Fine-Grained ICD Code Assignment Using Ontology-Based Classification

Joshua Carberry
Computer and Information Science Department
University of Massachusetts Dartmouth
Dartmouth, MA 02747, USA
jcarberry@umassd.edu

Haiping Xu
Computer and Information Science Department
University of Massachusetts Dartmouth
Dartmouth, MA 02747, USA
hxu@umassd.edu

***Abstract*—Assigning International Classification of Diseases (ICD) codes based on doctors' clinical diagnoses has historically been a difficult task performed by highly trained clinical coding experts. Recently, attempts have been made to use machine learning techniques at a coarse-grained level to automatically generate lists of medical codes from doctors' notes; however, the results are often difficult to interpret and validate. In this paper, we propose a fine-grained approach that focuses on one diagnosis at a time. We use ontology-based human knowledge to extract semantically related sentences from doctor's notes to support the use of deep learning for reliable training and classification. This fine-grained deep learning approach significantly reduces training load and improves scalability while providing users with a rationale for ICD code prediction. To demonstrate the effectiveness and advantages of our approach, we apply it to the MIMIC-III dataset and show how ICD-9 codes can be automatically assigned to clinical diagnoses.**

Keywords—doctor's notes, ICD code, ontology, deep learning, fine-grained code assignment

I. Introduction

Healthcare is one of the most important and fastest growing industries, and many data-rich problems related to healthcare have been actively studied in recent years. One such problem is medical coding, or the assignment of standardized medical codes to healthcare diagnoses, procedures, medical services and equipment. In a worldwide industry like healthcare, it is crucial to reduce facts to standardized codes that benefit not only communication between hospitals, but also financial institutions. Some of the most important uses of medical diagnosis codes are related to billing and insurance [1]. There are many coding standards; in this research, we focus on the standard presented by the International Classification of Diseases, Ninth Revision (ICD-9), a robust system of 16 chapters, each with many sections for disease diagnoses [2]. Due to the complexity of the healthcare industry and its interaction with financial institutions, medical coding is a non-trivial issue that has historically been performed only by highly trained clinical coding experts. When performed manually, medical coding can be a time-consuming, error-prone and expensive task. For this reason, there is a great deal of interest in developing automated solutions for medical coding and, more specifically, for the automatic assignment of medical codes from electronic medical records, such as doctors' notes, that have been generated during the medical procedures.

In this paper, we specifically examine the problem of coding diagnoses using doctor's notes associated with a patient's hospital visit. When patients are discharged from hospital visits, they are typically given diagnoses that describe their condition at the time of the visit. For example, a patient with high blood pressure might receive a diagnosis of "hypertension." In addition, during a given patient's hospitalization, doctors also make extensive notes about their condition, their experience, and any procedures or interventions performed. These notes are written in natural language and are usually unstructured. Often, they contain many details that are not directly useful for diagnoses, such as descriptions of peripheral events, superfluous tracking of dates and times, and so on. A trained medical coding expert is tasked with reading these notes and assigning the appropriate diagnosis codes. Due to the sheer volume of text and specialized language usage, this manual approach may lead to significant time expenditures. In this paper, we attempt to mitigate these issues by introducing a method that can assist the diagnosis code assignment process by predicting medical codes for the diagnoses in the doctor's notes.

In recent years, with the adoption of electronic medical records and the release of large-scale medical datasets such as MIMIC-III (Medical Information Mart for Intensive Care), research has shifted from more explicit solutions (e.g., rule-based systems) to deep learning-based approaches. Existing methods have achieved reasonable accuracy in assigning ICD-9 diagnosis codes to clinical doctors' notes; however, coarse-grained approaches that apply deep learning across entire doctor's notes raise concerns in the medical field because the results are typically not justifiable. In this paper, we present an evidence-based, fine-grained approach that automatically assigns an ICD code to each individual diagnosis. The proposed approach first uses human knowledge from existing ontologies to extract semantically related sentences from doctor's notes to support the code assignment for a given diagnosis. These semantically related sentences are combined with the diagnosis and then fed into a trained deep learning classifier to predict the medical code for the given diagnosis. With this fine-grained classification, the extracted semantically related sentences can be seen as evidence to justify the classification process, and help the user understand the information used in the prediction or resolve any doubts before a code is finally assigned. In addition, the method uses only one diagnosis and its semantically related sentences as input; therefore, it can significantly reduce the training load and improve the scalability of the classifier.

