Unsupervised Extraction of Diagnosis Codes from EMRs Using Knowledge-Based and Extractive Text Summarization Techniques

Ramakanth Kavuluru*[1,2], Sifei Han[2], and Daniel Harris[2]

[1] Division of Biomedical Informatics, Department of Biostatistics
[2] Department of Computer Science
University of Kentucky, Lexington, KY
{ramakanth.kavuluru,eric.s.han,daniel.harris}@uky.edu

Abstract. Diagnosis codes are extracted from medical records for billing and reimbursement and for secondary uses such as quality control and cohort identification. In the US, these codes come from the standard terminology ICD-9-CM derived from the international classification of diseases (ICD). ICD-9 codes are generally extracted by trained human coders by reading all artifacts available in a patient's medical record following specific coding guidelines. To assist coders in this manual process, this paper proposes an unsupervised ensemble approach to automatically extract ICD-9 diagnosis codes from textual narratives included in electronic medical records (EMRs). Earlier attempts on automatic extraction focused on individual documents such as radiology reports and discharge summaries. Here we use a more realistic dataset and extract ICD-9 codes from EMRs of 1000 inpatient visits at the University of Kentucky Medical Center. Using named entity recognition (NER), graph-based concept-mapping of medical concepts, and extractive text summarization techniques, we achieve an example based average recall of 0.42 with average precision 0.47; compared with a baseline of using only NER, we notice a 12% improvement in recall with the graph-based approach and a 7% improvement in precision using the extractive text summarization approach. Although diagnosis codes are complex concepts often expressed in text with significant long range non-local dependencies, our present work shows the potential of unsupervised methods in extracting a portion of codes. As such, our findings are especially relevant for code extraction tasks where obtaining large amounts of training data is difficult.

1 Introduction

Extracting codes from standard terminologies is a regular and indispensable task often encountered in medical and healthcare fields. Diagnosis codes, procedure codes, cancer site and morphology codes are all manually extracted from patient records by trained human coders. The extracted codes serve multiple purposes including billing and reimbursement, quality control, epidemiological studies,

* corresponding author

and cohort identification for clinical trials. In this paper we focus on extracting international classification of diseases, clinical modification, 9th revision (ICD-9-CM) diagnosis codes from electronic medical records (EMRs), although our methods are general and also apply to other medical code extraction tasks.

Diagnosis codes are the primary means to systematically encode patient conditions treated in healthcare facilities both for billing purposes and for secondary data usage. In the US, ICD-9-CM (just ICD-9 henceforth) is the coding scheme still used by many healthcare providers while they are required to comply with ICD-10-CM, the next and latest revision, by October 1, 2014. Regardless of the coding scheme used, both ICD code sets are very large, with ICD-9 having a total of 13,000 diagnoses while ICD-10 has 68,000 diagnosis codes [1] and as will be made clear in the rest of the paper, our methods will also apply to ICD-10 extraction tasks. ICD-9 codes contain 3 to 5 digits and are organized hierarchically: they take the form `abc.xy` where the first three character part before the period `abc` is the main disease category, while the `x` and `y` components represents subdivisions of the `abc` category. For example, the code `530.12` is for the condition *reflux esophagitis* and its parent code `530.1` is for the broader condition of *esophagitis* and the three character code `530` subsumes all *diseases of esophagus*. Any allowed code assignment should at least assign codes at the category level (that is, the first three digits). At the category levels there are nearly 1300 different ICD-9 codes. In our current work, we only work on predicting the category level codes. That is, if the actual code is `abc.xy`, our methods will only be able to generate `abc` as the correct category code.

The process of assigning diagnosis codes is carried out by trained human coders who look at the entire EMR for a patient visit to assign codes. Majority of the artifacts in an EMR are textual documents such as discharge summaries, operative reports, and progress notes authored by physicians, nurses, or social workers who attended the patient. The codes are assigned based on a set of guidelines [2] established by the National Center for Health Statistics and the Centers for Medicare and Medicaid Services. The guidelines contain rules that state how coding should be done in specific cases. For example, the signs and symptoms (`780-799`) codes are often not coded if the underlying causal condition is determined and coded.

