ICD Code Retrieval: Novel Approach for Assisted Disease Classification

Stefano Giovanni Rizzo[1(✉)], Danilo Montesi[1], Andrea Fabbri[2], and Giulio Marchesini[3]

[1] Department of Computer Science and Engineering, University of Bologna,
Mura Anteo Zamboni 7, 40127 Bologna, Italy
stefano.rizzo8@unibo.it

[2] Local Public Health Unit of Forlì, Emergency Department,
Hospital Morgagni-Pierantoni, via Forlanini 34, 40121 Forlì, Italy

[3] Department of Medicine, University of Bologna,
via Massarenti 9, 40138 Bologna, Italy

Abstract. The task of assigning classification codes to short medical text is a hard text classification problem, especially when the set of possible codes is as big as the ICD-9-CM set. The problem, which has been only partially tamed for a subset of ICD-9-CM, becomes even harder in real world applications, where the labeled data are scarce and noisy. In this paper we first show the ineffectivenesss of current Text Classification algorithms on large datasets, then we present a novel incremental approach to clinical Text Classification, which overcomes the low accuracy problem through the top-K retrieval, exploits Transfer Learning techniques in order to expand a skewed dataset and improves the overall accuracy over time, learning from user selection.

Keywords: ICD-9-CM · Text classification · Transfer learning · Learning to rank · Document expansion · Icd coding task

1 Introduction

The International Classification of Diseases (ICD) is a standard, broadly used classification system, that codes a large number of specific diseases, symptoms, injuries and medical procedures into numerical classes. Assigning a code to a clinical case means classifying that case into one or more particular discrete class, hence allowing further statistics studies and automated calculations. The possibility to have a discrete code instead of a text in natural language is intuitively a great advantage for data processing systems. The use of such classification is becoming increasingly important for, but not limited to, epidemiological, economic and policy-making purposes.

The presentation of this work has been partly funded by FIRB project Information monitoring, propagation analysis and community detection in Social Network Sites.

N. Ashish and J.-L. Ambite (Eds.): DILS 2015, LNBI 9162, pp. 147–161, 2015.
DOI: 10.1007/978-3-319-21843-4_12

While the ICD Classification is clearly useful on many aspects, physicians and clinical personnel think and write in natural language and, after that, assign the right code to their text description aided by manuals, guidelines, or their own memory.

The ICD-9-CM contains more than 14 thousand classification codes for diseases, meaning that manual methods are inadequate to locate the right classes in a real-world scenario, even for expert clinical coders. In some medical departments the codes used are just a tiny subset of the ICD classification set, hence the problem is reduced, but in many other and in generic departments like the Emergency, this subset covers a big portion of the classification codes. An accurate system that assist the medical personnel in the task of coding is needed to reduce costs and to provide better standardization of the medical data (Table 1).

Table 1. Samples of medical text with the associated ICD-9-CM codes

Clinical short text	ICD-9-CM labels
5-year-old male with cough, normal slightly hypoventilatory chest x-ray, no pneumonia	786.2
Urinary tract infections. Normal sonographic appearance of the kidney bilaterally. Trace amount of nonspecific free fluid in the pelvis	599.0, 780.31, 780.39
Vesicoureteral reflux followup. Normal renal ultrasound. Mild intermittent left hydroureter proximally at the renal pelvis	593.70, V13.09, 593.5

Among the many attempts to simplify or automate the coding task of medical text we can distinguish between two approaches: the **Information Retrieval** (IR) of codes from a dictionary and the machine learning or rule-based **Text Classification** (TC).

In the first approach a typical boolean IR model allows the personnel to search the dictionary for a set of one or more terms. Often these systems allow also to search for disjunction and conjunction of terms (boolean queries), exact text matching (full-text search) and the use of jolly characters, to expand the queries (regular expressions). Nevertheless these methods represent the most used techniques in real world applications, due to their simplicity of implementation and their ability to cover seamlessly an entire set with thousands entries.

Over the last years, TC has received attention as a valuable solution to medical text coding [3,8]. The described problem fall into a TC problem with some properties:

1. Multi-class Classification: the number of output classes (ICD codes) can be very high, contrary to the simplest binary classification.
2. Multi-label Classification: a text instance can be associated with more than one label. This is true for two reasons: because a text can include different diseases (e.g. injuries to different arms) and because there might need more than one code to describe a clinical condition (primary and secondary).

The TC approach to the problem is the most promising one, since it aims at providing automatic code assignment, without any user interaction. Unfortunately, even getting a clean and balanced training set of labeled medical text, TC achieved great results on small datasets, but almost fails in classifying large-scale taxonomies, like the ICD, in both classification accuracy and performance. The effectivenesss of cutting-edge classification algorithms is heavily reduced when applied to very large taxonomies. We will conduct a short survey on classification accuracy in Sect. 3.1, showing the accuracy degradation over increasing number of classes.

