Eksplorium *p*-ISSN 0854-1418
Volume 46 No. 1, May 2025: 813–835 *e*-ISSN 2503-426X

Sensitivity Evaluation of Healthcare Data Using Biomedical NLP Using Weighted Features Analysis

[1]Brajesh Chaturvedi, [2]Dr. Harish Patidar
[1]Department of Computer Science and Engineering, Mandsaur University
Mandsaur, India
brajeshatindore@gmail.com
Brajesh Chaturvedi is a Research Scholar at Mandsaur University, Mandsaur, India. His research area includes Big Data Analytics, Natural Language Processing, Information Security and Machine Learning,
[2]Department of Computer Science and Engineering
Mandsaur University, Mandsaur, India
harish.patidar1@meu.edu.in
Harish Patidar, Ph.D is Associate Professor & Head, Department of Computer Science and Engineering, Mandsaur University, Mandsaur, India. His research area includes Information Security, Artificial Intelligence, Machine Learning, and Data Analytics.

Article Received: 25 Feb 2025, Revised: 27 April 2025, Accepted: 07 May 2025

Abstract—This paper demonstrates the use of transfer learning in biomedical NLP to identify sensitive data in electronic healthcare records. This research aims to improve the efficiency of multiclass classification of biological texts for sensitivity evaluation by combining two distinct feature representation methodologies. Multiple statistical weighting techniques, including as class probability (CP), inverse document frequency (IDF), and term frequency (TF), were considered for use with each component of the WE vectors in an effort to unify the two feature representations. Application of transfer learning is biomedical NLP opens up a great opportunity to exploit a lot of insights from the electronic medical records (EMR). BioALBERT, a variant of A Lite Bidirectional Encoder Representations from Transformers, was used in this investigation (ALBERT). It was taught with medical and biological databases. To classify all possible actions on the feature vector combinations we looked at, we developed a BioALBERT-based multiclass classification model. Experimental testing backs up the findings of the theoretical study of the proposed system. This research analyzed the usefulness and practicability of the proposed task using the MIMIC-III database. The MIMIC III and the PubMed dataset were utilized to construct the linguistic model. Our deep neural network model and other cutting-edge ML methods were used to test the efficacy of our weighted feature representation strategies for multiclass classification.

Keywords-Natural language processing; transfer learning; Biomedical NLP; BioALBERT.

I. INTRODUCTION

The EHR comprises a patient's medical history, vitals, lab results, and doctor's notes (EHR). These data can help doctors and patients communicate better. Predictive analysis of EHR data improves clinical care. EHRs are the ideal resource for evidence-based care because they contain

detailed patient information such as symptoms, primary complaints, treatments, procedures, tests, final diagnosis, discharge drugs, care notes, and referrals. These indicators can all be tracked. EHRs can help clinical informatics specialists increase their knowledge. The Medical Information Mart for Intensive Care (MIMIC) database is being mined and analyzed in several ways. MIMIC-III (Johnson et.al, 2016) is a huge, publicly accessible collection of de-identified healthcare data from intensive care unit patients from a large tertiary hospital. Clinical notes and discharge summaries are unstructured data, unlike diagnosis and test results. Organized data can be analyzed with statistical tools and machine learning programmes in ways unstructured text can't. Each patient's medical record includes admissions, clinical, transfer, and discharge notes. Traditional applications are constrained because extracting knowledge from unstructured data involves substantial human feature engineering and mapping to ontologies.

Text categorization (TC) has become an intriguing machine learning application in the previous decade. Classifying unknown data by finding commonalities between classes is TC's biggest problem. A classification model must complete "feature representation" before categorizing data. "Feature representation" explains the process of translating textual information into floating-point numerical vectors (Azar et.al, 2012; Hannah et.al, 2014; Jothi et.al, 2019; Jothi et.al, 2013; Anter et.al, 2013; Emary et.al, 2014; Banu et.al, 2017). More characteristics increase the classifier's pattern recognition (Meystre et.al, 2008; Bui et.al, 2022). Word embeddings and word bags are popular TC encoding methods. Both classification approaches are beneficial, but in different ways. The bag of words (BoW) technique lists the complete text, including phrases and documents. These words are recorded in a matrix regardless of grammatical, syntactic, or contextual relationships. BoW approaches include TF and TF-IDF analyses (TF IDF). Word embedding algorithms can uncover syntactic and contextual relationships (Cohen et.al, 2005; Chifu et.al, 2019; Chen et.al, 2021). Combining the two approaches may produce a more robust weighted feature representation model. When building a classification system, consider both approaches. Improved weighted feature representation can be employed in multiclass and multilabel classification systems.

Transfer learning is a relatively recent but strong NLP approach. Less fine-tuning of datasets improves performance. Transfer learning's domain adaptability is achieved by translating semantically related words into identical vectors. Pre-training a model for a data-intensive activity creates generic knowledge and transferrable skills. This affects future tasks. In computer

vision applications, the transfer learning model is pre-trained on a large labelled dataset, however in NLP applications, unsupervised learning is used. Transfer learning could be used to train the network utilizing online text data. Transfer learning models for NLP include GPT, ELMo, BERT, XLNET, ALBERT, and RoBERTa (Stanfill et.al, 2010; Nigam, 2016; Huang et.al, 2019; Li et.al, 2018; Shickel et.al, 2017). Each model proved effective for some tasks but not others. Just as these models' tactics, methods, and procedures vary in effectiveness, so do their ultimate results. Transfer learning requires a systematic strategy.

Increasing volumes of clinical reports and health literacy require more precise and generalized information extraction algorithms in biological NLP (BioNLP). Deep learning (DL) for NLP has revolutionized pre-trained language models (LMs) for BioNLP applications (NLP). Existing pre-trained LMs perform poorly in biological environments because they were taught on wordnet, Wikipedia, etc. LMs have been used in biomedical NLP algorithms for a decade. These include ELMo, BERT, and A Lite BERT. BioNLP researchers trained these LMs on biological and clinical corpora to improve biomedical data performance. (Peters et.al, 2018; Tiwari et.al, 2021)

This study applies transfer learning to biological NLP to evaluate data sensitivity in context. Biological (PubMed and PubMed Central) and clinical (MIMIC-III) corpora were fine-tuned for LM BioALBERT's domain-specific adaption. This research proposes and utilizes a weighted feature representation technique for classifying biological text. Our study combines WE and BoW to improve a biological multiclass text categorization system. First, we built a feature representation language model using MIMIC-III and PubMed data. We presented weighting techniques to blend the two feature representations. Our weighted feature representation method successfully classified biomedical texts after a battery of machine learning experiments.The remaining paper is structured as follows: The review of pertinent research that has lately been presented by a number of different academics takes up the second part of the paper. ALBERT and its biomedical variant are architecturally described in Section III. Section IV introduces a BioALBERT-based sensitivity retrieval mechanism for the supplied dataset. Section VII concludes the work after the experimental inquiry in part V reviews and discusses the effectiveness of the suggested technique.

