1 August 2023, Preprint v1 · CC-BY 4.0

Qeios

PREPRINT

Research Article

NER Sequence Embedding of Unified Medical Corpora to Incorporate Semantic Intelligence in Big Data Healthcare Diagnostics

Sarah Shafqat[1], Zahid Anwar[2], Qaisar Javaid[1], Hafiz Farooq Ahmad[3]

1. Faculty of Computing and Information Technology (FOCIT), International Islamic University, Islamabad, Islamabad, Pakistan; 2. Department of Computer Science, North Dakota State University, United States; 3. King Faisal University, Saudi Arabia

Clinical diagnosis is a challenging task for which high expertise is required at the doctors' end. It is recognized that technology integration with the clinical domain would facilitate the diagnostic process. A semantic understanding of the medical domain and clinical context is needed to make intelligent analytics. These analytics need to learn the medical context for different purposes of diagnosing and treating patients. Traditional diagnoses are made through phenotype features from patients' profiles. It is also a known fact that diabetes mellitus (DM) is widely affecting the population and is a chronic disease that requires timely diagnosis. The motivation for this research comes from the gap found in discovering the common ground for medical context learning in analytics to diagnose DM and its comorbidity diseases. Therefore, a unified medical knowledge base is found significantly important to learning contextual Named Entity Recognition (NER) embedding for semantic intelligence. Our search for possible solutions for medical context learning told us that unified corpora tagged with medical terms were missing to train the analytics for diagnoses of DM and its comorbidities. Hence, we put effort into collecting endocrine diagnostic electronic health records (EHR) corpora for clinical purposes that are labeled with ICD-10-CM international coding scheme. International Codes for Diseases (ICD) by the World Health Organization (WHO) is a known schema to represent medical codes for diagnoses. The complete endocrine EHR corpora make DM-Comorbid-EHR-ICD-10 Corpora. DM-Comorbid-EHR-ICD-10 Corpora is tagged for understanding the medical context with uniformity. We experimented with different NER sequence embedding approaches using advanced ML integrated with NLP techniques. Different experiments used common frameworks like; Spacy, Flair, and TensorFlow, Keras. In our experiments albeit label sets in the form of (instance, label)

doi.org/10.32388/HPAUYJ

pair for diagnoses were tagged with the Sequential() model found in TensorFlow.Keras using Bi-LSTM and dense layers. The maximum accuracy achieved was 0.9 for Corpus14407_DM_pts_33185 with a maximum number of diagnostic features taken as input. The sequential DNN NER model diagnostic accuracy increased as the size of the corpus grew from 100 to 14407 DM patients suffering from comorbidity diseases. The significance of clinical notes and practitioner comments available as free text is clearly seen in the diagnostic accuracy.

Corresponding authors: Sarah Shafqat, sarah.shafqat@gmail.com; Zahid Anwar, zahid.anwar@ndsu.edu

1. Introduction

Automated clinical diagnoses are challenging. Machines are trained on healthcare information that is usually in free running text. Hence, information in free text is labeled to understand the context. Recently, researchers have focused on learning the context in the clinical domain. Special consideration is given to diagnosing cancer, retinopathy or pediatric issues amongst others in previous studies [1][2][3][4]. In our case, we classified diabetes mellitus (DM) and its comorbidity diseases. Natural Language Processing (NLP) is a promising technique for text mining. We need it to train machines or analytics to learn the context in a domain as humans do. Named Entity Recognition (NER) is an NLP technique among others that gives meaning to words or sequences of words in a sentence for contextual learning. This intelligence that we incorporate in analytics is called semantic intelligence. NER techniques are being applied to embed semantic intelligence in big data healthcare analytics for decision-making in a clinical context. Manual or automated NER annotation techniques are mostly used for embedding domain vocabulary in the analytics models [5], [6].

Labeled or annotated text with healthcare information is required for diagnoses, treatments, procedures and admissions. However, mostly labeled text is not available. But there are some unstructured texts available known as corpora. There are some corpora collected in the medical domain by the organizations such as; Multi-parameter Intelligent Monitoring in Intensive Care (MIMIC [7] and National Center for Biotechnology Information (NCBI) [8]. These corpora are regularly updated and maintained at the institutional level. These corpora need to be labeled in order to learn clinical context. This is a challenging task because manual or automated annotation is required which is difficult and requires lots

doi.org/10.32388/HPAUYJ

of computation and memory. In this paper, we proposed a novel technique that uses semantic tagging/embedding [9], [10] of tabular corpora for diagnoses.

Clinical entity classification in EHRs is challenging and takes time starting from collecting labeled training and validation datasets that are private and keeping patients' identities anonymous [11]. Custom corpora are developed through the collection of real-time datasets in Excel formats in a normalized tabular form provided by Shifa International Hospital, Pakistan. These datasets consist of endocrine patients diagnosed with diabetes mellitus (DM) and its comorbidity diseases. These are converted into three de-normalized flat datasets. There are standards for diagnoses known as the International Classification of Diseases (ICD-10) and for interoperability and generalizability of data, there is Fast Healthcare Interoperability Resources (FHIR) given by Health Level Seven (HL7). We kept the standardization to the global perspective for analytics to be deployed on the cloud and therefore, used FHIR 4.0 HL7 schema to model data. ICD-10-CM codes were used to label the endocrine EHR datasets for diagnoses. These standardized endocrine corpora labeled with ICD-10-CM codes are being analyzed to diagnose diabetes mellitus (DM) and its comorbidities in this paper. These corpora are tabular and domain-specific instead of general therefore providing better semantic representation of terms which makes it a good candidate for embedding. The corpora contain clinical notes and practitioner comments fields and the ICD-10 corpus as free text inputs for sequential NER annotation. These free text corpora were annotated for sequence NER embedding. Manual annotation is done using spaCy. Automated sequence NER annotation was done using a previously built model that was trained on medical data in Flair. Finally, we used the corpora in CSV format for sequence embedding to diagnose endocrine diseases as a case study. Evaluation of sequence NER embedding techniques is done through validation of results by experimental design. Experimentation is done on our proposed mechanism to annotate and implement sequential embedding to get corpora that are semantically intelligent. We used three techniques for NER sequence embedding. Manual and automated annotations using Spacy and Flair respectively were time-consuming. Correct classification is important, and it requires including all important features and attributes of a patient medical profile for achieving the correct diagnosis. Open-source cloud platforms like; Anaconda, Gradient Paperspace and Google Collab, provided us with high throughput and speed to train large unified medical corpora for multi-classification and embedding for semantic contextual intelligence. The standard corpora in CSV format have multiple columns or fields labeling the corresponding textual values that we need to sequence using NLP NER tagging and cluster to reach the right diagnosis. The patients in our dataset have multiple co-existing diagnoses with DM from multiple visits hence we have multiple classes to classify. Hence, we are trying to solve a multi-class and semi-supervised learning problem to diagnose DM and its comorbidity diseases in a patient or set of patients in large unified standardized corpora. The corpora in tabular form gave labeled/annotated features in sequence to classify and diagnose multiple endocrine diseases. These datasets formed an input to TensorFlow.Keras Sequential model embedded with Bi-LSTM layers and dense layers to give diagnoses. TensorFlow Keras DNN Sequential model gave a maximum accuracy of 0.9 for multiclass diagnoses of endocrine diseases.

Our contributions to this research entail (i) unified corpora built, (ii) a proposed sequential NER embedding mechanism and, (iii) achieved diagnostic accuracy of 0.9 for multiclass classification of DM and its comorbidities.

The rest of the paper is organized as follows. Section 2 discusses some related work that is done to further organize our experiments for NER tagging. The Proposed mechanism for NER embedding with Corpora details is given in section 3 of this paper. Section 4 explores ML algorithms and NLP tools and techniques by experimenting with traditional NER embedding approaches. Section 5 sheds light on sequential NER approaches and coming up with a custom NER model using Spacy for manual and automated annotation. Section 6 elaborates our proposed sequential DNN model for NER-embedded diagnoses of DM and its comorbidities. Section 7 evaluates all the experiments done showing that the DNN model gives better accuracy on a large corpus with an increased number of features. The conclusion is drawn in section 8 with future intentions to take this work further and diagnose all the diseases present in the corpora using multi-label and multiclass embedding.

2. Related Work

We did an in-depth study of previous research done in this domain of Natural Language Processing (NLP). It plays a significant role in embedding free text. Recent research is catering domain domain-specific NER embedding, and clinical contextual learning is gaining attention. Structured and unstructured clinical data is annotated to train analytics for NER embedding. Medical information extracted as Electronic Health Records (EHRs) or free text forms is labeled for getting clinical insights.

Natural Language Processing (NLP) [12] plays an important role in extracting rich information using deep phenotyping. Deep Phenotyping needs to be more expressive and interoperable for semantic intelligence. These phenotypes need to be in normalized form for decision-making or interpretation of meaningful information. Hence, the dataset or corpus needs standardization using the Human Phenotype Ontology

(HPO) or coding scheme like; ICD-10 or SNOMED, etc. Therefore, the whole process of application of NLP is challenging and a very cautious task for the best explainability of the clinical context. Textual clinical notes are also a good resource for data extraction for e-Phenotyping however challenging due to their free form for which two ways of extraction are there; symbolic and statistical [13]. Symbolic focuses on predefined relations where statistics annotates the corpus of text for finding semantic relationships. A study on the Natural Language Processing (NLP) tools and techniques was done by Ruas et al. in their doctoral thesis [3] and by Qureshi et al. for M-Health [4] that we referred to for semantic contextual embedding in the medical domain. There have been growing platforms for NLP processing on text in clinical notes to form interoperable data models but the first one was MedLEE (Medical Language Extraction and Encoding). Mayo Clinic [13], [14], while working on Learning Healthcare System also devised an NLP pipeline cTAKES (clinical text analysis and knowledge extraction system) that is open source to get clinical rules for symptoms, diagnosis, medication, etc. Research has led to the development of a large corpus of clinical text taken from Mayo Clinic in syntactic form.

The first machine learning application was applied to Phenotyping in 2007 [15] on a cohort of diabetic patients using feature selection via supervised model construction (FSSMC) with 47 filtered features ranked on the scale for their significance. At that time Naïve Bayes, C4.5 and IBl (Instance Bases Learning algorithms) were used to identify diabetic patients. In another study [16], prescription data, ICD-9 coding and clinical notes from the Unified Modeling Language System (UMLS) were employed to come up with a Phenotyping model using SVM for rheumatoid arthritis. This study took all feature structures and unstructured ones to show that SVM as in [17][18][19] was as good on unrefined feature sets as was on engineered. Noise in data could not be ignored for which Halpern et al. [20] used the framework of Agarwal et al. [21] XPRESS (extraction of phenotypes from records using silver standard) to build a platform for extraction of features and building models. These researchers assumed that large datasets would mitigate the effect of label errors by setting bounds and would generate results as good as in small data that is clearly labeled (Gold Standard). Phenotyping was defined as three pillars; (i) a complex relationship between multiple features, (ii) it is understandable by medical knowledge domain experts, and (iii) its definition is transferable into new domain knowledge. Researchers used this definition to introduce high-throughput Phenotyping [22] that was unsupervised and transformed in a scalable format. These phenotypes were clustered in correspondence with the diseases and validated by medical experts. PheKnow cloud tool by Henderson et al. [23] evaluated phenotypes derived from previous medical literature and associates them to the biomedical standard codes; latest International Classification of

Diseases (ICD) codes, SNOMED-CT (Systemized Nomenclature of Medical – Clinical Terms), or MeSH, etc. and ranks as per relativity thus limiting the need of medical expert review. Automated Feature Extraction for Phenotyping (AFEP) extracted features from medical resources like; Wikipedia and Medscape, to list UMLS concepts to train the classifier. Feature sets are more refined using NLP and ICD codes are given to develop hybrid applications like; ElasticNet on the Logistic Regression Model. SAFE (Surrogate-Assisted Feature Extraction) extended AFEP to include other resources like; Merck Manuals, Mayo Clinic UMLS and MedlinePlus Encyclopedia removing noise from phenotypes to classify manually labeled patients on gold standard.

Embedding applied on sparse text in a general or clinical context to multidimensional arrays/vectors is a known task and a recent survey addresses it in a clinical context [24]. This concept of contextual text embedding is understood as a de facto standard. Details of some medical corpora having certain characteristics are chosen for review and known embedding models are compared. Nine types of clinical embeddings are discussed with evaluation methods and solutions. Distributed vector representations are recent additions to the knowledge of natural language processing (NLP). Word embedding puts a word as part of hundreds of dimensions to learn semantic similarity with other similar words. Each dimension represents a feature itself. Word embeddings represent words in fixed-length vectors, and are dense in low dimensions. Word embedding [25] in sparse continuous vector space needs deep learning models for quantifying high-level textual representations. Bag-of-words has previously been used by researchers for NLP problems that represented a dimension related to the word as 1 and others as 0. These sets of 0s and 1s can be replaced by word frequencies, term frequency-inverse document frequency (TF-IDF) n-gram measures, etc. These previous traditional NLP methods did not consider the semantic similarity of words. Embedding solved this problem with an application related to unlabeled corpus and is used to map the text to dense vector representations overcoming the issue of dimensionality and adhering to finding semantic similarity in context. The survey [25] contains classification and comparison of medical corpora. The quality of embedding models is based on the size and type of corpus. In a large general corpus, there is a large vocabulary that can be inferred. A domain-specific corpus is inferred for the semantic similarity of terms. Medical corpora are categorized into four types; electronic health records (EHRs), social media medical corpus, online medical sources and scientific literature. Embedding models are compared that are; word2vec, paragraph2vec, glove, fasttext and Elmo. Embedding applications are being looked into for unsupervised and transfer learning as they infer an unlabeled corpus to map onto smaller datasets for smaller tasks. Embedding models are of two types; prediction and count-based models. Prediction-based embedding learns a context to predict target labels whereas count-based models learn the context to know word counts or their co-occurrences in any document or a corpus. Tanh and softmax activations are mostly used by previously proposed models for hidden and output layers to predict the next word of all the possible outputs for an unseen sentence of unknown dimensionality. Elmo builds on Bi-LSTM or CNN architecture is found best for word embedding to give context-level vector representations and understands similar contextual words and out-of-vocabulary (OOV) or misspelled words. Its drawback is intensive training time for massive computation.

Clinical embedding is classified as; Char, Word, Code, Concept Unique Identifier (CUI), Augmented, Patient, Phrase, Sentence and Document embedding. The resultant embedding is evaluated as intrinsic or extrinsic. Intrinsic evaluation of embedding for encoding similar/related contextual information is done using nearest neighbor search (NNS), clustering and similarity measures. Extrinsic evaluation is done by testing the model accuracy for input text for an expected output for name entity recognition (NER), medical text classification, medical concept normalization, etc. Known NLP methods listed for clinical predictions are; word2vec [24] and stacked de-noising auto-encoders, for medical coding; Glove [26], fasttext and word2vec have been preferred before, for NER in the clinical domain; word2vec and fasttext were chosen, for patient de-identification; Glove [26] or RNN encoder/decoder are used and for patient similarity word2vec. It is understood in [24] that NLP tasks vary from corpus to corpus and expertise of embedding applied by researchers. The size of the unlabeled corpus also influences the quality of embedding. A huge amount of medical text is developed by combining multiple corpora from different sources into corpora. Domain knowledge can further endorse embedding using ICD-10 or other standards like; RxNorm or SNOMED [27], or update embedding using other NLP methods like; word2vec, etc. Domain-specific embedding is improved by adding task-specific knowledge.

Information Extraction from Medical Data is an open issue discussed in detail [28] focusing on the challenges that hinder its progression. There are two concerns; (1) whether to develop a clinical decision support system (CDSS) or (2) design search engines for health-related queries; recommending patients for possible diagnosis and treatment or facilitating experts, clinicians, and doctors. Although there is a great deal of hype surrounding research in the health informatics field during our search for a solution to diagnose DM and its comorbidities having a large EHR diagnostic data of patients we came to know there was a lot of room for performance improvement in these systems. Tamine and Goeuriot [28] focused on information retrieval (IR) through state-of-the-art semantic search techniques to facilitate health informatics tasks. The semantic search capability also coincides with the feature tagging methods for medical search systems. Tamine and Goeuriot [28] emphasized on future direction for development in deep learning while elaborating on current research trends that open several issues and challenges in the field. There are two methods that assist in semantic search within in text or document and that are semantic gap analysis and finding vocabulary mismatch. A semantic gap refers to the difference in conceptual meaning of two sentences, phrases, or documents where vocabulary mismatch relates to the difference in lexical representations of two texts. WordNet, DBPedia or MeSH for the medical domain are known examples that relate words and create associations to understand literature. Wu et al. [29] had worked on automated free text mining to find phenotypes in a medical context. NLP process implementation or transfer has always remained difficult on new data or settings. Paper [29] proposes a distributed representation mechanism to train and reuse NLP models through identified phenotype embeddings in patient profiles. 23 phenotypes were extracted from 17 million documents of anonymous medical records from South London Maudsley, NHS Trust in the UK, for application on 6 morbidities. The experiments were done to reevaluate NLP models for the identification of 4 phenotypes. The proposed approach selects the best NLP model using two measures for quantification of reductions in duplicate and imbalance wastes. The proposed approach also guides in the validation and retraining of NLP models to perform up to 93% to 97% accuracy. Recent advancements show that text mining is being used to find associating features with diseases [30]. Electronic Health Records (EHRs) are there to keep these phenotypes in a structured format. Language models are best at finding patterns and relationships between dependable features. Accuracy increases by adding additional dense layers to be used as deep learning heuristics to analyze big EHR diagnostic data. These Language Models [11], [31] like bidirectional Long Short Term Memory (Bi-LSTM) combined with dense layers analyze big EHR data in a generalized way when given categorical feature vectors. Dense layers [32] of variable sizes of neurons build deep learning heuristics to analyze big complex datasets as in our case. A deep learning model is built and trained with the perspective of continuous data coming in to predict. Recently deep learning heuristics are seen embedded in automated clinical diagnoses architecture for higher accuracy [33]. NER Embedding [11] in EHR patients' diagnostic records to label disease names with associated attributes rely on annotated labeled data. Task-specific rules-based NER embedding is a known technique in clinical text. Recent promising research [7] has compared NLP techniques enhanced with neural networks for contextual embedding with traditional embedding in a clinical setting but could not provide a common standard mechanism to follow.

