Enhancing Automatic ICD-9-CM Code Assignment for Medical Texts with PubMed

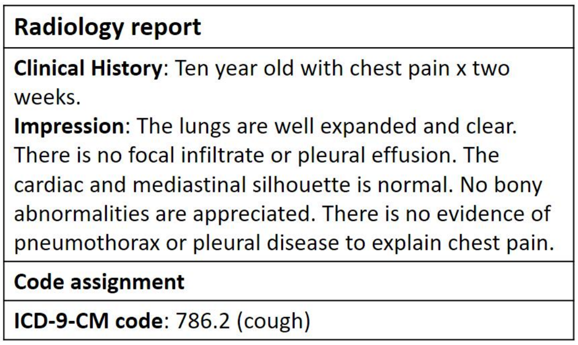
Danchen Zhang[1]**, Daqing He**[1]**, Sanqiang Zhao**[1]**, Lei Li**[2]
[1]School of Information Sciences, University of Pittsburgh
[2]School of Economics and Management, Nanjing University of Science and Technology
{daz45, dah44, saz31}@pitt.edu, leili@njust.edu.cn

Abstract

Assigning a standard ICD-9-CM code to disease symptoms in medical texts is an important task in the medical domain. Automating this process could greatly reduce the costs. However, the effectiveness of an automatic ICD-9-CM code classifier faces a serious problem, which can be triggered by unbalanced training data. Frequent diseases often have more training data, which helps its classification to perform better than that of an infrequent disease. However, a diseases frequency does not necessarily reflect its importance. To resolve this training data shortage problem, we propose to strategically draw data from PubMed to enrich the training data when there is such need. We validate our method on the CMC dataset, and the evaluation results indicate that our method can significantly improve the code assignment classifiers' performance at the macro-averaging level.

1 Introduction and Background

The rapid computerization of medical content such as electronic medical records (EMRs), doctors notes and death certificates, drives a crucial need to apply automatic techniques to better assist medical professionals in creating and managing medical information. A standard procedure in hospital is to assign the International Classification of Diseases (ICD) codes to diseases appearing in medical texts by professional coders. As a result, several recent studies have been devoted to automatically extracting ICD code from medical texts to help manual coders (Crammer et al., 2007; Farkas and Szarvas, 2008; Aronson et al., 2007; Kavuluru et al., 2015, 2013; Zuccon and Nguyen, 2013; Koopman et al., 2015).



Radiology report

Clinical History: Ten year old with chest pain x two weeks.
Impression: The lungs are well expanded and clear. There is no focal infiltrate or pleural effusion. The cardiac and mediastinal silhouette is normal. No bony abnormalities are appreciated. There is no evidence of pneumothorax or pleural disease to explain chest pain.

Code assignment

ICD-9-CM code: 786.2 (cough)

Figure 1: An example radiology report with manually labeled ICD-9-CM code from CMC dataset.

In this paper, we focus on ICD-9-CM (the 9th version ICD, Clinical Modification), although our work is portable to ICD-10-CM (the 10th version ICD). The reason to conduct our study on ICD-9-CM is to compare with the state-of-art methods, whose evaluations have mostly conducted on ICD-9-CM code (Aronson et al., 2007; Kavuluru et al., 2015, 2013; Patrick et al., 2007; Ira et al., 2007; Zhang, 2008). ICD-9-CM codes are organized hierarchically, and each code corresponds to a textual description, such as ”786.2, cough”. Multiple codes can be assigned to a medical text, and a specific ICD-9-CM code is preferred than a more generic one when both are suitable (Pestian et al., 2007). Figure 1 shows a code assignment example where a radiology report is labeled with ”786.2, cough”.

Existing methods for automatic ICD-9-CM assignment have been mostly supervised methods because of the effectiveness the training; however, classification performance heavily relies on the sufficiency of training data (He and Garcia, 2009). To certain degree, micro-average measures, commonly used to evaluate the classification performance of existing algorithms, pays at-

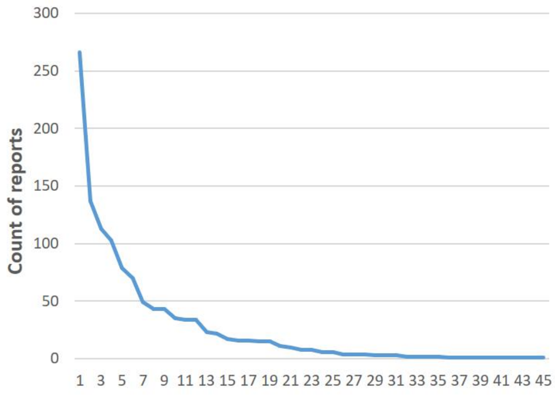


Figure 2: The distribution of radiology reports for 45 ICD-9-CM codes in the CMC dataset.

