Thesis No. CSER-22-43

Medical Named Entity Recognition From Twitter Text

By

Md. Monarul Islam

Roll: 1607104

&

Md. Firoz Ahmed

Roll: 1607118

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

March, 2022

Medical Named Entity Recognition From Twitter Text

By

Md. Monarul Islam
Roll:1607104
&
Md. Firoz Ahmed
Roll:1607118

A thesis submitted in partial fulfillment of the requirements for the degree of
"Bachelor of Science in Computer Science & Engineering"

Supervisor:
Dr. K. M. Azharul Hasan
Professor
Department of Computer Science and Engineering
Khulna University of Engineering & Technology

____
Signature

Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna 9203, Bangladesh
March, 2022

Acknowledgement

First and foremost, we give thanks to Allah, the Almighty, who created us and will always be with us. We have completed our thesis with the mercy of Allah Almighty. We are grateful to Prof. Dr. K. M. Azharul Hasan, Department of Computer Science and Engineering, Khulna University of Engineering and Technology, for guiding and directing us through the completion of our thesis work. He made materials and tools available to us that will aid in the improvement of our thesis work. He was a well-behaved, well-disciplined individual who assisted us in becoming more responsible.

Next, we thank our parents, who are the true heroes and heroines in our lives. They will always be with us till we are born. They are even more supportive and helpful in ensuring our success. For us, their prayer was more powerful.

Next, we want to express our gratitude to all of our class teachers who have shared their knowledge with us, which has aided us in completing our thesis. Also, a big thank you to our classmates for sharing their unique skills, which helped us finish our thesis.

Abstract

Medical Named Entity (MNER) is a critical phrase for anyone working in the medical area, especially doctors and researchers. Medical Named Entity Recognition has been proven to be a critical step in achieving effective medical text analysis. Medical entities can be a single word or a series of words. Disease-related terminology, biological chemical agents, healthcare-related terms, and so on are all examples of medical entities. Our research focuses on extracting disease-related phrases from natural writing on Twitter, such as diseases, medicines, symptoms, tests, and treatments. Medical Named Entity Recognition (MNER) is a critical problem in Natural Language Processing (NLP) that involves finding text spans that express entity references. More medical information can be found on social media sites such as Twitter. NLP, a more advanced field of language processing, can be used to retrieve this information. We suggested the probabilistic language model Conditional Random Fields (CRF) and the deep learning model Bidirectional LSTM (Bi-LSTM) to extract the medical entity from twitter content. Deep leaning has demonstrated exceptional performance in entity recognition, and the probabilistic model has also demonstrated its ability to recognize entities. We used two datasets in our thesis: the CADEC corpus (122581 instances) and the twitter datasets (12326 instances). Each word was tagged or identified in the datasets (B-Disease, I-Disease, B-drug, I-Drug , O etc). Our suggested model Bi-LSTM-CRF has an accuracy of 62.70% on datasets, whereas ME (Maximum Entropy Model) has an accuracy of 72.31%, which is significantly less better than Bi-LSTM-CRF. This process becomes more challenging when dealing with tweets, which are known to be full of slang and provide insufficient context for information extraction.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviations	**Description**
MNER	**M**edical **N**amed **E**ntity **R**ecognition
CRF	**C**onditional **R**andom **F**ield
LSTM	**L**ong **S**hort **T**erm **M**emory
Bi-LSTM	**Bi**-directional **L**ong **S**hort **T**erm **M**emory
NLP	**N**atural **L**anguage **P**rocessing
NER	**N**amed **E**ntity **R**ecognition
BP	**B**ack **P**ropagation
RNN	**R**ecurrent **N**eural **N**etwork
EHR	**E**lectronic **H**ealth **R**ecord
ME	**M**aximum **E**ntropy

Chapter 1

Introduction

1.1 Introduction

Relevant information extraction has become an essential component of our daily lives in the current era. More information is available on social media, particularly medical information. Named Entity Recognition (NER) is a sub-field of information extraction [1]. MNER (Medical Named Entity Recognition) is a prominent method for extracting medical entities. Medical entities contain more crucial medical information. Diseases, drugs, symptoms, tests, and treatments are examples of medical entities that might be a single term or a sequence of terms. Twitter is a social media platform with a significant amount of data that may also be considered a big data repository[2]. With the use of natural language processing, we were able to extract necessity data from Twitter. In our task, medical diseases-related entities has been extracted from twitter text. In our thesis, we study data from the social media site Twitter, with the data domain being medical. In our study, we used two datasets. The medical corpus is one dataset that includes occurrences related to diseases. The other dataset contains tweets come from twitter. A language model and machine learning model are more suitable to this context. With the use of a deep learning model and natural language processing, we will be able to create a learning model to extract medical disease related information from Twitter. Some language and machine learning methods are better suited for this situation. Our mission is difficult because the majority of the data on Twitter is in a disorganized pattern.

1.2 Background

Named entity recognition (NER) makes it simple to recognize key aspects in a document, such as people's names, places, brands, monetary values, and so on. NER is often used in biomedical data to identify genes, DNA, and medication and disease names. Much of the research in biomedical informatics has centred on named entity recognition. Another way of detecting medical entities from a big medical corpus or medical-related content is Medical Entity Recognition(MNER)[3]. NLP has improved information extraction from big data sources in modern science, which would be more time demanding if done manually. Twitter is a social networking platform that has a wealth of information about medications, sports, entertainment, education, and other topics. Our goal is to extract medical disease information from tweets, which will be useful to both clinicians and medical researchers. They may be supplemented with fresh medical information by gathering medical data. MNER has the potential to make a difference in today's medical environment by making it easier to extract entities.

1.3 Motivation

Because of the large amount of data in Twitter, recognizing medical entities is a more difficult task. It will be time consuming and costly to manually find medical entities from a big volume of data. With the help of a learning model called Bidirectional LSTM, we were able to extract medical named items from Twitter in a suitable manner[18]. They extract entities from natural text or a corpus in certain entity extraction activity. We're inspired by such tasks, and we have accepted Twitter's challenges. In different works, they have proposed different learning model like Maximum Entropy model, Probabilistic model and deep learning model. We offered a deep learning model for our work due to its numerous benefits, and we also used a probabilistic model called the CRF model. Finally, we used the CRF layer in the deep learning model to obtain accurate entity recognition predictions. Without knowledge of NLP, entity recognition is impossible. The work of data pre-processing is critical before moving on to the training model. A model's result is improved through better data processing.