II. RELATED WORK

Since deep learning can sift biological literature for useful nuggets of knowledge, several researchers have focused on undiscovered medical paradigms. NLP has enhanced processing

power. Biomedical NLP has impacted drug discovery and sequence estimation, for example. In the past decade, many advanced LMs have been developed for NLP (Erdil, 2019). When these approaches were applied to biomedical literature, the expected gains did not materialize, perhaps due to their initial training on generic corpora. BioNLP researchers resolved this restriction by training LMs on biological and clinical corpora and testing them on downstream tasks.

Jin et.al, (2019) built BioELMo from PubMed abstracts. Beltagy et al. (2019) developed the Scientific BERT paradigm to recover biomedical data attributes as relational information across various entities, which encouraged the authors to argue for its use in scientific education. (SciBERT). Si et al. (2019) trained BERT-based transfer learning models with clinical notes. Combining non-contextual and contextual word embedding improved named-entity-recognition for biological data. Numerous studies have used Peng et al (2019) Biomedical Language Understanding Evaluation (BLUE) score to measure their model's performance. Training LMs on clinical notes and PubMed abstracts improved bioNLP performance. BioBERT is the most common LM for biological applications because it pretrains on PubMed and PubMed Central data (PMC). The proposed model was modified to perform relationship extraction, named entity identification, and question answering (QA). PubMedBERT, Gu et al.'s (2020) new LM, used medical journal data. Using an LM trained on large corporas to translate topic-specific language, the suggested model's performance was evaluated. KeBioLM (Yuan et.al, 2021) was used to extract information from the UMLS corpus. Two BioNLP studies used KeBioLM. A domain-specific LM outperforms SOTA on biological NLP tasks (Naseem et.al, 2021).

Even if they've undergone previous training, BERT architecture makes retraining LMs costly and time-consuming. These LMs were only tested on a small fraction of BioNLP tasks, making generalization problematic. LMs have only been exposed to a limited collection of domain-specific corpora during their first training, therefore greater exposure can increase performance. Many tasks require clinical and biological terminology. We believe that ALBERT, like BERT, can be trained to enhance BioNLP tasks. Pretraining ALBERT on biological and clinical notes and then modifying it to provide context-aware summaries can give beneficial results for data sensitivity evaluation.

Word embedding research for classifying biological literature has been studied extensively. Sentence2Vec has enabled vector-based features. This method is like WEs (i.e., each vector represents a whole sentence instead of a word). Later, researchers tested the word embedding

method for multiclass classification of cancer corpus hallmarks. Sent2Vec *WE* was proposed in (Mikolov et.al, 2013). It combines sentences by averaging word vectors. A document-based WE system (Yuan et.al, 2021) can turn a single document into a vector. All document word vectors were combined. A recommendation algorithm (Jin et.al, 2019) helps discover comparable bug reports. The algorithm used similarity ratings from four vector approaches. Combining word embedding systems with TF-IDF weighting algorithms advanced word embedding research. Several weighting algorithms were utilized, including term frequency (TF), inverse document frequency (IDF), and smooth inverse frequency (SIDF) (SIF). Word embedding vectors collected all relevant word vectors for a particular document. Finally, we computed the word embedding vectors' weighted sums. IDF weighting won ROC and AUC.According to the research, a domain-specific WE is preferable in the biomedical industry. Many researchers have focused on improving and expanding WE infrastructure. Transfer learning can be a strong method for retrieving meaningful EHR data.

III. ARCHITECTURAL FRAMEWORK OF ALBERT

The expanding body of published biomedical literature, including clinical reports and health literacy, is facilitating the development of text mining algorithms. The goal of biological natural language processing (NLP) is to automate the process by which medical literature is sifted for references to diseases, drugs, genes, proteins, etc. Consequently, the improvement of methods for automatically detecting and extracting biological components is a critical step toward achieving this objective. Numerous text mining applications for biomedical NLP exist, such as the extraction of drug-drug interactions and disease-treatment correlations. The field of biological NLP has historically relied on feature engineering methods (e.g., lexicon-based, rules-based and statistics-based). Unfortunately, feature engineering relies heavily on domain knowledge, which biological NLP does not provide.

As more people turn to Deep Learning (DL) to automatically extract the most difficult aspects of a data field, interest in Biomedical NLP has skyrocketed. Performance in biological NLP has benefited greatly by enhanced LM's ability to obtain the vector representation of each word in a sentence (Lan et.al, 2019). Recent state-of-the-art (SOTA) DL-based language models have shown SOTA superior performance in a variety of NLP tasks. NLP was modified using transformer models, most notably BERT. The problem of sparse annotation in text data was ultimately resolved. Since it is now able to make adjustments to trained models, there is no

longer a need to retrain whole new models from scratch. BERT (340M parameters), however, is a little out of reach because of its vast size. Spending a lot of time and energy on BERT inference runs is a need. Reduced in size and weight without sacrificing any of BERT's useful features, ALBERT is a more portable and convenient alternative. Compared to the widely used BERT framework, the ALBERT model requires far fewer inputs. Two parameter reduction methods are employed to address the primary challenges of scaling pre-trained models. To get started, we have a factorized parameterization of the embedding. Dividing the massive vocabulary embedding matrix into two smaller matrices allows us to clearly see the differences in scale between the vocabulary embedding and the hidden layers. This dissection makes it easier to increase the hidden size without considerably increasing the parameter size of the word embeddings. Layer-to-layer parameter sharing is the second strategy. This strategy prevents the parameter from increasing linearly with the size of the network. Both techniques drastically cut down on BERT's parameter count without sacrificing performance, vastly improving parameter efficiency. Training on the ALBERT configuration for big is 1.7 times faster and requires 18 fewer parameters than the BERT setup (Devlin et.al, 2019). Regularization techniques used in parameter minimization strategies guarantees stable and transferable training.