tention to the correctness of the code assignment to each EHR (individual case), which helps to hide the impact of unbalanced training data. However, a useful classification system should perform consistently across all ICD-9-CM codes regardless of the popularity of the codes (Jackson and Moulinier, 2007). This motivated us to examine the imbalanced training data and its impacts to the classifier. Specifically, we pay more attention to macro-average measures, which helps to examine the consistency across all codes.

Unfortunately, in a real dataset for studying ICD-9-CM code classification, the data available for each code is highly imbalanced. For example, Figure 2 shows the count of available radiology reports for each of the 45 ICD-9-CM codes in the CMC dataset (Pestian et al., 2007). Common diseases like ”786.2, cough”, can have 266 reports as the training data, whereas unpopular disease ”758.6, Gonadal dysgenesis” only has one. Similarly, Kavuluru et al. (2015) found that 874 of 1,231 ICD-9-CM code in their UKLarge dataset have less than 350 supporting data, and only 92 codes have more than 1,430 supporting data. In another example, Koopman et al. (2015) found that 85% of the whole death certificate dataset are related to top 20 common cancers, and only rest 15% is associated with 65 rarer cancers. These long tail supporting data problems are very common, which makes data imbalance an noticeable problem.

Our approach for resolving this problem is to introduce additional information resources. Furthermore, due to the privacy concern of medical-related content, this study is particularly interested in obtaining extra relevant training data from publicly available medical datasets. PubMed[1], as a vast and broad medical literature dataset, covers a great number of disease related information and imposes few restrictions on data access. Therefore, it is a perfect starting point to explore our approach. The hypothesis is that training data can be obtained from PubMed articles that talk about a disease corresponding to a ICD-9-CM code. With the abundant PubMed articles, we would be able to alleviate the training data imbalance problem.

There are several contributions in our study. Firstly, we examine the data imbalance problem in ICD-9-CM code assignment. Secondly, we propose and compare several methods to resolve the data imbalance problem. Thirdly, we give a comprehensive discussion on the current classification challenges. Finally, our method can be adapted to ICD-10-CM code assignment task with minor modifications.

The rest of this paper is organized as follows. In Section 2, we will discuss related research. Our methods and experiments will appear in Section 3 and 4. Limitations are discussed in Section 5 and the conclusion is provided in Section 6.

2 Related Work

The existing studies of automating ICD-9 code assignment can be classified into two groups. Through examining how professional coders assigning ICD codes, the first one used rule-based approaches. Ira et al. (2007) developed a rule-based system considering factors such as uncertainty, negation, synonymy, and lexical elements. Farkas and Szarvas (2008) used Decision Tree (DT) and Maximum Entropy (ME) to automatically generate a rule-based coding system. Crammer et al. (2007) composed a hybrid system consisting of a machine learning system with natural language features, a rule-based system based on the overlap between the reports and code descriptions, and an automatic policy system. Their results showed better performance than each single system.

The second group employed supervised machine learning methods for the assignment task, and their performance has been being equivalent or even better than those rule-based systems that need experts manually crafting knowledge. Aronson et al. (2007) used a stacked model to combine the results of four modules: Support Vector Ma-

[1] https://www.ncbi.nlm.nih.gov/pubmed/

chine (SVM), K-Nearest Neighbors (KNN), Pattern Matching (PM) and a hybrid Medical Text Indexer (MTI) system. Patrick et al. (2007) used ME and SVM classifiers, enhanced by a feature-engineering module that explores the best combination of several types of features. Zhang (2008) proposed a hierarchical text categorization method utilizing the ICD-9-CM codes structure. Zuccon and Nguyen (2013) conducted a comparison study on four classifiers (SVM, Adaboost, DT, and Naive Bayes) and different features on a 5,000 free-text death certificate dataset, and found that SVM with a stemmed unigram feature performed the best.