In this paper we propose an unsupervised ensemble approach to extract ICD-9 codes and test it on a realistic dataset curated from the University of Kentucky Medical Center. Our approach is based on named entity recognition (NER), knowledge-based graph mining, and extractive text summarization methods. We emphasize that automatic medical coding systems, including our current attempt, are generally not intended to replace trained coders but are mainly motivated to expedite the coding process and increase the productivity of medical record coding and management. Hence we take a recall oriented approach with a lesser emphasis on precision. In the rest of the paper, we first discuss related work and the context of our paper in Section 2. We describe our dataset in Section 3 and elaborate our methods in Section 4. We provide an overview of the evaluation measures in Section 5 and present our results in Section 6.

2 Related Work

Several attempts have been made to extract ICD-9 codes from clinical documents since the 1990s. Advances in natural language and semantic processing techniques contributed to a recent surge in automatic extraction. de Lima et al. [3] use a hierarchical approach utilizing the alphabetical index provided with the ICD-9-CM resource. Although completely unsupervised, this approach is limited by the index not being able to capture all synonymous occurrences and also the inability to code both specific exclusions and other condition specific guidelines. Gunderson et al. [4] extracted ICD-9 codes from short free text diagnosis statements that were generated at the time of patient admission using a Bayesian network to encode semantic information. However, in the recent past, concept extraction from longer documents such as discharge summaries has gained interest. Especially for ICD-9 code extraction, recent results are mostly based on the systems and dataset developed for the BioNLP workshop shared task on multi-label classification of clinical texts [5] in 2007.

An important issue in clinical document analysis is the absence of datasets that are free to use by other researchers due to patient privacy concerns and regulations. The BioNLP shared task [5] takes an important first step in providing such a dataset which consists of 1954 radiology reports arising from outpatient chest x-ray and renal procedures and are observed to cover a substantial portion of pediatric radiology activity. The radiology reports were also formatted in XML with explicit tags for *history* and *impression* fields. Finally, there were a total of 45 unique codes and 94 distinct combinations of these codes in the dataset. The dataset was split into training and testing sets of nearly equal size where example reports for all possible codes and combinations occur in both sets. This means that all possible combinations that will be encountered in the test set are known ahead of time. The top system obtained a micro-average F-score of 0.89 and 21 of the 44 participating systems scored between 0.8 and 0.9. Next we list some notable results that fall in this range obtained by various participants and others who used the dataset later. The techniques used range from completely handcrafted rules to fully automated machine learning approaches. Aronson et al. [6] adapted a hybrid MeSH term indexing program MTI that is in use at the National Library of Medicine (NLM) and included it with SVM and k nearest neighbor classifiers for a hybrid *stacked* model. Goldstein et al. [7] applied three different classification approaches - traditional information retrieval using the search engine library Apache Lucene, Boosting, and rule-based approaches. Crammer et al. [8] use an online learning approach in combination with a rule-based system. Farkas and Szarvas [9] use an interesting approach to induce new rules and acquire synonyms using decision trees.

We believe that the coverage of pediatric radiology activity, the small number of codes and their combinations where code combinations are known ahead of time, and the clear demarcation of history and impression fields do not provide a realistic representation of EMRs, especially for in-patient visits. It is well known that ICD-9 codes are extracted from the full EMR [10] for each in-patient visit where the EMR includes documents such as emergency department

notes, discharge summaries, radiology reports, pathology reports, operative reports, progress notes, and multiple flow sheets. Aronson et al. [6] also discuss the narrow focus on cough/fever/pneumonia and urinary/kidney problems and the relatively error-free clinical text present in the BioNLP radiology report dataset as a possible limitation for the extensibility of techniques to generalized EMRs.