1.4 Problem Statement

Named Entity Recognition can automatically scan whole articles and identify the most important persons, organizations, and places mentioned. Knowing the necessary tags for each article aids in the automatic categorization of articles into established hierarchies and facilitates content discovery. The goal of Medical Entity Recognition (MER) is to recognize and classify text segments that span medical entity mentions. The entity types are predetermined and known. Drug, Disease, Symptom, Treatment, and Test are the entity types to be identified in this research. Traditional MERs (Medical Entity Recognition) are used on formal medical texts such published medical studies, research articles, clinical reports, discharge summaries, and EHRs (Electronic Health Records). We will analyze the performance of various existing MER's on Twitter data in this project and try to improve it.

1.5 Organization of Thesis

Chapter 2: It represents the literature review and describes some related works with approaches on Medical Named Entity Recognition (MNER).

Chapter 3: This chapter will present the proposed methodologies.

Chapter 4: This section will clarify the experimental result analysis and the model's performance.

Chapter 5: This chapter outlines the future scope of our work as well as concluding remarks on the method.

Chapter 2

Literature Review

2.1 Introduction

Natural language processing has had a significant impact on several academic topics in recent decades. Many research fields are enriched with natural language processing. The NLP adventure has a significant impact on big data research. Twitter is a social media platform with a significant amount of data that may also be considered a big data repository. With the use of natural language processing, we were able to extract necessity data from Twitter. With the use of a deep learning model and natural language processing, we will be able to create a learning model to extract medical disease related information from Twitter.

2.2 Introduction of NLP

Natural language processing (NLP) is the ability of a computer software to interpret spoken and written human language, often known as natural language. NLP is a part of Artificial Intelligence (AI).

Data is created in a variety of ways, including speech, photos, massages, and many others. The majority of data is accessed through text, which is relatively unstructured. Twitter data is likewise unstructured, lacking in grammatical structure and containing emoji and other unique symbols. These data structures can be handled by NLP's tools. NLP may now be used for data mining and information collecting in a wide range of real-world situations.

2.2.1 Applications of NLP

List of some applications of NLP are:

- Semantic analysis
- Machine translation
- Speech recognition
- Named Entity Recognition
- Question answering
- Web searching suggestion

- Spell correction
- Keyword extraction
- Sentiment analysis

2.2.2 Steps of NLP

There are two major components of NLP :

- Natural language understanding
- Natural language generation

Understanding natural language entails converting natural language input into meaningful representations. Examining numerous aspects of the language is a part of natural language generation. The following are the steps in natural language processing:

- Tokenization
- Stemming
- Lemmatization
- POS tagging
- Name Entity Recognition
- Chunking

There are general five steps of natural language processing:

The source code is scanned as a stream of characters and converted into intelligible lexeme in this phase. The entire text is divided into paragraphs, phrases, and words. Syntactic analysis is used to check grammar and word layouts, as well as to show the relationships between words. The representation of meaning is the focus of semantic analysis. The literal meaning of words, phrases, and sentences is the main focus. Discourse Integration is influenced by the sentences that come before it, as well as the meaning of the ones that come after it. The fifth and final phase of NLP is pragmatic. It uses a set of rules that characterize cooperative discussions to assist you in discovering the desired impact.

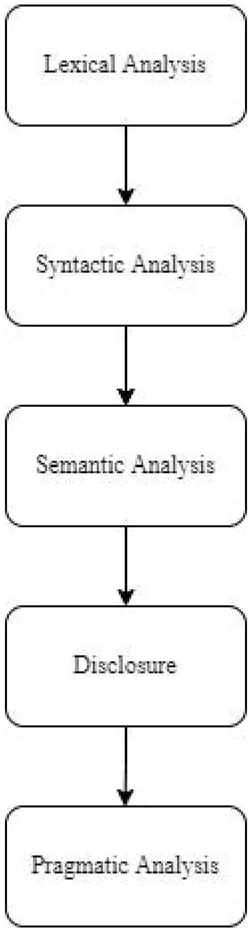

Figure 2.1 General Steps of NLP

2.3 Related Works

More work had been done in the past on medical text data to locate medical entities. In January 2005, a study on Medical Named Entity Recognition was released on the ResearchGate site. The GENIA corpus was used in their research. They worked with the GENIA 1.0 and GENIA 3.0 corpora. To complete their work, the Maximum Entropy (ME) model was proposed. POS tagging, tokenization, stemming, and other NLP procedures were used in their work. There were around 2000 dataset instances, with 80% of them being training and 20% being instances. Their model's F-1 score was approximately 68.2%[4]. Another research from the University of Oxford was released in 2017 and used a deep learning model to perform medical named entity recognition on a real-world challenge. They used the datasets i2b2 and ConLL which contain medical domains such as Medication names , Dosages, Modes, Frequencies, duration and reasons. They suggested the RNN and one-directional LSTM machine learning models. The experimental F-1 score was 69.3%[5].

Another paper has been published from university of California Berkeley, USA in 2017. They perform medical named entity recognition using the neural multi-task learning framework. They used the several datasets such as BC5CDR, NCBI disease corpus. They proposed machine learning model Bi-LSTM and CNN. The experimental F-1 score was 76%[3]. In 2020, a paper has been published from Coastal Highway university, Lewes, DE, USA 19958. They proposed datasets BC4CHEMD and JNLPBA which were medical corpus. They used the machine learning model CNN and Bi-LSTM. The experimental F-1 score was 84.26%[6]. In November 4-7, 2019, Coimbra, Portugal a paper has been published on medical named entity recognition. They used the GENIA corpus 3.0.2 as a dataset. They applied the CRF probabilistic model to learn the model. N-gram approach was also used to annotate their data in the corpus. They performed POS tagging and other NLP tasks before applying the learning model. The experimental of their work was about 90%[7]. In 2019, University of Toronto, Canada published a paper on medical named entity recognition in the biomedical domain. They proposed a learning method LSTM-CNN. They used the corpus such as BC4CHEMD, CRAFT, BC5CDR, BC2GM. The result was 47% on BC4CHEMD, 39.55% on BC5CDR, 71.81% on CRAFT[8]. In December 31, 2021 a paper has been published from Lanzhi Modern Service Industry Digital Engineering Technology Research Center, China. They used the medical datasets FSCBR and CMP to extract the medical entity. The F-1 score of the model was about 89.10% on the FSCBR dataset and 72.35 % on the CMP dataset[9]. A paper had been published in Research gate from university of California 2021 on medical text. They proposed the datasets BC2GM, BC4CHEMD, BC5CDR of medical information. They worked on existing paper and calculated model performance by increasing number of datasets. The STM (single task model) and Best-MO-MTM models were used. The model performance was 81.55% for STM and 81.68% for Best-MO-MTM[10] . In 7 March 2022, a paper had been published from College of Electronic information, Qiangdao University, China on the medical text to extract medical entity. They used the medical corpus GENIA 3.0.1 and GENIA 2.0. They proposed the machine learning model Maximum Entropy (ME) model to complete the task. They performed the NLP steps such as Tagging of each word on medical corpus. The model performance was near about 79.80%[11]. Another paper had been published from School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Korea. They worked on Biomedical fields with the help of deep learning model. They proposed model CLSTM and Bi-LSTM with CRF. They used several medical corpus such as NCBI, GM, CDR. The model performance for Bi-LSTM was 78.91% and 82.19% for Bi-LSTM with CRF[12]. In our research, we have worked with CADEC medical corpus and twitter datasets. The corpus was annotated with medical disease related information. The twitter dataset was manually tagged like B-drug, I-drug, B-disease, I-disease. We proposed and Bi-LSTM-CRF learning model. The model accuracy was about 86.24%.