Along with the introduction of supervised methods, many past studies indicated that data imbalance problem can severely affect the classifier's performance. For example, Kavuluru et al. (2015) found that 874 of 1,231 ICD-9-CM codes in UKLarge dataset have less than 350 supporting data, whereas only 92 codes have more than 1,430 supporting data. The former group has macro F1 value of 51.3%, but the latter group only has 16.1%. To resolve data imbalance problem, they used optimal training set (OTS) selection approach to sample negative instance subset that provides best performance on validation set. However, OTS did not work on UKLarge dataset because several codes have so few training examples that even carefully selecting negative instances could not help. When Koopman et al. (2015) found that 85% of the whole death certificate dataset is associated with only top 20 common cancers, whereas the other 65 rarer cancers only have the rest 15% of the dataset, they tried to construct the balanced training set by randomly sampling a static number of negative examples for each class. Their results reflected the benefits of having more training data in improving the classifiers' performance. Since result of original model learned with imbalanced data is not provided, we cannot know the actual improvement. In addition, to deal with codes that only appear once in the dataset, Patrick et al. (2007) used a rule-based module to supplement ME and SVM classifiers.

Consistent to the existing works, our approach utilizes supervised methods for automatic ICD-9-CM code assignment, and our focus is on addressing the training data imbalance problem. But our work tries to solve the data imbalance problem by adding extra positive instances, which is not limited to the existing training data distribution or expert's knowledge. Adding positive instances have been proven to be effective in supervised machine learning in other domains(Caruana, 2000; He and Garcia, 2009), and we are first to find open source dataset as supplementary data for improving ICD-9-CM assignment performance.

3 Methods

In this section, we will first introduce the dataset on which our methods will be evaluated, then we propose two methods of collecting supplementary training data from PubMed dataset.

3.1 Dataset

We validate our methods on the official dataset of the 2007 Computational Medicine Challenge (CMC dataset), collected by Cincinnati Children's Hospital Medical Center (Pestian et al., 2007), which is frequently used by researchers working on the ICD-9-CM code assignment task. The CMC dataset consists of training and testing dataset, but only training dataset is accessible for us. Fortunately, most studies publish their system performance on both training and testing dataset, and then we can compare our methods with state-of-art methods. This corpus consists of 978 radiological reports taken from real medical records, and each report has been manually labeled with ICD-9-CM codes by professional companies. The example in Figure 1 comes from this dataset. In total, there are 45 ICD-9-CM codes appearing in the CMC dataset, and each report is labeled with one or more ICD-9-CM codes. This is a very imbalanced collection, with around half codes having less than 10 training data (see Figure 2).

3.2 Method I: Retrieving PubMed articles using ICD-9-CM code official description

Through examining the reports available to us, and also based on the discussions in previous work (Farkas and Szarvas, 2008; Ira et al., 2007; Crammer et al., 2007; Farkas and Szarvas, 2008), we hypothesize that the text description part of ICD-9-CM code can play important role for code assigners to build up the connection between a medical text and a ICD-9 code. Therefore, this motivated us to view the identifying extra training data in PubMed for an ICD-9-CM code as a retrieval problem where the text description part of an ICD-9-CM code can act as the query, and the whole PubMed dataset as a document collection. For example, based on ICD-9-CM code ”786.2, cough”, we can retrieve PubMed articles with a query ”cough”. Our initial informal testing confirmed our hypothesis.

To avoid bring back too much noise, we restricted the PubMed retrieval to only search on the article title field. Our motivation is that the title generally introduces the main topic of the whole paper. For the same reason, we also only utilized the title and abstract of top returned articles as the supplementary training data. In case of empty retrieval result, certain ICD-9-CM description terms that would not appear in PubMed article titles, such as ”other”, ”unspecified”, ”specified”, ”NOS”, and ”nonspecific”, are removed from the query. For example, ICD-9-CM code ”599.0”, whose description is ”urinary tract infection, site not specified”, will generate a cleaned query ”urinary tract infection”, and ICD-9-CM code ”596.54”, whose description is ”neurogenic bladder NOS”, will generate a cleaned query ”neurogenic bladder”.

3.3 Method II: Retrieving PubMed articles with both official and synonyms ICD-9-CM code description

Despite great overlap among them, ICD-9-CM code descriptions and the radiology reports in the CMC collection are written by different groups of people with different purposes. Therefore, there could be term mis-match problems between them. When this happens, it is actually better to not use the terms in the ICD-9-CM official description as the query for finding relevant PubMed articles, but actually to use the related terms in the CMC dataset as the query terms instead. This would enable the model trained on these returned PubMed articles can be more effectively classifying CMC reports. For example, the description ”Anorexia” of code ”783.0” does not appear in CMC dataset. Instead, ”loss of appetite” exists in the radiology reports labeled with ”783.0”, while according to data in ICD9Data.com, ”loss of appetite” is the synonym of ”Anorexia”. Therefore, in this case, it is better to use ”loss of appetite” rather than ”Anorexia” to be the query when search for training data in PubMed.