3 In-Patient EMR Dataset

As a first step to study automatic diagnosis coding at the EMR level, we curated a dataset of 1000 clinical document sets corresponding to a randomly chosen set of 1000 in-patient visits to the University of Kentucky (UKY) Medical Center in the month of February, 2012. We also collected the ICD-9-CM codes for these EMRs assigned by trained coders at the UKY medical records office. Aggregating all billing data, this dataset has a total of 7480 diagnoses leading to 1811 unique ICD-9 codes that map to 633 top level codes (three character categories). Using the (code, label, count) representation the top 5 most frequent codes are (`401`, *essential hypertension*, 325), (`276`, *Disorders of fluid electrolyte and acid-base balance*, 239), (`305`, *nondependent abuse of drugs*, 236), (`272`, *disorders of lipoid metabolism*, 188), and (`530`, *diseases of esophagus*, 169). The average number of codes is 7.5 per EMR with a median of 6 codes. There are EMRs with only one code, while the maximum number assigned to an EMR is 49 codes. For each in-patient visit, the original EMR consisted of several documents, some of which are not conventional text files but are stored in the RTF format. Some documents, like care flowsheets, vital signs sheets, ventilator records were not considered for this analysis. We have a total of 5583 documents for all 1000 EMRs. While our correct codes arise from billing data where different trained coders code each EMR (one per coder), the BioNLP shared task dataset is a high quality dataset coded by three different companies with a final correct code set generated by consolidation of the three sets of codes.

Before we proceed to our methods, we note that after a discussion with our medical records officer, we learned that the coders do not necessarily code conditions purely from a billing perspective. On the contrary, they are trained to extract all codes following the coding guidelines even if the patient may not be billed for them eventually. However, as explained towards the end of Section 1, the coding guidelines might not allow certain codes even though they are discussed in the EMRs because of some specific restrictions on how coding should be done. In our unsupervised methods we do not model the logic behind the coding guidelines and hence our approach is primarily a recall oriented one. However, we use text summarization techniques to weed out codes (so, to improve precision) that are extracted from noun phrases (in EMR narratives) using certain statistical measures (more on this later).

4 Our Approach

To extract diagnosis codes, we used a combination of three methods: NER, knowledge-based graph mining, and extractive text summarization. In this sec-

tion we elaborate on the specifics of each of these methods. Before we proceed, we first discuss the Unified Medical Language System (UMLS), a biomedical knowledge base used in our NER and graph mining methods.

4.1 Unified Medical Language System

The UMLS[3] is a large domain expert driven aggregation of over 160 biomedical terminologies and standards. It functions as a comprehensive knowledge base and facilitates interoperability between information systems that deal with biomedical terms. It has has three main components: Metathesaurus, Semantic Network, and SPECIALIST lexicon. The Metathesaurus has terms and codes, henceforth called *concepts*, from different terminologies. Biomedical terms from different vocabularies that are deemed synonymous by domain experts are mapped to the same Concept Unique Identifier (CUI) in the Metathesaurus. The semantic network acts as a typing system that is organized as a hierarchy with 133 *semantic types* such as *disease or syndrome, pharmacologic substance,* or *diagnostic procedure*. It also captures 54 important relationships (or relation types) between biomedical entities in the form of a relationship hierarchy with relationships such as *treats, causes,* and *indicates*. The Metathesaurus currently has about 2.8 million concepts with more than 12 million relations connecting these concepts. Although relations in the Metathesaurus have relation types that are beyond the 54 available through the semantic network, here we would like to limit ourselves to high level relation types such as *parent, child, rel_narrow,* and *rel_broad*. The high level relations can be represented as $C1 \rightarrow < rel - type > \rightarrow C2$ where $C1$ and $C2$ are concepts in the UMLS and $< rel - type > \in \{parent, child, rel\_narrow, rel\_broad\}$. The semantic interpretation of these relations (or triples) is that the $C1$ is related to $C2$ via the relation type $< rel - type >$. The *child* (resp. *parent*) relationship means that concept $C1$ has $C2$ as a child (resp. *parent*). The *rel_broad* (resp. *rel_narrow*) type means that $C1$ represents a broader (resp. narrower) concept than $C2$. For example, the concept *hypertensive disease* is a broader concept compared to *systolic hypertension*. These broad and narrow relationships are created by experts to capture those relationships that cannot be captured by the more rigid parent/child relationships in different source vocabularies. The SPECIALIST lexicon is useful for lexical processing and variant generation of different biomedical terms.

