Automating ICD-10 Tagging of Short Pieces of Text using Deep Learning Techniques

Brandon Fan, Weiguo Fan, Harold "Skip" Garner

Abstract

Background: The 10th revision of the International Statistical Classification of Disease and Related Health Problems (ICD-10) offers a set of codes that are associated with diseases and medical procedures. These ICD-10 codes can be used to classify sets of clinically-related pieces of text (papers, facts, questions etc.). However, with over 200,000 ICD-10 codes with varying levels of specificity, it becomes a tedious task to label these pieces of texts of differing length with their proper associated ICD-10 codes. This task of classification is exacerbated when a piece of text can be associated with multiple ICD-10 codes with equal confidence. This results in large amounts of human judgement, labor and time to properly assign ICD-10 codes to a given piece of text.

Purpose: We wish to solve this problem by introducing an automated system that can properly classify short-pieces of text with their associated ICD-10 Codes

Methods: In this paper, we introduce a novel deep-learning-based algorithm that utilizes trained Paragraph2Vec embeddings combined with a custom sentence importance algorithm to form an ensemble model. We train this model on a custom short-text (factoid) dataset conglomerated from powerpoints and lectures. This dataset was then divided into train and test datasets to compare against other state-of-the-art models.

Results: The model can determine with up to 82% accuracy and 0.848 F1-Score, outperforming all baseline methods, on a short piece of text and its associated ICD-10 codes by comparing the short piece of text with the associated ICD-10 code description and crawled synonyms.

Conclusion: This model can be implanted into existing systems to assist in text classification and association to provide meaningful results in medical and clinical decision support systems.

Keywords: ICD-10, Medical Tagging, Factoid, Machine Learning, Deep Learning

1. Introduction

Textual classification, or tagging, has become a hot topic in natural language processing (NLP) literatures. Various corpuses have been utilized for tagging purposes ranging from newspaper articles, customer reviews and linguistic part-of-speech tagging. Because of this, textual classification has seen a recent rise in the fields of medicine and the medical tagging of medical literature. Various statistical models such as the Brill's rule-based tagger [1] have been used to classify a medical text corpus. These papers recognize the ability to transfer similar algorithms from other tagging problems (such as newspaper-tagging) to medical literature.

Standard medical diagnostic tags have been provided by the International Statistical Classification of Disease and Related Health Problems. Surveys have shown that ICD-10 tagging and medical coding has recently become a necessity for hopsital management and in helping doctors make improved patient diagnoses [2]. ICD-10 codes are considered the staple and standard for medical diagnoses and tagging [3]. Past medical tagging research has utilized these ICD codes for classification and has achieved significant results [4]. Such examples include the classification of patient notes [5], medical records [6, 7], online health forums [8] and unstructured text [9], as well as general medical free text including various languages [10, 11, 12, 2], and semantic and deep learning approaches to automate the tagging process. However, past research has focused on longer lengths of text instead of short pieces of texts. This introduces a new problem as short pieces of text often have abbreviations and contain less information in comparison to medical records or free text. Due to the short lengths, data is sparse and often contains less information in comparison to feature rich medical records or free text. An example of a short medical piece of text is shown below.

> *Hematogenous osteomyelitis primarily affects the metaphyseal area of the long bones.*

Thus, the problem cannot be composed a simple tagging problem (traditional classification problem attempting to map x to y by estimating a function $f(x)$ with $\tilde{f}(x)$ that is learned through training or optimization); rather, it must be approached from a ranking perspective (attempting to find the best x that has attains the max relevancy score with a certain y).

> Therefore, we attempt to find a function $s(D_i, F_i)$ such that if $D_i \in D$ is the most relevant document to factoid $F_i \in F$, then $s(D_i, F_i) = \max s(D, F_i)$

Because of this, tagging short excerpts of text poses a new problem to solve in the medical tagging paradigm. Classifying short pieces of text allows for the automation of tagging in medical search databases and search engines. Additionally, factoid classification can be used in medical student management systems [13].