The code retrieval approach that we propose is a mixed approach, as it shares features and ideas of both IR and TC. The proposed approach is based on learning from labeled samples and auxiliary sources, retrieving the K most relevant classes based on term-frequency similarities and improving the ranking by learning from the users feedback. The impossibility to achieve a reasonable accuracy on a large class space, together with the online assisted coding approach, leads to prefer a top-K retrieval model over a strict text classifier. Instead of precisely selecting the right number of labels for a medical text, we are interested in showing the most relevant codes, and then let the user to choose the appropriate ones. Moreover this ensures associations with the right codes, allowing a running system to further learn and improve itself, using the users' selection as a continue flow of training data.

In order to address the lack of high-quality annotated examples we took some ideas from Transfer Learning, that is a set of methods to extract useful knowledge from different but related auxiliary domains [16]. Using the ICD code as an attribute to match related contents, we augmented our training set with knowledge from auxiliary sources (e.g. Wikipedia, ICD Manuals, etc.), thus obtaining a model with a greater accuracy.

In Sect. 2 we present the related work. In Sect. 3 we outline the addressed problem and we lay the foundations for the proposed approach. In Sect. 4 we present the approach in further details showing our implementation. In Sect. 5 we evaluate the accuracy of the system in different settings on medical datasets. In Sect. 6 we summarize the contributions of the paper and propose further experiments.

2 Related Work

The goal of fully automating the ICD-9-CM assignment of codes to medical text is unrealistic for many practical reasons we will outline hereafter. Nevertheless some attempts and studies have been made by researchers in the last two decades, most of which have been conducted on a small subset of the coding classes. Larkey and Croft [10] trained three statistical classifiers for the automatic assignment of ICD-9 codes, and then combined their results to obtain a better classification. Their work is based on discharge summaries, for which the number of labels per document is from 1 to 15. This combined classifier produces a ranked list of the top-K most relevant codes, which makes it very similar to our approach, but the instances domain is different, as discharge summaries have different terms distribution than short diagnosis. Lussier et al. [13] studied the feasibility of automating the ICD-9-CM coding task, concluding that more external knowledge bases and manual revisions where needed to improve accuracy.

The lack of a shared, publicly available training and testing dataset with labeled medical text discouraged further reasearches, until the CMC Challenge in 2007 [17]. The challenge consisted in building a classifier that could automatically encode medical text in ICD-9-CM classification. For the challenge purposes, data were collected from the Cincinnati Childrens Hospital Medical Centers (CCHMC) Department of Radiology. Since code annotation is a difficult task, each document in the corpus was evaluated by three expert annotators. A gold annotation was created by taking the majority of the annotators. With only 45 ICD codes, this corpus is still really far from the ideal. A group of 50 teams and individuals submitted their results for the challenge. The best results for classification accuracy have been scored by rules-based systems [7], which dominated the challenge. These systems were based entirely or partly on hand-crafted expert rules. In the challenge context this was a feasible approach and has been proved to be the best model in terms of prediction accuracy. However it would be very time-consuming, if not impossible, to hand-craft expert rules for all ICD codes. Another approach is the machine learning one [20], in which the classifier is automatically built from the training data, without the need for human intervention. We will show in Sect. 3.1 that the accuracy tends to drop dramatically as the number of classes increases [12].

3 ICD Code Retrieval

Our work is focused on the practical problem that medical personnel face on a daily basis. Medical personnel manually assign ICD codes while or after examinations and procedures. The code assignment task has become part of the procedure and is not an a posterieri practice, therefore a coding system should help the personnel during the coding. This significant property of the problem should be exploited with a new approach to overcome the low accuracy of automated solutions.

3.1 Text Classification Accuracy Decay

In the task of Multi-label TC we have a set of text instances such that each instance must be associated with a subset of all possible classes $\mathcal{Y} = \{c_1, ..., c_n\}$.

TC on large taxonomies, like ICD-9-CM codes set, is a major challenge for state-of-the-art machine learning algorithms, including Support Vector Machines (SVM). Machine learning algorithms, like SVM, have achieved great results in classifying small text collections [4], but proved to be less and less accurate when the number of classes starts growing [4,12].

As discussed in [11,18], current machine learning methods need significant improvement when applied to very large-scale datasets. Effectiveness of state-of-the-art models is unacceptable on large-scale applications, partially due to the data sparseness in rare classes.

In order to show the relation between number of classes and classification accuracy, we conducted a short survey on classification performance, summarized in Fig. 1. Accuracy values are measured as F1 score, a popular measure of accuracy in classification problems. Given the number of true positive results (TP), false positives (FP) and false negatives (FN), the F1 score is calculated as:

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{1}$$

The Macro-averaged F1 is the average of the F1 scores of each instance in the specific test collection.