4.2 Named Entity Recognition: MetaMap

NER is a well known application of natural language processing (NLP) techniques where different entities of interest such as people, locations, and institutions are automatically recognized from mentions in free text (see [11] for a survey). Named entity recognition in biomedical text is difficult because linguistic features that are normally useful (e.g., upper case first letter, prepositions before an entity) in identifying generic named entities are not useful when

[3] UMLS Reference Manual: `http://www.ncbi.nlm.nih.gov/books/NBK9676/`

identifying biomedical named entities, several of which are not proper nouns. Hence, NER systems in biomedicine rely on expert curated lexicons and thesauri. In this work, we use MetaMap [12], a biomedical NER system developed by researchers at the NLM. MetaMap uses a dictionary based approach (using the UMLS concept names as the dictionary) in combination with heuristics for partial mapping (based on lexical information in the SPECIALIST lexicon) to extract UMLS concepts. MetaMap can process a textual document as a whole but can also generate UMLS concepts from individual noun phrases that are passed as input to it. The latter option is more helpful to identify more specific concepts from longer phrases. Since more specific diagnosis codes are more valuable than generic codes (systolic hypertension vs hypertension), we used the latter approach called "term processing" in MetaMap's manual. MetaMap also identifies negations of concepts and hence we only used non-negated disorders when extracting codes. So as the first step in our code extraction pipeline, we extract biomedical named entities by running MetaMap on noun phrases that satisfy the following regular expressions based on those used in the paper [13].

1. `Noun* Noun`.
2. `(Adj|Noun)+ Noun`, and
3. `((Adj|Noun)+ | ((Adj|Noun)* (NounPrep)?)(Adj|Noun)*)Noun`

Here `Adj` stands for adjective and `NounPrep` stands for a noun followed by a preposition. Note that we allow the presence of a single preposition to capture phases like "malignant neoplasm of colon". We also allow single token noun phrases that just consist of one noun. For instance, both "hypertension" and "systolic hypertension" will be processed by MetaMap for concept extraction when the latter phrase occurs in text. However, we have a way of assigning more weight to specific codes using key phrase extraction covered in Section 4.4. Once we obtain non-negated UMLS concepts using MetaMap from these phrases, we convert these concepts to ICD-9 diagnosis codes when possible as explained next.

ICD-9-CM is one of the over 160 source vocabularies integrated into the UMLS Metathesaurus. As such, concepts in ICD-9-CM also have a concept unique identifier (CUI) in the Metathesaurus. As part of its output, for each concept, MetaMap also gives the source vocabulary. The concepts from MetaMap with source vocabulary ICD-9-CM finally become the set of extracted codes for each EMR document set. However, this code set may not be complete because of missing relationships between UMLS concepts. That is, in our experience, although MetaMap identifies a disorder concept, it might not always map it to a CUI associated with an ICD-9 code; it might map it to some other terminology different from ICD-9, in which case we miss a potential ICD-9 code because the UMLS mapping is incomplete. We deal with this problem and explore a graph based approach in the next section.

4.3 UMLS Knowledge-Based Graph Mining

As discussed in Section 4.2, the NER approach might result in poor recall because of lack of completeness in capturing synonymy in the UMLS. However, using

the UMLS graph with concepts (or equivalently CUIs) as nodes and the inter-concept relationships connected by relation types *parent* and *rel_broad* as edges, we can map a original CUI without an associated ICD-9 code to a CUI with an associated diagnosis code. We adapt the approach originally proposed by Bodenreider et al. [14] for this purpose. The mapping algorithm starts with a CUI c output by MetaMap that is not associated with an ICD-9 code and tries to map it to an ICD-9 codes as follows.