Recent developments in machine learning and deep learning have resulted in countless applications in a variety of industries and fields. Given the significant progress in image classification [14] and image-based applications (such as captioning and segmentation) [15, 16], the application of deep learning is poised to enter the natural language processing (NLP) realm. Additional deep learning research has been done in the medical sect including computer-aided diagnosis systems [17], lung cancer prediction [18], pharmacovigilance [19, 20, 21] and drug side effects through social media mining [22, 23] and internet mining [24]. More specifically, the intersection between NLP and medical literature is showing promise and has been illustrated in publications and the press. Text classification has long since been a problem and has been seen to have compelling results with the use of Long Short-Term Memory Networks [25]. Additionally, the development of word embeddings enabled by Word2Vec [26] which provides semantic vectors to capture contextual meanings of words in numerical representation has become the staple of deep learning in text. In this paper, we pose a new text classification problem framed as a text similarity problem, the classification of short pieces of text with associated ICD-10 codes and their descriptors. ICD-10 codes provide a set of codes that are associated with various diseases and medical procedures. These can be used to classify various pieces of text, more specifically assign ICD-10 codes to various sets of text, thereby enabling reliable clustering of those text items or the putative assignment of actionable/billable codes to free text that is often part of a medical record. Current methods involve human annotation of short pieces of text [7]. However, this method of annotation is highly prone to human error and is very time consuming as there are over 200,000+ ICD-10 codes. Therefore, in this paper, we utilize the descriptions of the ICD-10 codes to perform a text similarity search for the most relevant ICD-10 code that can be assigned to a short piece of text or factoid (a small factual statement or phrase; we will use factoid for the rest of the paper).

Our goal is to introduce a novel deep learning approach (1) to address the ICD-10 classification problem. More specifically, we utilize custom trained word embeddings combined with an importance algorithm (1) to propose a similarity function to find the most relevant ICD-10 code for an associated text item, a factoid. This model significantly reduces the amount of human work required to annotate factoids and subsequently reduces human error. The annotation process can be automated and online learning can be implemented for future updates in ICD codes. These putative quantitative and rank-able relationship assignments are reviewed by human experts to measure the accuracy, and in regular use make the final judgments as to the desired classification assignment.

2. Related Work

*2.1. **Automated Tagging***

As aforementioned, there has been much research done in the field of automated tagging. Simple models such as Naive Bayes have been implemented for email spam detection and more complex models have been used to classify newspaper headlines. Deep learning methods have been utilized to classify complex data such as sound in music classification tasks [27]. In that paper, Choi et. al. utilizes a custom convolutional neural network (NN) to automatically tag music. The paper's results beat state-of-the art statistical models. Additionally, paper tagging has seen growth as well [28]. The authors utilized key phrase extraction methods to classify papers with their associated tags. E-commerce product tagging was done through the use of visual features and the use of a convolutional neural network [29]. This increases the throughput of search results: providing an additional filtering option and enhancing search. Thus, based on past and ongoing research, the automation of tagging tasks allows for less human annotation and human error, reducing the amount of time required for classification and reducing potential error. This becomes critical in certain industries such as medicine as an incorrect classification of a document or text can lead to irrelevancy.

2.2. Medical Tagging

With increasing development in deep learning, new tasks of text classification have emerged: one of which is medical tagging. More specifically, the tagging of corpuses in varying lengths with their corresponding ICD codes. A method proposed by Chen et al. involved the use of an improved version of the longest common subsequence algorithm [6, 30]. This LCS algorithm is then coupled with a semantic similarity calculation to determine final similarity. However, the research done by Chen et. al. focused entirely upon the classification of Chinese medical documents and thus was designed specifically with the Chinese language in mind. Two students at Stanford proposed a method of deep learning through Long Short Term Memory networks [5, 25]. The students utilized pre-trained word embeddings from the GloVe dataset [26]. However, this method assumes that every diagnostic label has an associated text (so that every label may be accounted for), however this cannot always be assumed and so other methods must be used instead. Additionally, the use of LSTM networks is often better at predicting labels on longer pieces of text due to more feature-rich vectors. However short text classification is often more difficult to train and to generalize. Thus, based on the uniqueness of the problem and the limited data, we introduce a novel approach that utilizes word embeddings combined with a cosine similarity function and a custom importance function to determine the most relevant ICD-10 code given a short piece of text.

3. Methods

3.1. Approach

The primary goal for the model is that it must properly annotate a given piece of text, a factoid, with the most appropriate ICD-10. Typically, this is approached as a classification problem. However, due to the limited dataset and the sparse text, the problem must be posed as a ranking problem. With such in mind, the approach utilizes a word embedding model coupled with an importance algorithm to properly annotate factoids given ICD-10 descriptions. The model's source inputs are a factoid and the descriptions of all ICD-10 codes and subsequently outputs a similarity or relevancy score. The most appropriate ICD-10 codes will be given the highest similarity (implying relevancy) scores and the least relevant the lowest scores. The similarity measure used to compute similarity is a combination of the cosine similarity function (Equation 2) and a custom importance algorithm.

$$\frac{A \cdot B}{||A|| \cdot ||B||} \tag{1}$$

The overarching research approach is shown below.