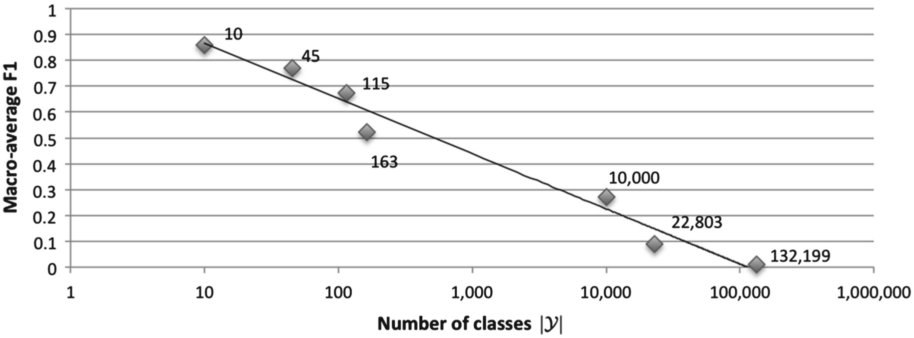


Fig. 1. Values of accuracy (Macro-averaged F1) using SVM on datasets with different $|\mathcal{Y}|$ value (classes space size). Trend and class space scale are logarithmic.

Results from various multi-label classification experiments on small, medium and large popular datasets are shown. In particular, we selected results of Support Vector Machines (SVM) algorithms on collections with different size $|\mathcal{Y}|$ of the target class space $\mathcal{Y}$: $|\mathcal{Y}| = 10$ and $|\mathcal{Y}| = 115$ are from *Reuters 21578* [4], $|\mathcal{Y}| = 45$ is the best result for the CMC Challenge [20], $|\mathcal{Y}| = 163$ is from the *LookSmart* web directory [2], $|\mathcal{Y}| = 22,803$ is from the *MERG* subset of *Yahoo! Directory* [12], $|\mathcal{Y}| = 132,199$ is from *Yahoo! Directory* [11]. All the results are obtained using SVM algorithms except for the CMC Challenge ($|\mathcal{Y}| = 45$), which derives from a rule-based system. Those results come from different works on text categorization, therefore the SVM implementation may vary slightly, but the overall degradation of accuracy on larger datasets is evident in Fig. 1.

3.2 Top-K Code Retrieval

As pointed out in [15,21], there is no obvious winner in multi-class classification techniques. For practical problems, the choice of approach will have to be made depending on the constraints, e.g. the desired accuracy level, the time available, and the nature of the problem.

In the hard version of the classification problem, a particular set of labels is explicitly assigned to the instance, whereas in the soft version of the classification problem, a score is assigned to the each label. The approach for interactive TC that we refer to as the code retrieval approach solve a soft version of the TC problem. It has been applied on the ICD-9-CM classification problem by Larkley and Croft [10] in 1995. A similar approach is found more recently in [14] in which a semi-automatic approach is proposed to automatically classifying the easiest associations while hardest instances are left to the user judgement.

In a top-K IR model, results are displayed in a ranked order to the user. Similarly, in code retrieval, most probable ICD codes for a medical text are retrieved and displayed in ranked order. In most multi-label TC algorithms, a ranked set of the best scoring classes is also produced, however a thresholding strategy exists to select how many codes, from the best scoring ranked set, should be assigned to the text instance.

In multi-label TC, different choices of the threshold strategy lead to different accuracy results [6], while in our approach results are presented in ranked order, without a thresholding strategy. Since no specific set of codes is assigned in code retrieval, accuracy measures are evaluated over the first K ranked results returned, for different values of K. Each value of K can be considered as the number of ranked results to be shown in the first Search Engine Results Page (SERP). It is important to note that the total lack of the right code in the retrieved results is unacceptable: the end-user must be able to get more than K results whenever he asks to. However, it is desirable to obtain the right codes in the first SERP: from a user point of view, earlier researches have shown that only 30 % of users view results past the first SERP in search engines [9], which in the average case counts 10 results.

3.3 Transfer Learning

Dealing with a collection of real labeled data provided by partner hospitals, we came across different issues regarding its use as a training set:

1. **Data Sparseness**: uncommon or specific clinical conditions are never or rarely present in the data.
2. **Unreliable Labels Association**: due to the coding task complexity, the chosen labels (ICD codes) are not always objectively accurate. The construction of a reliable ground truth would involve several experts to individually vote for every association, as in the CMC Challenge for radiology department [17].
3. **Unbalanced Distribution of Labels**: while less common diseases or very specific codes are missing or scarce, generic codes and codes related to common clinical conditions are used very often resulting in over abundance of positive samples for a small subset of the labels space.