1. Recursively, construct a subgraph G_c (of the UMLS graph) consisting of ancestors of the input non-ICD-9 CUI c, using the *parent* and *rel_broad* edges. Build a set I_c of all the ICD-9 concepts associated with nodes added to G_c along the way in the process of building G_c. Note that many nodes added to G_c may not have associated ICD-9 codes.
2. Delete any concept c_1 from I_c if there exists another concept c_2 such that
   - c_1 is an ancestor of c_2, and
   - The length of the shortest path from c to c_2 is less than the length of the shortest path from c to c_1.
3. Return the ICD-9 codes of remaining concepts in I_c and the corresponding shortest distances from c.

Note that the algorithm essentially captures ancestors of the input concept and tries to find ICD-9 concepts in them. We also see that instead of returning a single code, the algorithm returns a set of ICD-9 codes (possibly singleton or empty). If the set has more than one code, all resulting ICD-9 codes are included in the extracted code set for performance evaluation purposes.

4.4 Extractive Text Summarization: C-value Method

Extractive text summarization is an approach where short summaries of a collection of documents are generated by selecting a few sentences or phrases from those documents that represent the gist of the collection in some way. Key phrase extraction algorithms including the C-value method [13] and TextRank [15] belong to a category of summarization algorithms that extract top phrases that capture a summary of a collection of documents. In this paper, we apply the C-value method to rank the noun phrases that were used to extract ICD-9 codes in Section 4.2. The ranking on the noun phrases automatically imposes a ranking on the codes extracted from them using the approaches outlined in Sections 4.2 and 4.3. The C-value of a noun phrase is computed based on its frequency and the frequencies of longer phrases that contain it in the given set of documents. We use all the documents in an EMR as the corpus to extract candidate noun phrases for code extraction. Hence, we also use the same set of documents to compute the frequencies of all phrases required to compute the C-value of a given phrase. The C-value formula can be written as

$$C(p) = \begin{cases} \log_2(len(p)) \cdot f(p) & \text{if } p \text{ is not nested} \\ \log_2(len(p)) \cdot \left(f(p) - \frac{1}{|T_p|} \sum_{q \in T_p} f(q) \right) & \text{if } p \text{ is nested} \end{cases}$$

where $C(p)$ is the C-value of phrase p, $len(p)$ is number of words in p, and T_p is the set of the longer noun phrases that contain p, and $f(p)$ is the frequency of p in all documents from an EMR. If p is not nested, it implies that it does not appear in longer phrases. When it is nested, we discount its C-value based on the number of its occurrences in longer phrases (the $\sum_{q \in T_p} f(q)$ part) and dampen this discount based on the number of unique longer phrases that contain it (the $\frac{1}{|T_p|}$ part). We chose to include all codes arising from phrases whose C-value is ≥ 1. Although we had phrases that had C-value as high as 20, including only codes whose phrases had very high C-values resulted in many missed codes. Hence we chose those codes arising from phrases with C-values ≥ 1 based on the needs of our recall oriented task. Before applying the C-value filter to eliminate nonsignificant codes extracted using MetaMap and graph based methods, we also applied a different filter that eliminated codes that are extracted from a set of very frequent phrases (mostly single nouns) that result in common symptoms like cold. This is akin to a stop word list used in information retrieval and text classification research.

5 Evaluation Measures

Before we discuss our findings, we establish notation to be used for evaluation measures. Let M be the set of all EMR records; here $|M| = 1000$ since we have 1000 EMRs. Let E_i and B_i, $i = 1, \ldots, 1000$, be the set of extracted codes using our methods from EMR documents and the corresponding set of billing codes respectively for the EMR of the i-th in-patient visit. Since the task of assigning multiple codes to an EMR is the multi-label classification problem, there are multiple complementary methods [16] for evaluating automatic approaches for this task. Here we use EMR-based precision and recall and code label based micro and macro precision, recall, and F-score. First we discuss the EMR-based measures. The average EMR-based precision P_{emr} and recall R_{emr} are

$$P_{emr} = \frac{1}{|M|} \cdot \sum_{i=1}^{|M|} \frac{|E_i \cap B_i|}{|E_i|} \quad \text{and} \quad R_{emr} = \frac{1}{|M|} \cdot \sum_{i=1}^{|M|} \frac{|E_i \cap B_i|}{|B_i|}.$$