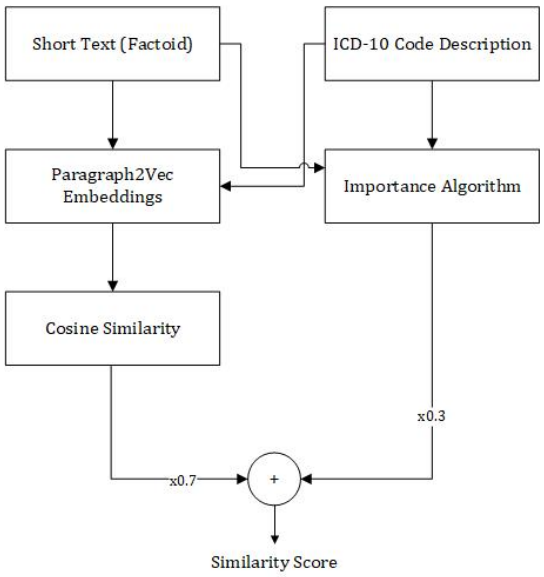

Figure 1: Research Workflow

Research workflow representing the overall structure of the algorithm to determine the most relevant ICD-10 code for a corresponding factoid.

3.2. Model Design and Determination

As seen in Figure 1, the model design consists of two components. The Paragraph2Vec and the Keyword Matching importance algorithm. The results are then weighted in an ensemble manner to produce a similarity score between 0 and 1. An algorithmic representation of this model is shown below.

Algorithm 1: Determination of Similarity Score

Input: Factoid Text and ICD-10 Code Descriptions
Result: Sorted Similarity Scores
for *each word* $w_F \in$ *Factoid* F **do**
  find associated word vector v_F
end
average all word vectors F_{avg}
for *each ICD-10 code description* D **do**
  for *each word* $W_D \in D$ **do**
    find associated word vector v_D
  end
  average all word vectors D_{avg}
  $S_{D_{avg},F_{avg}} = similarity(D_{avg}, F_{avg})$
  $I_{D_{avg},F_{avg}} = importance(D_{avg}, F_{avg})$
  calculate similarity: $S_{D_{avg},F_{avg}} \cdot 0.7 + I_{D_{avg},F_{avg}} \cdot 0.3$
end

3.2.1. Paragraph2Vec Embeddings

The team decided to fully train an untrained model of Paragraph2Vec: opting not to pre-train it on Medline or some other biomedical article databases. This was determined by analyzing the performance of a custom trained embedding in comparison to a pretrained embedding on Medline. Therefore, a custom Paragraph2Vec embedding was used as the model approach to translate tokenized text into proper semantic vectors to determine text similarity. The embeddings offer a lookup table for the most common words in the corpus. This lookup table captures the semantic relationships between words and are used to create the paragraph vector. Our embedding averages the word vectors for each sentence. This produces a one-dimensional vector that is used to calculate cosine similarity. The Paragraph2Vec was trained on the training set of ICD-10 code descriptions and factoids. The optimal dimension of the paragraph vectors was determined to be 1 by 200. These custom trained word embeddings offered substantial results, however, was not fully accurate. Thus, the importance algorithm was introduced to enhance performance and to make the model an ensemble-based weighting model.

3.2.2. Importance Algorithm

Due to the shortness of the text, word embeddings often have difficulty capturing relationships due to the sparse data present in the corpus. Thus, we introduce the importance algorithm to help solve the issue with short text. The importance algorithm is a critical aspect of the algorithm as it allows a method of "normalization" of the results. As previously mentioned, the Paragraph2Vec embeddings can be skewed at times and return results that contain more text offering more relevancy. Thus, utilizing the importance algorithm, these results will be further normalized to determine the most relevant results. The importance algorithm is based on a simple keyword matching of the lemmatized versions of each factoid A and ICD-10 code B. The number of matches is then divided by the length of the ICD-10 code description. This equation is shown below (Equation 3).

$$\frac{A \cap B}{|B|} \tag{2}$$

The importance algorithm improved results of the model and offered a secondary decision maker to assist in the ICD-10 code tagging.

3.2.3. Ensemble Weighting

As seen in Figure 1 and Algorithm 1, custom weighting of the cosine similarities and the importance algorithm are implemented. These weights were determined through experimentation and were determined to maximize the metrics described. The weighting gives more to the Paragraph2Vec and Cosine Similarity pathway (70%) and less to the importance algorithm (30%). The importance algorithm was added after the first set of results were determined (i.e. the word embedding results). Thus, adding the importance algorithm enhanced the performance of the overall algorithm.