When a valid training set is not available, one strategy to improve the learning is to expand the training set with text-code associations from auxiliary sources, a practice that falls under the Transfer Learning category. Transfer Learning refers to the framework of methods for machine learning where training data or classification model are extracted from an auxiliary source to augment the original learning model. In a Transfer Learning setting a *transfer* of knowledge occurs from a source domain (the auxiliary source) to the target domain (the domain of the model you want to learn). Apart from this common meaning, many different settings and definitions of the Transfer Learning model exist and found application in different contexts of classification [16]. Our scenario fits in an *inductive* Transfer Learning setting, in which labeled data from the source domain are used to induce a predictive model for the target domain. Since a lot of labeled data are available in the source domain, the *inductive* Transfer Learning setting aims at improving the learning task in the source domain by transferring knowledge from the source task.

4 Implementation

The overall architecture of the implementation is shown in Fig. 2. The main processes of the implementation are:

1. **Training Set Learning**: labeled samples, consisting of diagnoses labeled with codes from the ICD standard, compose the training set of the original domain.
2. **Trasfer Learning**: external sources, like dictionary entries and encyclopedia articles, are extracted along with the related ICD codes. Generic codes are mapped onto a subset of the labels set $\mathcal{Y}$.
3. **Text Preprocessing**: a set of filters is applied on the text data from both the training set and the auxiliary domains, in order to improve the final accuracy and reduce the index size.
4. **TF/IDF Indexing**: the preprocessed text data, with the associated labels, is indexed in a vector space using standard term weighting based on terms frequency.
5. **Top-K Retrieval**: when a user issue a set of words describing a disease (query), the K best scoring labels are selected using a textual similarity model, and provided to the user for manual picking. We evaluated three different similarities: Vector Space Model, Language Model and Okapi BM25, in the implementations provided by the *Apache Lucene*™framework[1]. We found the BM25 similarity to be the most effective similarity model for this task.
6. **Learning to Rank Cycle**: from the set of K relevant codes, the user select the right ones. The user selection feedback allows further improvement of future scoring: the issued query text is used as a positive training sample for the hand-picked labels.

[1] http://lucene.apache.org.

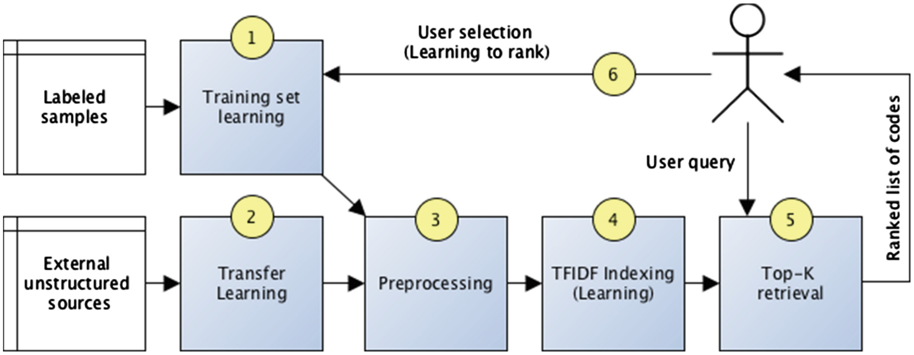


Fig. 2. The architecture of the ICD code retrieval implementation, showing the data flow involved in the main processes.

4.1 Preprocessing and Indexing

In order to improve efficiency and effectivenesss of the classification and reduce the index size, some pre-processing actions must be taken. The pre-processing filters are applied in sequence to the textual data. Apart from the HTML filter, which is applied only to the auxiliary domain instances coming from web sources, the rest of the pipeline is applied on all the text involved: labeled samples, auxiliary instances, test instances (user queries).

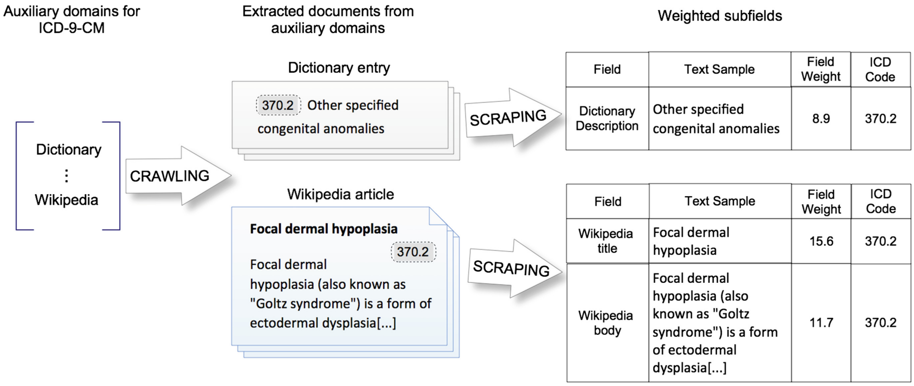
All the modules in the pre-processing pipeline are:

1. HTML Code Removal: this filter applies only on auxiliary instances coming from web sources. If the text data is in unformatted form, the filter is ignored.
2. Keep Word Filter: this module ensures that the words on a list are not discarded or altered by the pre-processing. The keep list is populated with abbreviations and expressions from the medical jargon.
3. Stop Word Removal: common words (e.g. “the”, “that”, “a”, “an”) are discarded in order to reduce index size and improve effectivenesss.
4. Lowercase Filter: transforms the letters in each term to lowercase only letters, in order to reduce the number of tokens.
5. Porter Stemmer: Porter’s stemming algorithm is applied to remove the commoner morphological and inflexional endings of the terms, improving recall and reducing the index size.
6. Shingle Filter: combine together adjacent terms to form n-grams of terms, producing a new token for every combination, therefore improving precision without affecting the recall of single terms.

The last two filters significantly improved accuracy on both general and medical TC.

4.2 Cross-Domain Transfer Learning

For each auxiliary domain a specific *crawler* is required to retrieve all the documents associated with ICD codes. Each document is then processed with a *scraper*, built on a set of hand-crafted regular expression, which extract different fields of a semi-structured document, along with the attribute related to the ICD-9-CM code (see Fig. 3).


Auxiliary domains for ICD-9-CM
Extracted documents from auxiliary domains
Weighted subfields
Dictionary
⋮
Wikipedia
CRAWLING
Dictionary entry
370.2 Other specified congenital anomalies
Wikipedia article
Focal dermal hypoplasia
370.2
Focal dermal hypoplasia (also known as "Goltz syndrome") is a form of ectodermal dysplasia[...]
SCRAPING
SCRAPING

Field	Text Sample	Field Weight	ICD Code
Dictionary Description	Other specified congenital anomalies	8.9	370.2

Field	Text Sample	Field Weight	ICD Code
Wikipedia title	Focal dermal hypoplasia	15.6	370.2
Wikipedia body	Focal dermal hypoplasia (also known as "Goltz syndrome") is a form of ectodermal dysplasia[...]	11.7	370.2

Fig. 3. Cross-domain data extraction and mapping for Transfer Learning. New features from different domains are extracted and decomposed into fields with specific weights.

Transfer learning is a valuable solution when the training set is small relatively to the number of classes and the labels distribution is unbalanced. It improves the recall of the system, expanding the terms in the training set with synonyms and related words. However this augmentation may associates terms which are not strictly related to a disease, whose relationship makes sense only in the context of the source domain.

A strategy to address the degradation of precision is to weight differently the domains and the fields involved in the training, as in Fig. 3. Given the set of all fields $T = \{t_1, ..., t_r\}$ from all domains, a weight vector $W = \{w_1, ..., w_r\}$ is computed, denoting the relative significance of every field. At retrieval time, the probability of a code c with respect to a text is computed as the linear combination of the probabilities of each single field t_i associated with c, with w_i the coefficient for t_i. The Apache DisMax query parser allows to alter the similarity model by specifying different weights for different fields of a structured document, therefore implementing the linear combination described.

The optimal weight vector W depends on the involved auxiliary domains and can be determined empirically, selecting the vector that maximize the overall accuracy. While this exhaustive search can be viable for the small CMC Corpus, it becomes extremely time-consuming for larger datasets. In this case the weight w_i of each field t_i can be approximated as the accuracy produced by the system trained with t_i only. This means training the system r times with a different binary permutation of the vector W.

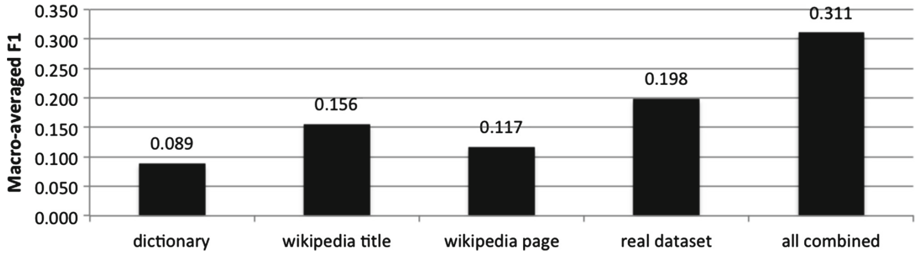


Fig. 4. Accuracies (macro-averaged F1 scores) obtained training with each single field only. The first field is the description of official italian dictionary of all ICD-9-CM codes. From italian wikipedia pages of diseases are extracted the title and the whole article. The real dataset are text medical reports from the *ITA50*.