EHRs [24] is defined as a publicly available form of the known corpus. MIMIC is a public dataset fetched from Intensive Care Units (ICU) and prepared by MIT lab. The MIMIC dataset has three versions and the latest version collected data from 2001 to 2012. Access to this dataset is given with permission and a training course. Electronic libraries are being maintained to tag prominent medical vocabulary for named entities. Some medical resources mentioned in the paper [28] are; ICD-10, SNOMED-CT, MeSH and UMLS. NER tagging requires annotating the raw text dataset to be used as train and test sets. Annotation makes the raw text semantically intelligent to understand vocabulary and the contextual relationship of words in the document. Annotation [10] is either done manually or automatically for NER tagging. In the medical domain, it is mostly manual and ICD-10 codes are also given to related diagnoses by humans. Mostly it is seen that NER embedding follows Part-of-Speech (POS) tagging for best textual learning approaches [34]. In our study, we omitted the use of POS tagging and applied NER embedding only that minimized the code and kept it simple. Our experimental study took inspiration from the experiment run in [35] that predicted miscoded diabetes ICD-10 labels in a large EHR dataset extracted from CERNER Health Facts, a HIPPA-compliant repository maintained by the University of South California.

This previous research helped us to take it further and apply it to the custom DM_Comorbid_EHR_ICD10 corpora for diagnoses of endocrine diseases. Study of big data classification tools and techniques [36] was done to direct our experimental study for diagnosing DM and its comorbidity diseases labeled with related ICD-10-CM diagnostic codes. The data tables specifically for diagnostics of endocrine patients were fetched from the Shifa International Hospital EHR system. This normalized data was converted to flat tables using data warehousing techniques. It was cleaned and pruned where the target diagnostic labels were missing. We started off with some traditional ML algorithms like; multinomial logistic regression, decision tree, naïve Bayes, ada boost and light gradient boosting machine (Light GBM) as in [37]. Our previous explorations in [37] and [38] showed us some good results using deep learning heuristics. ML algorithms integrated with traditional NLP methods were also experimented with and results were obtained. Free text fields in a single patient profile in Corpus100_DM_pts_2844, were manually annotated using spaCy which gave us 789 lines of information-rich annotated text. Manual annotation was a time-consuming task and we used it to train a custom NER model for automated annotation of the corpora using the Spacy and Sequential DNN model available in the TensorFlow Keras framework.

MetaMap [39], cTakes [39] or QuickUMLS [28] would be looked into in future as cTakes [28] was not downloadable from ctakes.apache.org and our use of Interactive MetaMap [28] yielded us no results.

3. Architecture and Design

A high-level diagram of our diagnosis framework is shown in Figure 1. The figure illustrates that there are five main modules in our system. Starting from the left we provide EHR corpora as input to the extraction module. The extraction module covers the corpora in an input format suitable for machine learning and NLP. Various algorithms are employed in the ML module which returns trained classifiers. Finally, the model is executed on a cloud platform to return the primary diagnosis and comorbidity diseases. An abstraction of this figure has recently been presented by researchers at an international conference [40]. Detailed descriptions of the modules comprising our system are described in the subsections below.

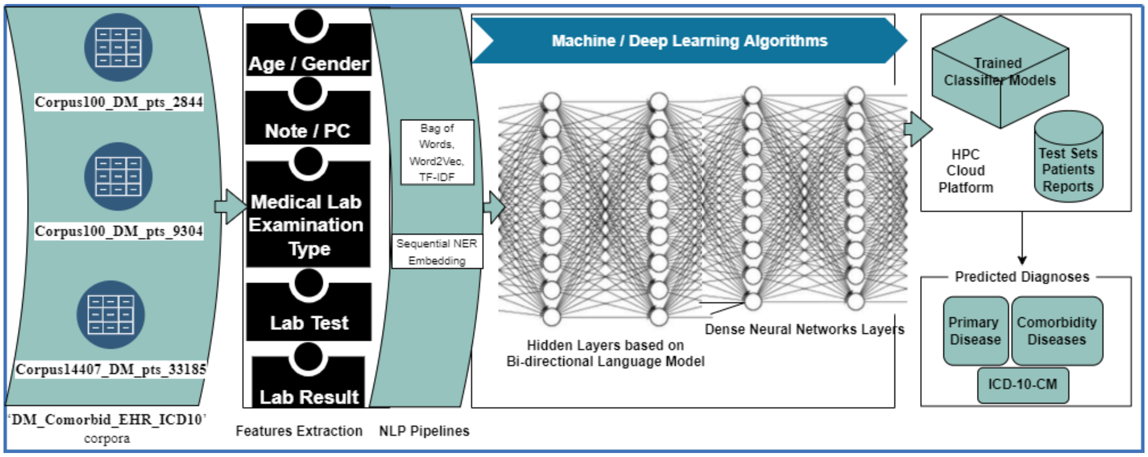


Figure 1. The Architectural Design for Clinical Diagnoses.

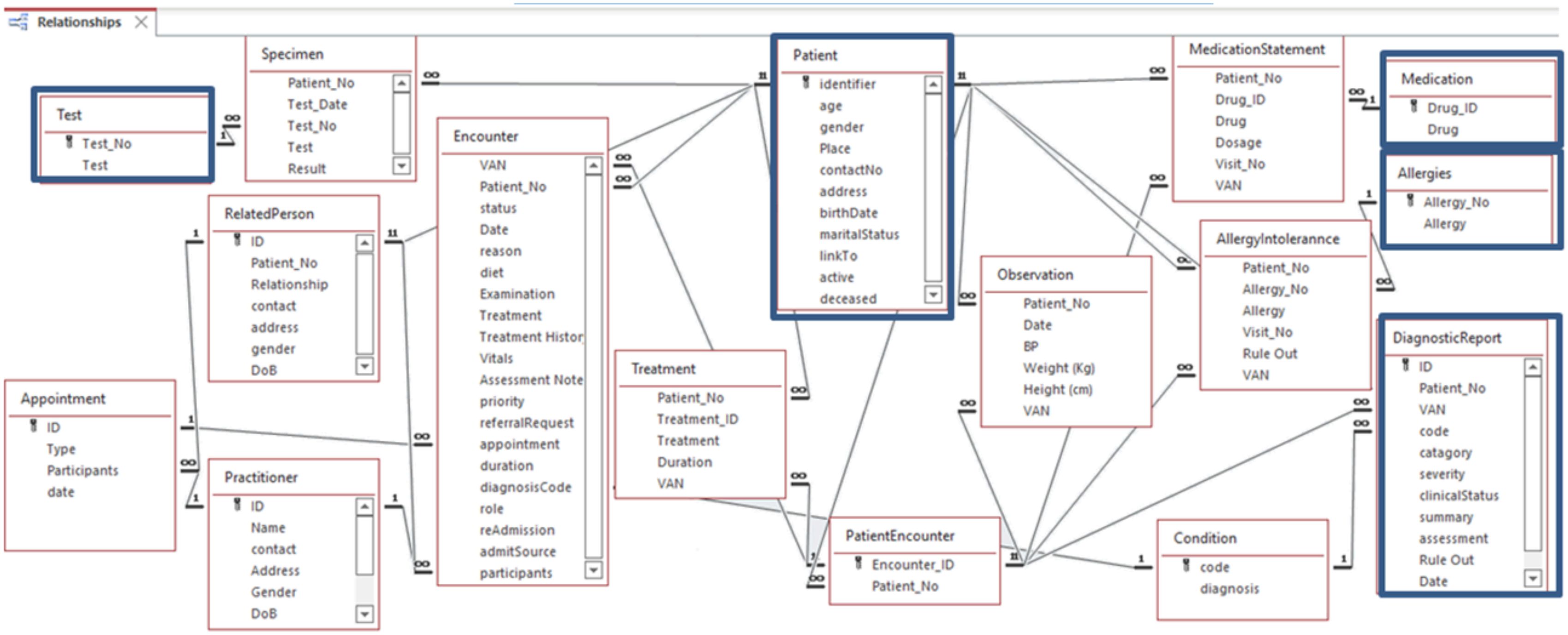

Relationships
Test
Test_No
Test
Specimen
Patient_No
Test_Date
Test_No
Test
Result
Encounter
VAN
Patient_No
status
Date
reason
diet
Examination
Treatment
Treatment Histor
Vitals
Assessment Note
priority
referralRequest
appointment
duration
diagnosisCode
role
reAdmission
admitSource
participants
RelatedPerson
ID
Patient_No
Relationship
contact
address
gender
DoB
Appointment
ID
Type
Participants
date
Practitioner
ID
Name
contact
Address
Gender
DoB
Patient
identifier
age
gender
Place
contactNo
address
birthDate
maritalStatus
linkTo
active
deceased
Treatment
Patient_No
Treatment_ID
Treatment
Duration
VAN
PatientEncounter
Encounter_ID
Patient_No
Observation
Patient_No
Date
BP
Weight (Kg)
Height (cm)
VAN
MedicationStatement
Patient_No
Drug_ID
Drug
Dosage
Visit_No
VAN
AllergyIntolerannce
Patient_No
Allergy_No
Allergy
Visit_No
Rule Out
VAN
Condition
code
diagnosis
Medication
Drug_ID
Drug
Allergies
Allergy_No
Allergy
DiagnosticReport
ID
Patient_No
VAN
code
catagory
severity
clinicalStatus
summary
assessment
Rule Out
Date
1
∞

Figure 2. Entity Relationship Diagram (ERD) for Diagnostic Data Model.

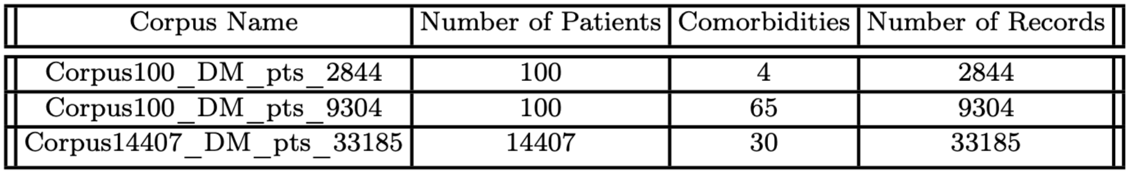
3.1. Unified Corpora

The work on building a knowledge base in healthcare is initiated. The corpora were prepared from health care data made available to us from the Management Information System (MIS) of Shifa International Hospital, Pakistan. They were provided Excel sheet templates which the health professionals filled out and returned to us. in three parts. It is to be noted that the patient IDs were anonymized for privacy purposes. The data model of these Excel sheets is shown as an Entity Relationship Diagram (ERD) in Figure 2. As can be seen in the figure these entities were designed based on the HL7 FHIR v4.0 schema. There are a total of sixteen normalized entities. The main entities are Patients, Tests, Allergies, Medication and DiagnosticReport. These were also the main entities that were used for features in our machine-learning algorithms. These Excel sheets were first imported into an access database as normalized tables. Subsequently, these were de-normalized into three flat data sheets. Data wrangling and preprocessing steps were carried out for cleaning and pruning where the labeled columns had missing or misspelled diagnoses.

To keep the naming convention uniform and meaningful throughout the rest of the paper the corpora and three corpora are named as follows:

'DiseaseName_Comorbid_EHR_StandardLabelConvention'

In this paper, we are diagnosing primarily DM patients and their comorbidity diseases labeled with ICD-10 codes Therefore, our corpora are named 'DM_Comorbid_EHR_ICD10'. The descriptions of the corpora comprising our 'DM_Comorbid_EHR_ICD10' corpora are provided in Table 1. As can be seen, the corpus size gradually increases with the number of patients.

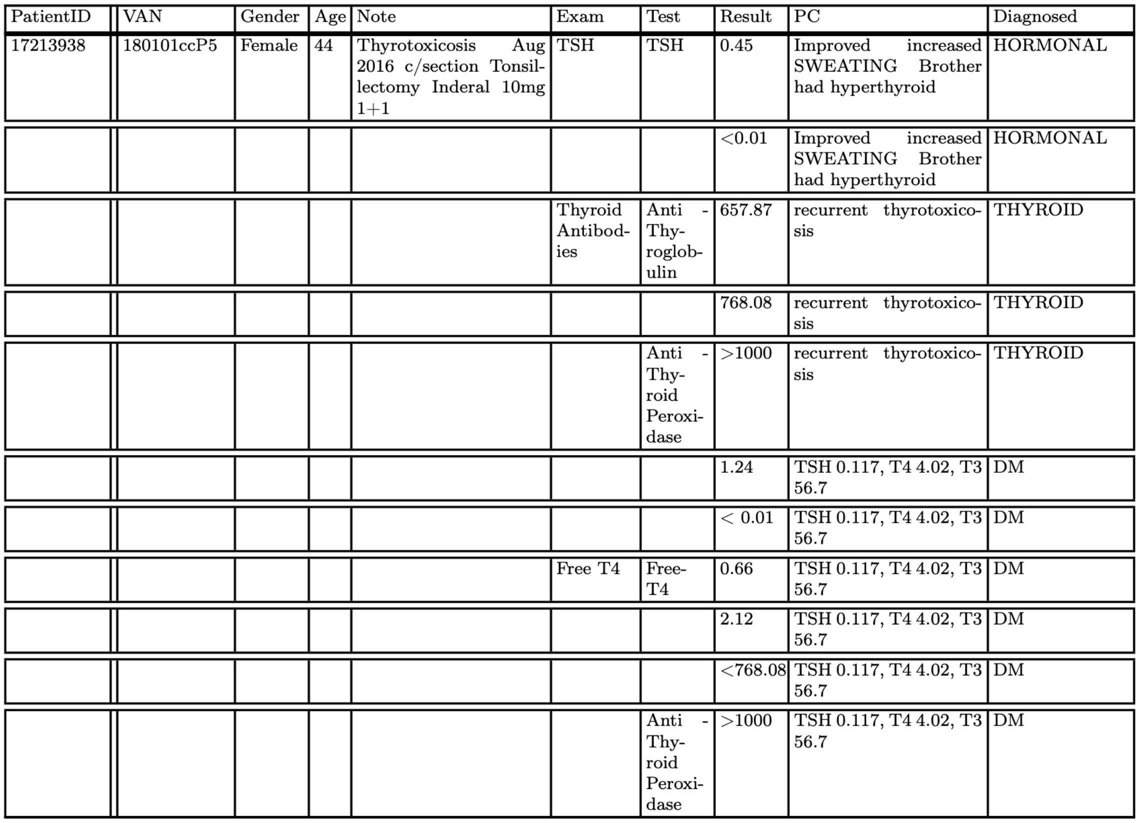


Corpus Name	Number of Patients	Comorbidities	Number of Records
Corpus100_DM_pts_2844	100	4	2844
Corpus100_DM_pts_9304	100	65	9304
Corpus14407_DM_pts_33185	14407	30	33185

Table 1. Corpus Descriptions.

3.2. Feature Extraction

Table 2 illustrates through the example of a single patient record what these corpora contain. An individual patient ID is connected to multiple visits and is denoted as Visit Account Numbers (VAN) as can be seen in column two of Table 2 along with the medical examinations recommended by the clinician over the course of these visits. Further, the laboratory examinations contain multiple prescribed lab tests with their corresponding results as well as the diagnoses. Raw text columns comprising clinical notes and practitioner comments are also included in the example.



PatientID	VAN	Gender	Age	Note	Exam	Test	Result	PC	Diagnosed
17213938	180101ccP5	Female	44	Thyrotoxicosis Aug 2016 c/section Tonsillectomy Inderal 10mg 1+1	TSH	TSH	0.45	Improved increased SWEATING Brother had hyperthyroid	HORMONAL
							<0.01	Improved increased SWEATING Brother had hyperthyroid	HORMONAL
					Thyroid Antibodies	Anti - Thyroglobulin	657.87	recurrent thyrotoxicosis	THYROID
							768.08	recurrent thyrotoxicosis	THYROID
						Anti - Thyroid Peroxidase	>1000	recurrent thyrotoxicosis	THYROID
							1.24	TSH 0.117, T4 4.02, T3 56.7	DM
							< 0.01	TSH 0.117, T4 4.02, T3 56.7	DM
					Free T4	Free-T4	0.66	TSH 0.117, T4 4.02, T3 56.7	DM
							2.12	TSH 0.117, T4 4.02, T3 56.7	DM
							<768.08	TSH 0.117, T4 4.02, T3 56.7	DM
						Anti - Thyroid Peroxidase	>1000	TSH 0.117, T4 4.02, T3 56.7	DM

Table 2. Example of a Single patient record in our corpus.

3.3. Advanced Machine / Deep Learning algorithms

This module houses a collection of advanced machine-learning algorithms with the goal of maximizing the accuracy and the speed of diagnosis. To build an optimal ML model, diagnostic data is required for correct prediction. However, the challenge for the ML algorithm is to understand and interpret the variety of formats in medical data that are quantitative e.g., tests like pH or are categorical e.g., negative or positive bilirubin. Therefore, we used fuzzification to unify the input features. In addition, the data is multidimensional because there are multiple classes of diagnosis for which we employed multinomial methods in the ML models. In fact, medical data also contains plain text e.g., clinical notes and practitioner comments but to process these we employed NLP techniques mentioned in the next section.