Next, the data source and corresponding data preprocessing methods are discussed.

3.3. Dataset and Preprocessing

3.3.1. Factoid Dataset

The factoid dataset was compiled and written by Edward Via College of Osteopathic Medicine students. A total of over 36,000 Power Point slides used in the first two years of medical school were reviewed for opportunities to annotate them with ICD-10 diagnostic or procedure codes. Approximately, 4,300 slides were identified as containing diagnoses and procedures to which an ICD-10 code could be assigned. Then for each slide, a declarative sentence of fact, a factoid, was composed as a quick review item for students as they continuously study for school and board exams. These factoids are presented to students in real-time as they are logging/documenting their clinical patient encounters, and since such encounters are captured in an "app" anchored in ICD-10 codes, an association was needed between the ICD-10 codes and each factoid. Data sources to generate factoids came from these presentations and medical textbooks and literature. Students were assigned specific topics to compose these factoids and then properly annotate the factoids with their associated ICD-10 codes. The data was then conjoined into a spreadsheet for analysis and training. Below are examples of factoids from the factoid dataset.

> *E. coli replication is inhibited by ciprofloxacin , which inhibits bacterial DNA gyrase.*
>
> *Mutations may results from the non-repair of pyrimidine dimers that produce melanomas*
>
> *Cockayne syndrome is specifically linked to defects in nucleotide excision and transcription-couple repair systems.*

As seen by the examples, factoids are short in length and do not contain enough information for traditional machine learning methods and deep learning methods. Additionally, the dataset that was conglomerated contained only 2,000 total datapoints. Thus, due to the eager training required for traditional approaches, a new approach had to be considered by the authors.

3.3.2. ICD-10 Dataset

The ICD-10 dataset of codes and descriptions was retrieved from the World Health Organization compiled set available from the Center for Disease Control web site and additional items were added to the dataset to enhance the currently existing dataset. These additions included synonyms and medical terms that corresponded with the short descriptions associated code. The description combined with these additions were used together to provide a feature-rich dataset for model development. The distribution of lengths and depths of ICD-10 codes are shown below.

Statistic	**Value**
Count	22569
Mean	4.96
Standard Deviation	4.72
Min	1
Max	100

Table 1: ICD-10 Description Length Information

Table shows the description length statistics in the ICD-10 code dataset. These descriptions were used to determine similarity. Based on the distribution, it can be clearly seen that the majority of descriptions were short, one to two sentences, descriptions making the classification task more complex.

Statistic	**Value**
Count	240896
Mean	5.88
Standard Deviation	1.03
Min	1
Max	7

Table 2: ICD-10 Depth Distribution

Table highlights the depth statistics in the ICD-10 code dataset. ICD-10 codes are divided up into assorted depths (or specificities). The depth attributes can be used a method of reduction to reduce the number of comparisons required and to further speed up the algorithm.

3.3.3. Data Preprocessing

As aforementioned, both datasets were used in conjunction for model training (as the model takes in an input from each dataset to compute similarity). Thus, equal measures were taken for preprocessing of the datasets. All text data was first tokenized, standardized into a common format, and then lemmatized to normalize word stems. This resulted in a preprocessed corpus that was used to train the Paragraph2Vec embeddings.

3.4. Benchmarks

Due to the uniqueness of the problem, five benchmarks were chosen to compare our results. These benchmarks are explained in the following sections. The problem consists of a limited dataset (max of 2,000 data points) and short lengths of text, resulting in sparse and feature-poor data points. This makes classification of ICD-10 codes very difficult for factoids that only consist of 10-20 words. Traditional classification techniques such as Support Vector Machines [31], and Probabilistic Modelling [32] often fail to gather enough information from sparse data to make accurate classifications. Thus, the classfication problem of factoid tagging with ICD-10 codes is posed as a ranking problem where we attempt to find the most relevant ICD-10 code by comparing a factoid with an ICD-10 code description (See Figure 1).

3.4.1. Count Vectorization

Count Vectorization consists of counting the number of related text and computing cosine similarity from the count matrix. This is commonly discussed in introductory information retrieval courses and books [33, 34]. Thus, this algorithm can be seen as counting up the number of keywords to compose a matrix that, when compared to another count matrix, produces a similarity measure that can be calculated through cosine similarity. The concept revolves around the aspect that the greater the number of common words in two vectors, the more similar the two sentences are. However, count vectorization algorithms do not take into account semantic elements and term frequency. Thus, it is imperative that stop words (i.e. and, or, but, etc.) are removed for better results. Count vectorization is considered the standard baseline and benchmark for NLP tasks and provides a comparison metric for more complex models.