ICD9Data.com is an online website, providing rich and free ICD-9-CM coding information. It contains code definition, hierarchy structure, approximate synonyms, etc. We crawled the 45 codes’ synonyms from the website. In method II, besides the queries from the official description, we also conducted PubMed searches with queries based on the synonyms of the descriptions. Each synonym is an individual PubMed query, and only when all its terms appear in CMC dataset, the query is considered. If one ICDcode has n queries and totally needs m supplementary documents for training, only top m/n retrieved PubMed articles from each query are considered.

4 EXPERIMENTS

4.1 Evaluation metrics

Following the past studies (Pestian et al., 2007; Kavuluru et al., 2015), we evaluate the classification performance through a micro F1 score (i.e., sum of the individual classification performance and divided by the individual amount) and a macro F1 score (i.e., sum of the classifiers performance and divided by the classifiers amount). We expect that by alleviating the data imbalance problem, macro F1 scores can increase significantly. All experiments in this study have gone through 10-fold cross validation, because it can provide a reliable result when data is limited (Witten et al., 2016).

4.2 Pre-process and Features

Following the past studies (Crammer et al., 2007; Aronson et al., 2007; Kavuluru et al., 2015, 2013; Koopman et al., 2015; Patrick et al., 2007; Ira et al., 2007), the CMC dataset is preprocessed with following steps:

- Full name restoration. Medical abbreviation restoration is a hard topic, which is not explored in this study. We manually generate a list of full names for abbreviations appearing in CMC dataset [2].

- Word lemmatization. Lemmatization of words are restored with WordNet 3.0 (Miller, 1995).

- Negation detection and removal. Negex (Chapman et al., 2001) is used to detect the negation expression, and negation target terms are removed after detection.

[2] https://github.com/daz45/CMC-data-set-abbreviations/tree/master

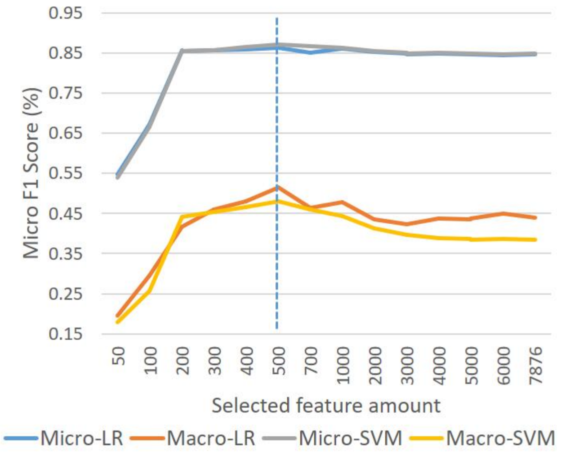


Figure 3: Feature selection on LR and SVM.

- Phrase recognition. MetaMap (Aronson and Lang, 2010) is utilized to extract the medical concept phrase appearing in the text, which is appended to the text.

After preprocessing, the example radiology report in Figure 1 will be ”ten year old with chest pain x two week. the lung be well expand and clear. there be. the cardiac and mediastinal silhouette be normal. there be. chest_pain”. Supplementary data collected from PubMed will be preprocessed in the same way.

4.3 Baselines

According to the past studies(Farkas and Szarvas, 2008; Aronson et al., 2007; Kavuluru et al., 2015), Support Vector Machine (SVM) and Logistic Regression (LR) are the most effective and commonly used classification models in this task. Therefore, we selected them as the two baselines. Each ICD-9-CM code has one binary classifier implemented using Scikit-Learn(Pedregosa et al., 2011). We name these two sets of baselines as Baseline_LR and Baseline_SVM.

Features consist of unigrams and bigrams appearing in preprocessed radiology reports, and the feature vector values are binary, indicating the appearance or absence of the word in text.

We performed feature selection on two baselines to avoid over-fit and extra computation cost. χ^2 based feature selection was employed for feature selection (Liu and Setiono, 1995). As shown in Figure 3, We find that 500 features can provide stable micro F1 performance and best macro F1 performance for Baseline_LR and Baseline_SVM. In all the following experiments, all classifiers are trained on these 500 selected features.