We conducted the experiment on the *ITA50* corpus [5] using the known minimum K for every instance of the testing set. The resulting accuracies are shown in Fig. 4 for 4 different domain fields, of which 3 fields comes from transfer learning. The first 4 accuracies are obtained training the sistem with each single field alone. The rightmost accuracy show the results of the linear combination of all the fields, proving the advantage of transfer learning.

4.3 Incremental Learning to Rank

The learning to rank paradigm allows a running system to improve the ranking relying on past user selections. Based on this idea, the proposed code retrieval approach increase its capabilities over time, using additional knowledge and users interaction.

In the typical scenario, when a coding assistant software is not yet deployed, there are no labeled instances for training yet, or the ones available are not reliable. With no other supplementary knowledge, the best and only help a non-expert code can get is a search engine on the dictionary.

Our approach permits, within a single framework, to first relying solely on a simple search engine and a provided dictionary. Then to increase the system capability providing other knowledge bases, like encyclopedia and manuals, assuming these are properly structured. Finally, every medical text issued in the system, along with the selected codes, will contribute at improving the system.

The user query text, together with the subset of codes in the ranked list selected by the user, is regarded as a labeled sample, in the same domain of the training set, therefore weighted accordingly.

5 Experimental Results

Since it has not been possible to test the implementation with medical personnel, we conducted several experiments on labeled corpuses to assess the benefits of the proposed approach. The indexing, preprocessing and scoring tasks have been carried out using the *Apache Lucene*^{TM} framework.

5.1 Dataset

Popular TC datasets, such as *Reuters 21578* and *20 Newsgroups* have been first used to evaluate the model as a classic text categorization algorithm, obtaining average results. The CMC corpus [17] and a set of 50 thousand text-label associations for short clinical reports from italian hospitals (*ITA50* [5]) have been used for accuracy testing on medical text data.

Since there is no publicly available English dataset for medical classification with a label space $\mathcal{Y}$ larger than 45 codes, the *ITA50* corpus represented the most reliable dataset to validate our approach in a realistic scenario. The *ITA50* corpus is a set of human labeled samples from real hospital clinical reports, edited in italian language and coded accordingly to the ICD-9-CM guidelines. Albeit the learning sources involved in training and Transfer Learning are obviously language dependent, the proposed approach abstracts from any specific language. The *ITA50* corpus is composed of 14,304 different medical records and 50,078 text-label associations, meaning an average of 3.5 labels per text record. The distribution of classes among the records is strongly unbalanced: 3,259 different classes of which 1,061 associated with only 1 record instance, while the 4 most frequent classes alone counts 5,187 records. The average number of words per text record is 18.

5.2 Evaluation

The commonly used performance evaluation criteria for multi-label classification is the F1 accuracy score. Since our approach is strongly related to Information Retrieval, significant measures considered in our tests comprise also precision and recall measures at specific K values. In fact, since a fixed K of results will be returned, it is crucial to investigate recall and precision over K.

The precision score denotes the fraction of TP in the returned results:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

The recall score denotes the fraction of TP in the set of right codes:

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

In the evaluation on the *ITA50* corpus we considered the total set of ICD-9-CM classes (14,170 in the italian dictionary), despite the *ITA50* comprises only 3,259 labels, of which 1,061 labels are either in the training or in the testing set. Given the imbalance in the number of samples per class, we splitted the corpus with a ratio 10/90 between testing and training set.

Using values of K from 1 to 100 we evaluated the overall system accuracy under precision, recall and macro-averaged F1, as shown in Fig. 5.

With only 3.5 right labels per test instance on average, accuracy measures taking into account false positives (i.e. precision and F1) are clearly disadvantaged for larger values of K. We are instead mostly interested in the recall of the

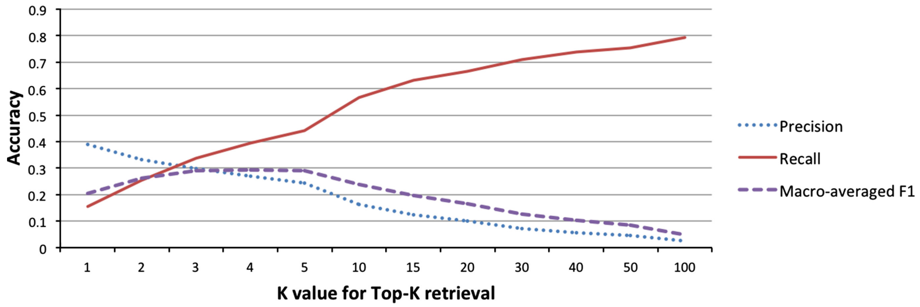


Fig. 5. Recall, precision and macro-averaged F1 evaluated for a different number K of returned scores, on the ITA50 corpus [5].

retrieval over K values, which can be viewed as the probability to find the entire set of right codes within the first K results.