3.4. Natural Language Processing Pipeline

As mentioned in the previous section the clinical notes and practitioner comments features in the corpus cannot be reliably processed by traditional machine learning algorithms. For this text data the NLP pipeline is used for tokenization and preprocessing using the popular NER embedding techniques [24] that includes bag of words, TF-IDF for 1-gram, 2-gram and word2vector frequency analysis. This data was then used to train ML models through stratified sampling and was cross-validated.

3.5. Deep Neural Net (DNN) Layered Architecture for NER Embedding

For effective NLP it is important to have sequential NER embedding because it helps understand the semantics by defining a sequence of categorical features for diagnosis. For example, the proper sequencing for recommended medical analysis is first examination, then the test and then its result in that order, gives a diagnosis. The module of the NLP pipeline for sequential NER embedding works as a DNN. Selected categorical fields are converted to vectors; Examination, Test and Result. These inputs (x) are passed to the sequential model adapted with an embedding layer, bidirectional language model layers (Bi-LSTM) and dense layers. The Output (y) dense layer then predicts a multiclass diagnosis for a single patient labeled with ICD-10-CM codes. NER-embedded tags are given on the true prediction of test sets. The various submodules of the DNN module are described below.

1. Language Model: find patterns and relationships between dependent features and are helpful for our use case of classifying diseases. Using language models, we associated lab results comprising of medical examinations, test and their results. We utilized the Bidirectional Long Short-Term Memory (Bi-LSTM) language model which when combined with dense layers allowed us to analyze big EHR data in a generalized way on a given set of categorical feature vectors. Annotations of features or events are used to create contextual embedding taking word vectors as inputs. In our case, we have tabular fields to annotate the values in each column vector and sequence them as albeit form (x, y) to classify diagnoses with corresponding ICD-10-CM code for NER embedding.

2. Dense Layers: The accuracy of a DNN increases by adding additional dense layers of configurable sizes of neurons to build a deep learning heuristic tailored to our healthcare dataset. The deep learning model is specifically trained with the intention of embedding it for automated disease diagnosis by streaming input data on the cloud. Our big datasets of endocrine patients having multi-class diagnoses for a single patient add the desired complexity to test the accuracy of such embedding models based on DNNs. The architecture in Figure 1 has a flexible number of dense layers of variable sizes with respect to the complexity of the problem at hand.
3. Sequential NER Embedding: in EHR patients' diagnostic records to label disease names with associated attributes rely on annotated labeled data. NER embedding using task-specific rules is a known technique for processing clinical text. In this framework, we propose a novel NER sequential model that uses vectored categorical features; clinical examination, test and results combined as laboratory results to tag diagnoses with ICD-10-CM codes.

3.6. HPC Cloud Platform

Different ML and deep learning models integrated with NLP pipelines produce various trained classifier models. These models were needed for experiments to achieve the maximum accuracy possible. The models were then validated on test datasets of different sizes and needed HPC cloud platforms for processing. We consider three cloud platforms to compare performance and speed achieved with selected classifier models. These three platforms are: (i) Anaconda, (ii) Google Colab and (iii) Gradient Paperspace for High-Performance Computing (HPC).

3.7. Predicted Diagnoses

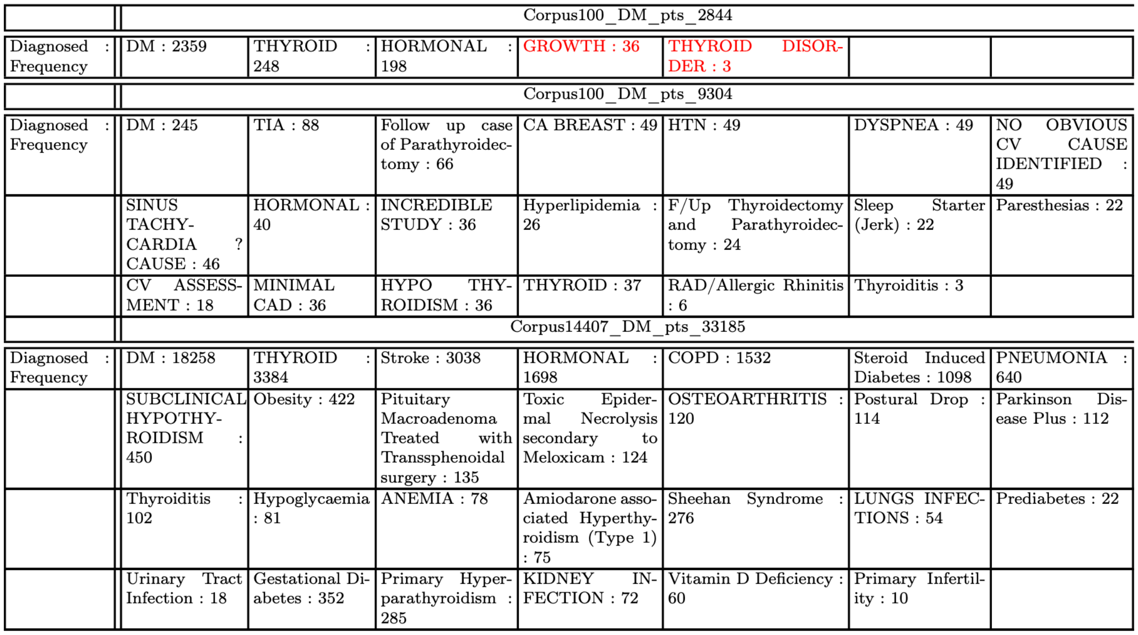
The test datasets had endocrine-diagnosed diseases with DM as primary and several other coexisting comorbid secondary diseases affecting individual patients. The outputs gained from different classifier models are presented in different forms showing corresponding diagnoses as primary and secondary diseases.

4. Diagnoses using Machine Learning Algorithms and NLP Pipeline

This section provides the implementation-level details of the architecture introduced previously. We start with an exploratory data analysis and discuss the machine learning and NLP algorithms applied.

4.1. Exploratory Data Analysis (EDA)

The corpora are used to diagnose corresponding diseases listed in Table 4 with their frequency of occurrences. The first partition of rows shows five disease labels for corpus100_DM_pts_2844. The two highlighted diseases were not included in the analysis due to repetition in the records and ambiguity in the terminology. The other two corpora likewise have 20 and 27 diseases listed respectively. The diseases shown in Table 4 are those that have all the features including clinical notes and practitioner comments. These corpora are explored for data and NLP preprocessing to be inputted into the ML algorithms to train classification models. For the exploratory data analysis, we used visualization techniques to characterize our data and understand the summary statistics of the patients. Practitioners' diagnoses listed for different endocrine diseases with the relevance of the respective corpora with the frequencies in Table 3 are relatable with figures 3, 4, 5 and 6.

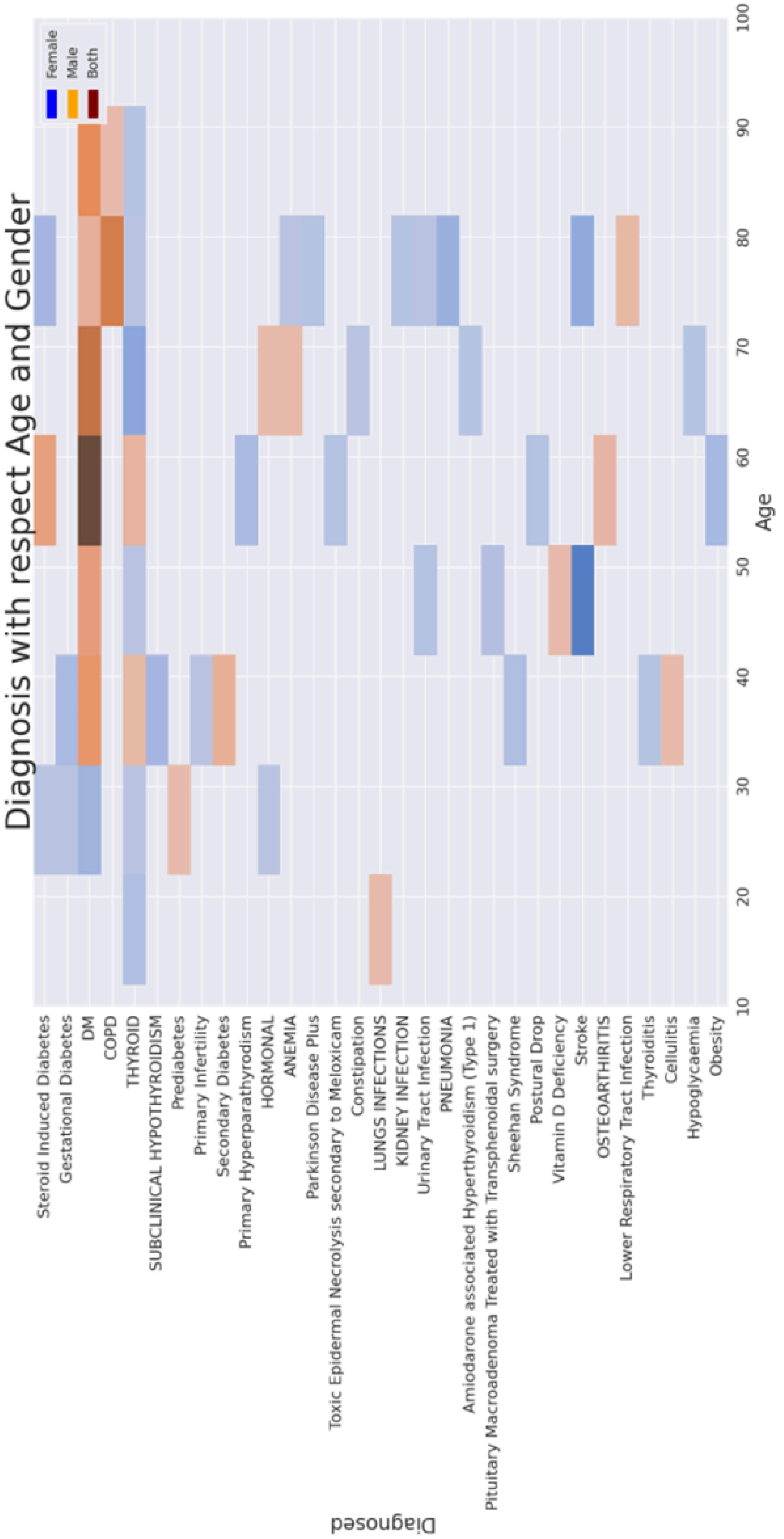


	Corpus100_DM_pts_2844						
Diagnosed : Frequency	DM : 2359	THYROID : 248	HORMONAL : 198	GROWTH : 36	THYROID DISORDER : 3		
	Corpus100_DM_pts_9304						
Diagnosed : Frequency	DM : 245	TIA : 88	Follow up case of Parathyroidectomy : 66	CA BREAST : 49	HTN : 49	DYSPNEA : 49	NO OBVIOUS CV CAUSE IDENTIFIED : 49
	SINUS TACHYCARDIA ? CAUSE : 46	HORMONAL : 40	INCREDIBLE STUDY : 36	Hyperlipidemia : 26	F/Up Thyroidectomy and Parathyroidectomy : 24	Sleep Starter (Jerk) : 22	Paresthesias : 22
	CV ASSESSMENT : 18	MINIMAL CAD : 36	HYPO THYROIDISM : 36	THYROID : 37	RAD/Allergic Rhinitis : 6	Thyroiditis : 3	
	Corpus14407_DM_pts_33185						
Diagnosed : Frequency	DM : 18258	THYROID : 3384	Stroke : 3038	HORMONAL : 1698	COPD : 1532	Steroid Induced Diabetes : 1098	PNEUMONIA : 640
	SUBCLINICAL HYPOTHYROIDISM : 450	Obesity : 422	Pituitary Macroadenoma Treated with Transsphenoidal surgery : 135	Toxic Epidermal Necrolysis secondary to Meloxicam : 124	OSTEOARTHRITIS : 120	Postural Drop : 114	Parkinson Disease Plus : 112
	Thyroiditis : 102	Hypoglycaemia : 81	ANEMIA : 78	Amiodarone associated Hyperthyroidism (Type 1) : 75	Sheehan Syndrome : 276	LUNGS INFECTIONS : 54	Prediabetes : 22
	Urinary Tract Infection : 18	Gestational Diabetes : 352	Primary Hyperparathyroidism : 285	KIDNEY INFECTION : 72	Vitamin D Deficiency : 60	Primary Infertility : 10	

Table 3. Diagnostic classes from multiple visits of endocrine patients with frequencies of occurrences.

1. Gender, Age, Examination Ratio with Diagnoses: Figure 3 illustrates the correlation between the patient gender and age with the practitioner diagnoses present in Corpus14407_DM_pts_33185. In endocrine patients, it is observed that the ratio of females is more compared to men. There are some diseases like urinary tract infection, obesity, infertility, and gestational diabetes generally seen only in females and chronic obstructive pulmonary disease (COPD) or pre-diabetes are identified mostly

in males. It can be observed that DM in females is typically diagnosed in the early middle ages from 25 to 40 and then in the later ages of 50 to 60. Similarly, males are diagnosed with DM after their 30s. Other comorbidity diseases evolve at later stages resulting from mismanagement of DM. Figure 4 provides a sunburst plot of gender that can be seen in the innermost circle for corresponding diagnoses in the middle circle and recommended examinations in the outermost circle. Amongst all the diseases in Corpus100_DM_pts_9304 shown in figure 4 affecting females, DM and breast cancer are the most prevalent. The data in figure 4 shows that practitioners mostly recommend several examinations to ascertain that a patient is suffering from a particular disease. For example, to accurately diagnose breast cancer there are 22 possible examinations. The HbA1c examination is important as shown expanded in the orange box on the right corner of figure 4. It is conducted to diagnose DM and its severity in any patient as here it is done for a male patient.



doi.org/10.32388/HPAUYJ

Figure 3. Correlation between gender and age with practitioner diagnosis.

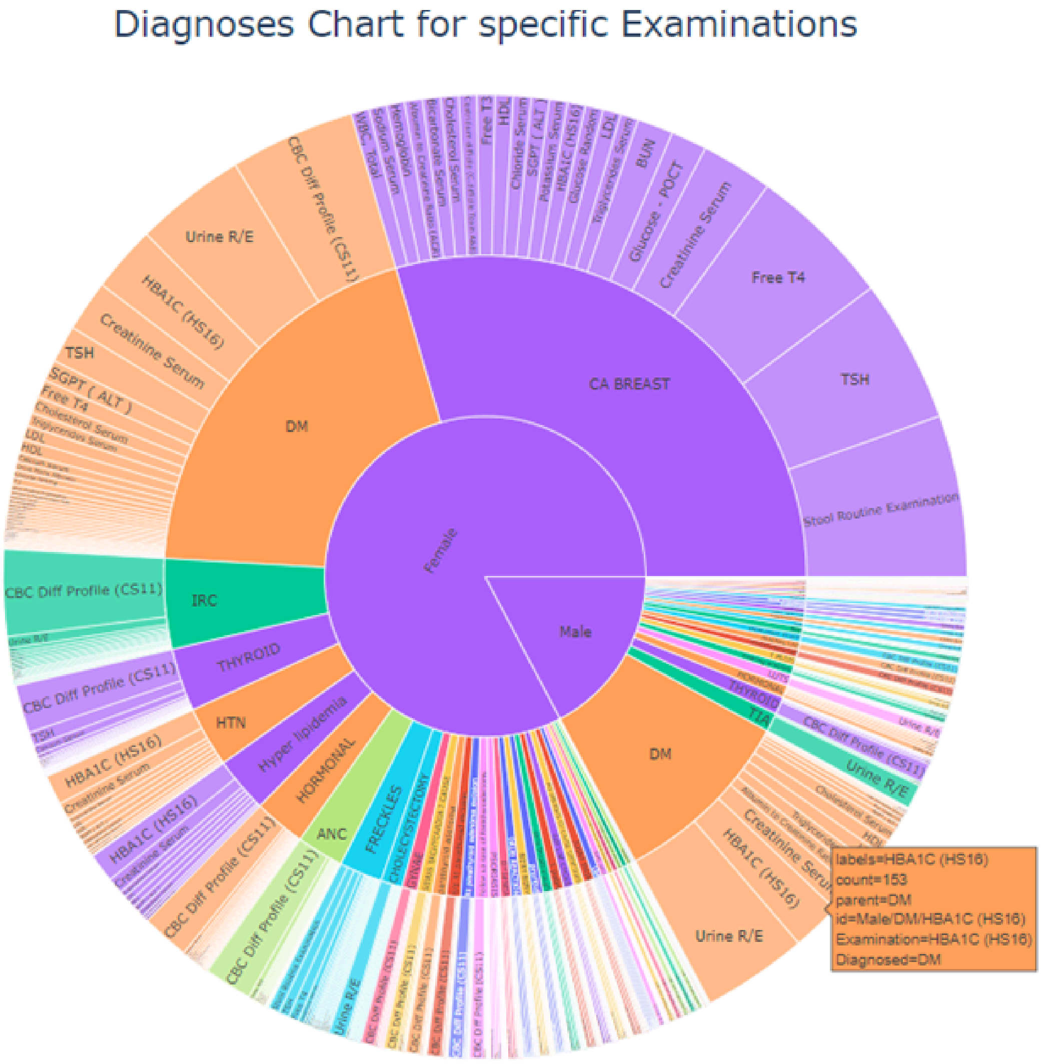
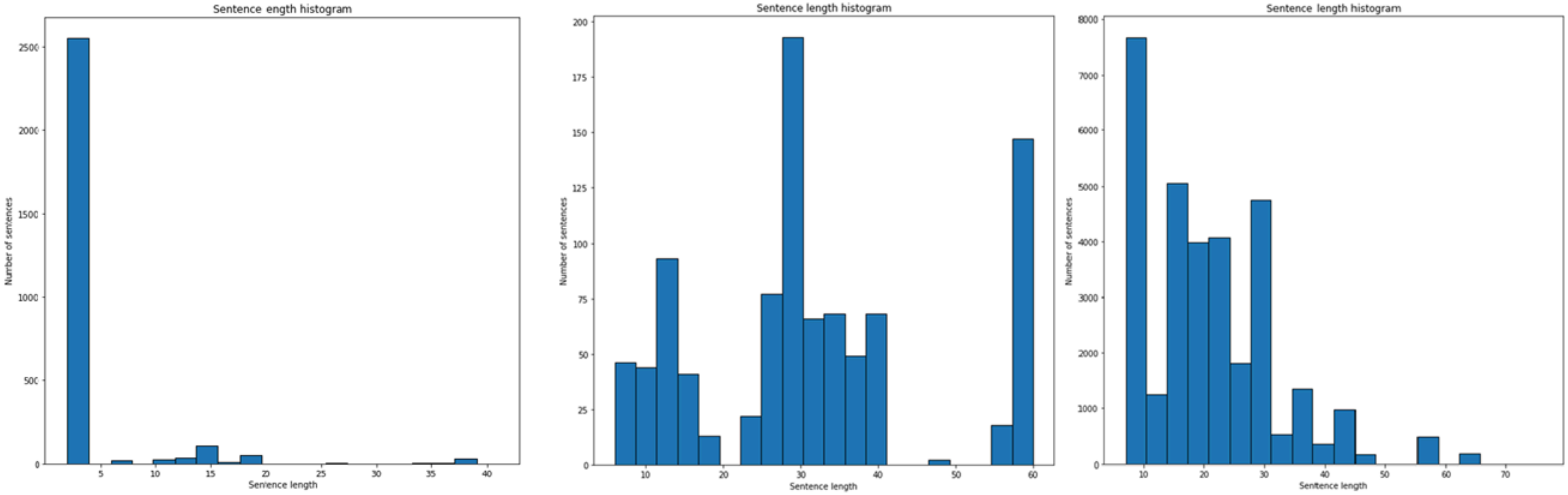


Figure 4. Sunburst for Examination and Gender with diagnosis.