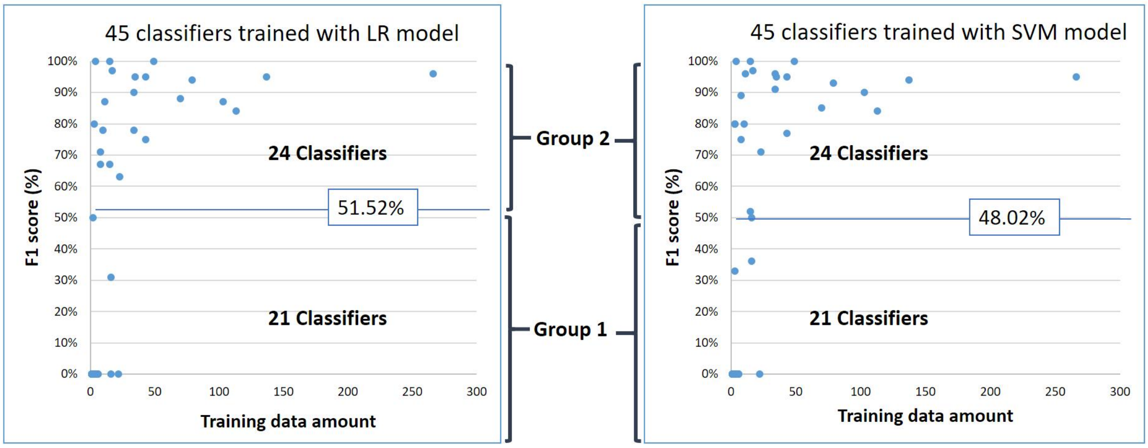
Our baseline performance were compared with the state-of-art methods in Table 1. Stacking is a stacked model combining four classification models (Aronson et al., 2007). Hybrid rule-based+MaxEnt is a hybrid system combining rule-based method with MaxEnt (Aronson et al., 2007). Although Table 1 shows that their performance is significantly better than our baselines, for the purpose of studying the methods for addressing imbalanced training data, we have to use the two current baselines since these advanced and complicated systems would hide the effects that we want to observe. In addition, any improvement we achieve in single classifier can be later integrated into these systems, which could be an interesting future work. These methods concentrated on micro averaging performance, while in this study we will explore the macro averaging performance.

Method	**Micro F1**
Baseline_LR	86.51%
Baseline_SVM	87.26%
Stacking	89.00%
Hybrid rule-based+MaxEnt	90.26%

Table 1: Baseline performance and existing best performed methods from related work.

Figure 4 shows the individual classification performance of 45 classifiers, and we can find an unstable performance across 45 classifiers. We use Macro F1 score as the split line, and we can find that, for both baseline system, there are 21 classifiers having a below-average performance, and all of them have relatively less training data than the classifiers with above-average performance. This indicates that the data imbalance leads to the performance instability across all classes.

With the Macro F1 score, we separate 45 classifiers into two groups: Group 1 consists of 24 classifiers, union set of classifiers with below-average performance in two baselines, and Group 2 consists of rest 21 classifiers with above-average classification performance. Though Group 1 has 24 classifier, radiological reports labeled with them only takes 11.56% of 978 reports. To deal with this data imbalance problem, we will introduce supplementary training data from PubMed dataset. Through adding additional data, we expect that classification performance of the whole system, especially Group 1, will be improved.



(a) (b)

Figure 4: Individual classification performance of 45 classifiers trained with LR and SVM model in Baseline.

4.4 Experiment I: retrieving PubMed articles with ICD code official description

In the first experiment, supplementary data is collected based on ICD-9-CM code official description, as described in method I. The supplementing document size is set to be 10, 20, 40 and 60. Supplementary training data is added to 24 classifiers in Group 1. We name these two new runs as Group_1_Description_LR (G1_desc_LR) and Group_1_Description_SVM (G1_desc_SVM), appended with supplementary data size. The results in Table 2 also show that supplementing 10 documents can generate best performance, and with more documents added, both macro and micro F1 will decrease.

Method	Micro F1	Macro F1
Baseline_LR	86.51%	51.52%
G1_desc_LR_10	86.68%	55.78%
G1_desc_LR_20	86.07%	55.23%
G1_desc_LR_40	85.18%	52.01%
G1_desc_LR_60	84.97%	51.40%
Baseline_SVM	87.26%	48.03%
G1_desc_SVM_10	86.96%	57.09%
G1_desc_SVM_20	86.67%	55.43%
G1_desc_SVM_40	85.87%	57.61%
G1_desc_SVM_60	86.25%	54.77%

Table 2: Enhance classifiers in Group 1 with supplementary data collected with method I, while the evaluation is performed on all classes.