3.4.2. Term Frequency-Inverse Document Frequency

Term Frequency-Inverse Document Frequency involves counting the number of occurrences of a single word i in a document j (i.e. term frequency $tf_{i,j}$) and multiplying it by a weight that correlates to the number of documents containing the word (i.e. document frequency df_i). Thus, each word in a piece of text is given a weight w as shown below.

$$w_{i,j} = tf_{i,j} \times \log(\frac{N}{df_i}) \tag{3}$$

Henceforth, this defines Term Frequency-Inverse Document Frequency (TF-IDF). TF-IDF is commonly used in search engine design and information retrieval [35, 36, 37]. Despite the common usage, TF-IDF has the limitation that semantics and word relationships are not properly captured.

3.4.3. Conditional Random Fields

A recent paper by Du et al. introduced the use of Conditional Random Fields (CRF) as a classification algorithm to classify Chinese Documents under certain ICD-10 Codes [11]. CRF models are predominantly used in pattern recognition and structured recognition including Named Entity-Recognition (NER). In this paper, the CRF model was designed and implemented according to the algorithm set forth by Du et al. Despite the similarities in classification, Du et al. presents a classification algorithm on Chinese EHR and clinical text that is feature rich. This is contrasted to the problem set forth in this paper of a sparse data source (short factoids). Thus, we analyze the effectiveness of the model presented in Du et al in comparison to the previous benchmarks, subsequent benchmark, and our model. We compare this current state-of-the-art classification on both multilingual and lengthy, feature-rich text to a feature-poor, short factoid dataset.

3.4.4. Okapi BM25

Okapi Best Matching 25 (or BM25 for short) is a standard information retrieval method that couples IDF with probabilistic retrieval methods [38]. BM25 is considered the state-of-the-art TF-IDF ranking function and is used as a baseline for comparison in medical and IR papers [39, 40]. Okapi BM25 was also used to reduce data corruption in electronic health records, showing the use of BM25 in medical informatics [41]. Okapi BM25 is used as a comparison to test traditional information retrieval methods (contrasting the traditional machine learning methods aforementioned) on the factoid dataset. The score function $S(D, Q)$ is shown below given a query Q containing keywords $q_1, q_2, ..., q_n$ and a document D:

$$S(D, Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})} \tag{4}$$

3.4.5. Word Embeddings

The word embeddings approach is the standard baseline for medical tagging among current papers [5]. The utilization of word embeddings coupled with a cosine similarity function is the most common method of similarity calculation and the determination of the most relevant ICD-10 code. This method involves the construction of a co-occurrence matrix X, often trained by the Word2Vec algorithm [26]. This matrix then allows for the conversion of words to the corresponding vector representation in the co-occurrence matrix. The word vectors can then be concatenated to form a paragraph matrix or can be averaged to form a 1-dimensional vector. This method retains semantic relationships in the vectors itself and thus offers a more robust calculation of similarity. Thus, this is used as the current standard or baseline.

3.5. Evaluation Metrics

Four metrics were utilized to determine model efficacy and accuracy: Top 1 Accuracy, Precision, Recall, and F1 Score. These metrics were chosen for effectiveness and correctness to guarantee that the most relevant ICD-10 codes are returned.

3.5.1. Top 1 Accuracy

Top 1 Accuracy is determined by how accurate the model is predicting the top ICD-10 code (i.e. is the 1st ICD-10 code predicted the labelled code). This offers a representation of how accurate the model is in comparison to other models. The Top 1 Accuracy measurement is a proper representation of the potential for automation. The algorithm for Top 1 Accuracy is shown below.

Algorithm 2: Top 1 Accuracy Algorithm

```
for each result r ∈ Results R do
    if r = correct label then
        increment sum S by 1
    end
end
divide S by length of R
```

3.5.2. Precision

Precision, or predicted positive value, is the measure of the number of truly relevant instances in all retrieved instances. Otherwise termed as the number of true positives over all returned documents. Precision helps recognize how accurate our model in returning relevant ICD-10 codes (as the classification is posed as a ranking problem). In multi-class classification, precision can also be considered number of items correctly labelled divided by the total of items that were labelled a certain label. The formula for precision is shown below:

$$\text{Precision} = \text{PPV} = \frac{TP}{TP + FP} \tag{5}$$

3.5.3. Recall

Recall, or sensitivity, the measure of the number of relevant instances that were retrieved in all relevant results. Recall sees how many relevant codes were actually classified. In multi-class classification, recall can also be considered the number of items that were correctly labelled divided by the total items that were actually labelled that label. The formula for recall is shown below:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