On the other hand, considering each code as a label, we define the code-based measures. For each code C_j in the billing code set C of the dataset, we have code-based precision $P(C_j)$, recall $R(C_j)$, and F-score $F(C_j)$ defined as

$$P(C_j) = \frac{TP_j}{TP_j + FP_j},\ R(C_j) = \frac{TP_j}{TP_j + FN_j},\ \text{and}\ F(C_j) = \frac{2P(C_j)R(C_j)}{P(C_j) + R(C_j)},$$

where TP_j, FP_j, and FN_j are true positives, false positives, and false negatives, respectively of code C_j. Now code-based macro average precision, recall, and F-score are defined as

$$P_c^{macro} = \frac{\sum_{j=1}^{|C|} P(C_j)}{|C|},\ R_c^{macro} = \frac{\sum_{j=1}^{|C|} R(C_j)}{|C|},\ \text{and}\ F_c^{macro} = \frac{\sum_{j=1}^{|C|} F(C_j)}{|C|},$$

respectively. Finally, the code-based micro precision, recall, and F-score are defined as

$$P_c^{micro} = \frac{\sum_{j=1}^{|C|} TP_j}{\sum_{j=1}^{|C|}(TP_j + FP_j)}, \; R_c^{micro} = \frac{\sum_{j=1}^{|C|} TP_j}{\sum_{j=1}^{|C|}(TP_j + FN_j)},$$

$$\text{and } F_c^{micro} = \frac{2P_c^{micro}R_c^{micro}}{P_c^{micro} + R_c^{micro}},$$

respectively. While the macro measures consider all codes as equally important, micro measures tend to give more importance to codes that are more frequent.

6 Results and Discussion

First we present our EMR based average precision and recall results in Table 1. In our experiments, ICD-9 codes that are associated with concepts at a distance greater than 1 from the input concept in the graph mining approach (Section 4.3) improved recall by only 1% with a 2% decrease in precision. Hence here we only report results when the shortest distance between the input concept and the ICD-9 ancestors is ≤ 1. Distance zero codes are those that are directly obtained from MetaMap output without having to use the graph mining method. The "No C-value" column in all the tables in this section means that C-value restriction is not applied to the noun phrases used for code extraction.

	without graph-mining		graph distance ≤ 1	
	No C-value	C-value ≥ 1	No C-value	C-value ≥ 1
R_{emr}	0.30	0.30	0.42	0.42
P_{emr}	0.47	0.53	0.40	0.47

Table 1: EMR-Based Average Precision and Recall

Using the graph mining approach we see an improvement of 12% in recall from 0.3 to 0.42. Without any recall loss, the C-value method improves precision by 7% when using the graph mining approach and by 6% when not using it. Thus we see a clear advantages of the key phrase scoring approach in increasing precision and the knowledge based graph mining approach in increasing recall. The 99% confidence interval ranges when using C-value ≥ 1 without graph mining are $0.28 \leq R_{emr} \leq 0.32$ and $0.50 \leq P_{emr} \leq 0.56$; the same ranges using both C-value and graph mining are $0.38 \leq R_{emr} \leq 0.44$ and $0.45 \leq P_{emr} \leq 0.49$.

Next we present our macro averaged recall, precision, and F-scores in Table 2. These results provide a contrast to the observations made in the EMR based measures. Although there is an 8% increase in recall using the graph mining approach, we notice that the recall gain comes at an expense of 10% loss in precision. However, we believe this is a still a reasonable although not ideal situation especially considering that our goal is a recall oriented approach to expedite the coding process. The C-value method increases precision by an amount equal to the loss in recall. However, this is not ideal as recall is more important to us.

