortness of Breath Major Surgical or Invasive Procedure: Intubation arterial line placement PICC line placement Esophagogastroduodenoscopy History of Present Illness: 87 yo F with h/o CHF, COPD on 5 L oxygen at baseline, tracheobronchomalacia s/p stent, presents with acute dyspnea over several days, and lethargy. [...]	1	0	…	0	0

Dataset Splitting, Pre-processing, and Feature Extraction. Out of 53.018 discharge summaries, 80% were used for training and 20% for testing. The definition of the sets was performed in a stratified manner to maintain the proportion of classes in both sets. The training set was then used to define a vocabulary of tokens. Before tokenization, we implemented pre-processing actions expecting to improve the quality of classification and to reduce the index size of the training set. The following pre-processing tasks were performed: stop word removal, lemmatization, number removal, and special characters removal.

In stop word removal task, words that occur commonly across all the documents in the corpus are removed instead of being considered as a token. Generally, articles and pronouns are considered as stop words because they are not very discriminative. lemmatization which consists in a linguistic normalization. The variant forms of a term are reduced to a common form (lemma). The lemmatization process acts removing prefixes or suffixes of a term, or even transforming a verb to its infinitive form [21]. For stop word removal, we used the stop word removal function of the feature

extraction module of the scikit-learn[5] library. For lemmatization, we used the class WordNetLemmatizer from Natural Language Toolkit (NLTK)[6] library.

The processed discharge summaries were then tokenized using unigram and bigram with TF-IDF weighting as features. The tokens with a document frequency strictly higher than 70% or lower than 1% were ignored resulting in 12.703 tokens. In this sense, we took the decision that the vocabulary as features does not contain too-frequent or too-rare unigrams and bigrams.

Parameters Searching and Prediction Model Creation. In this study, the classification task consisted in a multi-label classification in which one or more labels are assigned to a given record from the dataset. We used the Binary Relevance method to transform the multi-label problem into several binary classification problems. Therefore, we created one classifier per ICD-9 code.

We explored the SVM algorithm. SVM has important parameters like kernel, C, and gamma, which values have to be chosen for the training task. The majority of the studies found in literature for the code assignment problem, according to our knowledge, select parameters values arbitrarily. We assume that this decision might decrease the algorithm effectiveness. In this work, we performed a parameter search step, in which the training process was performed for each possible combination of predefined parameter values. The range of values for each parameter was defined as follows:

- Parameter kernel: [Linear, Radial Basis Function (RBF)]
- Parameter C: [0.02, 0.2, 1.0, 2.0, 4.0]
- Parameter gamma: [0.02, 0.2, 1.0, 2.0, 4.0]. Applicable only to the RBF kernel.

The parameter kernel specifies whether the SVM will perform a linear or a non-linear classification. To perform a linear classification, the kernel should be 'linear' while performing a non-linear classification requires a non-linear kernel, such as RBF [22]. The parameter C is related to the size of the margin of the SVM hyperplane, where low values of C will result in a large margin and high values of C result in a small margin. The size of the margin is strongly related to misclassification, because the smaller the margin, the smaller the misclassification [22]. However, lower misclassification on training set does not implicate in lower misclassification on testing set. Therefore, a larger margin may result in a more generalized classifier. Gamma is a free parameter of the Gaussian function of the RBF kernel.

Due to the unbalance of classes, another important parameter we considered was class weight. With this parameter, it was possible to penalize mistakes on the minority class proportionally to how under-represented it is. The *initial weight* for a class was computed as N / (2 x M), in which N is the number of records and M refers to the number of records in the respective class. This formula is widely used to deal with

5 http://scikit-learn.org/stable/
6 https://www.nltk.org/

imbalanced classes in classification problems, because the lower the number of samples in a particular class, the higher is the *initial weight*.

The *initial weight* might be not enough to obtain a good effectiveness for too imbalanced classes. Therefore, besides using the *initial weight* value, we also used two higher values. We used the following range for class weight parameter: [None, *initial weight*, *initial weight* + 2, *initial weight* + 4].

According to the number of parameters and respective range of values, it was necessary to perform 90 SVM trainings (15 for linear kernel and 75 for RBF kernel). Due to computational power limitations, the parameter searching was performed in a subset corresponding to 30% of samples of the training set (12.724 samples). That subset was split in a second training set (80%) and validation (20%) set. Figure 2 illustrates the dataset splitting process.