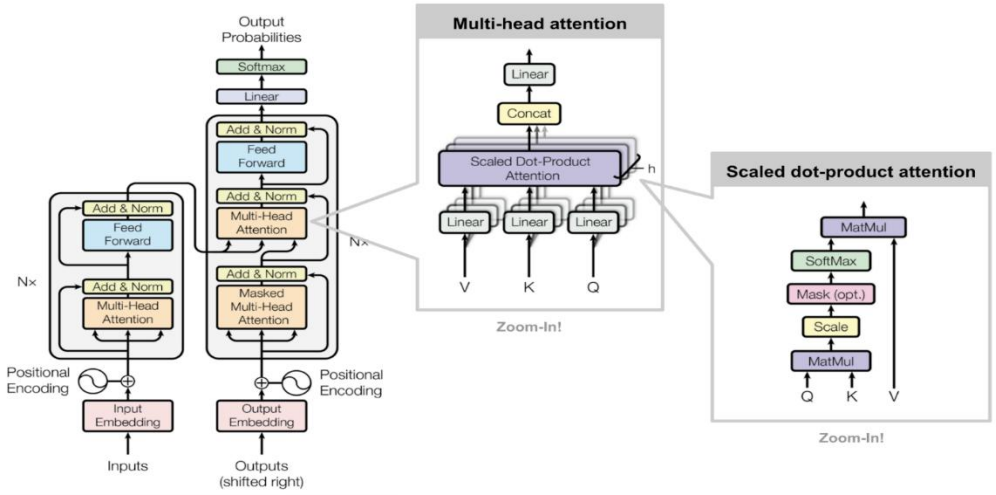

Figure1. Typical Transformer model

The building's framework, depicted in Fig. 1, is a multi-headed and multi-layered Transformer. In the ALBERT encoder-decoder architecture, the decoder is concerned with the encoder's outputs while the encoder is concerned with itself. The bricks used to create this structure were stacked vertically. A feedforward network and a multi-head attention block make up each of these sections. Contextual data and embedding data at the word level require large

representations in the hidden layer. Explosive variables, however, increase with increasing depth of the buried layer. If V is the total number of tokens in the vocabulary and H is the depth of the hidden layer, then we require parameters of the form V*H.Through the ALBERT factorization method, the dimensionality of these word-level input embeddings is reduced. In this context, we assume that E is the factorised embedding size.At this point, the required number of parameters is approximately equal to V*E + E*H.

The value of V is very large in most natural languages, allowing us to drastically reduce the number of parameters. While stacking multiple independent layers improves the models' learning ability, it also leads to a dramatic increase in redundancy. In many cases, characteristics that serve the same purpose are picked up by multiple tiers. Therefore, ALBERT is able to cut down on redundancy by sharing parameters across multiple tiers. A reduction in parameters is applied while the number of layers remains unchanged. Therefore, ALBERT is a relatively minor yet extremely beneficial alteration of BERT. It can reduce computational load and improve the effectiveness of language-understanding tasks in a number of contexts.

IV. PROPOSED BIOALBERT BASED DATA SENSITIVITY EVALUATION FRAMEWORK

Electronic medical records need a standardized system of document naming, linkage, and classification. Clients should be able to correctly identify the intended component of a request and be familiar with the data format of the given answer by using the EHR structure, which should be publicly available. The primary challenge of EHR interoperability lies in the development of a generic method for expressing any kind of healthcare data in a uniform data format. Methodical progress of the inquiry is shown in Figure 2. Our methodology incorporates a number of processes, including feature extraction, model training, and model testing. Spark was used for data preprocessing, and then Spark, Sklearn, and Gensim were used to extract the features.

The models were trained and tested using Spark ML and Keras.

a) Data Preprocessing

A sizable database of patients who were admitted to the critical care units of a significant tertiary care hospital makes up the MIMIC-III dataset. This dataset contains the medical records of patients who received treatment in the intensive care units at Beth Israel Deaconess Medical Center between 2001 and 2012. The goal of this research is to identify relevant semantic

information in unstructured data. The whole dataset was used, but only the free-text clinic notes and the noteevents table were examined. Discharge summaries stood out from other forms of summaries because, in contrast to those of other types, they included free text and genuine facts. Any reference of patient classification was removed before being included in discharge summaries because they were written after the diagnosis was determined (ICD-9 codes).

Table 1 lists the number of patients, hospital admissions, ICD-9 codes, and ICD-9 category counts from the MIMIC-III dataset. While noteevents and discharge summaries only cover the pertinent subgroups, MIMIC-III has access to the entire dataset. Methods we employed included the following:

- Removed punctuation marks.
- Removed numbers.
- Normalized all characters into lowercase.
- Word tokenization.