2. Word Embedding for Clinical Notes and Practitioner Comments:}

NLP is used to preprocess clinical notes and practitioner comments in all records in the corpora to find embedding in sentences for input into ML algorithms. In Figure 5 there are three bar graphs for each corpus showing sentence length on the x-axis and their frequencies on the y-axis. There were five classes of diseases in Corpus100_DM_pts_2844 out of which we only considered three discarding the other two erroneous disease labels that were repeating themselves or were not understandable.

The three diseases taken for analysis are DM, Thyroid and Hormonal as was shown in the first row of Table 4. The clinical notes and practitioner comments on these three diseases are analyzed for sentence length. The sentences had a maximum length of 41 words and a mean length of 3.56 which were tokenized into a total of 10114 words with a vocabulary size of 208. On tokenizing clinical notes and practitioner comments in Corpus100_DM_pts_9304, we get 29670 words total from sentences ranging from a maximum length of 60 to a mean sentence length of 31.33 with a vocabulary size of 464. In Corpus14407_DM_pts_33185, the sentences ranging from a maximum length of 76 and mean sentence length of 20.9 are tokenized into 681676 words total, with a vocabulary size of 1151. In our understanding the varying sentence lengths could affect the performance of ML models and diagnostic results therefore we used NLP word embedding techniques to tokenize. Tokenization into unigrams (i.e. tokenized into separate words), and removal of stop words and bad symbols were done with the goal of balancing the data and for fast processing in training multinomial ML models with weighted average [41]. This way of preprocessing clinical notes and practitioner comments did not let sentence length and vocabulary affect the diagnostic results and performance of the ML models.



(a) *Corpus100_DM_pts_2844*

(b) *Corpus100_DM_pts_9304*

(c) *Corpus14407_DM_pts_33185*

Figure 5. (a) Minimum sentence length = 2, Maximum sentence length = 41 in Corpus100_DM_pts_2844 (b) Minimum sentence length = 5, Maximum sentence length = 60 in Corpus100_DM_pts_9304 (c) Minimum sentence length = 5, Maximum sentence length = 76 in Corpus14407_DM_pts_33185.

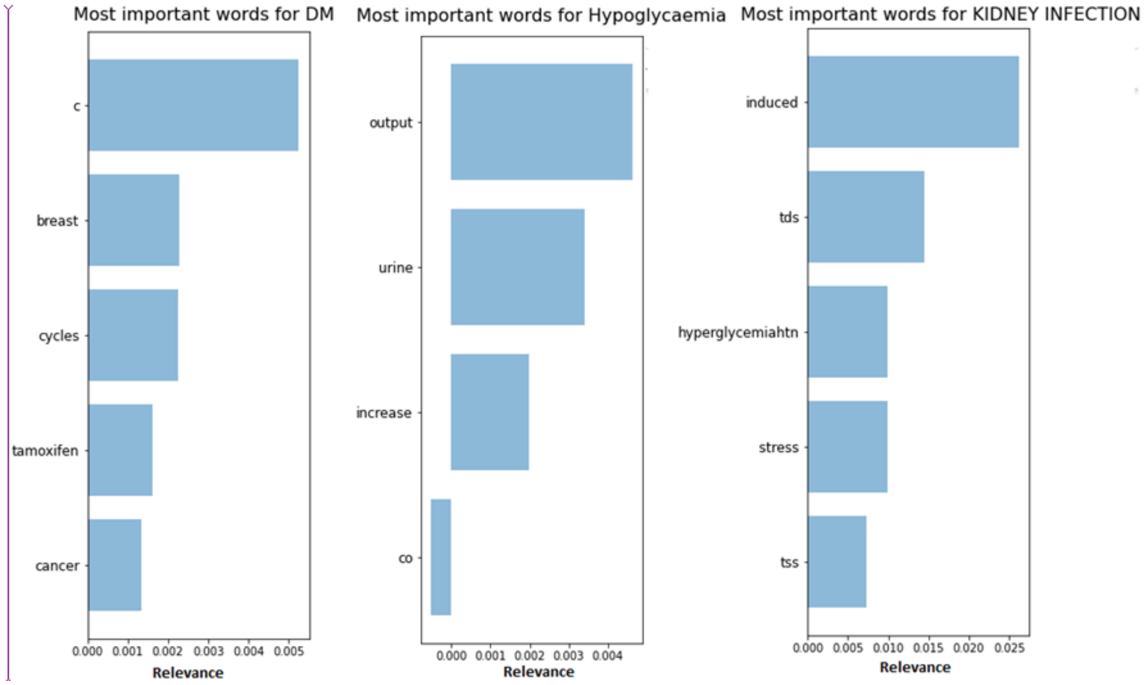


Figure 6. Extracted words relevant to DM and comorbidities.

2. In Figure 6, three bar graphs are plotted for each of the major diseases namely DM, hypo-glycaemia and kidney infection. Relevance to the diagnosis of these diseases is shown on the x-axis of some important keywords (on the y-axis) extracted from tokenization. Note the keywords ‘tamoxifen’, ‘breast’ and ‘cancer’ have a high degree of importance. This is because Breast cancer is treated with the medication tamoxifen and has a strong relationship with DM. Medical research shows that patients who recovered from breast cancer formed DM later due to chemotherapy treatment. Similarly in the second graph, hypoglycemia means low blood glucose that DM patients often undergo. ‘Increased’ and ‘urine’ are other keywords related to hypoglycemia and in the medical literature, this test is shown to be a very important measure that doctors recommend to patients for such diagnoses. This condition of low glucose results in several complications as can be seen in the

third graph of Figure 6 resulting in stress, HTN or even kidney infection. All these correlations between words related to DM and its comorbidities were also manually validated from the relevant literature in the discipline. It is also interesting to note that inverse relationships typically exist as well where for example patients prone to DM also form diseases like breast cancer, hypoglycaemia, HTN or kidney infection.

4.2. Traditional Machine Learning Algorithms Applied

Multi-label encoding is a technique for fuzzification. We have a human-understandable dataset having multiple categorical column fields that contain labels in text. Traditional machine learning algorithms process quantitative data more accurately and therefore label encoding converts the labels into numerical form. This preprocessing of structured data is an important step in supervised learning. We used the LabelEncoder() method in Scikit-learn which takes multiple columns as arguments and returns a numbered matrix for input into ML algorithms.

Our ML models comprised of the below algorithms were run successfully on our input corpora.

1. Logistic Regression [42][43][44][45][46] model is taken from statistical method to evaluate or distinguish between classes or events through probabilistic distribution. It may be binary in the case of two classes or multinomial if multiple target classes exist with regard to any specific classification problem as diagnosed in our case.
2. Decision Tree algorithm [47], [45], [48][49][50] breaks the problem or features (nodes) in a tree-like structure to find sequence or association between each node to reach a decision or a consequence and finally a target class (leaf).
3. Adaptive Boosting (AdaBoost) algorithm [48], [51][52][53][54] when combined with other types of learning algorithms boosts the overall performance of a classifier.

4.3. Analytics applied using NLP and ML techniques

NLP methods are applied to preprocess the textual fields in the datasets before the application of ML algorithms. Distributed vector representations are recent additions to the knowledge of natural language processing (NLP) [24]. The previous traditional NLP methods did not consider the semantic similarity of words. Embedding solved this problem with application on unlabeled corpus and is used to map the text to dense vector representations overcoming the issue of dimensionality and adhering to find semantic similarity in context. The quality of embedding models is based on the size and type of corpus whether general or domain-specific. In a large general corpus, there is a large vocabulary to infer. The domain-specific corpus is inferred for semantic similarity of terms as in our case of clinical diagnoses. Data is preprocessed taking different NLP embedding as vector lists from a bag of words, word2vec and TF-IDF. Analytics is then applied using cross-validation with stratified sampling in logistic regression, Naïve Bayes and Light Gradient Boosting Machine (GBM). These embedding techniques and ML algorithms are detailed below.

1. Bag of Words [10], [55][56][57] is a multi-set representation of all the vocabulary present in the text or a document ready to be used in NLP for information extraction. A bag of words has previously been used by researchers for NLP problems that represented a dimension to related words as 1 and unrelated as 0. These sets of 0s and 1s can be replaced by word frequencies, TF-IDF and n-gram measures.
2. Term Frequency Inverse Document Frequency (TF-IDF) [55], [58][59][60] is a statistical evaluation of a word in relevance of its importance in a document or a corpus. It is important to assign weights to a word for its relevance in a particular finding or prediction.
3. Word2Vec [7], [55], [58], [61], [62] is an NLP technique that uses neural networks to learn relationships between words in a textual corpus. Each word has a particular number to form a list represented as a vector. All the words listed in the vector have some sort of semantic similarity that can be deduced using a mathematical function. Word embedding puts a word as part of hundreds of dimensions to learn semantic similarity with other similar words. Each dimension represents a feature itself. Word embedding represents words in fixed-length vectors, dense in low dimensions. Word embedding in sparse continuous vector space needs deep learning models for quantifying high-level textual representations.
4. Naïve Bayes [63][64][65][66] is a simple probabilistic classifier based on the Bayesian statistical model. The Bayes classifier considers all features independently contributing to the target label without any correlation. A multinomial Naïve Bayes is chosen for this multi-class diagnostic problem.
5. Multinomial logistic regression [46], [66][67][68][69] generalizes the classifier to multi-class problems where there are more than two target classes. It is applied to predict categorical or nominal class variables.
6. Light Gradient Boosting Machine (Light GBM) [70][71][72][73] is a fast decision tree algorithm that supports a high-performance distributed gradient boosting framework. It is used with multi-class objectives here. It is fast as it puts continuous features in discrete bins for efficient memory usage.

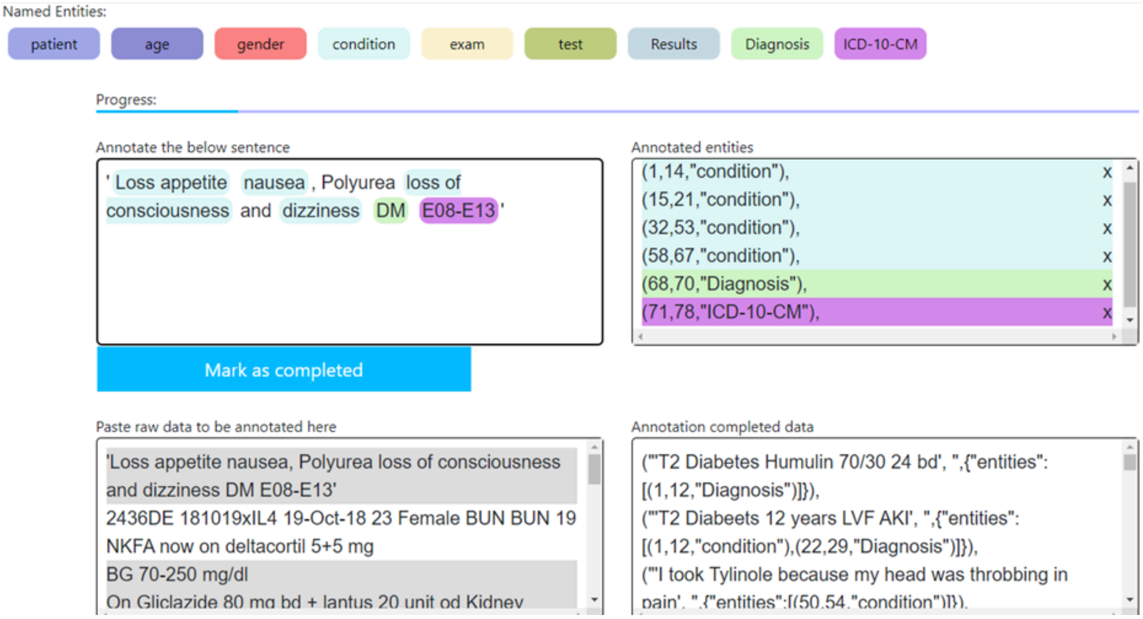
5. Sequential NER Embedding Techniques

Anaconda, Google Colab and Gradient Paperspace are cloud platforms with GPU and high-performance computing (HPC) data science frameworks that were used to implement the NLP models for sequential NER embedding. Anaconda and Google Colab are open-source and widely used by the research community, whereas Gradient Paperspace is based on a subscription-based pricing model and all three provide analytical processing capability by supporting frameworks like tensorflow. A comparative analysis of the efficiency of these three platforms was performed as is detailed in the later sections. The sequential NER analytical techniques used in our research are applied to the DM_Comorbid_EHR_ICD10 corpora. Phenotyping means to extract patient characteristics like age, gender, vitals and symptoms from datasets. Our goal is to perform phenotyping of our corpus using text mining techniques. There are three ways of doing this namely manual annotation through spaCy [9], pre-trained NLP models like HunFlair [74], [75] and training of custom NLP models using keras sequential model [76].

5.1. Manual Annotation

To extract clinical embedding from the textual fields in the dataset manual annotation using spacy was employed. Figure 7 elaborates on this process where nine named entities were defined for clinical diagnosis. These entities are as follows: patient, age, gender, condition, exam, test, results, diagnosis, and ICD-10-CM. Our naming convention is based on the FHIR HL7 schema that has defined a comprehensive set of entities for clinical diagnosis. The ‘patient’ entity captures variables such as patient identifier, age, and gender. Our labels ‘exam’ and ‘test’ are based on events from the HL7 ‘observation’ entity which captures different categories and methods some important ones being vital signs, BMI, Triglyceride, HDL, and LDL. Then ‘results’ are drawn for these observations to reach a final diagnosis. Finally, our ‘ICD-10-CM’ is based on the ‘code’ entity in HL7. Other standardized coding schemes include SNOMED and RxNorm.

We employed a team of nursing staff to annotate some textual data. The free text present in our dataset is taken as sentences and individual words or sequences of words that are annotated with related named entities as positional arguments. In figure 7, there are four compartments. Raw data is entered into the bottom left corner box ready to be annotated. The upper left window shows the current sentence that is being annotated where ‘loss of appetite’, ’nausea’, ‘loss of consciousness’ and ‘dizziness’ are tagged with the label ‘condition’. ‘DM’ is tagged with the ‘diagnosis’ label and ‘E08-E13’ as ‘ICD-10-CM’. The right-hand side window shows the spans of labeled entities that are identifiable in different colors. Finally, on tagging each sentence when the ‘Mark as completed’ tab is pressed all the annotated data gets collected in the bottom right window in annotated format to be stored as a dataset. It was observed that one staff member was able to annotate on average 70 comments per hour. With an average of 6000 comments per dataset, it is not hard to imagine that this process can easily become intractable. However, it is to note that if this process is conducted comprehensively, it can be very useful to build future knowledge bases as annotated datasets to train NER models with human precision. Due to increased man hours, we preferred to use automated annotation for our problem and finally came up with a custom bi-LSTM DNN sequential model for NER embedding built on categorically annotated tabular datasets.