Chapter 3

Proposed Methodology

3.1 Introduction

We proposed a Bi-directional LSTM neural network model with a Conditional Random Field (CRF) probabilistic model in our research. Now we shall go through the fundamentals of neural networks and probabilistic models. Then we shall go on to our proposed model and method for solving the Medical Name Entity Recognition (MNER) problem.

3.2 Neural Networks

Neural networks are computer systems made up of interconnected nodes that function similarly to neurons in the brain. They can discover hidden patterns and correlations in raw data using algorithms, cluster and categorize it, and learn and improve over time. In real-world circumstances, neural networks are also well-suited to assisting humans in solving complex challenges. Nodes in neural network assume similar to human neurons.

3.2.1 Components of Neural Network

There are three basic components of neural network given below:

- Input layer
- Processing layer or Hidden layer
- Output layer

Input layer

Artificial input neurons make up the input layer of a neural network, which brings in the first data to be processed by future layers of artificial neurons. Each node accepts input such as x1, x2, x3, and so on. Each node has a bias value associated with it.The input layer is the first step in the artificial neural network's workflow.

Hidden Layer

This layer is sometimes called processing layer. Hidden layers are necessary in artificial neural networks if and only if non-linear data separation is required. Because it updates its weights and bias value to make correct boundary decisions, the hidden layer is referred to as the processing layer. It's also known as a black box. Hidden layer is the intermediate layer of input and output layer. Between input neurons and output neurons is the number of hidden nodes in the hidden layer.

Output layer

In an artificial neural network, the output layer is the final layer of neurons that generates the program's outputs. The feedback is needed or not depends on the output value of this layer. If the desired output value is not achieved, back-propagation (BP) is used to update the bias and weight of the nodes.

3.2.2 Some Key Terms of Neural Network

Nodes: Neural network nodes resemble human neurons. The nodes are identified by drawing a circle.

Weight value: The strength of the relationship between units is represented by a weight. The importance of an input value is reduced by a weight.

Bias value: The bias value is used to shift the activation function to the left or right in order to better suit the data.

Activation function: In a neural network, an activation function specifies how the weighted sum of the input is turned into an output from a node or nodes in a layer. Some activation functions are given below:

- Linear activation function
- Sigmoid activation function
- Tanh activation function (Hyperbolic tangent)
- Binary step activation function etc.

Sigmoid and Tanh functions will be used as activation functions in nodes in our LSTM investigation.

$$f(x) = \frac{1}{1+e^{-x}}$$

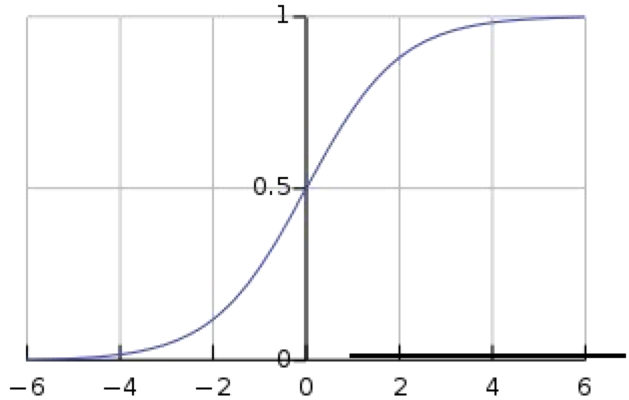

Figure 3.1 Sigmoid Function

3.2.3 Types of Neural Network

There are three types of neural network

- Artificial Neural Network (ANN)
- Convolutional Neural Network (CNN)
- Recurrent Neural Network (RNN)

Artificial Neural Network (ANN) An artificial neural network (ANN) is a computer model made up of many processing components that accept inputs and outputs based on their activation functions. Artificial Neural Networks function in a similar fashion to their biological counterparts. They can be thought of as weighted directed networks, with neurons acting as nodes and connections between neurons acting as weighted edges.The signatures are verified using Artificial Neural Networks.

The inputs are X1, X2, X3; the weights are W1, W2, …, Wi; the bias values are B1, B2, ..., Bi; and the outputs are Y1, Y2, Y3 in the upward figure.

The output equation is : $y_i = w^T x_i + b_i$; Where, w^T= weight vector, b_i= bias vector, x_i= input vector, y_i= output vector.

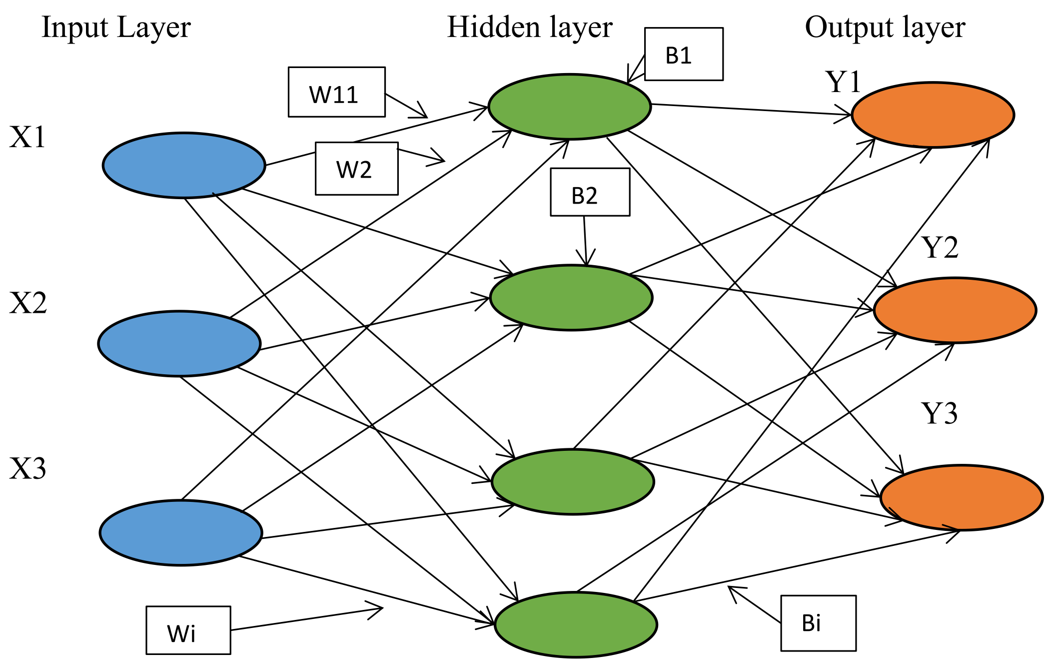

Figure 3.2 ANN Architecture

Convolutional Neural Network (CNN)