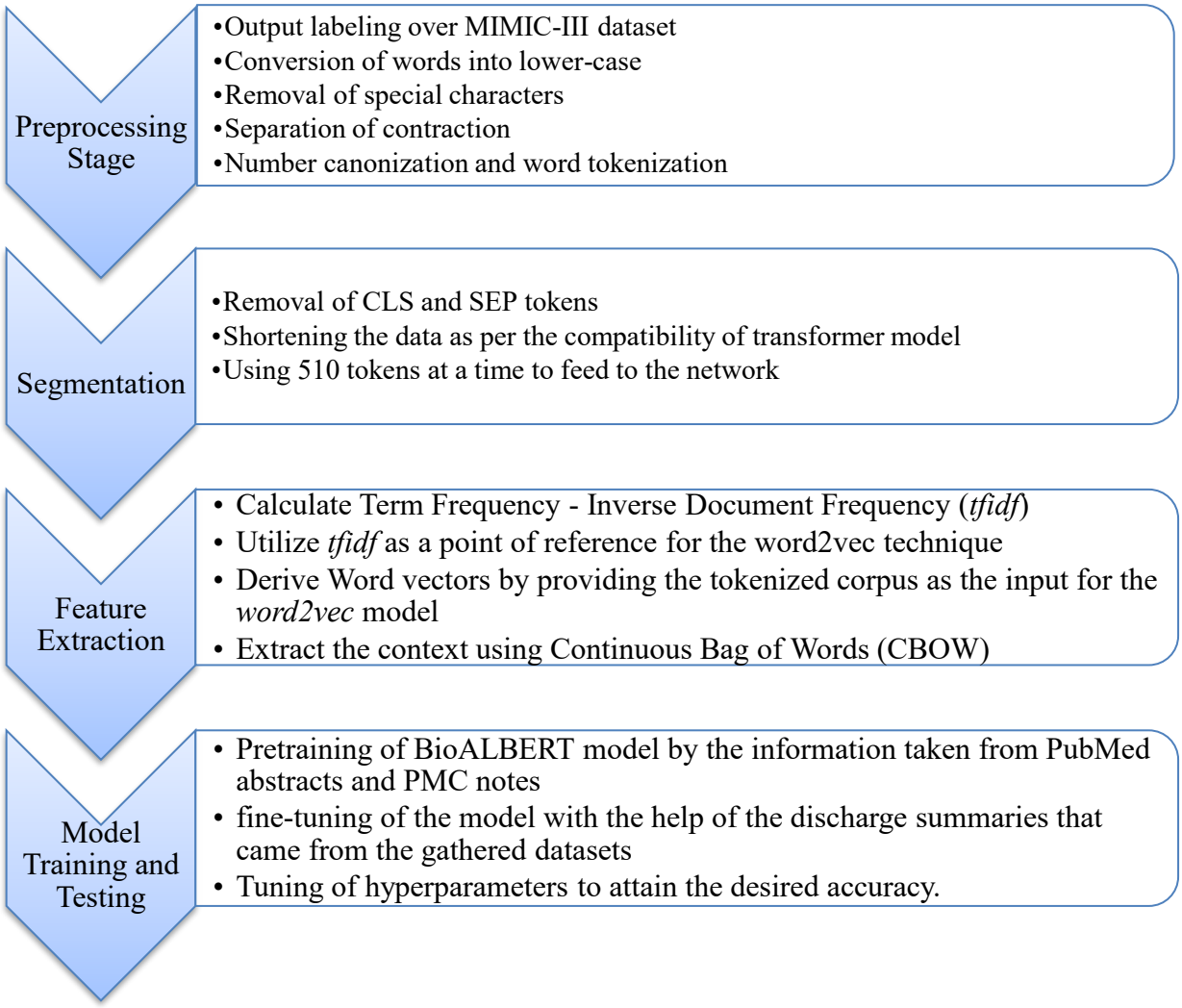

Figure 2. Proposed methodology

b) Segmentation

The method of distinguishing distinct sentences from a collection of words is known as sentence segmentation or sentence tokenization. Sentence segmentation is carried out with far more precision using the Spacy library, which was created for Natural Language Processing. Using the existing tokenization libraries and the CLS and SEP tokens, the sentences in the discharge summaries of the preprocessed dataset are segmented. According to Romanov et al. (2018), the real limit for the input layer in transformer models is 510 tokens, down from 512 with the removal of the CLS and SEP tokens. The data must be condensed in order to be incorporated into the Transformer model because the average length of the MIMIC-III discharge summaries is 1,947 tokens, and only 11.67 percent of all papers are longer than 510 tokens. Despite the summaries' narrative style, the majority of the documents adhere to a standard

format, which starts with a Chief Complaint and continues with a Historical Background section that provides details about the patient's social and ethnic backgrounds, past and current medical conditions, and other relevant information. A few subsections in Diagnostic tests and Relevant Results are more case-specific, and the overall structure is less uniform. The documentation is divided into two sections: "Discharge Instructions" at the end that details the conditions for the patient's departure from the hospital, and a "Brief Hospital Course" section in the middle that details the patient's stay in the intensive care unit. The Brief Hospital Course's content up to this point will be removed, and the remaining text will be used sequentially until all 510 tokens have been used. Those summaries were not included since the Brief Hospital Course was missing from 822 publications.

c) Statistical Feature Representation

Then, based on the *tf-idf* and *CP* values of every word in the corpus, the statistically weighted features are extracted. The word embedding vectors are then produced using these attributes. Considering a dataset D of n text summaries to classify the data sensitivity levels $D = \{d_1, d_2, d_3, \ldots, d_n\}$ where each text summary is represented using one weighted score in terms of *tf-tdf* or *CP* as $d_i = \{w_1, w_2, w_3, \ldots, w_n\}$. All the terms appearing in the dataset D build a vocabulary, $V = \{t_1, t_2, t_3, \ldots, t_p\}$ where a feature set $F_i = \{s_i, we_i\}$ comprising of two features (weighted value and word embedding vectors) corresponding to each term t_i in the vocabulary. We computed all of the weighted scores for each distinct phrase in the evaluation dataset. For feature extraction, we will use the following two methods: There are two of them: Class Probability (*CP*) and Term Frequency - Inverse Document Frequency (*tfidf*). The *tfidf* is a benchmark for *word2vec*. The goal of the *tfidf* is to determine a word's importance to a piece of writing within a corpus of papers. The outcome of merging *tf* and *idf* statistics is this. *Idf* evaluates a word's frequency across the corpus as opposed to *tf*, which counts how often a word appears in a document. We'll base our computations on the following definition of *idf*:

$$idf(w) = log \frac{nd}{df(d,w)} + 1$$

where n_d denotes the total number of documents and $df(d,w)$ denotes the number of documents containing the word *w*. The baseline is the *tfidf* compared to *word2vec*. The *tfidf* is a technique for figuring out a word's importance within a corpus of texts. The *tf* and *idf* statistics produced it. The *idf* algorithm determines if a word is common or rare across the corpus, while the *tf* algorithm counts the number of times a word appears in a document.

To get *tfidf*, we tokenized each track from the filtered training data set. Following this, a document-word matrix was created, and the total number of words from each note was entered. Then, the *idf* was multiplied by each word individually. The top 40,000 words with the greatest *tfidf* scores were utilized as the bag of word features in one *tfidf* configuration, while the top 20,000 words were used in another configuration with a minimum document frequency of 10 and a maximum document frequency of 0.8 of the total number of documents.

On the other hand, the conditional probability $CP_{t_{j,C_k}}$ that indicates how frequently the word tj appears when the class C_k is appearing is known as the class probability. *CP* illustrates the significance of a particular term while taking into account class information. For instance, a phrase t_i is strongly associated with class A if it appears more than thrice in documents annotated with that class.

$$CP_{t_{j,C_k}} = \frac{t_j \cap c_k}{P(c_k)}$$

where $P(C_k)$ is the likelihood of the class C_k. As a result, three weighted scores are used to represent each term in document di in the collection.

$$s_j = \sum_{d=0}^{n} s_d = \left(TF_{t_j}, IDF_{t_j}, CP_{t_j} \right)$$

Together, the four million pages from MIMIC III and Pubmed were used to create a vector of word embeddings. In order to better understand the relationships between words in a vocabulary, the FastText WE software generated a 100-dimensional vector space for each word. FastText's powerful model has helped to solve a wide variety of issues with natural language, such as those involving word morphology and words that aren't in the dictionary.

To accomplish this, we utilised the Gensim Python software to construct word vectors from the four million papers. A vocabulary V was produced consisting of 100-dimensional WE vectors for each distinct word in dataset D. Next, the *WE* vectors are merged with the weighting scores to produce the Weighted Word Embedding Vectors, the second stage of the proposed statistical feature representation. This means that a simple weighting factor was used to mix the *WE* vectors with the three weighted scores for each phrase tj in the vocabulary V. In the vocabulary assignment T_j will be a substitute for the word *V*. Applying the weighted multiplication to an arbitrary term t_j and each item of its associated *WE* vector (we_j) yields the

three weighted feature vectors HTF_j, $HIDF_j$, and HCP_j, one for each of the weighting processes *TF*, *IDF*, and *CP*.These weighted feature vectors are defined as below:

$$HTF_j = \sum_{l=0}^{n} TF_{t_j} \times we_{j,l}$$

$$HIDF_j = \sum_{l=0}^{n} IDF_{t_j} \times we_{j,l}$$

$$HCP_j = \sum_{l=0}^{n} CP_{t_j} \times we_{j,l}$$

Figure 2 shows how the combination process is done in detail.

D) Model Training and Testing

While these models show great promise, they also have several limitations, such as a lack of training data, an excessive reliance on acronyms, and the fact that a single entity can stand in for several different entity types depending on the surrounding circumstances. Thus, modern EMR models mix language models that were trained on biological information with models that were trained independently of context. To address the limitations of newly reported domain-specific language models, we developed a context-dependent, fast, and effective language model (bioALBERT) for the biomedical domain. To determine how sensitive the data is, bioALBERT can be employed.