But we note that macro measures give equal importance to all codes. That is codes that occur very infrequently are also scored the same way frequent scores are scored.

	without graph-mining		graph distance ≤ 1	
	No C-value	C-value ≥ 1	No C-value	C-value ≥ 1
R_c^{macro}	0.58	0.53	0.66	0.62
P_c^{macro}	0.74	0.79	0.64	0.69
F_c^{macro}	0.57	0.56	0.57	0.58

Table 2: Code-Based Macro Precision and Recall

Before we discuss micro measures, we note that several codes that were in billing were never extracted using our methods (more on this in the Discussion section). The F-scores for all those codes will be zero. Thus, we chose to compute micro measures over subsets of the set of all billing codes C. We choose two particular subsets. The first set is the set of all codes in billing that were retrieved at least once using a particular configuration of our methods; so these are the set of codes C_j for which the F-score $F(C_j) > 0$, whose results are presented in Table 3. Since the F-score changes with the method, we also show the number of codes that satisfy $F(C_j) > 0$ for each technique as the last row.

	without graph-mining		graph distance ≤ 1	
	No C-value	C-value ≥ 1	No C-value	C-value ≥ 1
R_c^{micro}	0.48	0.40	0.57	0.48
P_c^{micro}	0.53	0.64	0.44	0.53
F_c^{micro}	0.50	0.49	0.50	0.50
$\lvert\{j : F(C_j) > 0)\}\rvert$	277	271	370	365

Table 3: Code-Based Micro Measures with $F(C_j) > 0$

	without graph-mining		graph distance ≤ 1	
	No C-value	C-value ≥ 1	No C-value	C-value ≥ 1
R_c^{micro}	0.64	0.61	0.69	0.62
P_c^{micro}	0.70	0.75	0.66	0.71
F_c^{micro}	0.67	0.67	0.68	0.66
$\lvert\{j : F(C_j) > 0.5)\}\rvert$	170	168	237	240

Table 4: Code-Based Micro Measures with $F(C_j) > 0.5$

We also computed micro measures over the set of codes that satisfy $F(C_j) > 0.5$ whose results are shown in Table 4. We realize that showing micro measures for codes that were extracted at least once may overestimate the performance of methods, which is not our intention. Our only purpose of showing these results is to demonstrate how our unsupervised approach works on a subset of the codes and to quantify the difference between different components of our approach.

Out of 633 total possible codes, the NER approach extracted 277 (43% of 633) and using the graph mining approach we have 370 (58% of 633). In both tables, the pattern we see in the macro measures repeats where the increase in recall due to the graph mining approach is offset by a decrease in precision.

To understand the nature of our errors, we went back and looked at some of the codes that were causing high recall and precision errors. For example, from Table 3, we can see that even with the graph based approach only 58% of the codes were extracted. We found out that this is because there are a set of codes that never get extracted due to the complex ways in which they manifest in free text. A main class of such codes is the set of E (external cause of injury or poison) and V (encounters with circumstances other than disease or injury) codes. These codes do not generally manifest in free text as noun phrases and have evidence spread throughout the document with non-local dependencies. Out of a total of 92 E and V codes in our dataset, 85 were never extracted; the micro average recall over all E and V codes is 0.01. Similar to E and V codes, there are other classes of codes that rely on non-local dependencies in textual documents. One of them is the set of codes that deal with pregnancy and childbirth (codes `630-670`). In our dataset, over a total of 28 codes that belong to this class, the average recall was 0.25. When it comes to precision errors, most of the errors were caused by common symptoms such as *pain* and *bruising* that are generic and are not coded in many cases based on coding guidelines.

7 Conclusion

In this paper we presented an unsupervised ensemble approach to extract diagnosis codes from EMRs. We used a biomedical NER system MetaMap for the basic recognition of biomedical concepts in EMRs and mapped them to ICD-9 codes using the UMLS Metathesaurus . We then used graph mining to exploit the UMLS relationship graph to extract candidate ICD-9 codes for those disorder concepts output by MetaMap that did not have an associated ICD-9 code. We show a 12% improvement in EMR based average recall with this approach. Next, we used key phrase extraction using the C-value method to improve EMR based average precision by 7%. To our knowledge, our results are the first to report on EMR level extraction of diagnosis codes using a large set of codes. Although machine learning approaches are important, we believe that unsupervised knowledge-based approaches are essential especially given that large amounts of biomedical data might not be available owing to privacy issues involving patient data. As future work, we plan to extract full ICD-9 codes instead of top level category codes and are working on combining our unsupervised methods with machine learning approaches to build a hybrid extraction system.