Through Wilcoxon Signed Ranks test, there is no significant difference between G1_desc_LR_10 and Baseline_LR. Nor does G1_desc_SVM_10. Further, we compare both methods against baseline on Group 1 and Group 2 separately. However, there is still no significant difference existing. Take G1_desc_SVM_10 for example, from Figure 5, we can see that 11 classes still have F1=0%, while 5 classes' performance decrease, and only 8 got F1 improved. It indicates that the method I is ineffective.

After exploring the results, we find sometimes the supplementary data does not help training. For example, for ICD-9-CM code "783.0 Anorexia", the classification performance stays 0%. The corresponding radiology report doesn't have term "Anorexia", making the supplementary data useless. It implies we need to collect PubMed articles containing same features with the radiology reports in CMC dataset.

4.5 Experiment II: Retrieving PubMed articles with ICD code official and synonyms descriptions

In this second experiment, we collect PubMed data through the ICD-9-CM code's both official and synonyms description that appears in CMC dataset. We name these two runs as Group_1_Synonym_LR (G1_syn_LR) and Group_1_Synonym_SVM (G1_syn_SVM). Due to the paper size limitation, here we only show the best results with supplementary document size being 10 in Table 3.

Through Wilcoxon Signed Ranks test, G1_syn_SVM_10 significantly outperforms baseline ($p - value < 0.01$), but has no significant difference compared with G1_desc_SVM_10. However, if only classifiers in Group 1 are considered, G1_syn_SVM_10 significantly outperforms G1_desc_SVM_10 ($p - value < 0.01$). This indicates that our propose method II can generate effective supplementary training data. On the other hand, G1_syn_LR_10 is found to outperform Baseline_LR significantly only on Group 1 classes ($p - value < 0.01$).

Method	Micro F1	Macro F1
Baseline_LR	86.51%	51.52%
G1_desc_LR_10	86.68%	55.78%
G1_syn_LR_10	86.30%	62.85%‡
All_syn_LR_10	86.60%	62.19%‡
Baseline_SVM	87.26%	48.03%
G1_desc_SVM_10	86.96%	57.09%
G1_syn_SVM_10	87.22%	67.43%†‡
All_syn_SVM_10	87.88%	64.54%†‡

Table 3: Experiment results. †means significantly outperform baseline. ‡means significant outperform baseline on Group 1.

It shows that on SVM model, PubMed data collected with ICD-9-CM code descriptions synonyms works better in solving the data imbalance problem than with the official descriptions. After data supplementation, there are still 6 classifiers with F1 score being 0%, which will be further discussed in Section 5.

4.6 Experiment III: adding supplementary training data to all classifiers

In the third experiment, we add supplementary data to all 45 classifiers to explore whether adding supplementary data to the classifiers that originally have sufficient training data still can gain performance improvement. We name these two runs as All_Synonym_LR (All_syn_LR) and All_Synonym_SVM (All_syn_SVM). Also, only the best results with supplementary document size being 10 is shown in Table 3. Through Wilcoxon Signed Ranks test, All_syn_SVM_10 significantly outperforms the baseline, and All_syn_LR_10 significantly outperforms the baseline only on Group 1 ($p - value < 0.01$), but both have no significant difference with G1_syn and G1_desc. These means that adding supplementary training data is effective on solving data imbalance problem, but for the classifiers that originally have sufficient training data, extra training data seems have no significant effect.

5 Discussion

Experiment results indicates that our proposed supplementing training data method can help the classifiers to reach to a relatively balanced performance. Such improvement mainly comes from changing the word weight ranking so that important words rank higher. For example, for code ”758.6 turner syndrome”, in the baseline_LR (F1=0%), top 3 features with highest weights are ”duplicate left, partially, turner_syndrome”. But in G1_syn_LR_10 (F1=67%), top 3 features are ”turner, turner syndrome, syndrome”. Supplementary data trains term in ”turner syndrome” a higher weight in LR model, explaining this code’s classification performance increase.

In addition, supplementary data will improve classification through boosting the weight of the features. For example, the top features for code ”786.59, Other chest pain” are basically similar in both baseline_LR (F1=0%) and G1_syn_LR_10 (F1=40%), including ”tightness, chest_tightness, chest pain”. However, the weight differs a lot. For baseline_LR, weights are all under 1.5, while in G1_syn_LR_10, top 5 features are all above 1.5, indicating the classification model have much higher confidence on these features.