978-1-6654-6603-5/22/$31.00 ©2022 IEEE
DOI 10.1109/IRI54793.2022.00058

II. Related Work

Automated medical coding has been actively investigated due to the high cost of using doctors' notes to assign ICD diagnosis codes. Traditional methods were initially employed due to the scarcity of domain data. Farkas and Szarvas used an enhanced rule-based approach to predict ICD-9 codes [3]. They enriched existing expert rules and used decision trees and a max entropy classifier to detect and address the false negatives they produced. Medori and Fairon used a naïve Bayes classifier to make predictions for ICD coding [4]. In a comparison between different methods based on naïve Bayes, they found that stemming and information encoding could greatly increase recall. Perotte et al. introduced several metrics for ICD code prediction and used them to rate the effectiveness of support vector machines (SVM) [5]. They introduced a hierarchical SVM that leveraged the hierarchical organization of ICD codes to improve performance. As large-scale deidentified datasets were gradually made available, deep learning approaches became more feasible beyond these earlier approaches. The public release of MIMIC-III in 2016 was a turning point for the application of deep learning to the ICD coding problem [6].

In general, deep learning has been used as a multi-label classification method to classify doctors' notes at a coarse-grained level into one or more matching ICD codes. Baumel et al. presented a case study on ICD code assignment using multi-label classification [7]. They investigated four models for assigning multiple ICD codes to discharge summaries and introduced a hierarchical attention mechanism to tag a document by identifying the relevant sentences. Falis et al. employed a hierarchical classifier incorporating parent and grandparent codes in addition to child code classification [8]. Their classifier starts with a multi-view convolution module whose output is pooled and fed into an ensemble of attention-based classifiers that output the code predictions. Recently, Biseda et al. introduced an approach using clinical BERT embeddings to improve classifications [9]. They used a hierarchical classifier that first predicts in 16 ICD code chapters and then in 50 selected ICD codes contained in each particular chapter. Although the above approaches employ hierarchical classifiers to break the classification into multiple steps, the number of final output nodes may still be high. When multi-label classification is used, the high number of label combinations can become potentially intractable due to high training costs and data scarcity, especially given the continued growth of medical knowledge and data. In addition, the above methods all predict ICD codes at a coarse-grained level by viewing doctors' notes in their entirety. Since doctors' notes can be very lengthy (e.g., thousands of words), the data points used to train their classifiers can become very large, leading to a significant increase in training costs. In contrast to existing work, we propose an approach that avoids multi-label classification and reduces scalability issues by treating each diagnosis separately. The processing of individual diagnoses also provides users with specific evidence in each prediction, improving human understanding and allowing quick resolution of doubts about code recommendations. Since this evidence is extracted using existing human knowledge, our approach does not require the use of an additional attention mechanism on the training data.

Healthcare is an extremely large and widespread industry. Ontologies are often used in healthcare to unify knowledge and information from heterogeneous sources [10]. Much of the previous work on medical ontologies has revolved around the construction of unified ontologies from these heterogeneous information sources, and they are usually evaluated by expert examination or by performance in applications. One example comes from a researcher group that rated the performance of the Gene Ontology in an enrichment analysis task [11]. Essentially, the goal of this task is to identify genomes that are associated with diseases, which could be very important in real medicine. Ong et al. used medical ontologies to model kidney diseases' underlying pathology and locate potential treatments [12]. They showed that the KPMP (Kidney Precision Medicine Project) ontologies can improve the concepts used to annotate kidney data and revise existing definitions of kidney disease to support precision medicine. Jusoh et al. tackled the task of information extraction from medical natural language texts [13]. They proposed a method to generate ontologies from mined texts by extracting key entities and relations from natural language. In this paper, we propose an approach using existing ontologies to address the medical coding issue. That is, we use existing disease ontologies to identify terms that are semantically related to a given diagnosis. Then the identified terms are used to extract content from a free text that can be used as evidence for the prediction of a diagnosis's ICD code. In this sense, our approach complements existing ontology-based approaches in medical fields and provides a practical solution to automated medical coding using well-developed disease ontologies.