' Loss appetite nausea , Polyurea loss of consciousness and dizziness DM E08-E13 '

```
(1,14,"condition"),
(15,21,"condition"),
(32,53,"condition"),
(58,67,"condition"),
(68,70,"Diagnosis"),
(71,78,"ICD-10-CM"),
```

```
'Loss appetite nausea, Polyurea loss of consciousness
and dizziness DM E08-E13'
2436DE 181019xIL4 19-Oct-18 23 Female BUN BUN 19
NKFA now on deltacortil 5+5 mg
BG 70-250 mg/dl
On Gliclazide 80 mg bd + lantus 20 unit od Kidney
```

```
('"T2 Diabetes Humulin 70/30 24 bd', ",{"entities":
[(1,12,"Diagnosis")]}),
('"T2 Diabeets 12 years LVF AKI', ",{"entities":
[(1,12,"condition"),(22,29,"Diagnosis")]}),
('"I took Tylinole because my head was throbbing in
pain', ",{"entities":[(50,54,"condition")]}),
```

Figure 7. The spacy manual annotation is shown tagging named entities.

5.2. Automated Annotation

Flair [74] is an easy-to-use NLP framework built on top of PyTorch. It boasts a unified interface for embedding and labeling sequences in contextual domain-specific data. Flair can integrate with manual annotation tools like Spacy and NLTK. Reusing pre-trained word embedding models is helpful for generalized learning from unlabeled data included in an already-learned approach. HunFlair [75] is a specialized version of Flair that is designed for embedding in a medical context and has been trained on the National Center for Biotechnology Information (NCBI-Disease) data. This model was tested and validated on clinical notes stored in an array. The HunFlair model tags the sequence of words in a disease entity as either a beginner word with <B-Disease>, an inner word with <I-Disease>, or an ending word with <E-Disease>. If it is just a single word, then it's tagged <S-Disease>. Non-relevant sentences were scanned as having no entities.

array =

['I took Tylinole because my head was throbbing in pain',

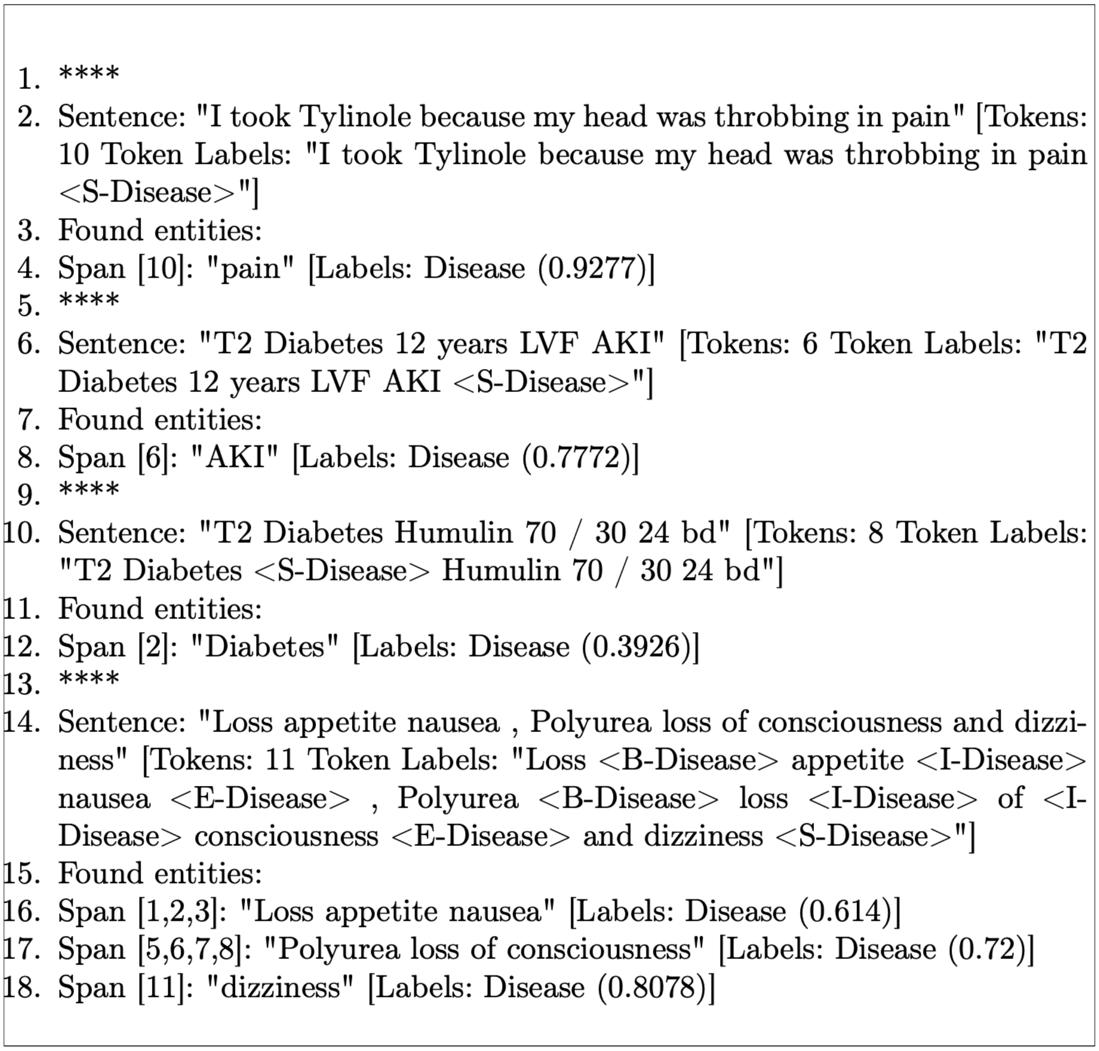
'T2 Diabetes 12 years LVF AKI',

'T2 Diabetes Humulin 70/30 24 bd',

'Loss appetite nausea, Polyurea loss of consciousness and dizziness']

We found some limitations in conducting our experiment with HunFlair in the medical learning context. It is observed in Table 4 that in some sentences where there were spelling errors as in; 'Diabeets' at line 6, it could not be recognised as a DM disease. It misinterpreted some symptoms as a disease, for example, pain (line 4), loss of appetite nausea (line 16), loss of consciousness (line 17) and dizziness (line 18).

On finding the above limitations in the HunFlair model we trained a custom NER model on the partially manually annotated dataset using spacy. This model could differentiate 'condition' from 'disease'. We further trained it on labels; age, gender, family, exam, test, result, medicine, procedure, and ICD-10-CM. Training of these embeddings would take some more time and effort to reach maximum accuracy.



```
****
Sentence: "I took Tylinole because my head was throbbing in pain" [Tokens: 10 Token Labels: "I took Tylinole because my head was throbbing in pain <S-Disease>"]
Found entities:
Span [10]: "pain" [Labels: Disease (0.9277)]
****
Sentence: "T2 Diabetes 12 years LVF AKI" [Tokens: 6 Token Labels: "T2 Diabetes 12 years LVF AKI <S-Disease>"]
Found entities:
Span [6]: "AKI" [Labels: Disease (0.7772)]
****
Sentence: "T2 Diabetes Humulin 70 / 30 24 bd" [Tokens: 8 Token Labels: "T2 Diabetes <S-Disease> Humulin 70 / 30 24 bd"]
Found entities:
Span [2]: "Diabetes" [Labels: Disease (0.3926)]
****
Sentence: "Loss appetite nausea , Polyurea loss of consciousness and dizziness" [Tokens: 11 Token Labels: "Loss <B-Disease> appetite <I-Disease> nausea <E-Disease> , Polyurea <B-Disease> loss <I-Disease> of <I-Disease> consciousness <E-Disease> and dizziness <S-Disease>"]
Found entities:
Span [1,2,3]: "Loss appetite nausea" [Labels: Disease (0.614)]
Span [5,6,7,8]: "Polyurea loss of consciousness" [Labels: Disease (0.72)]
Span [11]: "dizziness" [Labels: Disease (0.8078)]
```

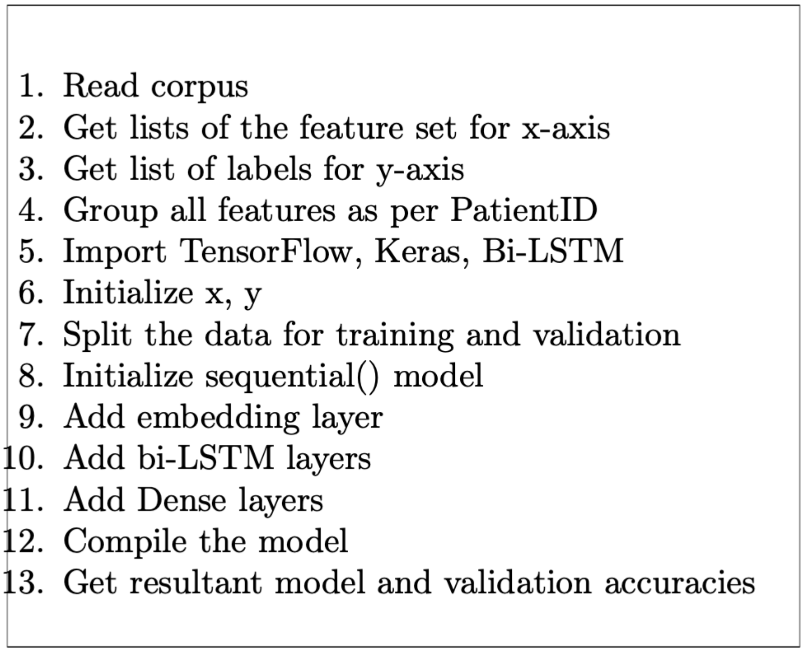
Table 4. Results for tagging tokenized array of sentences are depicted as Labels with mentioned accuracies.

6. TensorFlow.Keras NER Embeddings using Bi-LSTM Dense Layered Neural Networks Architecture

We used the categorically annotated ‘DM_Comorbid_EHR_ICD10’ corpora as input to understand the sequence of features to predict diagnosed patients individually for DM and its comorbid diseases with their relative ICD-10-CM codes. A custom NER model built with Spacy is used for annotation of free text fields; ‘Note’ and ‘PC’ to extract features ‘condition’, ‘disease’, and ‘medicine’. These features relate to the patient’s current condition with which he/she visited the doctor for consultation. A sequential model [77] based on the TensorFlow Keras library was used to hold a stack of embedding of bidirectional long short-term memory (Bi-LSTM) [78] and dense layers of varying sizes built on the recurrent neural networks architecture. TensorFlow.Keras [76], [79] was adopted as a neural networks interface to preprocess the finalized sequential columns: 'test', 'examine', 'result', 'condition', 'disease', 'medicine' and 'diagnosed' in our 'DM_Comorbid_EHR_ICD10' corpora.

Table 5 shows a dry run of the Python code. The selected feature is denoted as x, and the target feature diagnosed is denoted as y. We preferred Relu as the activation function for DNN layers and the outer dense layer is set softmax as its activation function. The optimizer chosen was Nesterov-accelerated Adaptive Momentum Estimation (NAdam). NAdam is a higher version of Adam and uses Nesterov momentum.

The hyper-parameters such as vocabulary size, number, and size of layers, learning rate, optimizers and accuracy metrics, in the sequential analytics model were learned on multiple runs on each corpus. The f1-score was not found significant.



```
 1. Read corpus
 2. Get lists of the feature set for x-axis
 3. Get list of labels for y-axis
 4. Group all features as per PatientID
 5. Import TensorFlow, Keras, Bi-LSTM
 6. Initialize x, y
 7. Split the data for training and validation
 8. Initialize sequential() model
 9. Add embedding layer
10. Add bi-LSTM layers
11. Add Dense layers
12. Compile the model
13. Get resultant model and validation accuracies
```

Table 5. The pseudo-code of TensorFlow.Keras Sequential() model to predict diagnosed diseases with ICD-10-CM codes.

Figure 8 evaluates the Sequential model with accuracies achieved for the three feature sets selected in the three corpora. The feature sets are; (i) (‘exam’,’test’,’result’), (ii) (‘exam’, ’test’, ’result’, ‘condition’), (iii) (‘exam’, ’test’, ’result’, ‘condition’, ‘disease’, ‘medicine’). It is observed that the overall accuracy increases as the corpus size grows. The maximum accuracy achieved amongst all feature sets for Corpus100_DM_pts_2844 is 0.4615 for the selected feature set (‘exam’,’test’,’result’) as other features in the corpus mostly hold null values and those records get dropped during analysis. The maximum accuracy of 0.6 and 0.9 is achieved in Corpus100_DM_pts_9304 and Corpus14407_DM_pts_33185 respectively with the maximum number of features selected. This observation reflects the importance of free-text clinical notes and practitioner comments from which the key attributes; condition, disease and medicine are extracted for accurate diagnoses. Another observation made is that the number of DNN layers increases with the size of the corpus being analyzed where the learning rate was set at 0.05.

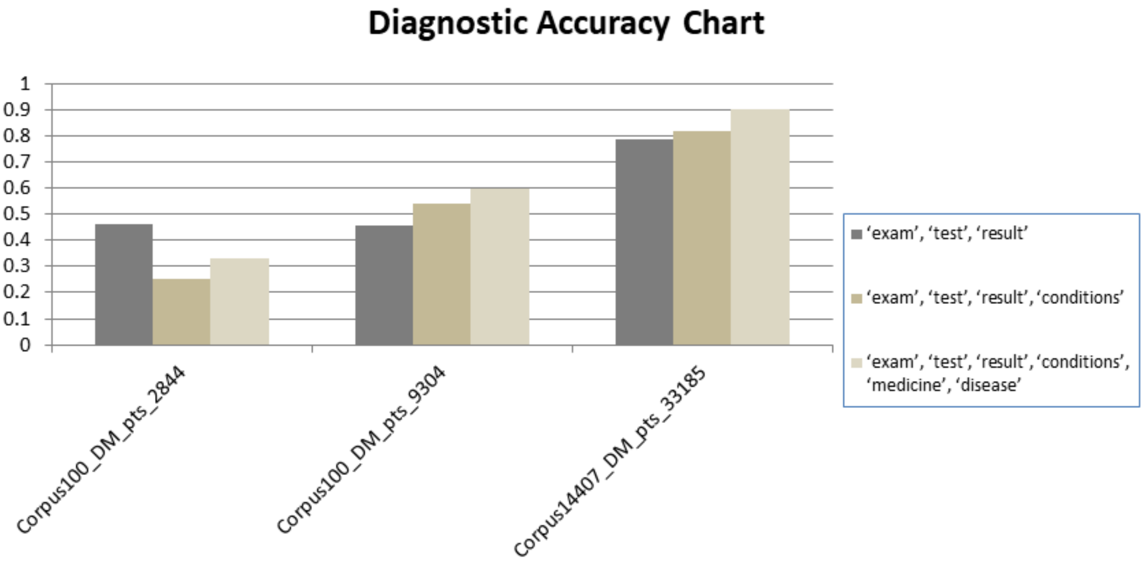


Figure 8. Sequential Model Accuracies achieved for the three corpora with hyper parameter setting and selection of features.

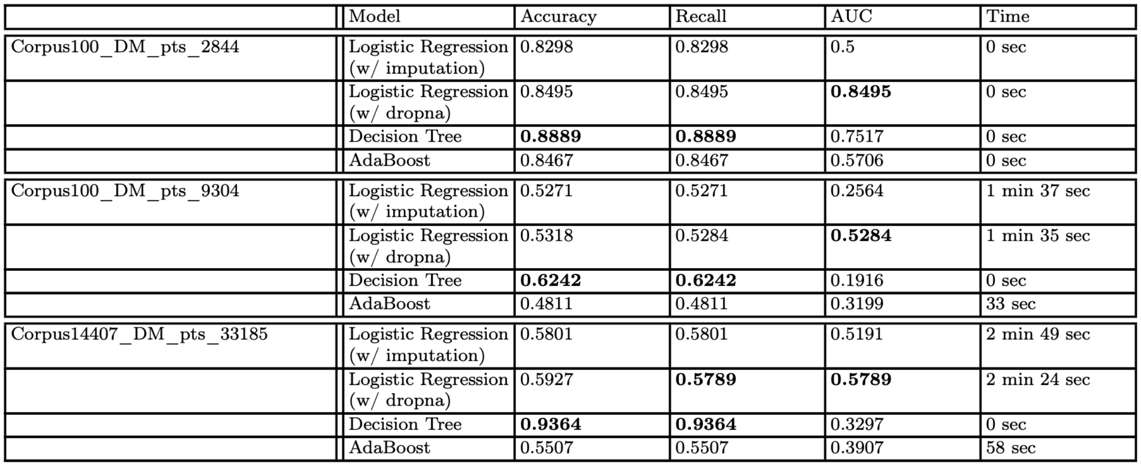
7. Evaluation of Analytics Performance

7.1. Evaluation and Validation Results for ML Diagnostic Algorithms

Accuracy results are stored for each input corpus and the selected classifier models. Results are compared based on the algorithmic performance and size of the corpus. The decision Tree algorithm in Table 6 is seen as outperforming but there is a significant decrease in validation accuracy where there are maximum numbers of diagnostic classes equal to 65 in 9304 instances of 100 patients. Maximum accuracy results are observed where there are larger numbers of classes equal to 32 but to balance it number of instances has also increased to 33185 from 14407 DM patients.