To make up for a lack of training data, BioALBERT makes advantage of large biomedical corpora during its training ((Lee et.al, 2020)). As an additional measure of sentence coherence loss, we developed the state-of-the-art cross-layer parameter sharing methodology. Learning the parameters of the first block allows us to apply those same settings to subsequent levels. Finally, it was found that in BERT-based models, the size of the embedding is related to the size of the buried layer of the transformer block sizes. To help the model construct more precise representations, the SOP randomly pairs two words from the training data. To distinguish between the dimensions of the hidden layers and the vocabulary embeddings, we adopt factorized embedding parameterization, which splits the embedding matrix into two smaller matrices. Because of this, we can increase the secret quantity without correspondingly raising the vocabulary embedding's value. Since it is optimized for EMR workloads, BioALBERT outperforms other SOTA models in terms of usefulness and productivity.

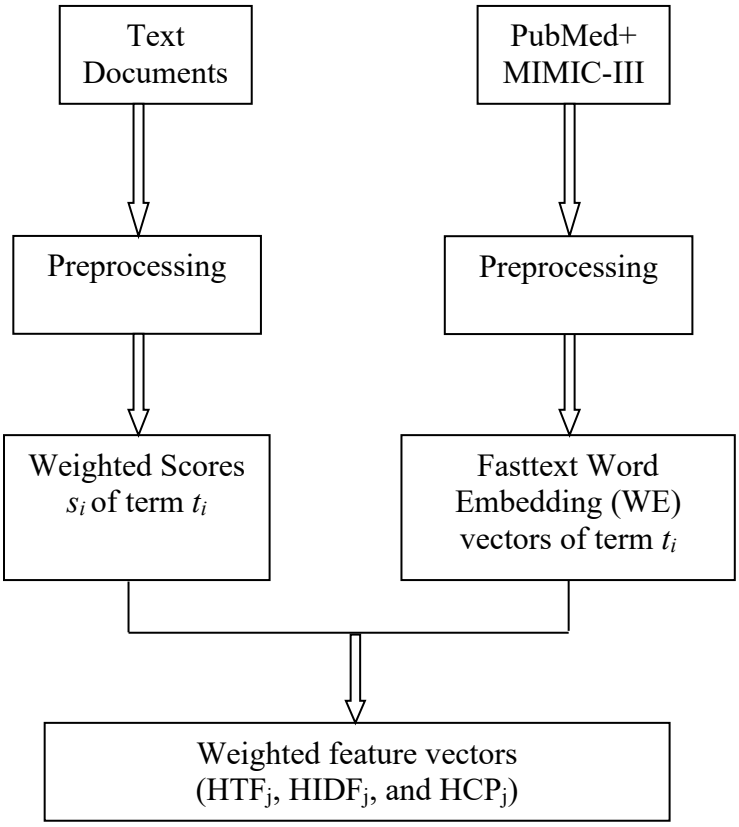

Fig. 2. Weighted feature representation

Below you'll find more details regarding BioALBERT's initial training and ongoing fine-tuning.

A. BioALBERT's pre-training

Due to the simplicity and similarity of its architecture to ALBERT's, we will outline the pre-training processes for BioALBERT in this section. BioALBERT permits the ALBERT model to be pre-trained on general text in biomedical corpora through the use of training data from PMC full-text articles and PubMed abstracts that contain biological terminology. After fusing all the source text files, we got a sentence with the following features: Since we were unable to pre-train the model with the PubMed and PMC raw text biomedical corpora, we added the following three features: When merging multiple files, a blank line will be added between each file, and lines with fewer than 20 characters will be discarded (when combining multiple files). Word count wise, PMC has 13.5 billion versus 4.5 billion for PubMed. The ALBERT model's base vocabulary was used in conjunction with pre-training on biological corpora to produce our final BioALBERT model. Since sentence embeddings are utilized by tokenization, the data was first

considered to be a string of sentences. Since each document in the input text is separated by a blank line, each line in the document is a sentence. It was capped at 512 characters per statement. Except for BioALBERT 1.1, all pre-trained models use a learning rate of 0.00176 and a total of 3125 warm-up steps.

B. BioALBERT fine-tuning

This study discusses a few ways in which the BioALBERT contextual summary task could be enhanced. It requires marking specific words in a sentence as "designated entities." Diseases, species, drugs/proteins, and drugs/chemicals were all represented in the datasets utilized for this task. The labels, which are the domain-specific nouns, are the primary inputs for this type of learning and prediction. As a clinical resource, please fill out the provided contextual summary exercise as accurately as possible using the provided symptoms and ICD 9 codes. When compared to pre-training, fine-tuning places far lower demands on computing resources.

BioALBERT is a great illustration because it uses fewer computational resources and less actual memory than similar approaches while still facilitating novel parameter swaps. Word embeddings, the foundation of the BioBERT model, are revealed through sentence tokenization-based fine-tuning for context-specific summary synthesis and the corresponding sensitivity scale. We created a fine-tuning task for each of the models we had previously trained using the unique dataset. Separate from the standard overview, the dataset now features a dedicated section on sensitivity. The weights from the trained model are used as inputs in the present setup. Training began with a total of 5336 steps, lower case texts, a batch size of 32, and a learning rate of 1e-5. In the past, BioALBERT models were trained using TPUv3-8. All hyper-parameters in ALBERT will be set to their default settings unless otherwise specified. There is a list of words in each dataset, labelled B, I, and O, where B represents the first word of an entity, I represents the words that follow it, and O represents the absence of the entity. To verify the efficacy of our pre-trained models for the downstream goal, we conducted experiments using the raw data. Using the holdout development dataset, the evaluation checkpoint, and the Adam optimizer, the model with the best overall performance was chosen. Specifically, PyTorch-transformers derived from BERT and XLNet deployments were used in the research. Figure 3 is a time line depicting the entirety of the training procedure. Since layer freezing didn't occur during model tuning, it's likely that this is what happened. Adam (Kingma et.al, 2014) performed well in the initial experiments, but Layer wise Adaptive Large Batch (LAMB) optimizer (Yang et.al, 2019) has

proven to be slightly more efficient in terms of training time. When choosing and refining the hyperparameters, we considered the outcomes from the training set. When tested with varying learning rates, the Transformer model fares poorly, indicating that it is highly sensitive to these parameters. Optimal performance was achieved at a learning rate of $7x10^{-4}$ or $6x10^{-4}$.

No matter what methods of warming were used, the results were the same. The cased version of BERT outperformed the uncased one. Nonetheless, the variations are still barely noticeable. At present, only Cased of XLNet is supported. With the significant computational cost in mind, we opted for the simpler XLNet version. When comparing BERT and XLNet models, the difference in training batch size was 16. Using binary cross entropy with logits, we sorted the codes and calculated our confidence in each one.