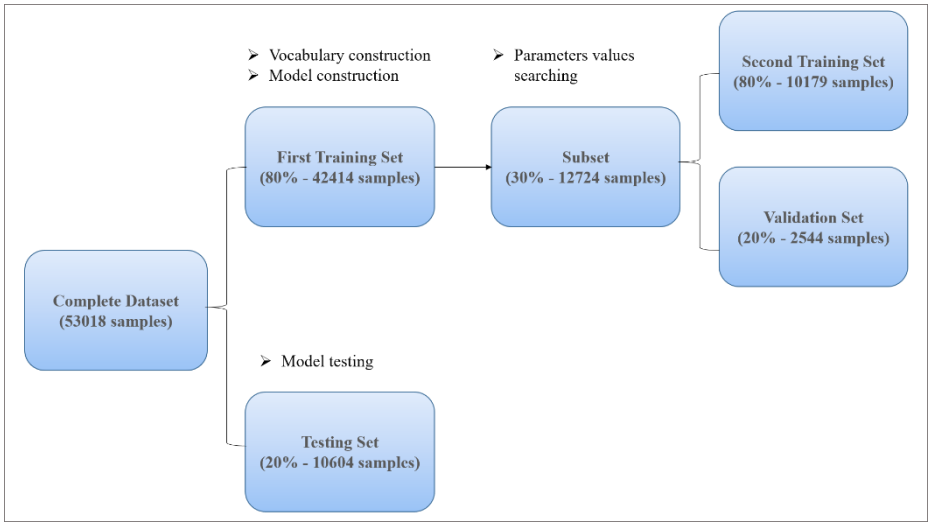

Fig. 2. Dataset splitting process

A SVM model was created for each parameter combination using the second training set. The analysis of the model was tested in the validation set through the calculation of f1-score. The parameter combination values with best results were then selected as parameter values in the creation of the prediction model. Once one model is created for each class, such values can be different for distinct classes.

After the study concerning the parameters, the creation of prediction model (for each class) was performed using the training set with 42414 records (80% of the 53018 discharge summaries). The effectiveness of the method was evaluated in the test set with 10.604 records. To this end, we explored the following evaluation metrics: recall, precision, and f1-score.

4 Results and Discussion

In this section, we present the results obtained with the construction of the classification model for ICD coding task. We highlight the influence of parameters optimization and the use of class weighting in the model construction.

4.1 Influence of Parameters and Class Weighting

After performing the searching for best combination of parameter values, we found that such values widely vary along the classes. The parameter 'C' varied between the following values: 1.0 (37 classes), 2.0 (31 classes), 0.2 (17 classes) and 4.0 (15 classes). For the 'gamma' parameter (applicable only to the RBF kernel), most classes presented the best results with the value 0.2 (52 classes), whereas five classes presented a value of 1.0 and three classes presented a value of 0.02.

For the 'kernel' parameter, 40 classes presented best results with a linear kernel, whereas 60 classes achieved better results with the RBF kernel. These results indicated the relevance of considering the RBF kernel. Usually, most studies in literature for ICD coding has approached the problem only using the linear kernel.

We addressed the problem of imbalanced classes with the use of class weighting. From 100 class in total, only two classes performed better without the need of using class weighting. These classes correspond to the diagnostics 276.8 – "Hypopotassemia" and 769 – "Respiratory distress syndrome in newborn" in ICD-9. The remaining 98 classes presented best results with the use of class weighting, highlighting the relevance of considering the class weighting as an approach to address the imbalanced class problem. According to the authors' knowledge, no other study has used this approach in literature for the studied problem.

4.2 Classifier effectiveness

As previously mentioned, we tested the effectiveness of each model using the testing set. Table 2 summarizes the obtained results. We reached 49.86% for the f1-macro metric, which represents the mean of f1-score for all classes. The mean for recall score was 68.61% and the mean for precision score was 41.94%.

Table 2. – Results summary

	Precision	**Recall**	**F1-macro**
Value	41.94%	68.61%	49.86%
Standard deviation	19.94%	14.67%	18.64%

Table 3 presents the five classes with worst f1-score whereas Table 4 presents the five classes with best f1-score. The column "frequency index" in Tables 3 and 4 represents the position of the diagnosis in the database. For example, the diagnosis 42731 – "Atrial fibrillation" is the third most frequent diagnosis, whereas 99591 – "Sepsis"

is the 92nd most frequent diagnosis. The higher the frequency index value, the lower the frequency of this diagnosis and, therefore, the more imbalanced is the respective class.