III. Ontology-Based Classification for ICD Coding

A. A Framework for Ontology-Based Classification

The ontology-based ICD code assignment process begins with the discharge diagnoses listed at the end of doctor's notes. Unlike existing approaches that treat the entire doctor's notes, including the discharge diagnoses, as a single entity, our approach takes a more granular view of these notes, generating a different view for each discharge diagnosis by extracting a set of semantically related sentences from the doctor's notes. Since the text fragments related to individual diagnosis are much shorter than the full text, this increased granularity improves scalability by significantly reducing the size of the input sequence. It also allows us to avoid performing multi-label classification, as each discharge diagnosis is assumed to correspond to exactly one ICD code label. In addition, the semantically related sentences extracted for each diagnosis provide users with context for final decisions on medical coding and help them resolve any issues in code assignment.

Fig. 1 shows a framework for ontology-based classification of ICD codes. As shown in the figure, the automatic code assignment process involves several steps. In the first step, for each discharge diagnosis, the semantically related sentences are extracted from the doctor's notes using ontology-based domain knowledge, i.e., a medical ontology. The extracted semantically related sentences are then appended to the corresponding diagnosis to constitute a new data point. Prior to classification, the natural language in the new data points is preprocessed, including the removal of useless words and stemming of words. Words in a data point are assigned word embeddings as an input to a trained classifier using a deep neural network such as a recurrent neural network (RNN). In this paper, we use long short-term memory (LSTM) artificial neural networks, which

are a type of RNN. In LSTM, each cell contains additional components to alleviate the vanishing gradient problem and improve the classifier's ability to exploit context during the classification process. After a prediction is generated, the user is able to see the assigned ICD code for a diagnosis as well as the semantically related sentences used in the classification process. Note that as shown in Fig. 1, the predictions of multiple data points can be performed in parallel using multiple processes.

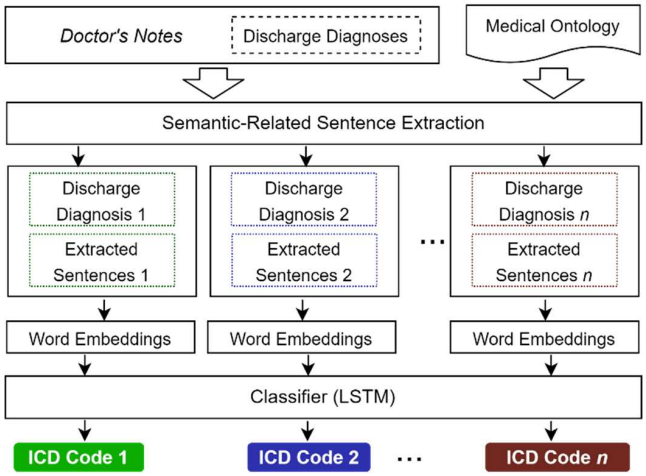

Fig. 1. A framework for ontology-based classification of ICD codes.

B. Training and Test Dataset

In this study, we use the MIMIC-III clinical database as the training and test dataset. The MIMIC-III database contains deidentified information for over 40,000 patients, including the free-text doctors' notes and ICD-9 diagnosis codes associated with unique hospital visits. For this research, only the free-text notes, including discharge diagnoses, and ICD-9 diagnosis codes are used. The other fields associated with a patient such as lab events and procedures are not considered in this approach.

Fig. 2 shows an example of the free-text doctor's notes including the discharge diagnoses. The doctor's notes are a text document ranging from a few paragraphs to dozens of pages in length, with some information expressed in natural language and some expressed in, for example, bulleted lists.

Cardiovascular: On telemetry, the patient was noted to have multiple premature ventricular contractions. These were asymptomatic and not treated. Due to the sudden episodes of pulmonary edema ...

DISCHARGE DIAGNOSES:
- Pulmonary edema.
- Congestive heart failure.
- Metastatic carcinoma.

Fig. 2. Example of free-text doctor's notes with discharge diagnoses.