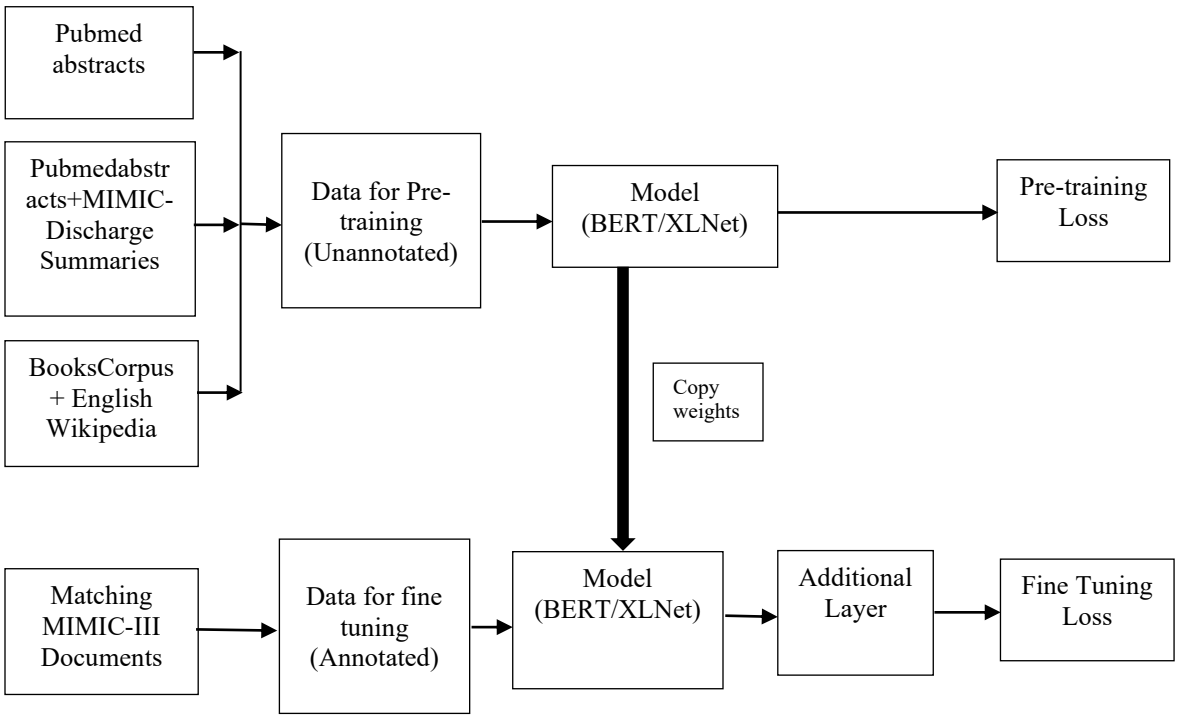

Figure 3. End-to-end training procedure

V. RESULTS AND DISCUSSION

An experimental study has made use of the MIMIC-III dataset, a substantial database of patients admitted to the intensive care units of a major tertiary care hospital. The patients who were admitted to the intensive care units at Beth Israel Deaconess Medical Center between 2001 and 2012 are represented in this dataset. Using the expertly annotated dataset, we took advantage of the ICD-9 codes that were included. In most cases, we used the preprocessing and data

separation methods from (You et.al, 2019). The model was trained using the most recent MIMIC-III dataset, validated using data from 1,632 patients, and tested using summaries from 3,372 patients who had been recently discharged. In addition, we employed the most widely used "MIMIC-III-50" parameter which orders the labels by frequency of occurrence.

Training discharge summaries numbered 8,066, test discharge reports numbered 1,729, validation discharge summaries numbered 1,573, and the aim of this research is to extract useful semantic information from large amounts of unstructured data. Our research included the entire dataset, with a special emphasis on the free-text sections of the clinic notes and the noteevents table. Unlike other types of summaries, discharge summaries integrated free-form prose with concrete data, giving them a distinct advantage. Discharge reports were written after a diagnosis was made, hence information pertaining to patient classification was removed before being presented (ICD-9 codes). For instance, we took the recommended number of sensitivity levels and distributed them as shown below. Sexual dysfunction and infertility (Level 4), victims of sexual violence (Level 5). Fever due to a virus, the common cold, a cough, etc. fall within the first category of illnesses. THREE: Extremely Malignant Cancer, Cancers of different types (blood, breast, et cetera) are at the Stage 2 level. The sensitivity scale goes from 1 (the least sensitive) to 5 (the most sensitive). In this case, the user will have control over how sensitive the approach is. Since there are N types of diseases and N categories of diseases (e.g., more than 150 subtypes of cancer), we've established sensitivity by dividing diseases into categories and assigning weights to each. The resulting sensitivity level can range from 1 to 5, as indicated by the range of values. The domain, the list of sensitive words, and the sensitivity level of the word are all taken into account. The information is arranged in rows for easy processing. The row is compared to the list of sensitive phrases to determine the overhead and sensitivity. Table 1 displays data on the total number of patients, hospital admissions, distinct ICD-9 codes, and ICD-9 categories.

Table 1. Analytics of MIMIC-III dataset

Dataset File	**Hospital Admissions**	**Number of patients**	**ICD-9 Codes**	**Categories of ICD-9**
Complete MIMIC-III	58976	46520	6984	943
Discharge Summaries	52726	41127	6918	942
Noteevents	58726	41127	6918	942

MIMIC-III has access to the full dataset, whereas noteevents and discharge summaries only cover the relevant subgroups. The top ten ICD-9 codes and top ten ICD-9 categories are shown in Table 2.For training, validation, and testing, the filtered datasets is divided into 50-25-25 groups.

Table 2. Top 10 ICD-9 codes and categories and respective admissions

ICD-9 Codes	Admissions		ICD-9 Category	Admissions
4019	20046		401	20646
4280	12842		427	16774
42731	12589		276	14712
41401	12178		272	14212
5849	8906		414	14081
25000	8783		250	13818
2724	8503		428	13330
51881	7249		518	12997
5990	6442		285	12404
53081	6154		584	11147

Due to computational resource limits, the training batch size was reduced from 1,024 for BioALBERT basic models to 256 for BioALBERT large models. The training variables are detailed in Table 3 (Sun et.al, 2019).

Table 3. Hyperparameters for pre-training

Parameters	**Values/Description**		**Parameters**	**Values/Description**
Baseline model	ALBERT		Size of vocab	30000
Kernel function	GeLU		Optimizer	LAMB
Number of attention heads	12		Size of training batch	1024
Number of layers	12		Size of evaluation batch	16
Hidden layers	768		Maximum length of sentence	512
Size of embedding	128		Warm-up steps	3125

When compared to other baseline models, BioALBERT's computational expenses were on par, and fine-tuning used a lot less processing power than pre-training did. BioALBERT was able to learn word embeddings by utilizing novel parameter-sharing strategies, memory-constrained hardware, and sentence-level tokenization. Time spent in training was reduced without a corresponding decrease in efficacy.