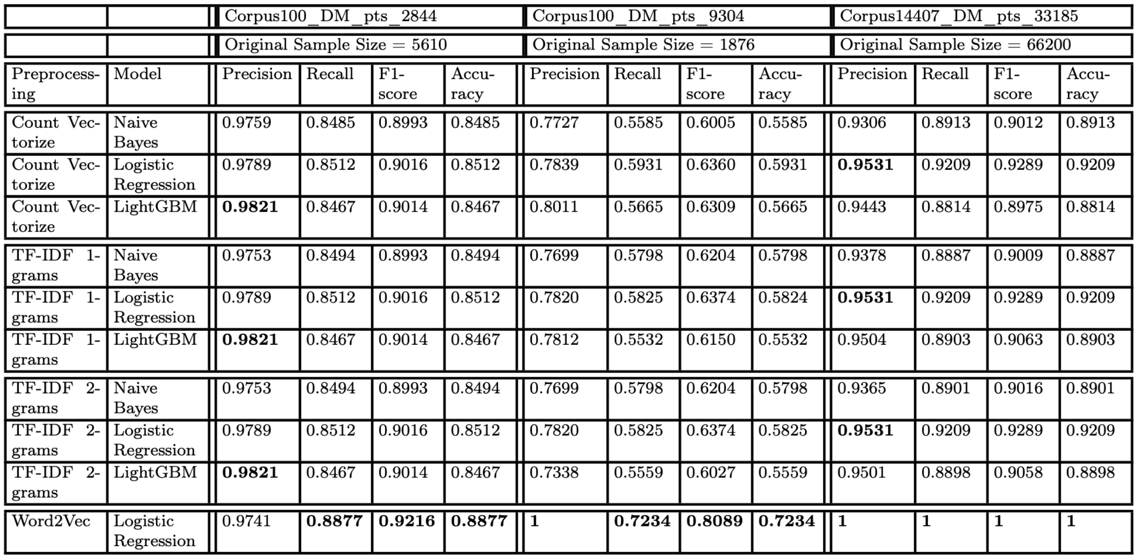
7.2. Evaluation and Validation Results for NLP embedded ML Diagnostic Algorithms

The resultant embedding is evaluated before as intrinsic or extrinsic [24]. Intrinsic evaluation of embedding for encoding similar/related contextual information is done using nearest neighbor search (NNS), clustering and similarity measures. Extrinsic evaluation is done by testing the model accuracy for input text for an expected output for name entity recognition (NER), medical text classification, medical concept normalization, etc. Known NLP methods are listed for clinical predictions; word2vec and stacked de-noising auto-encoders, for medical coding; Glove, fasttext and word2vec have been preferred before, for NER in the clinical domain; word2vec and fasttext were chosen, for patient de-identification; Glove or RNN encoder/decoder are used and for patient similarity word2vec. Word2vec is considered a popular technique for NER embedding therefore we chose it for experimentation on our corpora.



	Model	Accuracy	Recall	AUC	Time
Corpus100_DM_pts_2844	Logistic Regression (w/ imputation)	0.8298	0.8298	0.5	0 sec
	Logistic Regression (w/ dropna)	0.8495	0.8495	**0.8495**	0 sec
	Decision Tree	**0.8889**	**0.8889**	0.7517	0 sec
	AdaBoost	0.8467	0.8467	0.5706	0 sec
Corpus100_DM_pts_9304	Logistic Regression (w/ imputation)	0.5271	0.5271	0.2564	1 min 37 sec
	Logistic Regression (w/ dropna)	0.5318	0.5284	**0.5284**	1 min 35 sec
	Decision Tree	**0.6242**	**0.6242**	0.1916	0 sec
	AdaBoost	0.4811	0.4811	0.3199	33 sec
Corpus14407_DM_pts_33185	Logistic Regression (w/ imputation)	0.5801	0.5801	0.5191	2 min 49 sec
	Logistic Regression (w/ dropna)	0.5927	**0.5789**	**0.5789**	2 min 24 sec
	Decision Tree	**0.9364**	**0.9364**	0.3297	0 sec
	AdaBoost	0.5507	0.5507	0.3907	58 sec

Table 6. Logistic Regression, Decision Tree and AdaBoost performance are recorded with relative accuracies with time. Decision Tree is seen outperforming.



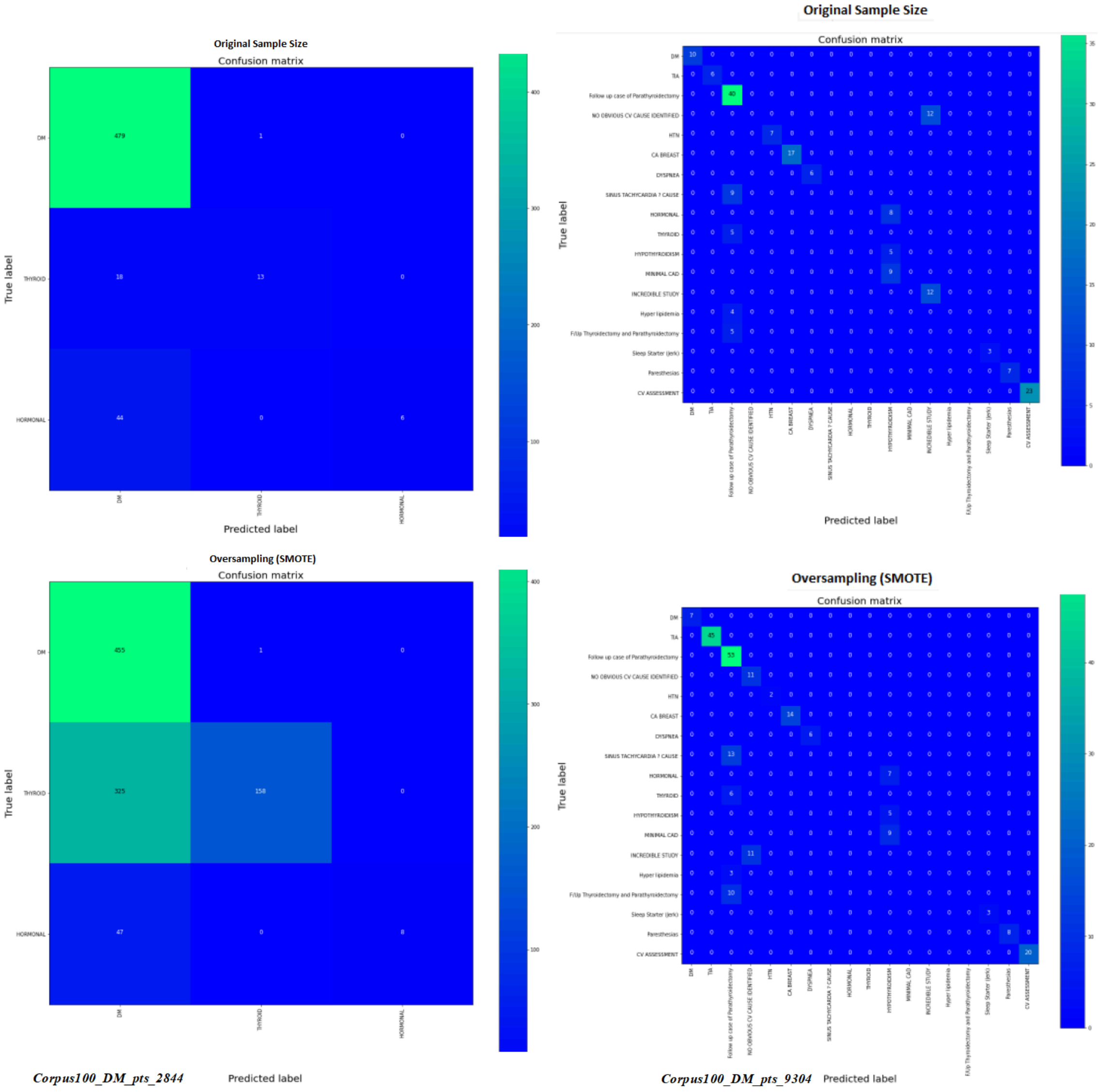
		Corpus100_DM_pts_2844				Corpus100_DM_pts_9304				Corpus14407_DM_pts_33185			
		Original Sample Size = 5610				Original Sample Size = 1876				Original Sample Size = 66200			
Preprocessing	Model	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy
Count Vectorize	Naive Bayes	0.9759	0.8485	0.8993	0.8485	0.7727	0.5585	0.6005	0.5585	0.9306	0.8913	0.9012	0.8913
Count Vectorize	Logistic Regression	0.9789	0.8512	0.9016	0.8512	0.7839	0.5931	0.6360	0.5931	**0.9531**	0.9209	0.9289	0.9209
Count Vectorize	LightGBM	**0.9821**	0.8467	0.9014	0.8467	0.8011	0.5665	0.6309	0.5665	0.9443	0.8814	0.8975	0.8814
TF-IDF 1-grams	Naive Bayes	0.9753	0.8494	0.8993	0.8494	0.7699	0.5798	0.6204	0.5798	0.9378	0.8887	0.9009	0.8887
TF-IDF 1-grams	Logistic Regression	0.9789	0.8512	0.9016	0.8512	0.7820	0.5825	0.6374	0.5824	**0.9531**	0.9209	0.9289	0.9209
TF-IDF 1-grams	LightGBM	**0.9821**	0.8467	0.9014	0.8467	0.7812	0.5532	0.6150	0.5532	0.9504	0.8903	0.9063	0.8903
TF-IDF 2-grams	Naive Bayes	0.9753	0.8494	0.8993	0.8494	0.7699	0.5798	0.6204	0.5798	0.9365	0.8901	0.9016	0.8901
TF-IDF 2-grams	Logistic Regression	0.9789	0.8512	0.9016	0.8512	0.7820	0.5825	0.6374	0.5825	**0.9531**	0.9209	0.9289	0.9209
TF-IDF 2-grams	LightGBM	**0.9821**	0.8467	0.9014	0.8467	0.7338	0.5559	0.6027	0.5559	0.9501	0.8898	0.9058	0.8898
Word2Vec	Logistic Regression	0.9741	**0.8877**	**0.9216**	**0.8877**	**1**	**0.7234**	**0.8089**	**0.7234**	**1**	**1**	**1**	**1**

Table 7. Base Accuracy results for different ML algorithms with embedded NLP techniques for original sized corpora.

1. Base Results on Original Corpora: In Table 7, we see that maximum accuracy is achieved with logistic regression with word embedding on Corpus14407_DM_pts_33185 where there are maximum instances of 33185 having 30 diagnostic classes. On smaller Corpus100_DM_pts_2844, the accuracy is 0.89 which is not bad. Accuracy for Corpus100_DM_pts_9304 having maximum classes and maximum mean sentence length falls to 0.7. We understand that large or multiple datasets would have a skewed class distribution that may affect the accuracy. We used undersampling and oversampling methods to balance the distribution of classes in corpora [80][81][82].
2. Undersampling using Naïve Approach: Undersampling is done with the assumption that any random sample taken from a majority class would balance the distribution of data while discarding the remaining. It is understood that the information that is lost is not significant for model training. This approach is called the naïve approach. We observed that overall accuracy decreases for all corpora but there is a significant decrease in accuracy in the smallest Corpus100_DM_pts_2844 having 2844 instances with three classes only. Therefore we can say that with a decrease in sample size, the accuracy deteriorates for the trained ML models.
3. Oversampling using SMOTE: The Synthetic Minority Oversampling Technique (SMOTE) duplicates the samples from the minority class to balance the distribution. The majority of classes remain untouched. The SMOTE oversampling technique is used with naïve Bayes, logistic regression and

light GBM in this diagnostic problem. The accuracy results with oversampling are also not very significant. Therefore, the original corpora prove best to train our classification models.

4. Results Analysis of Best Model: Logistic Regression gave the best accuracy results and confusion matrices (Figure 9) depict the comparison between its predictions on corpora with original sample size and oversampled sample size. It is seen that with oversampled corpora the rate of predicted classes increases.



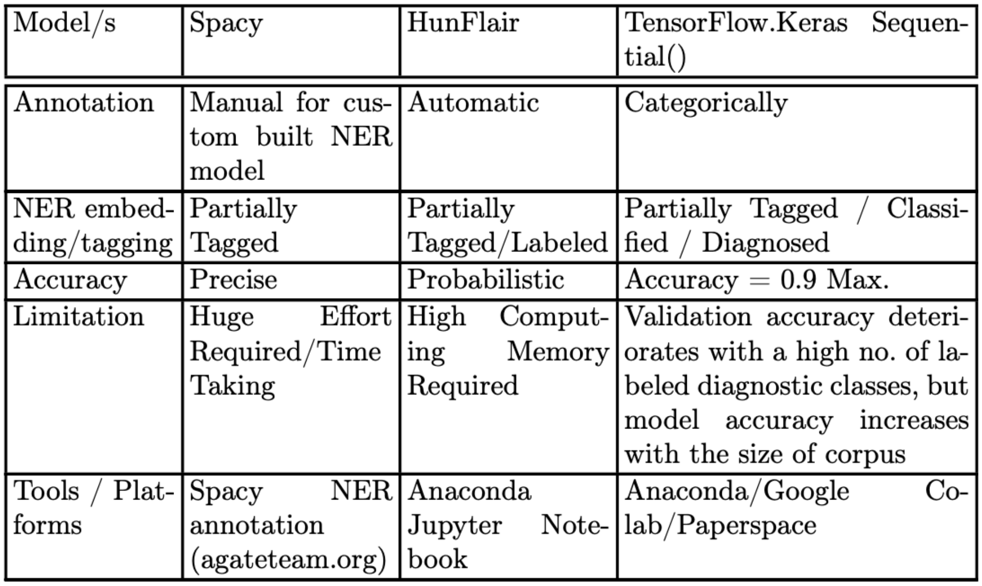
Corpus100_DM_pts_2844

Corpus100_DM_pts_9304

Figure 9. Resultant Confusion Matrix from analytics applied on two of the Corpora with original sample and oversampled SMOTE.

7.3. Evaluation for Sequential NER Embedding

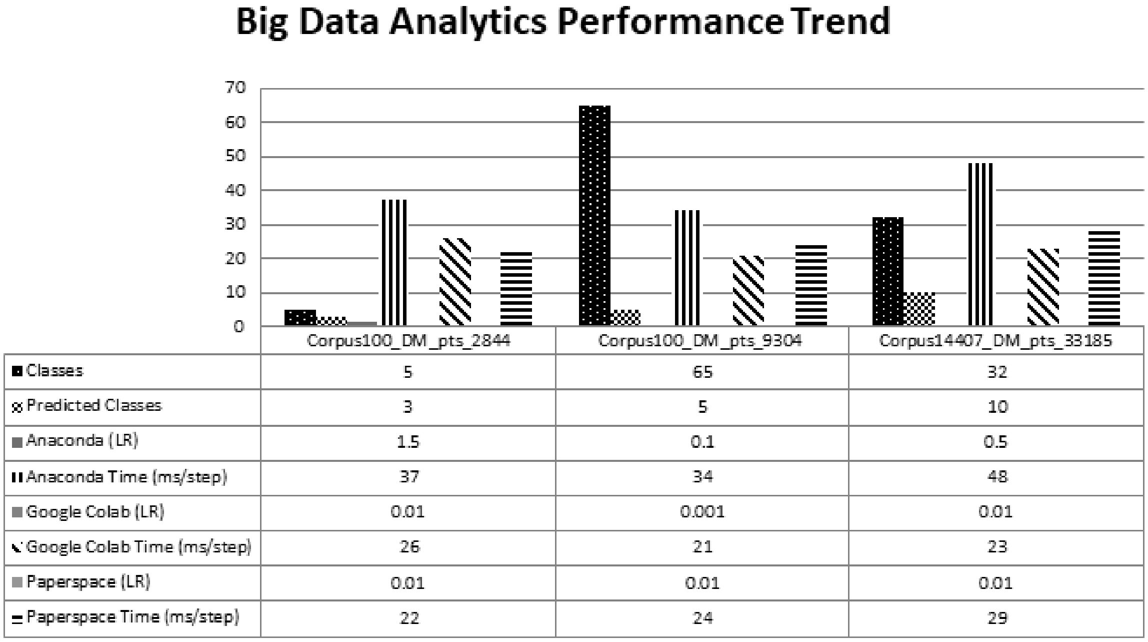
We implemented sequential NER embedding using three separate annotation tools namely spaCy [9], HunFlair [74], [75] and the Keras sequential model [76]. These tools were compared in terms of the support they provide for the annotation process and the amount of tagging required, the accuracy achieved and their limitations. These details are shown in Table 8. As can be seen in the first column manual annotation was conducted using spaCy and a custom NER model was built and trained to tag the phenotypes 'condition', 'disease' and 'medicine'. Manual annotation is time-consuming requiring increased man-hours and effort to reach maximum accuracy. The next column shows that the pre-trained HunFlair model in flair on Anaconda Jupyter Notebook was used to annotate free text in clinical notes and practitioner comments present in the tabular datasets that only labeled diseases with probabilistic accuracy. Tagging the complete dataset needed high computational memory. Finally, the third column is used to depict the performance of our proposed DNN sequential model for NER embedding.



Model/s	Spacy	HunFlair	TensorFlow.Keras Sequential()
Annotation	Manual for custom built NER model	Automatic	Categorically
NER embedding/tagging	Partially Tagged	Partially Tagged/Labeled	Partially Tagged / Classified / Diagnosed
Accuracy	Precise	Probabilistic	Accuracy = 0.9 Max.
Limitation	Huge Effort Required/Time Taking	High Computing Memory Required	Validation accuracy deteriorates with a high no. of labeled diagnostic classes, but model accuracy increases with the size of corpus
Tools / Platforms	Spacy NER annotation (agateteam.org)	Anaconda Jupyter Notebook	Anaconda/Google Colab/Paperspace

Table 8. Comparison of three Sequential NER Embedding Techniques experimented with in this paper.