A Convolutional Neural Network is a Deep Learning system that can take in an image as input, assign priority to distinct items in the image, and distinguish between them. It basically performs on image data. The convolutional layer, a type of layer found in convolutional neural networks, is what gives them their strength. A convolutional layer, a pooling layer, and a fully connected layer are the three layers that make up a CNN. Facial recognition is the major application of CNN. The architecture of a ConvNet is inspired by the organization of the Visual Cortex and is akin to the connectivity pattern of Neurons in the Human Brain. Individual neurons can only respond to stimuli in a small area of the visual field called the Receptive Field. A number of similar fields can be stacked on top of each other to span the full visual field.

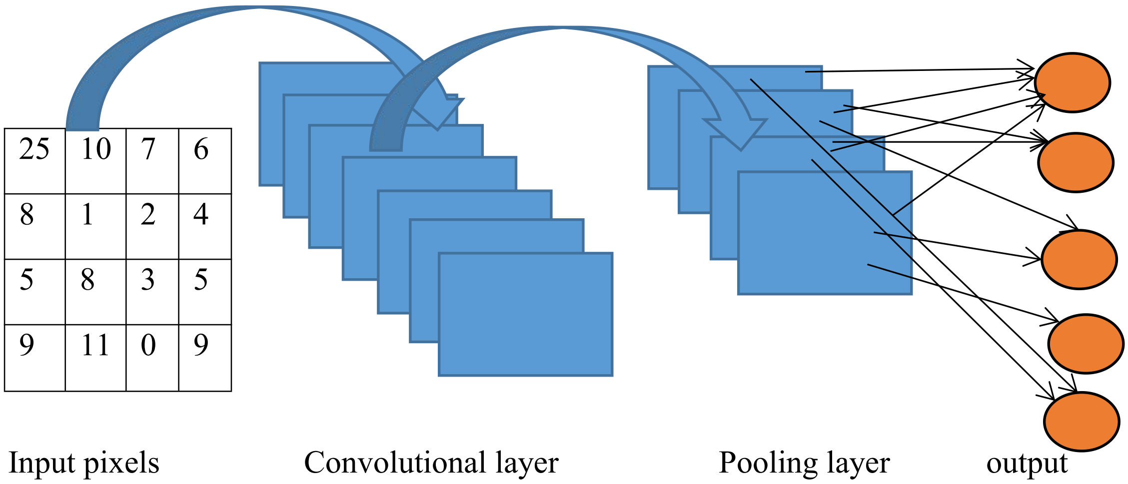

Figure 3.3 CNN Architecture

Recurrent Neural Network (RNN) A recurrent neural network (RNN) is a form of artificial neural network that works with time series or sequential data. The output from the previous stage is provided as input to the current step in the RNN. Because the order of data in a time series is more essential. All of the inputs and outputs in standard neural networks are independent of one another, however in some circumstances, such as when predicting the next word of a phrase, the prior words are necessary, and so the previous words must be remembered. RNN has a limited memory capacity and can only recall a small number of prior words. In our thesis, we presented an LSTM network to overcome this constraint. LSTM is the update version of RNN network.

Current state calculation formula of RNN is, $h_t = f(h_{t-1}, x_t)$ where h_t= current state, h_{t-1}= previous state, x_t= input state.

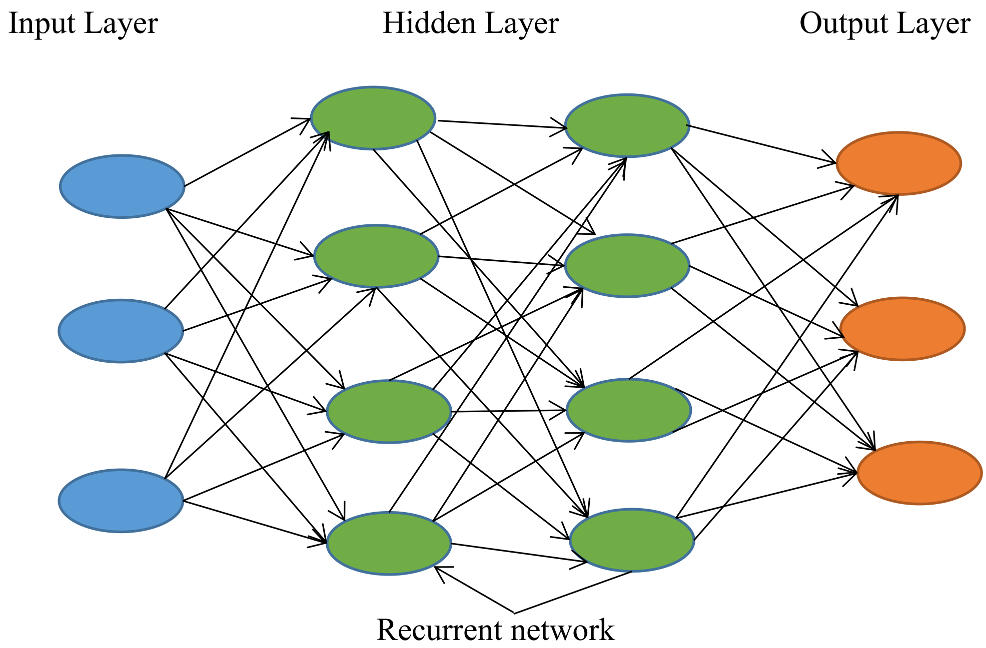

Figure 3.4 RNN Architecture

3.3 Long Short Term Memory (LSTM)

Long short-term memory (LSTM) is a deep learning architecture that uses an artificial recurrent neural network (RNN).Aside from that, LSTM is an updated version of RNN that has solved RNN's memory shortage problem. For time series or sequence data, LSTM is used. For time series data, the most critical issue is ordering. Memory cells and gate units are contained in the (completely) self-connected hidden layer. An LSTM layer is made up of memory blocks, which are recurrently connected blocks. An input layer, a single hidden layer, and a conventional feed-forward output layer make up a standard LSTM network.

3.3.1 Fundamental Components of LSTM Cell

LSTM is composed of components are:

- Input gate
- Forget gate
- Output gate

Input gate The input gate is in charge of updating the cell state with new information. Using a sigmoid function to control what values need to be added to the cell state. Creating a vector that contains all potential values for addition.

These potential values are added using the hyperbolic function, which has a range of values from -1 to 1. Sigmoid functional values are multiplied by tanh functional values.