During this stage, we employ the BioALBERT LM's training weights. To do this, we employed an AdamW optimizer with a learning rate of.000001%. After dividing into 32 groups, we dove into our homework. Each word's first letter was capitalized, and sentences were limited to a maximum of 128 characters. To get our trained models ready for the 10k training stages that came before the fine-tuning, we used a 512-step warm-up. The test data was used to make predictions, and the evaluation metric was compared to those of previous SOTA models. All tuning settings can be found in Table 4.

Table 4. Hyperparameters for fine tuning

Parameters	**Values/Description**
Optimizer	ADAMW
Size of training batch	32
Learning rate	0.00001
Number of training steps	10000
Warm-up steps	320

The typical validation accuracy of the CP weighting method is 0.494 and the typical loss is 1.901, as shown in Figure 4. We further highlight that the validation results showed a 0.01 percentage point error difference between CP and the other weighting techniques (TF and IDF). Initial findings, however, showed the best training accuracy and the least loss. All weighting strategies shown superior increases in validation accuracy over the control group. The accuracy of the proposed WE technique is superior than the other existing techniques.

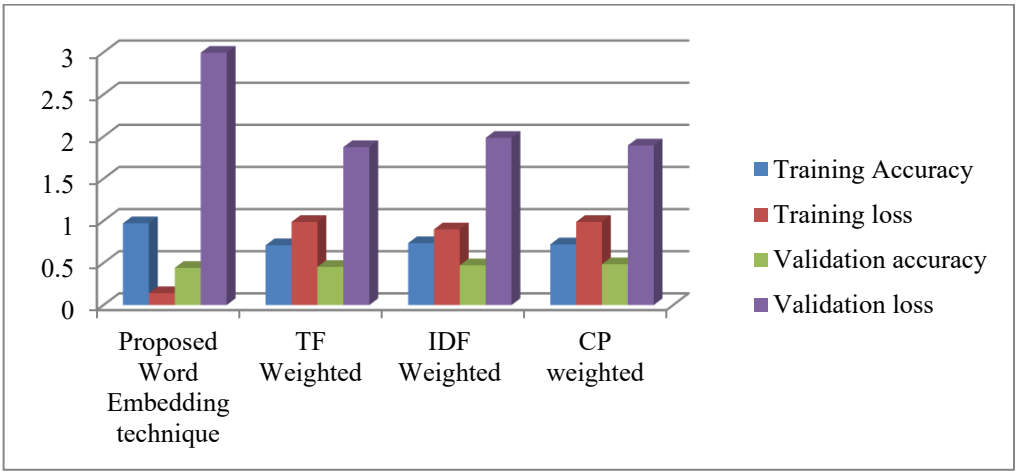

Fig. 4. Classification Performance metrics

Discussion: Success was achieved using the proposed classification model and the CP weighting method. About 1% separated CP from the other two methods of weight estimation. Results for all weighting methods were also better than in the control trial. To completely understand the efficacy of these weighting mechanisms, we must investigate the WE systems' work philosophy. Feature vectors can be generated by WE from anything from a single character to an entire document. Connections between words and between grammar structures occur instantly. Each of the syllables that make up the vector is a forceful argument in favour of the idea it defines. As a result, we postulate that altering this representation may have unintended consequences for the efficiency of the classification scheme. By combining the TF, IDF, and CP statistical data, we found that the weighting algorithms performed well in our situation. The fact that CP also performed exceptionally well in school is proof that prior knowledge can enhance accuracy.

VI. CONCLUSION

Our research here shows how the context-aware EHR was developed using transfer learning for biological NLP to evaluate the level of data sensitivity. New applications in biomedical NLP have been developed as a result of the availability of biomedical text data and advancements in natural language processing (NLP). Language models trained or improved with domain-specific corpora perform better than generic models. However, limited corpora and challenges are available for biological NLP research at the present time. BioALBERT, a variation of A Lite Bidirectional Encoder Representations from Transformers (ALBERT), was used since it has previously shown promise when trained on medical and scientific data. In this research, we developed a therapeutically relevant multiclass classification strategy for mining EHRs for sensitive data. The theoretical evaluation of the proposed system is supported by empirical study. The accuracy of the proposed WE technique (97%) is superior than the other existing techniques.This research determined the usefulness and efficiency of the proposed work by consulting the MIMIC-III database.

REFERENCES

[1] Johnson A. E. W., Pollard T. J., Shen L., Lehman H., Wei L., Feng M., Ghassemi M., Moody B., Szolovits P., Celi L. A. and Mark R. G. (2016), ‘Mimic-III, a freely accessible critical care database’, *Scientific data 3*:160035.

[2] Azar, A.T., Hassanien, A.E., Kim, Th. (2012), 'Expert System Based on Neural-Fuzzy Rules for Thyroid Diseases Diagnosis', *In: Kim, Th., Kang, JJ., Grosky, W.I., Arslan, T., Pissinou, N. (eds) Computer Applications for Bio-technology, Multimedia, and Ubiquitous City. BSBT MulGraB IUrC 2012. Communications in Computer and Information Science, vol 353*.

[3] Hannah Inbarani, H., Nizar Banu, P.K. & Azar, A.T. (2014), 'Feature selection using swarm-based relative reduct technique for fetal heart rate', *Neural Computing and Applications*, vol. 25, pp. 793–806.

[4] Jothi, G., Inbarani, H.H., Azar, A.T. *et al.* (2019), 'Rough set theory with Jaya optimization for acute lymphoblastic leukemia classification', *Neural Computing and Applications,* vol. 31, pp. 5175–5194.

[5] Jothi, G., Inbarani, H.H., Azar, A.T. (2013), 'Hybrid Tolerance Rough Set: PSO Based Supervised Feature Selection for Digital Mammogram Images', *International Journal of Fuzzy System Applications (IJFSA)* , vol.3, no. 4, pp.1-16.

[6] Anter A. M., Azar A. T., Hassanien A. E., El-Bendary N. and ElSoud M. A. (2013), "Automatic computer aided segmentation for liver and hepatic lesions using hybrid segmentations techniques," *2013 Federated Conference on Computer Science and Information Systems*, Krakow, Poland, pp. 193-198.

[7] Emary E., Zawbaa H. M., Hassanien A. E., Schaefer G. and Azar A. T. (2014), 'Retinal vessel segmentation based on possibilistic fuzzy c-means clustering optimised with cuckoo search', *2014 International Joint Conference on Neural Networks (IJCNN)*, pp. 1792-1796.