Finally, supplementary data mainly support code assignment effectively in Group 1, and we find that classification performance in Group 2 basically has no significant difference across all experiments. Meanwhile, 978 reports, dominated by Group 2 classes, also show no significant difference across all experiments. Therefore, extra training data does not improve Group 2’s performance, and hence supplementary data is not suggested for classes having sufficient training data.

Besides, our proposed methods can be directly used in ICD-10-CM classification with little modification. Just update the PubMed query with the ICD-10-CM textual descriptions and synonyms.

Though data imbalance problem has been largely alleviated, there are still a few classifiers in Group 1 have poor performance. After exploring, we think there are mainly four reasons:

- Word level feature matching limitation. For example, description of code ”V72.5 Radiological examination” does not appear in the collection, and it has no synonyms. Radiological examination actually means a variety

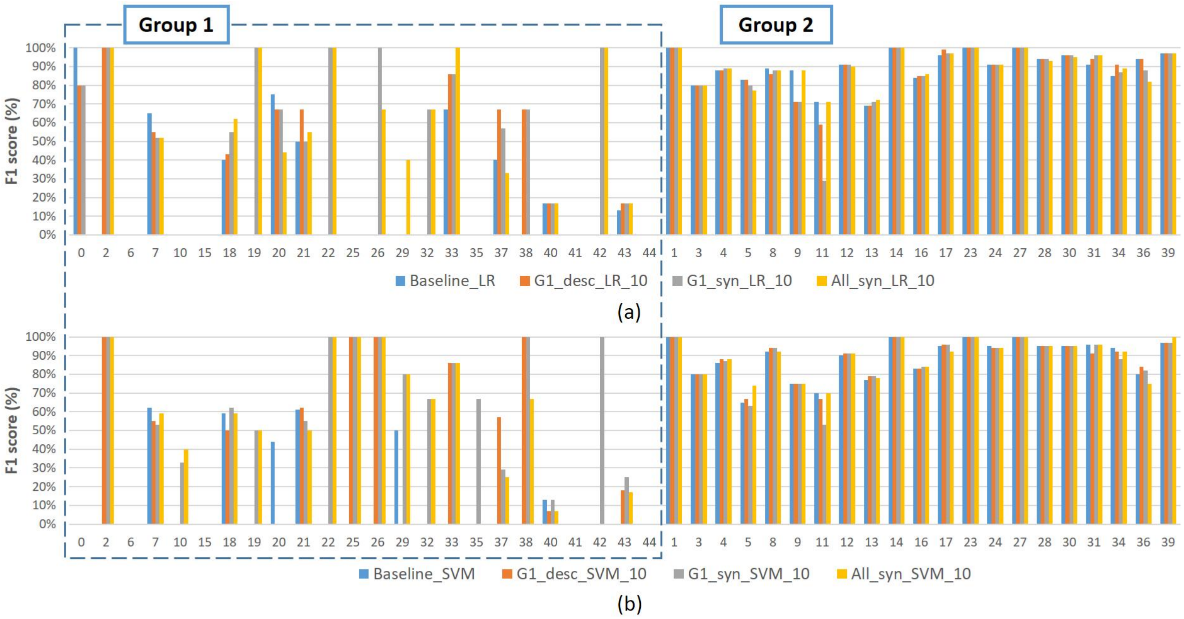


Figure 5: Individual classification performance of 45 classifiers on Baseline and three experiments.

of imaging techniques, and such word level feature matching cannot help classification.

- ”History of” ICD-9-CM codes. For codes ”V13.02, Personal history, urinary tract infection” and ”V13.09 Personal history of other specified urinary system disorders”, adding supplementary data doesn’t help their performance. We find their radiology reports are basically classified to ”599.0 urinary tract infection” and ”593.70 vesicoureteral reflux”. ”history of” feature is ignored. Extra training data has no effect on this problem.

- Speculative expression. In prepossessing procedure, negation terms are removed, but speculative expressions are kept. It results in that when doctor is not sure whether a patient may get a disease, but write it down to reports, classification results will rely on these speculative terms, and cause false positive. For example, code ”518.0” has a low F1 score because in many reports labeled with other codes, doctors write that the patient may have disease ”atelectasis”, while ”atelectasis” is a very important word to recognize ”518.0”.

- Data missing due to expert disagreement. In CMC dataset, three experts manually assign codes to 978 radiology reports. Only when two or more experts agree, code is approved. However, sometimes the conflict opinions results in code assignment failure. For example, reports 99619963 and 99803917 should be labeled with ”741.90 Spina bifida”. However, one expert assigned ”741.90”, another assigned ”741.9”, and the third expert miss this code at all. This led to ”741.90 Spina bifida” was not assigned to these two reports. However, with the supplementary data added into the training, our method correctly assigns ”741.90 Spina bifida” to these two reports, but this assignment was counted as wrong since the ground truth does not have this code due to expert disagreements.