At the end of many instances of doctor's notes, there is a numbered, bulleted, or comma-separated list indicating the major diagnoses associated with the doctor's visit. In our approach, the discharge diagnoses are split into separated diagnoses and processed independently. While not shown in Fig. 2, the database also contains a set of diagnosis codes matched to each visit. These diagnosis codes are helpful for labeling the data points, but they are often not sufficient. To build the training and test dataset, we matched each diagnosis and the extracted semantically related sentences with a medical code from a selected set. These matched codes serve as the labels of the data points. If a diagnosis does not have a matched medical code from the selected set, it is eliminated from the training and test dataset. Note that the diagnoses can be difficult to read, especially for non-experts, and often contain misspellings or abbreviations. Therefore, it is essential to use additional context from the natural language notes that precede the discharge diagnosis as supporting evidence during the labeling process.

C. Extracting Semantically Related Sentences

Since most of the doctors' notes are usually not related to the assignment of a specific diagnosis code, we extract only sentences that are semantically related to a particular diagnosis. In the sentence extraction procedure, we divide the doctors' notes into individual sentences and scan the sentences for terms that are semantically related to a diagnosis. Sentences with no related terms are skipped, while sentences containing one or more related terms are then extracted. Note that the detection of semantically related terms cannot be achieved without prior knowledge of the underlying relations between these terms and the targeted diagnosis. To detect semantically related terms, we use existing human knowledge in the form of an ontology.

Ontologies, as semantic data models, can be used to encode knowledge in a graph where the vertices represent entities (objects or abstract concepts) and the edges represent relations. The entities and relations can be encoded as lists of triples in the form <*head*, *relation*, *tail*>, where *head* is the first entity, *tail* is the second, and *relation* is the type of relationship that associates the two. For example, a piece of knowledge about the influenza virus could be encoded as the triple <*influenza*, *has_symptom*, *fever*>. In this paper, we used the Institute of Genome Science's Disease Ontology (DO), which contains knowledge on a wide range of diseases including classifications, symptoms, and synonyms [14]. Although DO is an ontology in which each link and entity type has a strong formal definition and meaning for semantic computing, we treat the ontology as a directed graph and use a simple neighborhood search algorithm that considers all entities and relations indiscriminately in searching for a set of semantically related terms.

Using ontologies, we can generate a set of semantically related terms within a certain number of relations to a specific entity. We first find the entity corresponding to a given diagnosis, then we recursively explore the graph within the desired number of links from the original diagnosis entity. Depending on the knowledge used, the ideal number of links to explore may vary. To simplify matters, in this study we only consider entities that are within one link of our target diagnosis entity, because information related to a specific disease entity is usually stored within one link and going further would involve unrelated diseases and unnecessarily reduce the effectiveness of sentence extraction. Fig. 3 shows an example of semantically related terms to "influenza" based on DO. Note that in our approach, it is necessary to consider both outgoing and incoming links in order to produce the best set of semantically related terms. As shown in the figure, "viral infectious disease" is not a symptom of "influenza"; however, the discussion of "viral infectious disease" in the doctor's notes can provide support for the identification of medical code for an influenza diagnosis.

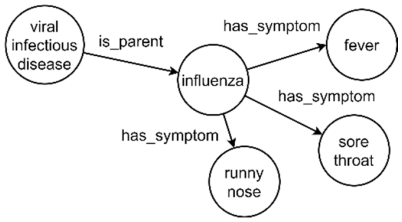

Fig. 3. An example of semantically related terms to "influenza."

Algorithm 1 shows how to generate unlabeled data points from doctor's notes using an ontology. As shown in the algorithm, for each diagnosis, semantically related terms identified using the ontology are added to set s_2. The terms in s_2 are then used to extract semantically related sentences from the doctor's notes. Finally, each diagnosis together with the list of extracted semantically related sentences constitutes a new unlabeled data point.