[8] Nizar Banu P.K., Azar A. T. and Inbarani H. H. (2017), 'Fuzzy firefly clustering for tumour and cancer analysis', *International Journal of Modelling, Identification and Control*, Vol. 27, No. 2, pp. 92-103.

[9] Meystre S. M., Savova G. K., Schuler K. C. and HurdleJ. F. (2008)'Extracting information from textual documents in the electronic health record: a review of recent research',*Yearbook of medical informatics*, vol. 17, no. 01, pp. 128–144.

[10] Bui, QT., Ngo, MP., Snasel, V. et al. (2022), 'The Sequence of Neutrosophic Soft Sets and a Decision-Making Problem in Medical Diagnosis', *International Journal of Fuzzy System*, vol. 24, pp. 2036–2053.

[11] Cohen A. M. and HershW. R. (2005),'A survey of current work in biomedical text mining',*Briefings in bioinformatics*, vol. 6, no. 1, pp. 57–71.

[12] Chifu V. R., Chifu E. S., Pop C. B., Salomie I. and M. Lupu (2019), 'Evolutionary-based method for risk stratification of diabetic patients', *International Journal of Intelligent Engineering Informatics*, vol.7, no.1, pp.37 – 60.

[13] Chen, R.J., Lu, M.Y., Chen, T.Y. et al. (2021), 'Synthetic data in machine learning for medicine and healthcare', *National Biomedical Engineering*, vol. 5, pp. 493–497.

[14] Stanfill M. H., WilliamsM., FentonS. H., JendersR. A. and HershW. R. (2010), 'A systematic literature review of automated clinical coding and classification systems', *Journal of the American Medical Informatics Association: JAMIA*, vol. 17, no. 6, pp. 646–651.

[15] Nigam P. (2016),'Applying deep learning to ICD-9 multi-label classification from medical records',*Technical report*, Stanford University.

[16] Huang J., Osorio C. and Wicent L. (2019),'An empirical evaluation of deep learning for ICD-9 code assignment using MIMIC-III clinical notes', *Computer Methods and Programs in Biomedicine*, vol. 177, pp.141–153.

[17] Li F.,Liu W. and YuH.(2018),'Extraction of information related to adverse drug events from electronic health record notes: design of an end-to-end model based on deep learning', *JMIR medical informatics*, vol. 6, no. 4, e12159.

[18] Shickel B., TigheP. J., Bihorac A. and RashidiP. (2017),'Deep EHR: a survey of recent advances in deep learning techniques for electronic health record (EHR) analysis', *IEEE journal of biomedical and health informatics*, vol. 22, no. 5, pp.1589–1604.

[19] Peters M., et al. (2018),'Deep contextualized word representations',*In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, *Association for Computational Linguistics,* Vol. 1 (Long Papers), pp. 2227–2237.

[20] Tiwari V., Verma L. K., Sharma P., Jain R. and Nagrath P. (2021), 'Neural network and NLP based chatbot for answering COVID-19 queries', *International Journal of Intelligent Engineering Informatics*, vol. 9, No. 2, pp 161-175.

[21] Erdil D. C. (2019), 'A comparison of health informatics education in the USA',*International Journal of Intelligent Engineering Informatics*, vol.7, No.4, pp. 366 – 383.

[22] Jin Q., DhingraB., Cohen W. and LuX. (2019), 'Probing biomedical embeddings from language models',*In Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pp. 82–89.

[23] Beltagy I., LoK., and CohanA. (2019), 'SCIBERT: A pretrained language model for scientific text', *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3606–3611.

[24] Si Y.,WangJ., Xu H. and RobertsK. (2019),'Enhancing clinical concept extraction with contextual embeddings',*Journal of American Medical Informatics Association*,vol. 26, pp. 1297–1304.

[25] Peng Y., Yan S. and LuZ. (2019), 'Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets',*In Proceedings of the 18th BioNLP Workshop and Shared Task*, pp. 58–65.

[26] Gu Y., et al. (2020),'Domain-specific language model pretraining for biomedical natural language processing',*arXiv preprint arXiv:2007.15779*.

[27] Yuan Z.,LiuY.,TanC., Huang S. and HuangF.(2021), 'Improving biomedical pretrained language models with knowledge',*arXiv preprint arXiv:2104.10344*.

[28] Naseem U., KhushiM., ReddyV., RajendranS., Razzak I.and KimJ. (2021), 'BioALBERT: A Simple and Effective Pre-trained Language Model for Biomedical Named Entity Recognition', *International Joint Conference on Neural Networks (IJCNN)*, pp. 1-7.

[29] Mikolov T., Sutskever I., Chen K., Corrado G. S. andDean J. (2013), 'Distributed representations of words and phrases and their compositionality', *In Advances in neural information processing systems*, pp. 3111–3119.

[30] Yuan Z.,LiuY., TanC., Huang S.and HuangF.(2021), 'Improving biomedical pretrained language models with knowledge',*arXiv preprint arXiv:2104.10344,* pp. 1-11.

[31] Jin Q.,DhingraB.,Cohen W. and LuX. (2019), 'Probing biomedical embeddings from language models',*In Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pp. 82–89.

[32] Lan Z., et al. (2019), 'Albert: A lite BERT for self-supervised learning of language representations', *In International Conference on Learning Representations*, pp. 1-17.

[33] Devlin J.,ChangM. W.,Lee K.and ToutanovaK. (2019), 'Bert: Pre-training of deep bidirectional transformers for language understanding',*In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 4171–4186.

[34] Romanov A. and ShivadeC. (2018),'Lessons from natural language inference in the clinical domain',*In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1586–1596.

[35] Mullenbach J., WiegreffeS., DukeJ., SunJ.and EisensteinJ.(2018), 'Explainable prediction of medical codes from clinical text', *in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana*, pp. 1101–1111.

[36] Lee J., YoonW., KimS., KimD., KimS., SoC. H., and KangJ. (2020), 'BioBERT: a pre-trained biomedical language representation model for biomedical text mining',*Bioinformatics*, Vol. 36, Issue 4, pp. 1234–1240.

[37] Kingma D. P. and BaJ. (2014), 'Adam: A method for stochastic optimization', *arXiv preprint arXiv:1412.6980*.

[38] Yang Y. et.al. (2019),'Large Batch Optimization for Deep Learning: Training BERT in 76 minutes',*arXiv:1904.00962 [cs.LG]*.

[39] You, Yang and Li, Jing and Reddi, Sashank and Hseu, Jonathan and Kumar, Sanjiv and Bhojanapalli, Srinadh and Song, Xiaodan and Demmel, James and Keutzer, Kurt and Hsieh, Cho-Jui (2019), 'Large Batch Optimization for Deep Learning: Training BERT in 76 minutes'.

[40] Sun C., Qiu X.,Xu Y. and Xuang X. (2019), 'How to fine-tune BERT for text classification?',*In: China National Conference on Chinese Computational Linguistics*, pp. 194-206.