Forget gate A forget gate is in charge of erasing data from the cell state. By multiplying a filter, information that is no longer required for the LSTM to understand things or that is of lesser value is removed. This is essential for the LSTM network's performance to be optimized.

Output gate The output gate is responsible for picking valuable information from the current cell state and displaying it as an output. After applying the tanh function to the cell state, the values are scaled to the range -1 to +1, resulting in a vector.

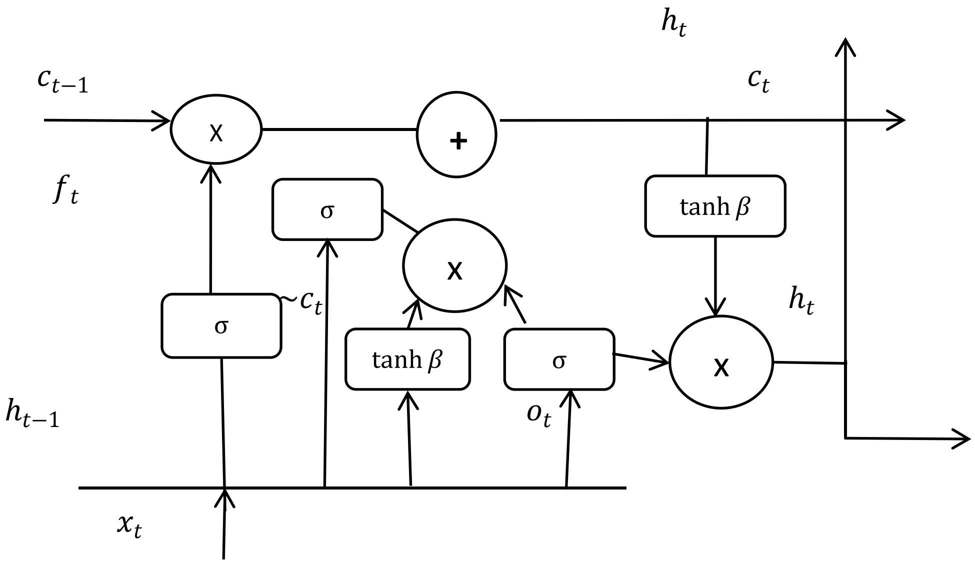

Figure 3.5 LSTM Architecture

A graphical illustration of an LSTM unit is shown in Figure 3.5. The LSTM unit contains numerous neural network layers, each of which is labeled with its activation function (σ ; tanh β). The pointwise operations (addition, multiplication) are represented by their proper mathematical symbol, whereas the connection is represented by arrows. A LSTM cell has two outputs: one h_t to the layer ahead of it, and another h_t and c_t to the next LSTM unit in temporal space t.

Notation about the upward figure:

x_t is means initial input, f_t denotes forget gate, o_t denotes output gate, i_t denotes input gate, f_t forget gate, c_t cell state.

LSTM working steps

Step1: The input data x_t goes through the forget gate f_t which decide whether to keep (f_t=1) or forget (f_t=0) as c_{t-1} as it multiplied with the output of f_t.

Step2: Input x_t is also passed to the sigmoid i_t gate and $\tanh \boldsymbol{\beta}$, $\sim c_t$ gate , where the sigmoid function decide whether to remember (i_t=1) or forget (i_t=0) the weight updates from $\sim c_t$. The long term memory c_t then,

c_t= f_t*c_{t-1}+i_t*c_t (3.1)

Step3: Lastly the input x_t is passed to the sigmoid gate o_t that decides whether to remember (o_t=1) or forget (o_t=0) the input, then multiplies
with the $\tanh \boldsymbol{\beta}$ gate. The output h_t of this multiplication is $h_t = o_t * \tanh(c_t)$ which is pass through as a final output of the cell and as input of the next LSTM cell. The formulas to update an LSTM unit at time t are formalized as:

i(t) = σ(w(i) [x(t); h(t-1)]+b(i)) (3.2)

f(t) = σ(w(f) [x(t); h(t-1)]+b(f)) (3.3)

o(t) = σ(w(o) [x(t); h(t-1)]+b(o)) (3.4)

c(t) = f(t)*c(t-1)+i(t)*c(t) (3.5)

h(t)= o(t)*tanh(c(t-1)) (3.6)

3.4 Bi-directional LSTM

The method of making any neural network include sequence information in both ways backwards (future to past) and forwards (past to future) is known as bidirectional long-short term memory (past to future). However, with bi-directional input, we can have the information flow in both directions, preserving both the future and the past. The prediction is much faster in the Bi-LSTM than in the one-directional LSTM. For the efficiency of our job, we offered a Bi-LSTM in our case. Forward and backward LSTM outputs are coupled to the same output layer in the Bi-LSTM. Bi-LSTM is made up of two directional LSTM's in two distinct RNN's, but both of its outputs are coupled to the same output layer. So, this is a means to know what will happen in the future and what happened in the past at a specific point in time.