Evaluation on the CMC corpus has been carried out using transfer learning from Wikipedia English (articles title and body) and from the Centers for Medicare and Medicaid Services (abbreviated and full descriptions in dictionary). As for the accuracy of strict TC classifiers, the accuracy of soft classifiers depends on the number of classes involved, as shown in the top-k experiment on the CMC corpus. Considering a SERP of 10 results, the probability of getting all the right codes in the first SERP is quite different in the two datasets: this probability is 97.3 % for the 45 codes of the CMC corpus (Fig. 6) and 56.7 % for the 3,259 codes in *ITA50*.

ICD code retrieval is a soft classifier in which it returns k classes sorted by probability of relevance, therefore no thresholding strategy is defined. Conversely, the CMC challenge systems were hard classifiers, returning a definite set of classes for each sample of the testing set. In order to compare a soft classifier with hard classifiers we defined two elementary thresholding:

- Fixed K: K is fixed to 1, i.e. only the first class is retrieved. Threshold is fixed for all samples of testing set, this can be seen as the worst case scenario.

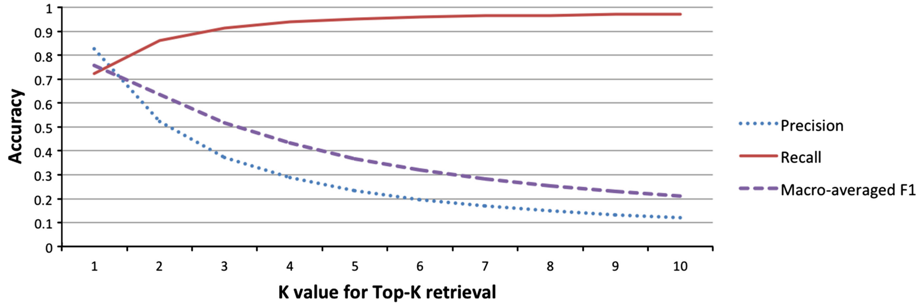


Fig. 6. Recall, precision and macro-averaged F1 evaluated for a different number K of returned scores, on the CMC corpus [17].

- Perfect K_s: for each sample s of the testing set, the top K_s are retrieved, where K_s is the exact number of labels for the sample s. This emulates an ideal thresholding strategy, therefore can be seen as the best case scenario.

Table 2. Final results for top 8 submission of CMC challenge [1], sorted by macro-averaged F1, in comparison with ICD Code Retrieval scores in the two different thresholding settings.

Team/System	Ma-F1	Challenge Rank	Approach
ICD Code Retrieval (Perfect K)	**0.806**		**BM25 + Transfer learning.**
LMCO-IS & S	0.776	5	N/A
Szeged [7]	0.7691	1	Rule based + C4.5 + Maximum entropy classifier
ICD Code Retrieval (Fixed K)	**0.756**		**BM25 + Transfer learning.**
LLX	0.7343	21	N/A
GMJ_JL	0.7334	6	N/A
SULTRG	0.7322	7	N/A
University at Albany [8]	0.7291	2	Rule-based + synonyms from www.icd9data.com
PENN [3]	0.721	4	Rule-based + synonyms from MeSH
University of Turku [20]	0.7034	3	SVM-like (RLS) + concepts from UMLS

Knowing the right number K of codes for each instance in the testing set, we selected the top-K codes from our implementation, thus yelding a macro-averaged F1 of 80.6 %, which is higher than the best scoring rule-based system in the challenge (macro-averaged F1 76.9 % [7]). Even setting a global fixed K to 1, the resulting macro-averaged F1 is 75.6 %, which is slightly lower than the best system, but still higher than any machine-learning approach in literature [19,20]. As shown in Fig. 6, we then evaluated precision, recall and F1 for each globally fixed K between 1 and 10 (Table 2).

6 Conclusion and Future Work

We have presented and evaluated a complete approach for assisting users in diseases coding. Our approach consider two related problems of ICD computer assisted coding systems: the low accuracy in automated TC for large labels space and the lack of balanced, well coded labeled samples.

The low accuracy problem is first established by surveying related works, showing that the accuracy of multi-label TC algorithms is strongly affected by the number of target classes. In order to overcome the difficulty, we have proposed an end-user oriented approach that aims at maximizing the recall of returned results, allowing the user to select the right labels in the smallest possible set of best-scoring matches. In the worst case, when the selected target subset of ICD is bigger than a few hundred codes, navigation through more than one results page could be necessary.

The unavailability of an adequate training set has been tackled through Transfer Learning techniques: the proposed incremental learning strategy allows to bootstrap with an acceptable search engine, which then improves its accuracy through machine learning on users selection feedback. We have shown the substantial benefit of using a combination of multiple sources with respect to a single source (e.g. training labeled samples).

Analysis of unreviewed labeled data coming from italian hospitals has provided a deeper understanding of the real problem hardness, addressing research towards more realistic solutions. Evaluation on the CMC corpus shown evidence of the accuracy of the proposed approach in comparison to the best-scoring systems in literature.