Table 3. – Five worst results and their respective classes

Diagnosis	Frequency index	Precision	Recall	F1-macro
E8788 - Other specified surgical operations and procedures causing abnormal patient reaction, or later complication, without mention of misadventure at time of operation	84	8.23%	76.40%	14.86%
27652 - Hypovolemia	81	9.91%	56.55%	16.87%
E8798 - Other specified procedures as the cause of abnormal reaction of patient, or of later complication, without mention of misadventure at time of procedure	69	13.22%	42.22%	20.14%
99591 - Sepsis	92	12.41%	68.85%	21.03%
2930 - Delirium due to conditions classified elsewhere	73	13.73%	65.14%	22.68%

Table 4. – Five best results and their respective classes

Diagnosis	Frequency index	Precision	Recall	F1-macro
42731 - Atrial fibrillation	3	84.49%	88.86%	86.62%
V3000 - Single liveborn, born in hospital, delivered without mention of cesarean section	24	83.94%	89.26%	85.52%
7742 - Neonatal jaundice associated with preterm delivery	48	76.17%	97.98%	85.71%
V3001 - Single liveborn, born in hospital, delivered by cesarean section	36	81.05%	90.31%	85.43%
V290 - Observation for suspected infectious condition	13	76.33%	93.80%	84.17%

Results indicated that the classes presenting the worst effectiveness correspond to the most imbalanced classes. This suggests that the more diagnoses we consider, the lower the effectiveness of the model (cf. Figure 3). For example, if we consider only the first 20 most frequent diagnostics, we obtain 65.43% of f1-macro against 49.86% if we consider the 100 most frequent diagnostics.

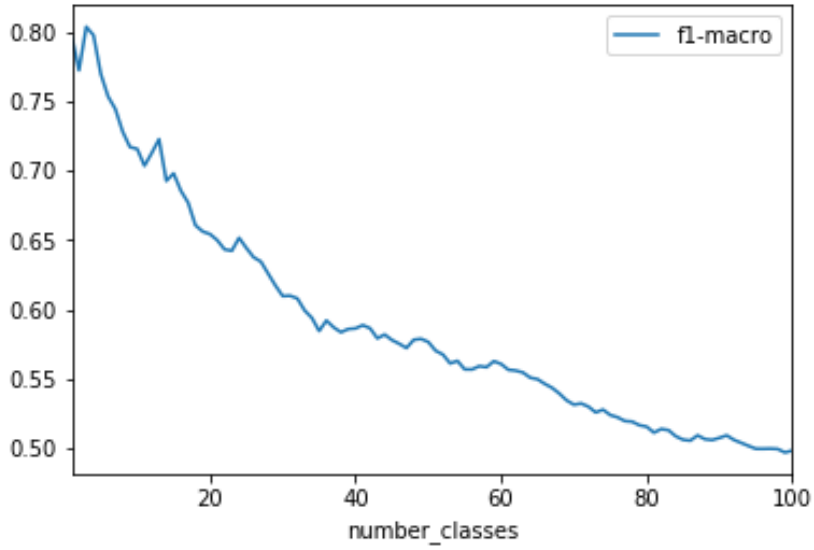

Fig. 3. – Variation of f1-macro in relation to the number of classes

5 Conclusion

In this work, we constructed a model based on machine learning approaches for the task of automated ICD coding from free-text discharge summaries. The results we obtained highlight the importance of optimization of parameter as well as the use of class weighting approach to deal with imbalanced class problem.

We also highlight some limitations of this work. The computational power restrictions limited the range of parameters values to test as well as the number of samples in the second training set used for parameter optimization. We considered only the 100 most frequent diagnostics out of 6,985 diagnostics present in the database. Therefore, the most imbalanced classes (the less frequent diagnosis) were not considered. However, it is important to note that 96.6% of the diagnostics were assigned to only 1% or less of the discharge summaries.

Another important limitation is related specifically to the database characteristics. Most of the free-text discharge summaries present misspelling and abbreviations, which may have impaired the model effectiveness. In addition, the process of manually coding itself may have errors, which may have led to incorrect or incomplete list of diagnostics.