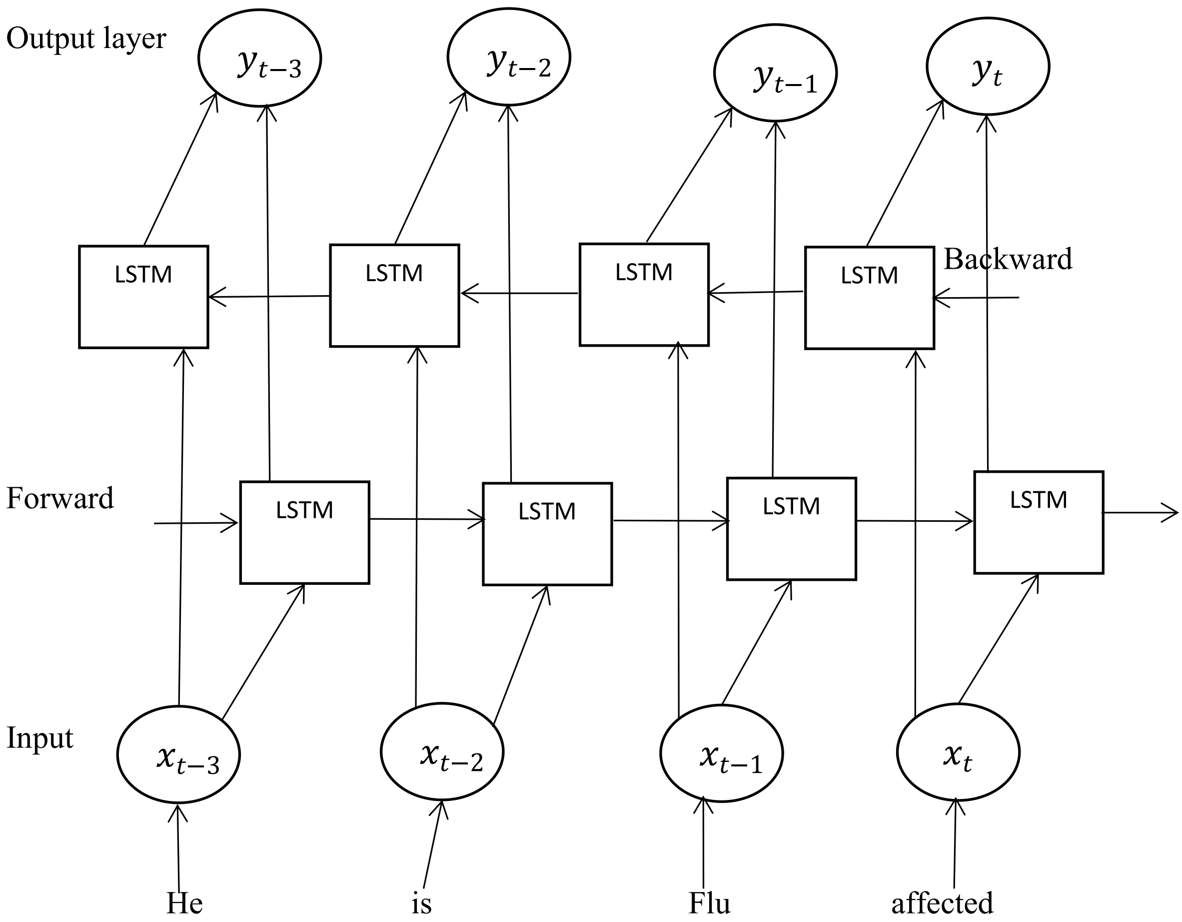

Figure 3.6 Bi-LSTM Architecture

In Bi-LSTM forward LSTM starts the perceive the in “He” at the same time stamp backward LSTM starts the perceive input “affected”. In the fashion, when forward LSTM perceive the input “is”, already it know the next word “Flu”. At the similar way backward LSTM already know the next word “is”. So, in this way Bi- LSTM predict the next word or word sequence.

3.5 Datasets

A dataset is a collection of instances that all have the same feature. In machine learning, a dataset is essentially a collection of data pieces that can be analyzed and predicted by a computer as a single entity. This means that the data gathered should be homogeneous and understandable to a machine that does not see data in the same manner that people do. Data is arranged in datasets so that computers can process it quickly. It should be emphasized that in the case of machine learning, data must originate from the same distribution. Data analysis is critical in research because it simplifies and improves data processing. It enables researchers to evaluate data in an easy manner, ensuring that nothing is overlooked that could aid in the discovery of new information. Every research effort including (measurement) data sets should consider the data availability in the short, medium, and long term. Data is organized in datasets in a structured way that machines can easily interpret. When a model is learnt using datasets that come from the same domain, the following prediction is considerably more accurate. The importance of datasets in machine learning models can be summarized in these terms.

3.5.1 Properties of Datasets

Some properties of datasets are listed below:

- Dimension of datasets
- Sparsity
- Resolution
- Center of data
- Skewness of data
- Spread among the data members
- Presence of Outlier
- Co-relation among the data

3.5.2 Datasets in Our Research Work

In our thesis, we study data from the social media site Twitter, with the data domain being medical. In our study, we used two datasets.The medical corpus is one dataset that includes occurrences related to diseases. The other dataset contains tweets come from twitter. One dataset contains 122581 instances and other dataset contains 12426 instances. Now we will describe our datasets used in our research.

3.5.3 CADEC Corpus

The CSIRO Adverse Drug Event Corpus (Cadec) is a large annotated corpus of patient-reported Adverse Drug Events from medical forums (ADEs). The disease, drug, symptoms, treatment, and test are all included in the CADEC dataset. In the CADEC dataset , there exits four columns. First column name is Document which include author. The second column contains the sentence number, which is the number of each tweet to be posted. Third column is text which is our main focus. Fourth column is tagging which helps to tag each word. Tagging is done in datasets in the following way: if any disease, drug, symptom, treatment, or test is discovered in the beginning or end of a sentence, it is tagged with B-diseases, B-drug, B-symptoms, B-treatment, or B-test. If medical terms appear within a sentence, they are labeled with I-disease, I-symptom, I-treatment, I-test, and I-drug. If a word is not found in medical terms, it is labeled with the letter O, which signifies it is out of context.

Examples:

He	is	affected	by	Covid19
O	O	O	O	B-disease

Flu	leads	to	the	fever	and	aspirin	is	needed.
B-disease	O	O	O	I-disease	O	I-drug	O	O

Table 3.1 Small Portion of CADEC Dataset

Document ID	Sentence No	Word	Tag
ARTHROTEC.12	97	I	O
ARTHROTEC.12	97	was	O
ARTHROTEC.12	97	severly	B-Symptom
ARTHROTEC.12	97	fever	I-disease
ARTHROTEC.12	97	and	O
ARTHROTEC.12	97	could	O
ARTHROTEC.12	97	only	O
ARTHROTEC.12	97	walk	O
ARTHROTEC.12	97	less	O
ARTHROTEC.12	97	than	O
ARTHROTEC.12	97	100	O
ARTHROTEC.12	97	meters	O

3.5.4 Twitter Dataset

The information gathered by the user, the access point, the content of the post, and how users view or utilize your post is referred to as Twitter data. Because tweets are posted by individuals, the data on Twitter is typically less ordered than formal text. In our work, we have used the twitter data from the exiting datasets. There are about 12226 instances in our twitter dataset. Normal text, emoji, and other characters are all included in the Twitter data. Our research focuses on text data in order to identify medical entities. As a result, the data cleaning process was completed in order to remove the less informative indicator. This may occur with high precision, and we will have to deal with less noisy data. Then the each word was tagged manually that is discussed in section 3.1. Now some portion of twitter data is show below:

Table 3.2 Small Portion of Twitter Dataset

Sentence number	Word	Tag
100	Joint	B-disease
100	pain	I-symptom
100	is	O
100	a	O
100	symptom	O
100	of	O
100	Osteoarthristis	B-disease

3.5.5 Classes in Datasets

In our twitter dataset, there are three columns. Sentence number is in the first column, word is in the second, and tag is in the third. Our goal is to determine whether or not any word belongs in a medical category. So, in our dataset “Tag” is the class label and there are three main category in that class B-disease, I-disease and O that means outside the medical entity. In the CADEC dataset, there are several category like B-drug, I-drug, B-symptom, I-symptom, B-ADR (adverse drug reaction), I-ADR and so more.

CADEC Corpus

Class name = Tag

Categories are B-ADR, I-ADR, B-drug, I-drug, I-ADR, B-symptom, I-symptom, B-disease, I-disease.