6 Conclusion and Future Work

In this study, we studied to address the data imbalance problem in ICD-9-CM code automatic assignment task. Using ICD-9-CM codes synonyms can accurately search medical texts relevant documents from PubMed. Collected data, used as supplementary training data, can significantly boost systems macro averaging performance as the data imbalance problem is largely alleviated. However, for the classifiers that originally have sufficient training data, additional data basically has no significant effect. As future work, we will modify the Context algorithm (Harkema et al., 2009) to detect the historical mentions and speculative expressions in the radiology reports. Also, we would explore the difference of same features extracted from different field of radiology reports.

References

Alan R Aronson, Olivier Bodenreider, Dina Demner-Fushman, Kin Wah Fung, Vivian K Lee, James G Mork, Aurélie Névéol, Lee Peters, and Willie J Rogers. 2007. From indexing the biomedical literature to coding clinical text: experience with mti and machine learning approaches. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*. Association for Computational Linguistics, pages 105–112.

Alan R Aronson and François-Michel Lang. 2010. An overview of metamap: historical perspective and recent advances. *Journal of the American Medical Informatics Association* 17(3):229–236.

Rich Caruana. 2000. Learning from imbalanced data: Rank metrics and extra tasks. In *Proc. Am. Assoc. for Artificial Intelligence (AAAI) Conf*. pages 51–57.

Wendy W Chapman, Will Bridewell, Paul Hanbury, Gregory F Cooper, and Bruce G Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of biomedical informatics* 34(5):301–310.

Koby Crammer, Mark Dredze, Kuzman Ganchev, Partha Pratim Talukdar, and Steven Carroll. 2007. Automatic code assignment to medical text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*. Association for Computational Linguistics, pages 129–136.

Richárd Farkas and György Szarvas. 2008. Automatic construction of rule-based icd-9-cm coding systems. *BMC bioinformatics* 9(3):S10.

Henk Harkema, John N Dowling, Tyler Thornblade, and Wendy W Chapman. 2009. Context: an algorithm for determining negation, experiencer, and temporal status from clinical reports. *Journal of biomedical informatics* 42(5):839–851.

Haibo He and Edwardo A Garcia. 2009. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering* 21(9):1263–1284.

Goldstein Ira, Arzumtsyan Anna, and Uzuner Ozlem. 2007. Three approaches to automatic assignment of icd-9-cm codes to radiology reports .

Peter Jackson and Isabelle Moulinier. 2007. *Natural language processing for online applications: Text retrieval, extraction and categorization*, volume 5. John Benjamins Publishing.

Ramakanth Kavuluru, Sifei Han, and Daniel Harris. 2013. Unsupervised extraction of diagnosis codes from emrs using knowledge-based and extractive text summarization techniques. In *Canadian Conference on Artificial Intelligence*. Springer, pages 77–88.

Ramakanth Kavuluru, Anthony Rios, and Yuan Lu. 2015. An empirical evaluation of supervised learning approaches in assigning diagnosis codes to electronic medical records. *Artificial intelligence in medicine* 65(2):155–166.

Bevan Koopman, Guido Zuccon, Anthony Nguyen, Anton Bergheim, and Narelle Grayson. 2015. Automatic icd-10 classification of cancers from free-text death certificates. *International journal of medical informatics* 84(11):956–965.

Huan Liu and Rudy Setiono. 1995. Chi2: Feature selection and discretization of numeric attributes. In *Tools with artificial intelligence, 1995. proceedings., seventh international conference on*. IEEE, pages 388–391.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

Jon Patrick, Yitao Zhang, and Yefeng Wang. 2007. Developing feature types for classifying clinical notes. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*. Association for Computational Linguistics, pages 191–192.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* 12(Oct):2825–2830.

John P Pestian, Christopher Brew, Paweł Matykiewicz, Dj J Hovermale, Neil Johnson, K Bretonnel Cohen, and Włodzisław Duch. 2007. A shared task involving multi-label classification of clinical free text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*. Association for Computational Linguistics, pages 97–104.

Ian H Witten, Eibe Frank, Mark A Hall, and Christopher J Pal. 2016. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Yitao Zhang. 2008. 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*. Association for Computational Linguistics, pages 67–72.

Guido Zuccon and Anthony Nguyen. 2013. Classification of cancer-related death certificates using machine learning .