Algorithm 1: Generate Unlabeled Data Points Using Ontology

Input: Doctor's notes Ξ and medical ontology Φ
Output: A list of data points *ldata*

1. split the notes prior to diagnoses in Ξ into a list of sentences Σ
2. **for each** diagnosis ρ in Ξ
3. identify a set of key terms s_1 in ρ
4. initialize a set of terms $s_2 = s_1$
5. **for each** *term* in s_1
6. identify all *entities* in Φ that are semantically related to *term*
7. add *entities* to s_2
8. create a new data point *ndp*
9. let *extracted* be an empty list of sentences
10. **for each** sentence σ in Σ
11. **if** σ does not contain any term in s_2 **then continue**
12. append σ to *extracted*
13. add ρ and *extracted* to *ndp*, and add *ndp* to *ldata*
14. **return** *ldata*

An unlabeled data point, after pre-processing, can be used as an input to a trained classifier for predicting a matching medical code. However, with supervised deep learning, the generated data points used for training and testing must be labeled. As shown in the case studies in Section IV, in this research, we select a small set of medical codes for demonstration purposes. We keep only data points where the discharge diagnosis can be mapped to a medical code from the selected set and discard all others that do not have a matched medical code.

D. Preprocessing Data Points for Training a Classifier

Typical preprocessing procedures are applied to reduce the length of the text and size of data points without altering the meaning of the text. For example, stop word removal is applied to remove common and less useful words like "a" and "the." Stemming is applied to group families of words into one word by removing inflections. For example, "coughs," "coughing," and "coughed" are all reduced to the root word "cough." Then, we assign word embeddings, a reduced dimensionality vector representation of word meanings, to each remaining word. To achieve this task, we generate word embeddings using a popular and established approach called word2vec. Word2vec is an unsupervised learning method that takes a large collection of text and generates embeddings for the words contained in it. Essentially, the word2vec method works by associating words with their contexts, which are composed of the surrounding words. Words that frequently occur in similar contexts produce embeddings with some similarity. Word2vec has been shown to reasonably encode semantic similarities and relationships in generated embeddings and is used successfully in many natural language processing applications.

We adopt LSTM as a deep learning classifier for training, validation and prediction. In an LSTM, each cell contains additional components to mitigate the vanishing gradient problem and improve the classifier's ability to utilize context in classification. More specifically, each cell of an LSTM contains a number of gates that control the information entering and leaving the cell. Input gates and forget gates work together to control which information is passed into a given cell by its predecessors, while an output gate controls which information is passed to the next cell. LSTM has been shown effective on a number of natural language processing tasks, particularly where input sequences are longer and earlier context tends to be forgotten by traditional RNNs. In our approach, we reduce each data point to contain only sentences related to a diagnosis, but the extracted sentences are long enough to demonstrate the improvement in long- and short-term memory of the LSTM.

IV. Experimental Results and Case Studies

A. Ontology-Based Classifier

To obtain experimental results, the methodology described in Section III was applied to seven medical codes all contained in Chapter 7 of ICD-9, as listed in Table I. The codes selected were closely related to highlight the potential use of the method and to emphasize the ability of the classifier to distinguish between similar codes, which makes classification more difficult. The selection of diagnosis codes was limited to those codes that were adequately represented in the knowledge base DO. In addition, codes were selected on a frequency basis to ensure that sufficient data points were available to properly train the classifier. Finally, 56,891 data points were generated from the MIMIC-III dataset for the experiments.

TABLE I. ICD-9 Codes Used In Experiments

ID	*Code*	*Description*	*Frequency*
1	4011	Benign hypertension	444
2	4019	Unspecified essential hypertension	19,117
3	41401	Coronary atherosclerosis of native coronary artery	11,392
4	4260	Atrioventricular block, complete	492
5	42731	Atrial fibrillation	12,122
6	4271	Paroxysmal ventricular tachycardia	1,635
7	4280	Congestive heart failure, unspecified	11,689