Acknowledgements

This publication was supported by the National Center for Research Resources and the National Center for Advancing Translational Sciences, US National Institutes of Health (NIH), through Grant UL1TR000117. The content is solely

the responsibility of the authors and does not necessarily represent the official views of the NIH.

References

1. American Medical Association: Preparing for the icd-10 code set. http://www.ama-assn.org/ama1/pub/upload/mm/399/icd10-icd9-differences-fact-sheet.pdf (2010)
2. National Center for Health Statistics and the Centers for Medicare and Medicaid Services. http://www.cdc.gov/nchs/icd/icd9cm.htm (2011)
3. de Lima, L.R.S., Laender, A.H.F., Ribeiro-Neto, B.A.: A hierarchical approach to the automatic categorization of medical documents. In: Proceedings of the 7th Iintl. Conf. on Inf. & Knowledge Mgmt. CIKM '98 132–139
4. Gundersen, M.L., Haug, P.J., Pryor, T.A., van Bree, R., Koehler, S., Bauer, K., Clemons, B.: Development and evaluation of a computerized admission diagnosis encoding system. Comput. Biomed. Res. **29**(5) (October 1996) 351–372
5. Pestian, J.P., Brew, C., Matykiewicz, P., Hovermale, D.J., 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. 97–104
6. Aronson, A.R., Bodenreider, O., Demner-Fushman, D., Fung, K.W., Lee, V.K., Mork, J.G., Neveol, A., Peters, L., Rogers, W.J.: From indexing the biomedical literature to coding clinical text: experience with mti and machine learning approaches. In: Biological, translational, and clinical language processing, Assc. for Comp. Ling. (2007) 105–112
7. Goldstein, I., Arzumtsyan, A., Uzuner, O.: Three approaches to automatic assignment of icd-9-cm codes to radiology reports. In: Proceedings of AMIA Symposium. (2007) 279–283
8. Crammer, K., Dredze, M., Ganchev, K., Pratim Talukdar, P., Carroll, S.: Automatic code assignment to medical text. In: Biological, translational, and clinical language processing, Assc. for Comp. Ling. (2007) 129–136
9. Farkas, R., Szarvas, G.: Automatic construction of rule-based icd-9-cm coding systems. BMC Bioinformatics **9**(S-3) (2008)
10. Pakhomov, S.V.S., Buntrock, J.D., Chute, C.G.: Automating the assignment of diagnosis codes to patient encounters using example-based and machine learning techniques. J. American Medical Informatics Assoc. **13**(5) (2006) 516–525
11. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **30**(1) (2007) 3–26
12. Aronson, A.R., Lang, F.M.: An overview of metamap: historical perspective and recent advances. J. American Medical Informatics Assoc. **17**(3) (2010) 229–236
13. Frantzi, K.T., Ananiadou, S., Tsujii, J.i.: The c-value/nc-value method of automatic recognition for multi-word terms. In: Second European Conf. on Research and Advanced Tech. for Digital Libraries. ECDL '98 (1998) 585–604
14. Bodenreider, O., Nelson, S., Hole, W., Chang, H.: Beyond synonymy: exploiting the umls semantics in mapping vocabularies. In: Proceedings of AMIA Symposium. (1998) 815–819
15. Mihalcea, R., Tarau, P.: Textrank: Bringing order into text. In: Proceedings of EMNLP. (2004) 404–411
16. Tsoumakas, G., Katakis, I., Vlahavas, I.P.: Mining multi-label data. In: Data Mining and Knowledge Discovery Handbook. (2010) 667–685