3.5.4. F1 Score

The final metric calculated was F1 Score, a conglomeration of both Precision and Recall. The F1 Score helps guarantee that the model does not succeed at only one task (such as precision or recall), rather combines both precision and recall into one combined metric. Mathematically, F1 Score is the harmonic mean of Precison and Recall. This metric is the standard in both ranking algorithms and classification algorithms and is used in comparing various models. The formula for F1 Score is shown below:

$$\text{F} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

4. Results

Multiple models were trained to determine the most optimal model. The results show three benchmark models (CV, TF-IDF, Word Embeddings) and the CRF Model presented in [11] in comparison to our model. Both training and test results are shown in the tables on the following page (3, 4).

The results show that the Word Embedding combined with Importance performs significantly better in all metrics in both training and test results. CountVectorization, TF-IDF, and Okapi BM25 models do not properly capture semantic information and key relationships between words, making it difficult to classify and find the most relevant ICD-10 code. This reduces the models' metrics. Due to the short lengths of the ICD-10 code descriptions and short lengths of the factoids, the Okapi BM25, traditional information retrieval method, does not have enough data to identify proper features for IR. As aforementioned, traditional machine learning and probabilistic techniques are unable to fully capture the feature-poor, short-lengthed factoids. This results in a poorer performance (as seen in the CRF model proposed by [11]). Top 1 Accuracy is more highly regarded as a metric as it allows for the automation of the tagging process. These results support support the findings that our model is a viable model for short text classification. This solidifies the efficacy of the word embedding with importance approach, emphasizes that traditional machine learning approaches do not properly classify short pieces of text, and provides a foundation for future development and model deployment.

5. Discussion

The experimental results show that the model is a viable and effective method for ICD-10 Code annotation of short pieces of text. The model performs significantly better than baselines and offers a basis for future developments as more data is collected. Online learning can be implemented for continuous model learning as new factoids are generated from data sources and ICD-10 codes are updated with new releases. Though results may not offer full autonomy, model improvements such as changing vector dimensions or the ensemble of baselines in conjunction with the model can offer enhanced performance that can approach full automation. Current results cannot offer full automation of tagging but can offer a method of human assistance: by filtering out extraneous results out of the 240,000+ ICD-10 codes and significantly reducing the time and error for human annotation.

Future developments include model deployment to industry by a development of an API platform for companies and medical systems to use the model for proper annotation using ICD-10 codes, model generalization to larger lengths of text (biomedical papers, press releases etc.), model fine-tuning (improving overall accuracy), and model translation (translation to a new problem in a new field and industry).

Model Name	Top 1 Accuracy	Precision	Recall	F1 Score
CV Model [33, 34]	11.65%	0.136	0.117	0.123
TF-IDF Model [35, 36, 37]	25.24%	0.272	0.252	0.259
CRF Model [11]	45.23%	0.572	0.571	0.548
Okapi BM25 [38]	29.23%	0.307	0.292	0.286
Word Embeddings [26]	69.55%	0.773	0.695	0.70
Word Embeddings + Importance	**81.55%**	**0.848**	**0.816**	**0.826**

Table 3: Numerical Training Results

Model Name	Top 1 Accuracy	Precision	Recall	F1 Score
CV Model [33, 34]	8.64%	0.111	0.086	0.091
TF-IDF Model [35, 36, 37]	20.58%	0.286	0.298	0.302
CRF Model [11]	39.64%	0.515	0.500	0.483
Okapi BM25 [38]	24.27%	0.261	0.243	0.237
Word Embeddings [26]	66.99%	0.742	0.670	0.688
Word Embeddings + Importance	**76.95%**	**0.844**	**0.770**	**0.785**

Table 4: Numerical Testing Results

6. Conclusion

This study shows that Paragraph2Vec combined with an importance algorithm can properly annotate short pieces of text with their associated ICD codes with improved accuracy and thus offers a method of reducing human time consumption and error during the annotation process. Further improvements can be made to improve model performance and generalization can be achieved through diversifying the datasets with short pieces of text combined with papers and press releases, offering an even more powerful model.

Contributors

W.F. and H.R.G. contributed to the conceptualization of this project, model and experimental design, data analysis, and the writing of this manuscript. B.F. was responsible for model design and development, model implementation, results generation and debugging. All authors read and approved the final manuscript.

Competing Interests

Authors have no competing interests.