Acknowledgements

This work is supported by the São Paulo Research Foundation (FAPESP) (Grant #2017/02325-5)[7]

References

1. B. Chaudhry, "Systematic Review: Impact of Health Information Technology on Quality, Efficiency, and Costs of Medical Care," Ann. Intern. Med., vol. 144, no. 10, p. 742, May 2006.
2. H. Navas, A. L. Osornio, A. Baum, A. Gomez, D. Luna, and F. G. B. de Quiros, "Creation and evaluation of a terminology server for the interactive coding of discharge summaries.," Stud. Health Technol. Inform., vol. 129, pp. 650–654, 2007.
3. A. Rios and R. Kavuluru, "Supervised Extraction of Diagnosis Codes from EMRs: Role of Feature Selection, Data Selection, and Probabilistic Thresholding," in 2013 IEEE International Conference on Healthcare Informatics, 2013, pp. 66–73.
4. E. Scheurwegs, K. Luyckx, L. Luyten, W. Daelemans, and T. Van den Bulcke, "Data integration of structured and unstructured sources for assigning clinical codes to patient stays," J. Am. Med. Informatics Assoc., vol. 23, no. e1, pp. 11–19, 2016.
5. A. Perotte, R. Pivovarov, K. Natarajan, N. Weiskopf, F. Wood, and N. Elhadad, "Diagnosis code assignment: models and evaluation metrics," J. Am. Med. Informatics Assoc., vol. 21, no. 2, pp. 231–237, 2014.
6. S. Hochreiter and J. Urgen Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.
7. J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," Proc. 2014 Conf. Empir. Methods Nat. Lang. Process., pp. 1532–1543, 2014.
8. S. V. S. Pakhomov, J. D. Buntrock, and C. G. Chute, "Automating the Assignment of Diagnosis Codes to Patient Encounters Using Example-based and Machine Learning Techniques," J. Am. Med. Informatics Assoc., vol. 13, no. 5, pp. 516–525, 2006.
9. L. Lefebure, "ICD-9 Coding of Discharge Summaries," AMIA Summits Transl. Sci. Proc., pp. 82–91, 2017.
10. R. Kavuluru, A. Rios, and Y. Lu, "An empirical evaluation of supervised learning approaches in assigning diagnosis codes to electronic medical records," Artif. Intell. Med., vol. 65, no. 2, pp. 155–166, 2015.
11. M. Dougherty, S. Seabold, and S. White, "Study Reveals Hard Facts on CAC," J. AHIMA, vol. 84, no. 7, pp. 54–56, 2013.
12. C. Helwe, S. Elbassuoni, M. Geha, E. Hitti, and C. Makhlouf Obermeyer, "CCS Coding of Discharge Diagnoses via Deep Neural Networks," Proc. 2017 Int. Conf. Digit. Heal. - DH '17, pp. 175–179, 2017.
13. Wang, X. Chang, X. Li, G. Long, L. Yao, and Q. Sheng, "Diagnosis Code Assignment Using Sparsity-based Disease Correlation Embedding," IEEE Trans. Knowl. Data Eng., vol. PP, no. 99, pp. 3191–3202, 2016.
14. S. G. Rizzo, D. Montesi, A. Fabbri, and G. Marchesini, "ICD Code Retrieval: Novel Approach for Assisted Disease Classification," in Lecture Notes in Computer Science (in-

[7] The opinions expressed in this work do not necessarily reflect those of the funding agencies.

cluding subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 9162, 2015, pp. 147–161.
15. R. Farkas and G. Szarvas, “Automatic construction of rule-based ICD-9-CM coding systems.,” BMC Bioinformatics, vol. 9 Suppl 3, no. Suppl 3, p. S10, Apr. 2008.
16. M. H. Stanfill, M. Williams, S. H. Fenton, R. A. Jenders, and W. R. Hersh, “A systematic literature review of automated clinical coding and classification systems,” J. Am. Med. Informatics Assoc., vol. 17, no. 6, pp. 646–651, 2010.
17. Y. Zhang, “A hierarchical approach to encoding medical concepts for clinical notes,” in Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies Student Research Workshop - HLT ’08, 2008, p. 67.
18. M. Subotin and A. R. Davis, “A method for modeling co-occurrence propensity of clinical codes with application to ICD-10-PCS auto-coding,” J. Am. Med. Informatics Assoc., vol. 23, no. 5, pp. 866–871, 2016.
19. S. Berndorfer and A. Henriksson, “Automated Diagnosis Coding with Combined Text Representations,” Stud. Health Technol. Inform., vol. 235, pp. 201–205, 2017.
20. H. Shi, P. Xie, Z. Hu, M. Zhang, and E. P. Xing, “Towards Automated ICD Coding Using Deep Learning,” pp. 1–11, 2017.
21. M. F. Porter, “An algorithm for suffix stripping,” *Program*, vol. 14, no. 3, pp. 130–137, Mar. 1980.
22. S. Haykin, *Neural networks and learning machines*, vol. 3. Pearson Upper Saddle River, NJ, USA:, 2009.