A training/testing ratio of 80/20 is used to train and evaluate the classifier. During training, one-fifth of the training data was used for validation. Due to the highly imbalanced nature of the data (with code 4011 or code 4260 represents less than 1% of the dataset), stratified sampling was used to ensure that each code was represented in the training and validation sets. The sigmoid focal loss function was used during training, which prioritizes a small subset of difficult examples while downplaying the majority of well-classified examples [15]. Focal loss, originally developed for the object detection

problem, is highly effective for classification problems with imbalanced data, such as the problem that occurs in our dataset. Furthermore, weights were applied to the loss function to incentivize the classifier to perform well on the less frequent codes as well as the dominant codes. Tables II shows the classifier's overall precision, recall, and F1-score calculated as in (1), (2) and (3), respectively. In the table, the weighted averages are calculated by weighting the metrics for each class with their relative frequencies; while the macro averages are unweighted averages that treat all classes the same regardless of size. We can see that the classifier achieved a high accuracy (number of correct predictions over all cases) of 0.935, as well as a macro average F1-score of 0.912, with similarly favorable results for macro average precision and recall. This indicates that the classifier performed well and successfully predicted both large and small classes despite the unbalanced data.

$$\text{Precision} = (\text{true positives}) / (\text{true positives} + \text{false positives}) \quad (1)$$

$$\text{Recall} = (\text{true positives}) / (\text{true positives} + \text{false negatives}) \quad (2)$$

$$\text{F1} - \text{score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (3)$$

TABLE II. OVERALL CLASSIFIER PERFORMANCE METRICS

	Precision	***Recall***	***F1-score***
Weighted Average	0.937	0.935	0.935
Macro Average	0.904	0.923	0.912
Accuracy		0.935	

Tables III shows the classifier's precision, recall, and F1-score for individual codes. From the table we can see that the classifier performed best on the most frequent classes (with 10,000+ data points). Despite the high data imbalance and general data scarcity of the less frequent classes, the classifier was able to classify them with F1-scores of 0.834 or higher. The classifier performed worst on code 4011 (Benign hypertension), which in addition to being the least frequent code, shares many characteristics with the dominant code 4019 (Unspecified essential hypertension), creating ambiguities and complicating classification results.

TABLE III. CLASSIFIER PERFORMANCE METRICS ON INDIVIDUAL CODES

ID	***Code***	***Precision***	***Recall***	***F1-score***
1	4011	0.807	0.863	0.834
2	4019	0.921	0.949	0.935
3	41401	0.921	0.985	0.952
4	4260	0.873	0.899	0.886
5	42731	0.978	0.949	0.963
6	4271	0.875	0.965	0.918
7	4280	0.950	0.853	0.899

B. Case Study 1: Typical Usage

An instance of doctor's notes usually contains several diagnoses, some of which are from the same ICD chapter, as in our experiment. One important aspect of medical coding is the ability to detect and distinguish between multiple diagnoses, especially when they are closely related. Because our approach uses different knowledge for each diagnosis code, a different set of sentences is extracted for each diagnosis. Fig. 4 shows an example of discharge diagnoses from doctor's notes. We start with the first diagnosis, "Complete heart block." This term is located in DO, which is used to generate a set of semantically related terms such as "AV", "AV block", "atrioventricular block", "dizziness", "chest pain". These semantically related terms are then used to extract related sentences from the entire text of the doctors' notes.

DISCHARGE DIAGNOSES:
1. Complete heart block
2. Heart failure
3. Acute on chronic renal failur
4. PAF
5. V. tach
...

Fig. 4. Discharge diagnoses present in Case Study 1.

Fig. 5 shows a sample of text found to be semantically related and extracted from the notes. Note that the extracted sentences contain information related to the type of block that can serve as evidence for classification. Furthermore, it contains supporting information regarding the symptoms experienced by the patient which are related to atrioventricular block. The extracted sentences are then fed into the trained classifier along with the original diagnosis. The classifier predicts ICD-9 code 4260*, Atrioventricular block, complete* for the first diagnosis.

... 1 day history of **chest pain** and **dizziness**.... has first degree **AV block** and is on amiodarone ...

Fig. 5. Sample of text extracted for the first diagnosis "Complete heart block".

Similarly, for the second diagnosis, "Heart failure," we generate a set of semantically related terms and gather evidence using the sentence extraction procedure on the full text. In this case, the evidence is helpful for disambiguating the general term "Heart failure" used to describe the failure, which may potentially indicate several different types of heart failure and therefore diagnosis codes. Sentences containing congestive heart failure symptoms like "fatigue" are also extracted and provide further support. Fig. 6 shows a sample of the text extracted for this diagnosis. The classifier predicts code 4280*, Congestive heart failure, unspecified*, and this code is assigned to the diagnosis.