The phenotype fields contained in DM_Comorbid_EHR_ICD10 corpora in .csv format were used to annotate the corresponding 'condition', 'disease', 'medicine', 'Examination', 'Test', 'Result' in NER format associated with the diagnoses of DM and its comorbidities. The sequential DNN model was run on three cloud platforms; Anaconda, Google Colab and Paperspace. It is notable in Figure 10 that Google Colab speeds up the processing with the increase in corpus size and the model learns fast. This model gave maximum accuracies of 0.4615, 0.6, and 0.9 where validation accuracies were 1, 1 and 0.8462 with respect to the size of three EHR corpora and features taken as input and diagnosed several endocrine diseases (Figure 8 in section 6). A single patient in a 100-patient corpus (named Corpus100_DM_pts_2844 having 2844 instances with CSV columns) is classified for multiple classes of diagnoses having DM and coexisting Hormonal and Thyroid diseases referred to as comorbidities. In Corpus100_DM_pts_9304 of 65 diagnosed classes, it classified eight endocrine diseases. In Corpus14407_DM_pts_33185 having 32 classes of diagnosed endocrine diseases it successfully diagnosed 17 classes with a learning rate set to 0.05 in Google Colab (Figure 11).



	Corpus100_DM_pts_2844	Corpus100_DM_pts_9304	Corpus14407_DM_pts_33185
■ Classes	5	65	32
▩ Predicted Classes	3	5	10
■ Anaconda (LR)	1.5	0.1	0.5
‖ Anaconda Time (ms/step)	37	34	48
■ Google Colab (LR)	0.01	0.001	0.01
▧ Google Colab Time (ms/step)	26	21	23
■ Paperspace (LR)	0.01	0.01	0.01
☰ Paperspace Time (ms/step)	22	24	29

Figure 10. Performance Comparison of three cloud platforms on different size Big EHR data. Anaconda, Google Colab and Gradient Paperspace Learning Rate (LR) with Time per iteration are shown for predicted classes in each dataset.

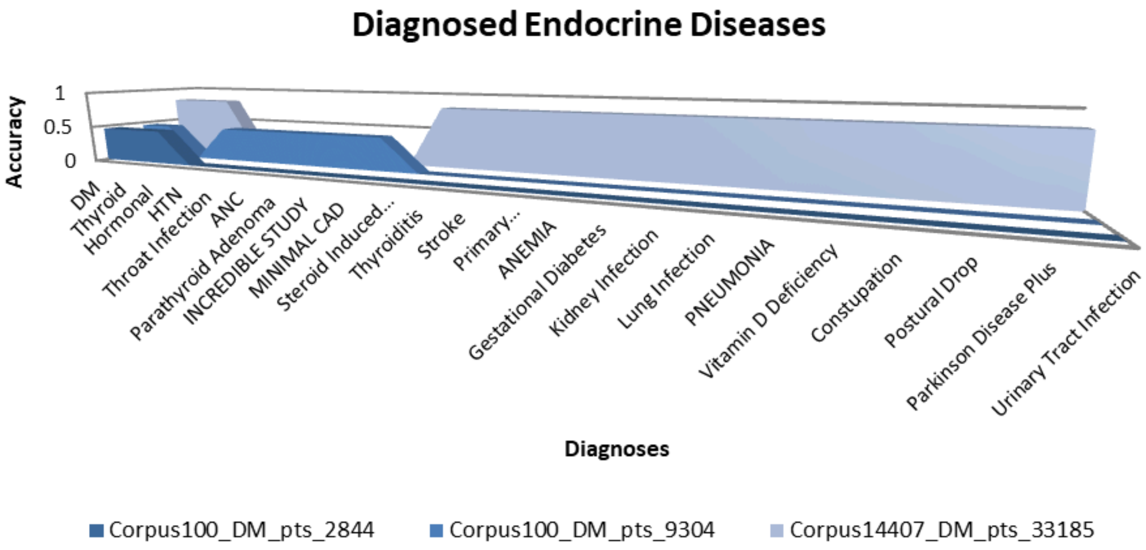


Figure 11. Multi-class classification for the patients' medical profiles led to diagnoses of primary disease DM and its comorbidities.

8. Conclusion and Future Work

This paper has initiated the extraction, preparation and maintenance of our unified knowledge base in the form of 'DM_Comorbid_EHR_ICD10'. We proposed the high-level architecture for a diagnostic framework that incorporates advanced ML integrated with NER embedding tools and techniques to learn semantics on our corpora as illustrated in Figure 1. The sequential NER embedding on these corpora let us deduce intelligent semantics to diagnose DM patient and their comorbidity diseases. The corpora would grow to maintain data for diagnoses of other diseases in the future and fill all the entities mentioned in Figure 2.

In this paper, we proposed the mechanism (Figure 1) in section 3 for NER tagging of unified medical corpora for standardized medical context learning. We specifically applied Spacy for manual annotation of a single patient profile extracted from Corpus100_DM_pts_2844. HunFlair's pre-trained NER model was reused and tested on raw medical data that only tagged diseases. Later we trained a custom NER model to extract attributes like 'condition', 'disease' and 'medicine' from free text fields; 'Note' and 'PC'. We trained our proposed DNN model based on Bi-LSTM with dense layers on TensorFlow.Keras with Corpus100_DM_pts_2844, Corpus100_DM_pts_9304 and Corpus14407_DM_pts_33185 taking key diagnostic features as inputs. The diagnosis problem was solved for DM as well as other comorbidity diseases in patients (Figure 11) using sequence embedding or tagging. The final features selected were 'exam, 'test', ‘result’, ‘condition’, ‘disease’ and ‘medicine’. It was observed that running the Sequential DNN model on Corpus100_DM_pts_2844, Corpus100_DM_pts_9304 and Corpus14407_DM_pts_33185 gave us the validation accuracies of 1, 1 and 0.8462 respectively. These accuracies signal the good quality of our corpora having real-time datasets. The differences in the validation accuracy results relate to the size of each corpus and the features that were the input. Model accuracy increased with added features and an increase in corpus size (Figure 8).

In this paper, where we explored some NER tagging schemes, we also identified some high-performance tools and techniques that would fasten the process of NER tagging and embedding for intelligent medical semantics in the future. In the future, we need to do more experimentation using other mechanisms like Auto ML, BERT or ELMo to solve multi-label and multiclass problems for diagnostics. These domain-specific medical corpora are structured on HL7 FHIR schema with labeled fields having untagged/unlabeled text values like clinical notes or practitioner comments. We extracted the selected feature set from raw corpora named ‘DM_Comorbid_EHR_ICD10’ as stated in representation learning. These extracted raw corpora would enable us to tag the medical vocabulary used for active learning in the future. Active deep learning would train these unified medical corpora using semi-supervised or reinforced learning techniques. Advanced learning techniques like; active deep learning through representation learning [36] have gained our interest. Active learning is semi-supervised where initial input is labeled dataset to train the model. Final evaluation would be done on unlabeled or undiagnosed classes (that were pruned in this paper) to get an efficient learning model after multiple iterations.

Acknowledgements

This research is supported by Shifa International Hospital, Pakistan. The data contributed by 14507 endocrine patients for diagnosis of Diabetes and its comorbidities holds a lot of worth to come up with the proposed semantically intelligent DNN Model.

Declarations

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Authors' Contributions

This is an original research that emerged with the collaborated efforts of the PhD(CS) Scholar and her professors.

- Sarah Shafqat conceptualized this study, designed the unified clinical data model, extracted, processed, and prepared the corpora for exhaustive experimentation and analysis.
- Prof Dr Zahid reviewed, proofread, and structured the paper with the scholar. He gave suggestions where necessary and assisted the scholar in drawing Figure 1 of high-level diagnostic architecture that forms the basis of her research.
- Prof Dr Qaisar managed the collaboration from the platform of the International Islamic University Islamabad (IIUI), Pakistan.
- Prof. Dr Hafiz Farooq Ahmad directed and led this research with the approval of the International Islamic University Islamabad (IIUI), Pakistan.

Data Confidentiality Statement

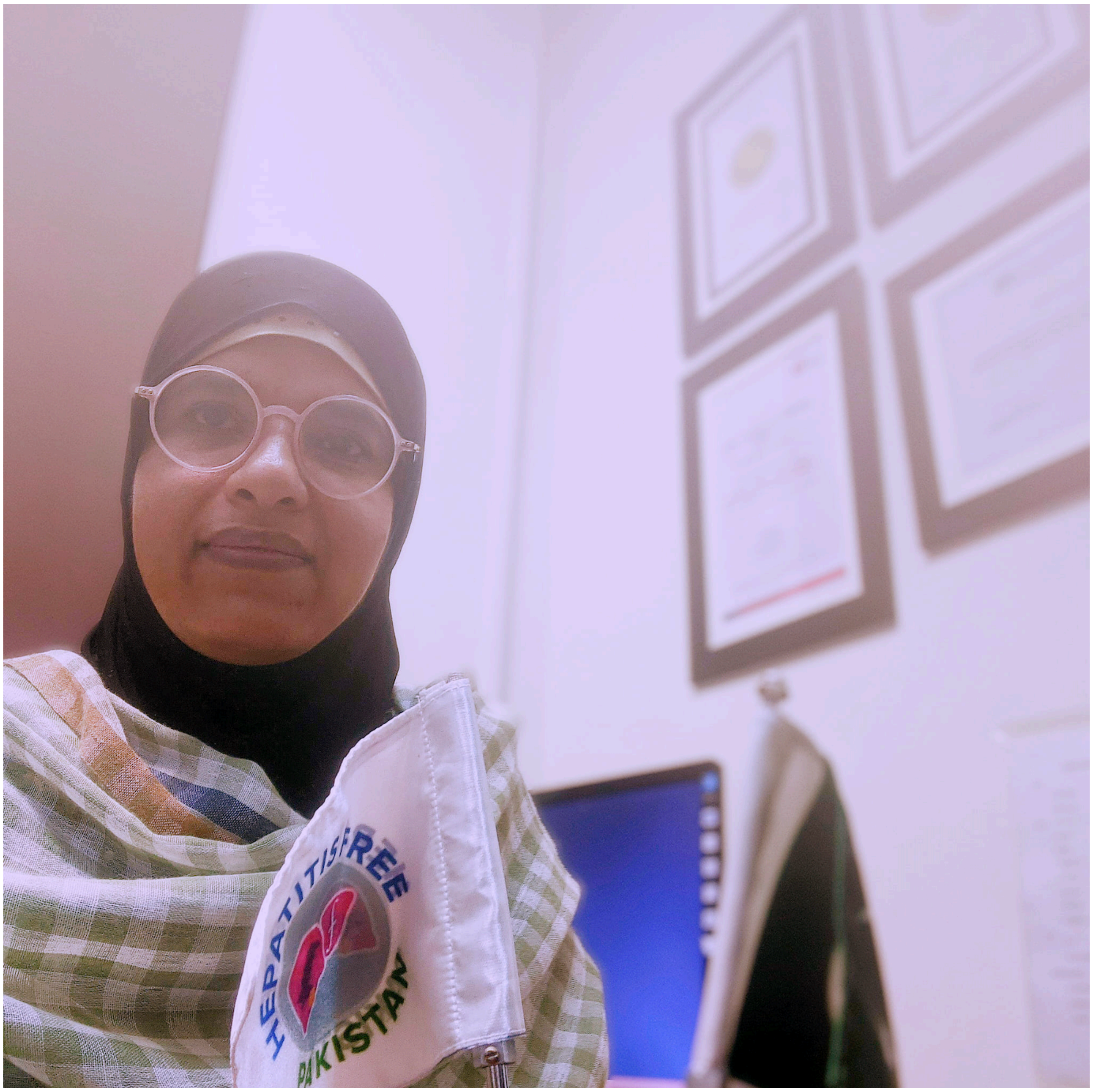
Due to the sensitive nature of the real-time patients' EHR data analyzed in this study, hospital and other executive stakeholders are assured that it would remain confidential and would not be shared.

Availability of Data

Sarah Shafqat, extracted endocrine EHR big data under Grant (IRB# 996-271-2018) approved by the hospital's MIS department. This dataset was further processed during the research for analysis and would become a unified knowledge base. Scholar plans to apply for a patent for the analytical models and the medical corpora for further research and collaboration. The knowledge domain experts feedback and supervision has to be there to mitigate chances of biasness and ensuring the ethical concerns in the clinical domain.

About the Authors

Sarah Shafqat. She is a profound research scholar in the field of Information Technology having sound knowledge of Business Processes. She received a Bachelor of Business (Information Technology) from Curtin University, Australia and continued her career with a Master's degree in Business Administration with majors in Human Resource Management and post-graduation in Software Engineering & Computer



Science. She is fully aware of the underlying risks in organizations residing over sensitive data like healthcare and individuals commuting over the cloud for in-time services. In spite of all the risks that are associated with the cloud she fully understands its importance and potential growth in the services industry. Therefore, the implementation of security in cloud infrastructure is her keen interest. Currently, she has some highly resourceful publications in renowned international journals. Her research area revolves around cloud computing, big data analytics and healthcare informatics.

doi.org/10.32388/HPAUYJ

Other References

- T. Ruas, “Semantic Feature Extraction Using Multi-Sense Embeddings and Lexical Chains,” 2019, Accessed: Aug. 11, 2019. [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/149647.

References

1. ^*L. Jiang, X. Sun, F. Mercaldo, and A. Santone, “DECAB-LSTM: Deep Contextualized Attentional Bidirectional LSTM for cancer hallmark classification,” Knowledge-Based Syst., vol. 210, Dec. 2020, doi: 10.1016/j.knosys.2020.106486.*
2. ^*B. E. Bejnordi et al., “Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer,” JAMA - J. Am. Med. Assoc., vol. 318, no. 22, pp. 2199–2210, Dec. 2017, doi: 10.1001/jama.2017.14585.*
3. a, b*A. V. Annapragada, M. M. K. Donaruma, A. V. Annapragada, and Z. A. Starosolski, “A natural language processing and deep learning approach to identify child abuse from pediatric electronic medical records,” PLoS One, vol. 16, no. 2 February, Feb. 2021, doi: 10.1371/JOURNAL.PONE.0247404.*
4. a, b*J. M. Brown et al., “Automated diagnosis of plus disease in retinopathy of prematurity using deep convolutional neural networks,” in JAMA Ophthalmology, 2018, vol. 136, no. 7, pp. 803–810, doi: 10.1001/jamaophthalmol.2018.1934.*
5. ^*S. Yadav, A. Ekbal, S. Saha, and P. Bhattacharyya, “Entity extraction in biomedical corpora: An approach to evaluate embedding features with PSO based feature selection,” in 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference, 2017, vol. 1, pp. 1159–1170, doi: 10.18653/v1/e17-1109.*
6. ^*K. Chehab, A. Kalboussi, and A. H. Kacem, “Study of healthcare annotation systems,” Int. J. E-Health Med. Commun., vol. 12, no. 3, pp. 74–89, 2021, doi: 10.4018/IJEHMC.20210501.oa5.*
7. a, b, c*Y. Si, J. Wang, H. Xu, and K. Roberts, “Enhancing clinical concept extraction with contextual embeddings,” J. Am. Med. Informatics Assoc., vol. 26, no. 11, pp. 1297–1304, Feb. 2019, doi: 10.1093/jamia/ocz096.*
8. ^*R. I. Doğan, R. Leaman, and Z. Lu, “NCBI disease corpus: a resource for disease name recognition and concept normalization,” J. Biomed. Inform., vol. 47, pp. 1–10, 2014, doi: 10.1016/J.JBI.2013.12.006.*
9. a, b, c*N. Sanprasit, K. Jampachaisri, T. Titijaroonroj, and K. Kesorn, “Intelligent approach to automated star-schema construction using a knowledge base,” Expert Syst. Appl., vol. 182, Nov. 2021, doi: 10.1016/j.eswa.2021.115226.*

10. a, b, c *P. Patel, D. Davey, V. Panchal, and P. Pathak, "Annotation of a large clinical entity corpus," in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, 2020, pp. 2033–2042, doi: 10.18653/v1/d18-1228.*

11. a, b, c *J. A. Fries et al., "Ontology-driven weak supervision for clinical entity classification in electronic health records," Nat. Commun. 2021 121, vol. 12, no. 1, pp. 1–11, Apr. 2021, doi: 10.1038/s41467-021-22328-4.*

12. ^ *C. Weng, N. H. Shah, and G. Hripcsak, "Deep phenotyping: Embracing complexity and temporality—Towards scalability, portability, and interoperability," Journal of Biomedical Informatics, vol. 105. Academic Press Inc., May 01, 2020, doi: 10.1016/j.jbi.2020.103433.*

13. a, b *J. M. Banda, M. Seneviratne, T. Hernandez-Boussard, and N. H. Shah, "Advances in Electronic Phenotyping: From Rule-Based Definitions to Machine Learning Models," Annu. Rev. Biomed. Data Sci., vol. 1, no. 1, pp. 53–68, Jul. 2018, doi: 10.1146/annurev-biodatasci-080917-013315.*

14. ^ *S. Shafqat, A. Abbasi, T. Amjad, and H. F. Ahmad, "Smarthealth simulation representing a hybrid architecture over cloud integrated with IoT: A modular approach," in Advances in Intelligent Systems and Computing, 2019, vol. 887, pp. 445–460, doi: 10.1007/978-3-030-03405-4_31.*

15. ^ *Y. Huang, P. McCullagh, N. Black, and R. Harper, "Feature selection and classification model construction on Type 2 diabetic patient's data," in Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), 2004, vol. 3275, pp. 153–162, doi: 10.1007/978-3-540-30185-1_17.*