Twitter Dataset

Class name = Tag

Categories are B-drug, I-drug, B-symptom, I-symptom, B-disease, I-disease.

3.6 Flow Chart of Methodology

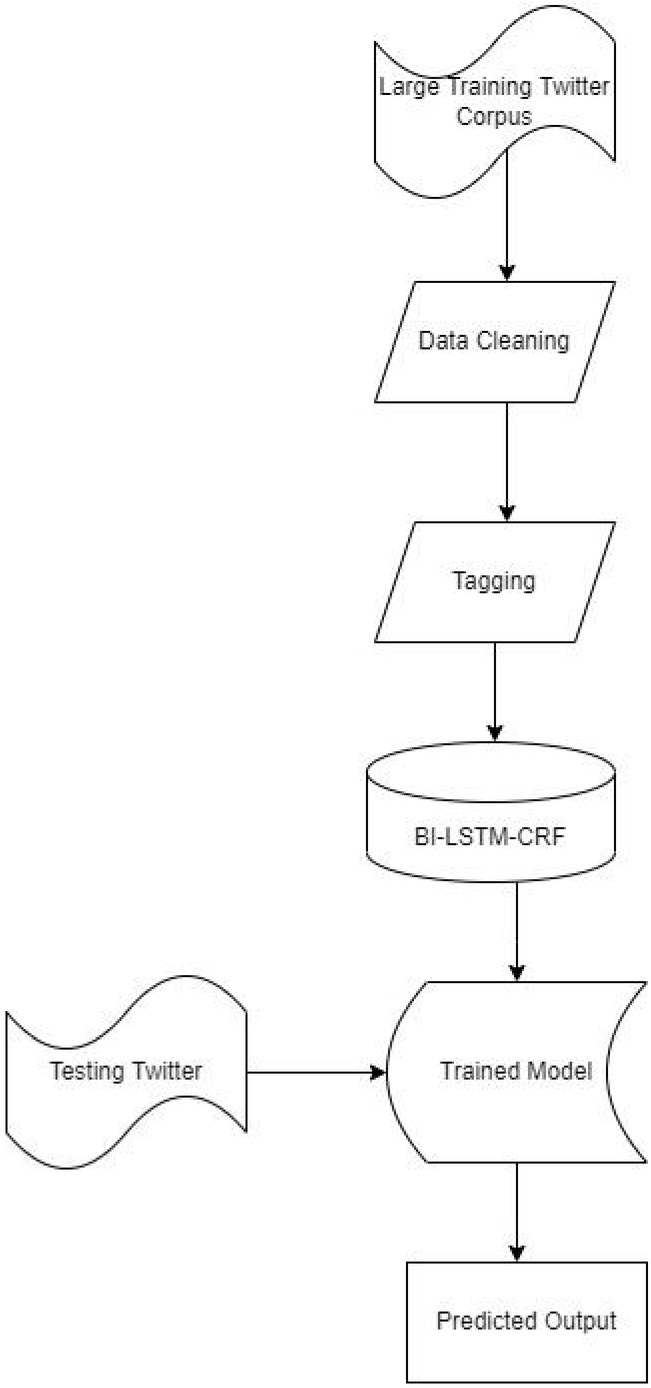

Figure 3.7 System Flowchart

3.6.1 Data Cleaning

Data cleaning is the process of removing or modifying data that is incorrect, incomplete, irrelevant, duplicated, or incorrectly formatted in order to prepare it for analysis. In our task, twitter data contains emoji and other special characters those are not our task context.So, by cleaning those data from the twitter dataset.

3.6.2 Tagging

Tagging is done in datasets in the following way: if any disease, drug, symptom, treatment, or test is discovered in the beginning or end of a sentence, it is tagged with B-diseases, B-drug, B-symptoms, I-disease, I-drug. Then next step is to train model with the prepared dataset.

3.7 Problem Solved with Bi- LSTM

Now, we will solve medical disease entity extraction with Bi- LSTM. Let two examples: "He is affected by covid19 virus and Flu"

" He is unwilling to buy a virus affected computer"

Through the first sentence, we will train a model and then using the second sentence we will test the trained model. Our target is to show how machine well treat the word "virus".

3.7.1 Training Step:

We know that neural network does work with numerical value but our training input is string. So, we need to convert each word with numerical representation. Assume the first sentence is strongly medical related where medical entities are covid19, virus and Flu.There are many technique to convert word to numerical representation. One hot encoding,

word embedding are the most popular technique. In our study, we will use both technique. One hot encoding will be used for rag encoding of each word. Word embedding technique will be used to convert word to numerical representation.

He	is	affected	by	covid19	virus	and	Flu
↓	↓	↓	↓	↓	↓	↓	↓
O	O	O	O	B-disease	I-disease	O	B-disease

Tag: O means outside of medical entity.

Tag: B-disease means first or last medical entity in the sentence.

Tag: I-disease means medical entity is in the inner of the sentence.

One hot encoding for tags

Table 3.3 One hot coding

	O	B-disease	I-disease
O	1	0	0
B-disease	0	1	0
I-disease	0	0	1

Word to vector

Each word is convert the numerical vector. This vector will be helpful in the nodes of the neural network. At each neuron embedding vector size will help the calculation.

0	0	0	0	1	1	0	1
He	is	affected	by	covid19	virus	and	Flu.

Then the vector will passed through the nodes or neurons and embedding vector will be calculated to each nodes. Embedding vector size of the network will be the number of neurons. In LSTM cell, the input in the previous output will be feedback to the current nodes. Based on the values embedding it will decide which term should be forget or not.