... presents with **fatigue** on exertion over past week ...
Chest x-ray indicated the presence of **congestive heart failure** and a possible left sided infiltrate ...

Fig. 6. Sample of text extracted for the second diagnosis "Heart failure".

We follow the same procedure for the other diagnoses to assign medical codes. Fig. 7 shows the full ICD code assignment for the diagnoses listed in Fig. 4. This full code assignment along with the evidence used for each prediction are provided to the user to help justify them.

Fig. 7. The completed ICD code assignment for the diagnoses listed in Fig. 4.

Note that for the third diagnosis, "Acute on chronic renal failure," when we follow the regular procedure, the classifier outputs low probability for all of the possible classifications and does not lead to a valid prediction. As the actual code is outside of the set of selected codes used to train the classifier, it is thus skipped with no matching code.

C. Case Study 2: Ambiguous Diagnoses

We now look into two instances of doctors' notes with the same diagnosis as shown in Fig. 8. Both of the instances contain a "Hypertension" diagnosis that may be associated with either one of the hypertension variant codes selected for our demonstration. In order to disambiguate which code is indicated by each diagnosis, semantically related sentences from the notes preceding the discharge diagnoses must be utilized.

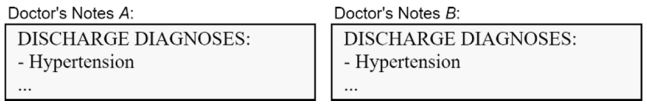

Fig. 8. The discharge diagnoses of the two separate discharge summaries.

As in Case Study 1, a set of semantically related terms is generated and used to extract semantically related sentences from the doctor's notes. The additional context collected as evidence during the sentence extraction phase can be used to disambiguate the two different ICD codes indicated by the diagnoses. Fig. 9 shows two examples of sentences extracted from two separate doctors' notes that can be useful in making the correct code assignment decisions.

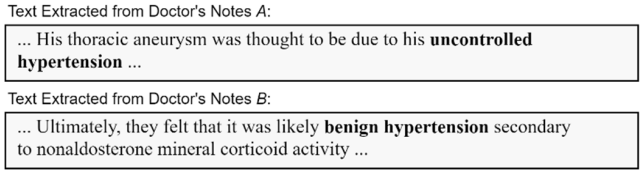

Fig. 9. Samples of text extracted from two separate doctors' notes.

Note that the text extracted from doctor' notes *A* contains the key term "uncontrolled hypertension," which is a synonym of unspecified hypertension. Meanwhile, the text extracted from doctor' notes *B* contains the key term "benign hypertension," which obviously contributes to a prediction of the code for benign hypertension. As shown in Fig. 10, the common diagnosis "Hypertension" in doctor' notes *A* is assigned code 4019 (Unspecified essential hypertension), while the same diagnosis in doctor' notes *B* is assigned code 4010 (Benign hypertension).

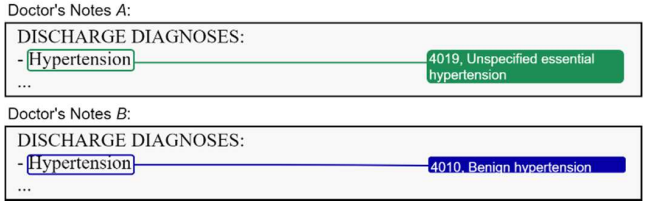

Fig. 10. ICD code assignments for two ambiguous diagnoses.

V. Conclusions and Future Work

In this paper, we introduced an ontology-based approach to assigning ICD diagnosis codes to diagnoses contained in doctors' notes. Unlike others, this approach is fine-grained, which focuses on one diagnosis at a time, increasing scalability and human understandability. Using domain knowledge encoded in ontologies, we can extract semantically related sentences from doctor's notes, seek further evidence for the predictions, and supply users with justification for decisions on code assignment. The experiments show that our approach is feasible and works accurately with the MIMIC-III dataset. Further case studies show that our approach can not only handle typical cases, but also special cases with ambiguous diagnoses.