Summary Points

- Current research has employed machine learning and probabilistic approaches to ICD-10 classification on longer and more feature-rich texts such as Electronic Health Records, Medical Free Texts, and Medical Literature. However, no research has been tested on short-pieces of text that are often one or two sentences long in length.
- State-of-art models such as Conditional Random Fields [11], Okapi BM25, TF-IDF, and CV are used as comparison models and are employed on the short-length text. These models are incapable of learning accurate representations on the data due to the sparsity and lack of data.
- We develop a novel deep-learning based approach to the ICD-10 tagging problem on short-pieces of text called factoids. We utilize a custom keyword matching algorithm to improve the model and compare it to existing state-of-the-art models.
- These factoids are often sparse and feature-poor, requiring a unique method for preprocessing and model inference. Due to the short-length and feature-poor text, the classification task of tagging is approached as a ranking problem attempting to find the most similar ICD-10 code for an associated factoid.
- The model can be applied to an automated tagging system and deployed in current medical databases and systems.

References

[1] U. Hahn, J. Wermter, Tagging medical documents with high accuracy, in: Pacific Rim International Conference on Artificial Intelligence, Springer, pp. 852–861.

[2] G. Nilsson, H. Åhlfeldt, L.-E. Strender, Computerisation, coding, data retrieval and related attitudes among swedish general practitioners—a survey of necessary conditions for a database of diseases and health problems, International Journal of Medical Informatics 65 (2002) 135 – 143.

[3] J. Stausberg, N. Lehmann, D. Kaczmarek, M. Stein, Reliability of diagnoses coding with icd-10, International Journal of Medical Informatics 77 (2008) 50 – 57.

[4] C. M. Hohl, A. Karpov, L. Reddekopp, J. Stausberg, Icd-10 codes used to identify adverse drug events in administrative data: a systematic review, Journal of the American Medical Informatics Association 21 (2014) 547–557.

[5] S. Ayyar, O. B. D. W. IV, Tagging patient notes with icd-9 codes (2016).

[6] Y. Chen, H. Lu, L. Li, Automatic icd-10 coding algorithm using an improved longest common subsequence based on semantic similarity, PloS one 12 (2017) e0173410.

[7] G. Mikkelsen, J. Aasly, Manual semantic tagging to improve access to information in narrative electronic medical records, International Journal of Medical Informatics 65 (2002) 17 – 29.

[8] L. O'Grady, C. N. Wathen, J. Charnaw-Burger, L. Betel, A. Shachak, R. Luke, S. Hockema, A. R. Jadad, The use of tags and tag clouds to discern credible content in online health message forums, International Journal of Medical Informatics 81 (2012) 36 – 44.

[9] S. P. Stenner, K. B. Johnson, J. C. Denny, Paste: patient-centered sms text tagging in a medication management system, Journal of the American Medical Informatics Association 19 (2012) 368–374.

[10] A. Coffman, N. Wharton, Clinical natural language processing: Auto-assigning icd-9 codes, Overview of the Computational Medicine Center's (2007).

[11] L. Du, C. Xia, Z. Deng, G. Lu, S. Xia, J. Ma, A machine learning based approach to identify protected health information in chinese clinical text, International Journal of Medical Informatics 116 (2018) 24 – 32.

[12] S. Velupillai, H. Dalianis, M. Hassel, G. H. Nilsson, Developing a standard for de-identifying electronic patient records written in swedish: Precision, recall and f-measure in a manual and computerized annotation trial, International Journal of Medical Informatics 78 (2009) e19 – e26. Mining of Clinical and Biomedical Text and Data Special Issue.

[13] F. Rawlins, C. Sumpter, D. Sutphin, H. R. Garner, Quantifying medical student clinical experiences via an icd code logging app, International Journal of Medical Informatics 111 (2018) 51 – 57.

[14] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, CoRR abs/1512.03385 (2015).

[15] Q. You, H. Jin, Z. Wang, C. Fang, J. Luo, Image captioning with semantic attention, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4651–4659.

[16] M.-S. Badea, I.-I. Felea, L. M. Florea, C. Vertan, The use of deep learning in image segmentation, classification and detection, arXiv preprint arXiv:1605.09612 (2016).

[17] M. A. Al-antari, M. A. Al-masni, M.-T. Choi, S.-M. Han, T.-S. Kim, A fully integrated computer-aided diagnosis system for digital x-ray mammograms via deep learning detection, segmentation, and classification, International Journal of Medical Informatics 117 (2018) 44 – 54.