16. ^ *R. J. Carroll, A. E. Eyler, and J. C. Denny, "Naïve Electronic Health Record phenotype identification for Rheumatoid arthritis.," AMIA Annu. Symp. Proc., vol. 2011, pp. 189–196, 2011, Accessed: Sep. 10, 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/pmc3243261/.*

17. ^ *J. S. Sartakhti, M. H. Zangooei, and K. Mozafari, "Hepatitis disease diagnosis using a novel hybrid method based on support vector machine and simulated annealing (SVM-SA)," Comput. Methods Programs Biomed., vol. 108, no. 2, pp. 570–579, 2012, doi: 10.1016/j.cmpb.2011.08.003.*

18. ^ *M. S. R. Nalluri, K. Kannan, M. Manisha, and D. S. Roy, "Hybrid Disease Diagnosis Using Multiobjective Optimization with Evolutionary Parameter Optimization," J. Healthc. Eng., vol. 2017, 2017, doi: 10.1155/2017/5907264.*

19. ^ *A. H. Osman and H. M. Aljahdali, "Diabetes Disease Diagnosis Method based on Feature Extraction using K-SVM," Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 1, pp. 236–244, 2017.*

20. ^ *Y. Halpern, S. Horng, Y. Choi, and D. Sontag, "Electronic medical record phenotyping using the anchor and learn framework," J. Am. Med. Informatics Assoc., vol. 23, no. 4, pp. 731–740, 2016, doi: 10.1093/jamia/ocw011.*

21. ^*V. Agarwal et al., “Learning statistical models of phenotypes using noisy labeled training data,” J. Am. Med. Informatics Assoc., vol. 23, no. 6, pp. 1166–1173, 2016, doi: 10.1093/jamia/ocw028.*

22. ^*G. Hripcsak and D. J. Albers, “Next-generation phenotyping of electronic health records,” J. Am. Med. Informatics Assoc., vol. 20, no. 1, pp. 117–121, 2013, doi: 10.1136/amiajnl-2012-001145.*

23. ^*J. Henderson, R. Bridges, J. C. Ho, B. C. Wallace, and J. Ghosh, “PheKnow-Cloud: A Tool for Evaluating High-Throughput Phenotype Candidates using Online Medical Literature.,” AMIA Jt. Summits Transl. Sci. proceedings. AMIA Jt. Summits Transl. Sci., vol. 2017, pp. 149–157, 2017, Accessed: Sep. 10, 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5543339/.*

24. a, b, c, d, e, f, g*K. S. Kalyan and S. Sangeetha, “SECNLP: A survey of embeddings in clinical natural language processing,” Journal of Biomedical Informatics, vol. 101. Mar. 03, 2020, doi: 10.1016/j.jbi.2019.103323.*

25. a, b*F. K. Khattak, S. Jeblee, C. Pou-Prom, M. Abdalla, C. Meaney, and F. Rudzicz, “A survey of word embeddings for clinical text,” J. Biomed. Inform., vol. 100, p. 100057, Jan. 2019, doi: 10.1016/J.YJBINX.2019.100057.*

26. a, b*J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global vectors for word representation,” in EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 2014, pp. 1532–1543, doi: 10.3115/v1/d14-1162.*

27. ^*M. Q. Stearns, C. Price, K. A. Spackman, and A. Y. Wang, “SNOMED clinical terms: overview of the development process and project status.,” Proc. AMIA Symp., p. 662, 2001, Accessed: Sep. 12, 2021. [Online]. Available: /pmc/articles/PMC2243297/?report=abstract.*

28. a, b, c, d, e, f, g*L. Tamine and L. Goeuriot, “Semantic Information Retrieval On Medical Texts: Research Challenges, Survey and Open Issues,” ACM Comput. Surv., 2021, doi: 10.1145/nnnnnnn.nnnnnnnï.*

29. a, b*H. Wu et al., “Contextualised concept embedding for efficiently adapting natural language processing models for phenotype identification,” arxiv.org, 2019, Accessed: Aug. 17, 2019. [Online]. Available: https://arxiv.org/abs/1903.03995.*

30. ^*Y. Park, J. Lee, H. Moon, Y. S. Choi, and M. Rho, “Discovering microbe-disease associations from the literature using a hierarchical long short-term memory network and an ensemble parser model,” Sci. Reports 2021 111, vol. 11, no. 1, pp. 1–12, Feb. 2021, doi: 10.1038/s41598-021-83966-8.*

31. ^*S. Yan and K. C. Wong, “Context awareness and embedding for biomedical event extraction,” Bioinformatics, vol. 36, no. 2, pp. 637–643, 2020, doi: 10.1093/bioinformatics/btz607.*

32. ^*N. Tomašev et al., “Use of deep learning to develop continuous-risk models for adverse event prediction from electronic health records,” Nat. Protoc. 2021 166, vol. 16, no. 6, pp. 2765–2787, May 2021, doi: 10.1038/s41596-021-00513-5.*

33. ^*J. J. Thiagarajan, D. Rajan, S. Katoch, and A. Spanias, "DDxNet: a deep learning model for automatic interpretation of electronic health records, electrocardiograms and electroencephalograms," Sci. Reports 2020 101, vol. 10, no. 1, pp. 1–11, Oct. 2020, doi: 10.1038/s41598-020-73126-9.*

34. ^*A. Lara-Clares and A. Garcia-Serrano, "LSI2 UNED at eHealth-KD Challenge 2019," 2019, Accessed: Sep. 12, 2021. [Online]. Available: http://ceur-ws.org/Vol-2421/eHealth-KD_paper_6.pdf.*

35. ^*S. Rashidian et al., "Detecting miscoded diabetes diagnosis codes in electronic health records for quality improvement: Temporal deep learning approach," JMIR Med. Informatics, vol. 8, no. 12, 2020, doi: 10.2196/22649.*

36. a, b*P. K. D. Pramanik, S. Pal, M. Mukhopadhyay, and S. P. Singh, "Big Data classification: techniques and tools," in Applications of Big Data in Healthcare, 2021, pp. 1–43.*

37. a, b*S. Shafqat et al., "Leveraging Deep Learning for Designing Healthcare Analytics Heuristic for Diagnostics," Neural Process. Lett., pp. 1–27, Feb. 2021, doi: 10.1007/s11063-021-10425-w.*

38. ^*Shafqat S, Anwar Z, Rasool RU, Javaid Q, Ahmad HF. Rules Extraction, Diagnoses and Prognosis of Diabetes and its Comorbidities using Deep Learning Analytics with Semantics on Big Data. Qeios; 2023. DOI: 10.32388/67kz7s.2.*

39. a, b*H. Alachram, "Knowledge Integration and Representation for Biomedical Analysis," 2021, Accessed: Aug. 27, 2021. [Online]. Available: https://ediss.uni-goettingen.de/bitstream/handle/21.11130/00-1735-0000-0005-158D-5/Thesis_Halima Alachram.pdf?sequence=1.*

40. ^*S. Shafqat, Z. Anwar, Q. Javaid, and H. F. Ahmad, "A Unified Deep Learning Diagnostic Architecture for Big Data Healthcare Analytics," 2023 IEEE 15th Int. Symp. Auton. Decentralized Syst., pp. 1–8, Mar. 2023, doi: 10.1109/ISADS56919.2023.10092137.*

41. ^*R. Y. Lee et al., "Identifying Goals of Care Conversations in the Electronic Health Record Using Natural Language Processing and Machine Learning," J. Pain Symptom Manage., vol. 61, no. 1, pp. 136-142.e2, Jan. 2021, doi: 10.1016/j.jpainsymman.2020.08.024.*

42. ^*A. Dagliati et al., "Machine Learning Methods to Predict Diabetes Complications," J. Diabetes Sci. Technol., vol. 12, no. 2, pp. 295–302, 2018, doi: 10.1177/1932296817706375.*

43. ^*J. Li, B. Jiang, and J. P. Fine, "Multicategory reclassification statistics for assessing improvements in diagnostic accuracy," Biostatistics, vol. 14, no. 2, pp. 382–394, 2013, doi: 10.1093/biostatistics/kxs047.*

44. ^*H. F. da Cruz, "Standardizing clinical predictive modeling: standardizing development, validation, and interpretation of clinical prediction models," 2021, doi: 10.25932/publishup-51496.*

45. a, bH. Ben Braiek and F. Khomh, “On testing machine learning programs,” J. Syst. Softw., vol. 164, Jun. 2020, doi: 10.1016/j.jss.2020.110542.

46. a, bN. Kallus, X. Mao, and A. Zhou, “Assessing Algorithmic Fairness with Unobserved Protected Class Using Data Combination,” Manage. Sci., Jun. 2021, doi: 10.1287/mnsc.2020.3850.

47. ^K. Naseer Qureshi, S. Din, G. Jeon, and F. Piccialli, “An accurate and dynamic predictive model for a smart M-Health system using machine learning,” Inf. Sci. (Ny)., vol. 538, pp. 486–502, Oct. 2020, doi: 10.1016/j.ins.2020.06.025.

48. a, bM. Shuja, S. Mittal, and M. Zaman, “Diabetes Mellitus and Data Mining Techniques A survey,” Int. J. Comput. Sci. Eng., vol. 7, no. 1, pp. 858–861, 2019, doi: 10.26438/ijcse/v7i1.858861.

49. ^P. Doupe, J. Faghmous, and S. Basu, “Machine Learning for Health Services Researchers,” Value Heal., vol. 22, no. 7, pp. 808–815, Jul. 2019, doi: 10.1016/j.jval.2019.02.012.

50. ^A. Talaei-Khoei, M. Tavana, and J. M. Wilson, “A predictive analytics framework for identifying patients at risk of developing multiple medical complications caused by chronic diseases,” Elsevier, 2019, doi: 10.1016/j.artmed.2019.101750.

51. ^M. Fatima and M. Pasha, “Survey of Machine Learning Algorithms for Disease Diagnostic,” J. Intell. Learn. Syst. Appl., vol. 09, no. 01, pp. 1–16, 2017, doi: 10.4236/jilsa.2017.91001.

52. ^M. I. Razzak, M. Imran, and G. Xu, “Big data analytics for preventive medicine,” Neural Comput. Appl., vol. 32, no. 9, pp. 4417–4451, May 2020, doi: 10.1007/s00521-019-04095-y.

53. ^J. Zhang, Y. Li, W. Xiao, and Z. Zhang, “Non-iterative and Fast Deep Learning: Multilayer Extreme Learning Machines,” J. Franklin Inst., vol. 357, no. 13, pp. 8925–8955, Sep. 2020, doi: 10.1016/j.jfranklin.2020.04.033.

54. ^L. Yang and A. Shami, “On hyperparameter optimization of machine learning algorithms: Theory and practice,” Neurocomputing, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.

55. a, b, cP. Goyal, S. Pandey, and K. Jain, Deep Learning for Natural Language Processing. 2018.

56. ^A. P. Ruiz, M. Flynn, J. Large, M. Middlehurst, and A. Bagnall, “The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances,” Data Min. Knowl. Discov., vol. 35, no. 2, pp. 401–449, Mar. 2021, doi: 10.1007/s10618-020-00727-3.

57. ^G. Desagulier, “Can word vectors help corpus linguists?,” Stud. Neophilol., vol. 91, no. 2, pp. 219–240, 2019, doi: 10.1080/00393274.2019.1616220.

58. a, bG. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi, “Integrating and evaluating neural word embeddings in information retrieval,” in ACM International Conference Proceeding Series, 2015, vol. 08-09-Dec-, doi: 10.1145/2838931.2838936.

59. ^S. Estevez-Velarde, Y. Gutiérrez, Y. Almeida-Cruz, and A. Montoyo, "General-purpose hierarchical optimisation of machine learning pipelines with grammatical evolution," Inf. Sci. (Ny)., vol. 543, pp. 58–71, Jan. 2021, doi: 10.1016/j.ins.2020.07.035.

60. ^M. A. Kader, A. P. Boedihardjo, S. M. Naim, and M. S. Hossain, "Contextual Embedding for Distributed Representations of Entities in a Text Corpus," jmlr.org, vol. XX, pp. 1–16, 2016, Accessed: Aug. 11, 2019. [Online]. Available: http://www.jmlr.org/proceedings/papers/v53/kader16.pdf.

61. ^H. El Boukkouri, O. Ferret, T. Lavergne, and P. Zweigenbaum, "Embedding strategies for specialized domains: Application to clinical entity recognition," in ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop, 2019, pp. 295–301, doi: 10.18653/v1/p19-2041.

62. ^T. Long, R. Lowe, J. C. K. Cheung, and D. Precup, "Leveraging lexical resources for learning entity embeddings in multi-relational data," in 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Short Papers, May 2016, pp. 112–117, doi: 10.18653/v1/p16-2019.

63. ^E. K. Y. Yapp, X. Li, W. F. Lu, and P. S. Tan, "Comparison of base classifiers for multi-label learning," Neurocomputing, vol. 394, pp. 51–60, Jun. 2020, doi: 10.1016/j.neucom.2020.01.102.

64. ^R. Manicka Chezian and C. Kanakalakshmi, "Performance Evaluation of Machine Learning Techniques for Text Classification," 2015. Accessed: Dec. 03, 2020.

65. ^S. Sa'di, A. Maleki, R. Hashemi, Z. Panbechi, and K. Chalabi, "Comparison of Data Mining Algorithms in the Diagnosis of Type Ii Diabetes," Int. J. Comput. Sci. Appl., vol. 5, no. 5, pp. 1–12, 2015, doi: 10.5121/ijcsa.2015.5501.

66. a, bJ. Wu and Y. Zhao, "Machine learning technology in the application of genome analysis: A systematic review," Gene, vol. 705. Elsevier B.V., pp. 149–156, Jul. 15, 2019, doi: 10.1016/j.gene.2019.04.062.

67. ^A. Palvanov and Y. I. Cho, "Comparisons of deep learning algorithms for MNIST in real-time environment," Int. J. Fuzzy Log. Intell. Syst., vol. 18, no. 2, pp. 126–134, 2018, doi: 10.5391/IJFIS.2018.18.2.126.

68. ^K. M. Kuo, P. Talley, Y. H. Kao, and C. H. Huang, "A multi-class classification model for supporting the diagnosis of type II diabetes mellitus," PeerJ, vol. 8, 2020, doi: 10.7717/peerj.9920.

69. ^S. Shafqat, S. Kishwer, R. U. Rasool, J. Qadir, T. Amjad, and H. F. Ahmad, "Big data analytics enhanced healthcare systems: a review," J. Supercomput., 2018, doi: 10.1007/s11227-017-2222-4.

70. ^N. Dunbray, R. Rane, S. Nimje, J. Katade, and S. Mavale, "A Novel Prediction Model for Diabetes Detection Using Gridsearch and A Voting Classifier between Lightgbm and KNN," 2021, doi: 10.1109/GCAT52182.2021.9587551.

71. ^D. D. Rufo, T. G. Debelee, A. Ibenthal, and W. G. Negera, "Diagnosis of diabetes mellitus using gradient boosting machine (Lightgbm)," Diagnostics, vol. 11, no. 9, 2021, doi: 10.3390/diagnostics11091714.

72. ^F. Hou, Z. X. Cheng, L. Y. Kang, and W. Zheng, "Prediction of Gestational Diabetes Based on LightGBM," in PervasiveHealth: Pervasive Computing Technologies for Healthcare, Oct. 2020, pp. 161–165, doi: 10.1145/3433996.3434025.

73. ^P. Xie et al., "An explainable machine learning model for predicting in-hospital amputation rate of patients with diabetic foot ulcer," Int. Wound J., 2021, doi: 10.1111/iwj.13691.

74. a, b, cA. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, "FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP," pp. 54–59, Accessed: Sep. 07, 2021. [Online]. Available: https://github.com/zalandoresearch/flair.

75. a, b, cL. Weber, M. Sänger, J. Münchmeyer, ... M. H. preprint arXiv, and undefined 2020, "HunFlair: an easy-to-use tool for state-of-the-art biomedical named entity recognition," arxiv.org, Accessed: Sep. 07, 2021. [Online]. Available: https://arxiv.org/abs/2008.07347.

76. a, b, cAurélien Géron, Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems. 2019.

77. ^S. Shafqat, H. Majeed, Q. Javaid, and H. F. Ahmad, "Standard NER Tagging Scheme for Big Data Healthcare Analytics Built on Unified Medical Corpora," J. Artif. Intell. Technol., Aug. 2022, doi: 10.37965/JAIT.2022.0127.

78. ^Z. Li, F. Yang, and Y. Luo, "Context Embedding Based on Bi-LSTM in Semi-Supervised Biomedical Word Sense Disambiguation," IEEE Access, vol. 7, pp. 72928–72935, 2019, doi: 10.1109/ACCESS.2019.2912584.

79. ^D. Machine and L. Data, "AI for Healthcare with Keras and Tensorflow 2.0," Springer, Accessed: Sep. 11, 2021. [Online]. Available: https://link.springer.com/content/pdf/10.1007/978-1-4842-7086-8.pdf.

80. ^S. Malmasi, W. Ge, N. Hosomura, and A. Turchin, "Comparison of Natural Language Processing Techniques in Analysis of Sparse Clinical Data: Insulin Decline by Patients.," AMIA Jt. Summits Transl. Sci. proceedings. AMIA Jt. Summits Transl. Sci., vol. 2019, pp. 610–619, 2019, Accessed: Aug. 17, 2019. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6568116/pdf/3051715.pdf.

81. ^A. Farhadi, "Classification Using Transfer Learning on Structured Healthcare Data."

82. ^B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," Progress in Artificial Intelligence, vol. 5, no. 4. Springer Verlag, pp. 221–232, Nov. 01, 2016, doi: 10.1007/s13748-016-0094-0

Declarations

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.