In future work, we will consider using more efficient models for text classification, as we did in our earlier work [16], and perform a comparative analysis with existing coarse-grained methods. We will explore the possibility of using more robust classifiers, such as a hierarchical classifier that follows the hierarchical structure of the ICD code books. The improved classifier will be scalable to include more medical codes and can be applied to predict a wider range of disease cases.

References

[1] M. A. Moisio, A guide to health insurance billing, Albany: Thomas Delmar Learning, 2001.

[2] National Center for Health Statistics, "International classification of diseases, ninth revision (ICD-9)," March 3, 2022. [Online]. Available: https://www.cdc.gov/nchs/icd/icd9.htm. [Accessed March 24, 2022].

[3] R. Farkas and G. Szarvas, "Automatic construction of rule-based ICD-9-CM coding systems," *BMC Bioinformatics,* vol. 9, suppl 3, no. S10, 2008.

[4] J. Medori and C. Fairon, "Machine learning and features selection for semi-automatic ICD-9-CM encoding," in *Proceedings of the NAACL HLT 2nd Louhi Workshop Text Data Mining Health Documents*, Los Angeles, June 2010, pp. 84-89.

[5] A. Perotte, R. Pivovarov, K. Natarajan, N. Weiskopf, F. Wood and N. Elhadad, "diagnosis code assignment: models and evaluation metrics," *J. Amer. Med. Informat. Assoc.,* vol. 21, no. 2, pp. 231-237, 2014.

[6] A. E. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi and R. G. Mark, "MIMIC-III, a freely accessible critical care database," *Scientific Data,* vol. 3, no. 1, p. 160035, 2016.

[7] T. Baumel, J. Nassour-Kassis, R. Cohen, M. Elhadad, and N. Elhadad, "Multi-label classification of patient notes: case study on ICD code assignment," in *Processings of the Workshops at the thirty-second AAAI conference on artificial intelligence*, New Orleans, 2017, pp. 409-416.

[8] M. Falis, M. Pajak, A. Lisowska, P. Schrempf, L. Deckers, S. Mikhael, S. Tsaftaris, and A. O'Neil, "Ontological attention ensembles for capturing semantic concepts in ICD code prediction from clinical text," in *Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*, Hong Kong, 2019, pp. 168-177.

[9] B. Biseda, G. Desai, H. Lin, and A. Philip, "Prediction of ICD codes with clinical BERT embeddings and text augmentation with label balancing using MIMIC-III," *arXiv preprint arXiv:2008.10492,* 2020.

[10] H. Shojaee-Mend, H. Ayatollahi, and A. Abdolahadi, "Development and evaluation of ontologies in traditional medicine: a review study," *Methods of Information in Medicine,* vol. 58, no. 6, pp. 194-204, 2019.

[11] E. L. Clarke, S. Loguercio, B. M. Good, and A. I. Su, "A task-based approach for Gene Ontology evaluation," *Journal of Biomed Semantics,* vol. 4, no. suppl 1, 2013, p. S4.

[12] E. Ong, L. L. Wang, J. Schaub, J. F. O'Toole, and B. Steck, "Modelling kidney disease using ontology: insights from the kidney precision medicine project," *Nature Reviews Nephrology,* vol. 16, no. 11, pp. 686-696, 2020.

[13] S. Jusoh, A. Awajan, and N. Obeid, "The use of ontology in clinical information extraction," *Journal of Physics: Conference Series,* vol. 1529, p. 052083, 2020.

[14] L. M. Schriml, E. Mitraka, J. Munro, et al., "Human disease ontology 2018 update: classification, content and workflow expansion," *Nucleic Acids Research,* vol. 47, no. D1, pp. D955-D962, November 2018.

[15] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 42, no. 2, pp. 318-327, 2020.

[16] R. de Groof, H. Xu, J. Zhang, and R. Liu, "Mining significant terminologies in online social media using parallelized LDA for the promotion of cultural products," in *Proceedings of the 14th International Conference on Data Science (ICDATA'18)*, Las Vegas, Nevada, USA, July 30 - August 2, 2018, pp. 3-9.