Future work will investigate hierarchical implementations of the proposed soft classifier, in order to leverage the taxonomy of the ICD-9-CM for improved accuracy. A more solid validation must be carried out on a large labeled corpus to show the effectiveness of the proposed approach. A period of expert usage is needed to assess the improvement of the system over time through the learning to rank process.

References

1. Results: Medical nlp challenge, computational medicine center (2007). https://web.archive.org/web/20080111141103/, http://www.computationalmedicine.org/challenge/res.php
2. Chen, H., Dumais, S.: Bringing order to the web: Automatically categorizing search results. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 145–152. ACM (2000)
3. Crammer, K., Dredze, M., Ganchev, K., Talukdar, P.P., Carroll, S.: Automatic code assignment to medical text. In: Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing, pp. 129–136. Association for Computational Linguistics (2007)
4. Debole, F., Sebastiani, F.: An analysis of the relative hardness of reuters-21578 subsets. J. Am. Soc. Inf. Sci. Technol. **56**(6), 584–596 (2005)
5. Fabbri, A., Montesi, D., Rizzo, S.G.: ITA50 corpus of 50 thousands icd-9 labeled medical text (2015). http://smartdata.cs.unibo.it/ITA50/
6. Fan, R.E., Lin, C.J.: A study on threshold selection for multi-labelclassification. Department of Computer Science, National Taiwan University,pp. 1–23 (2007)
7. Farkas, R., Szarvas, G.: Automatic construction of rule-based ICD-9-CM codingsystems. BMC Bioinform. **9**(Suppl 3), S10 (2008)
8. Goldstein, I., Arzumtsyan, A., Uzuner, Ö.: Three approaches to automatic assignment of icd-9-cm codes to radiology reports. In: AMIA Annual Symposium Proceedings. vol. 2007, p. 279. American Medical Informatics Association (2007)
9. Jansen, B.J., Spink, A.: How are we searching the world wide web? a comparison of nine search engine transaction logs. Inf. Process. Manag. **42**(1), 248–263 (2006)
10. Larkey, L.S., Croft, W.B.: Automatic assignment of icd9 codes to discharge summaries. Technical report (1995)
11. LIU, T.Y., Yang, Y., WAN, H., ZENG, H.J., CHEN, Z., MA, W.Y.: Support vector machines classification with a very large-scale taxonomy. ACM SIGKDD Explor. Newsl. **7**(1), 36–43 (2005)
12. Liu, T.Y., Yang, Y., Wan, H., Zhou, Q., Gao, B., Zeng, H.J., Chen, Z., Ma, W.Y.: An experimental study on large-scale web categorization. In: Special Interest Tracks and Posters of the 14th International Conference on World Wide Web, pp. 1106–1107. ACM (2005)

13. Lussier, Y.A., Shagina, L., Friedman, C.: Automating icd-9-cm encoding using medical language processing: A feasibility study. In: Proceedings of the AMIA Symposium, p. 1072. American Medical Informatics Association (2000)
14. Martinez-Alvarez, M., Yahyaei, S., Roelleke, T.: Semi-automatic Document Classification: Exploiting Document Difficulty. In: Baeza-Yates, R., de Vries, A.P., Zaragoza, H., Cambazoglu, B.B., Murdock, V., Lempel, R., Silvestri, F. (eds.) ECIR 2012. LNCS, vol. 7224, pp. 468–471. Springer, Heidelberg (2012)
15. Nigam, K., Lafferty, J., McCallum, A.: Using maximum entropy for text classification. In: IJCAI-99 Workshop on Machine Learning for Information Filtering, vol. 1, pp. 61–67 (1999)
16. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
17. Pestian, J.P., Brew, C., Matykiewicz, P., Hovermale, D., Johnson, N., Cohen, K.B., Duch, W.: 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, pp. 97–104. Association for Computational Linguistics (2007)
18. Sandu Popa, I., Zeitouni, K., Gardarin, G., Nakache, D., Métais, E.: Text categorization for multi-label documents and many categories. In: Twentieth IEEE International Symposium on Computer-Based Medical Systems. CBMS 2007, pp. 421–426. IEEE (2007)
19. Sujeevan, A., Youns, B.: Semi-structured document categorization with a semantic kernel. Pattern Recogn. **42**(9), 2067–2076 (2009)
20. Suominen, H., Ginter, F., Pyysalo, S., Airola, A., Pahikkala, T., Salanter, S., Salakoski, T.: Machine learning to automate the assignment of diagnosis codes to free-text radiology reports: a method description. In: Proceedings of the ICML/UAI/COLT Workshop on Machine Learning for Health-Care Applications (2008)
21. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 42–49. ACM (1999)