[18] C. M. Lynch, B. Abdollahi, J. D. Fuqua, A. R. de Carlo, J. A. Bartholomai, R. N. Balgemann, V. H. van Berkel, H. B. Frieboes, Prediction of lung cancer patient survival via supervised machine learning classification techniques, International Journal of Medical Informatics 108 (2017) 1 – 8.

[19] A. Cocos, A. G. Fiks, A. J. Masino, Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in twitter posts, Journal of the American Medical Informatics Association 24 (2017) 813–821.

[20] J. Liu, G. Wang, Pharmacovigilance from social media: An improved random subspace method for identifying adverse drug events, International Journal of Medical Informatics 117 (2018) 33 – 43.

[21] J. Mower, D. Subramanian, T. Cohen, Learning predictive models of drug side-effect relationships from distributed representations of literature-derived semantic predications, Journal of the American Medical Informatics Association (2018) ocy077.

[22] L. R. Kendra, S. Karki, L. J. Eickholt, L. Gandy, Characterizing the discussion of antibiotics in the twittersphere: What is the bigger picture?, J Med Internet Res 17 (2015) e154.

[23] L. Fernandez-Luque, R. Karlsen, J. Bonander, Review of extracting information from the social web for health personalization, J Med Internet Res 13 (2011) e15.

[24] T. Effland, A. Lawson, S. Balter, K. Devinney, V. Reddy, H. Waechter, L. Gravano, D. Hsu, Discovering foodborne illness in online restaurant reviews, Journal of the American Medical Informatics Association (2018) ocx093.

[25] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.

[26] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, CoRR abs/1301.3781 (2013).

[27] K. Choi, G. Fazekas, M. B. Sandler, Automatic tagging using deep convolutional neural networks, CoRR abs/1606.00298 (2016).

[28] M. G. Thushara, M. S. Krishnapriya, S. S. Nair, A model for auto-tagging of research papers based on keyphrase extraction methods, in: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1695–1700.

[29] V. Sharma, H. Karnick, Automatic tagging and retrieval of e-commerce products based on visual features, in: Proceedings of the NAACL Student Research Workshop, pp. 22–28.

[30] R. A. Wagner, M. J. Fischer, The string-to-string correction problem, Journal of the ACM (JACM) 21 (1974) 168–173.

[31] B. Koopman, G. Zuccon, A. Nguyen, A. Bergheim, N. Grayson, Automatic icd-10 classification of cancers from free-text death certificates, International Journal of Medical Informatics 84 (2015) 956 – 965.

[32] N. Afzal, V. P. Mallipeddi, S. Sohn, H. Liu, R. Chaudhry, C. G. Scott, I. J. Kullo, A. M. Arruda-Olson, Natural language processing of clinical notes for identification of critical limb ischemia, International Journal of Medical Informatics 111 (2018) 83 – 89.

[33] H. Schütze, C. D. Manning, P. Raghavan, Introduction to information retrieval, volume 39, Cambridge University Press, 2008.

[34] I. H. Witten, I. H. Witten, A. Moffat, T. C. Bell, T. C. Bell, Managing gigabytes: compressing and indexing documents and images, Morgan Kaufmann, 1999.

[35] J. Ramos, et al., Using tf-idf to determine word relevance in document queries, in: Proceedings of the first instructional conference on machine learning, volume 242, pp. 133–142.

[36] S. Tata, J. M. Patel, Estimating the selectivity of tf-idf based cosine similarity predicates, ACM Sigmod Record 36 (2007) 7–12.

[37] W. Zhang, T. Yoshida, X. Tang, A comparative study of tf* idf, lsi and multi-words for text classification, Expert Systems with Applications 38 (2011) 2758–2765.

[38] S. Robertson, H. Zaragoza, et al., The probabilistic relevance framework: Bm25 and beyond, Foundations and Trends® in Information Retrieval 3 (2009) 333–389.

[39] T.-Y. Liu, J. Xu, T. Qin, W. Xiong, H. Li, Letor: Benchmark dataset for research on learning to rank for information retrieval, in: Proceedings of SIGIR 2007 workshop on learning to rank for information retrieval, volume 310, ACM Amsterdam, The Netherlands.

[40] S. Jimenez, S.-P. Cucerzan, F. A. Gonzalez, A. Gelbukh, G. Dueñas, Bm25-ctf: Improving tf and idf factors in bm25 by using collection term frequencies, Journal of Intelligent & Fuzzy Systems (2018) 1–13.

[41] P. Ruch, R. Baud, A. Geissbühler, Evaluating and reducing the effect of data corruption when applying bag of words approaches to medical records, International Journal of Medical Informatics 67 (2002) 75 – 83.