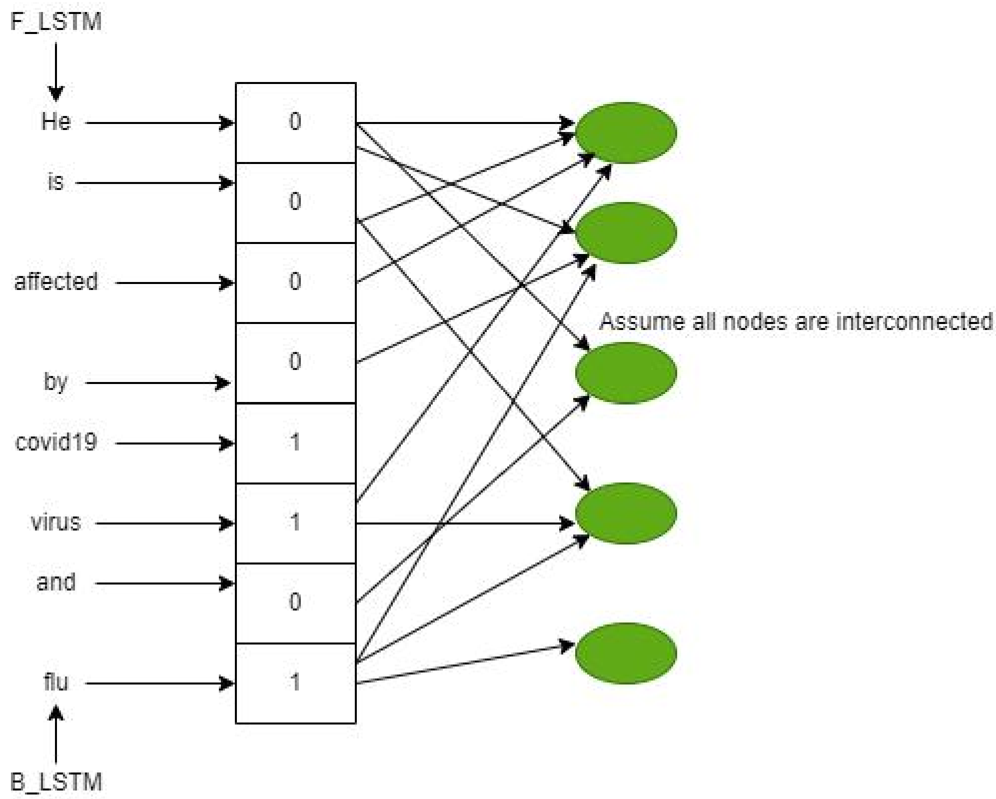

Figure 3.8 Training Process of Bi- LSTM

Forward LSTM starts to perceive input from left side to right side at the same time Backward LSTM will start to perceive the input from right to left. They produce the output. For every time stamp previous output will be feedback to current output node. Then it will update it's cell state value using the bias value and weight value. For forward LSTM when it perceive input "is" the previous word "he" will be feedback to the current state.Then forward LSTM will be hold the value for "He is".This process will go over in the similar fashion. At the same time stamp, the backward LSTM will hold the value for "Flu and". After some time stamp, when forward LSTM holds value for "He is affected by covid19" , it already know the next word is "virus". Let, after complete the training it will produce the output vector corresponding sentence.

He	is	affected	by	covid19	virus	and	Flu
0.01	0.01	0.04	0.02	0.7	0.9	0.01	0.8

3.7.2 Testing Step

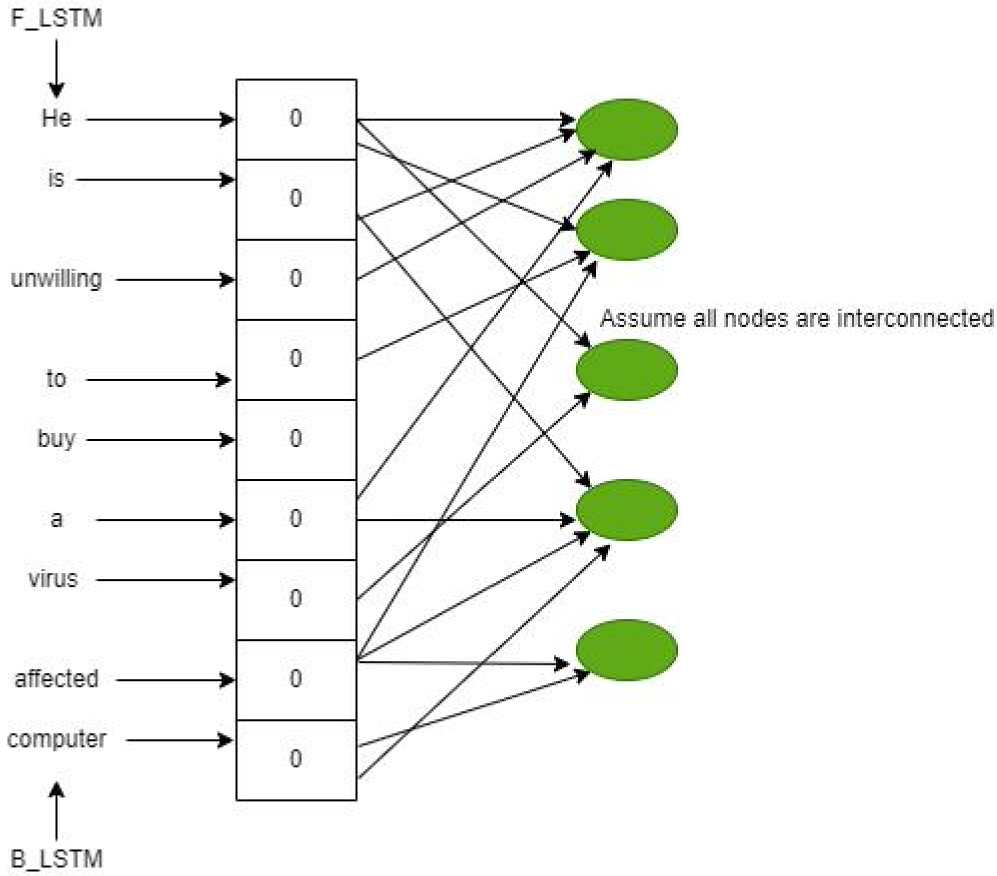

Figure 3.9 Testing Process of Bi- LSTM

The testing sentence was “He is unwilling to buy a virus affected virus”. In forward LSTM, it starts to perceive input from left to right and at the same time stamp right LSTM starts to perceive input from left to right. After complete the operation it produces a output vector like this:

He	is	unwilling	to	buy	a	virus	affected	computer
0.01	0.01	0.01	0.02	0.02	0.03	0.2	0.04	0.01

The word “virus” includes both training and testing sentence. When testing is starts, it will know the before and after word value. It will notice that the value of next word and before word value is less than the actual training value. So, it will decide that word “virus” in the second sentence may not medical entity.This is the way of training and testing using Bi- LSTM.

3.8 Bi-LSTM-CRF

In the application of Bi-LSTM, the entity recognition problem is solved but to get more accuracy another method is combined with the this model. The CRF hidden layer is added to the Bi-LSTM model. In training time, it learns more rather than Bi-LSTM. Bidirectional CRF provides the better output value of a node using the probabilistic method. The linear chain of CRF distribution p(y | x) is given by:

$$P(y \mid x) = \frac{1}{Z(x)} \prod P(y\,, x)$$

Where $Z(x)$ is the normalized function.

$$Z(x) = \sum_c \prod P(y, x)$$

The CRF layer learns only the transition probability of the output labels because the hidden layer on top of the Bi-LSTM provides the score matrix P for a given sequence. The LSTM hidden layer outputs are also passed to the following Bi-directional CRF layer in the Bi-LSTM and CRF architecture. On POS, chunking, and NER data sets, the Bi- LSTM-CRF model can deliver state-of-the-art (or close to it) accuracy. Furthermore, when compared to earlier observations, it is more resilient and less reliant on word embedding.

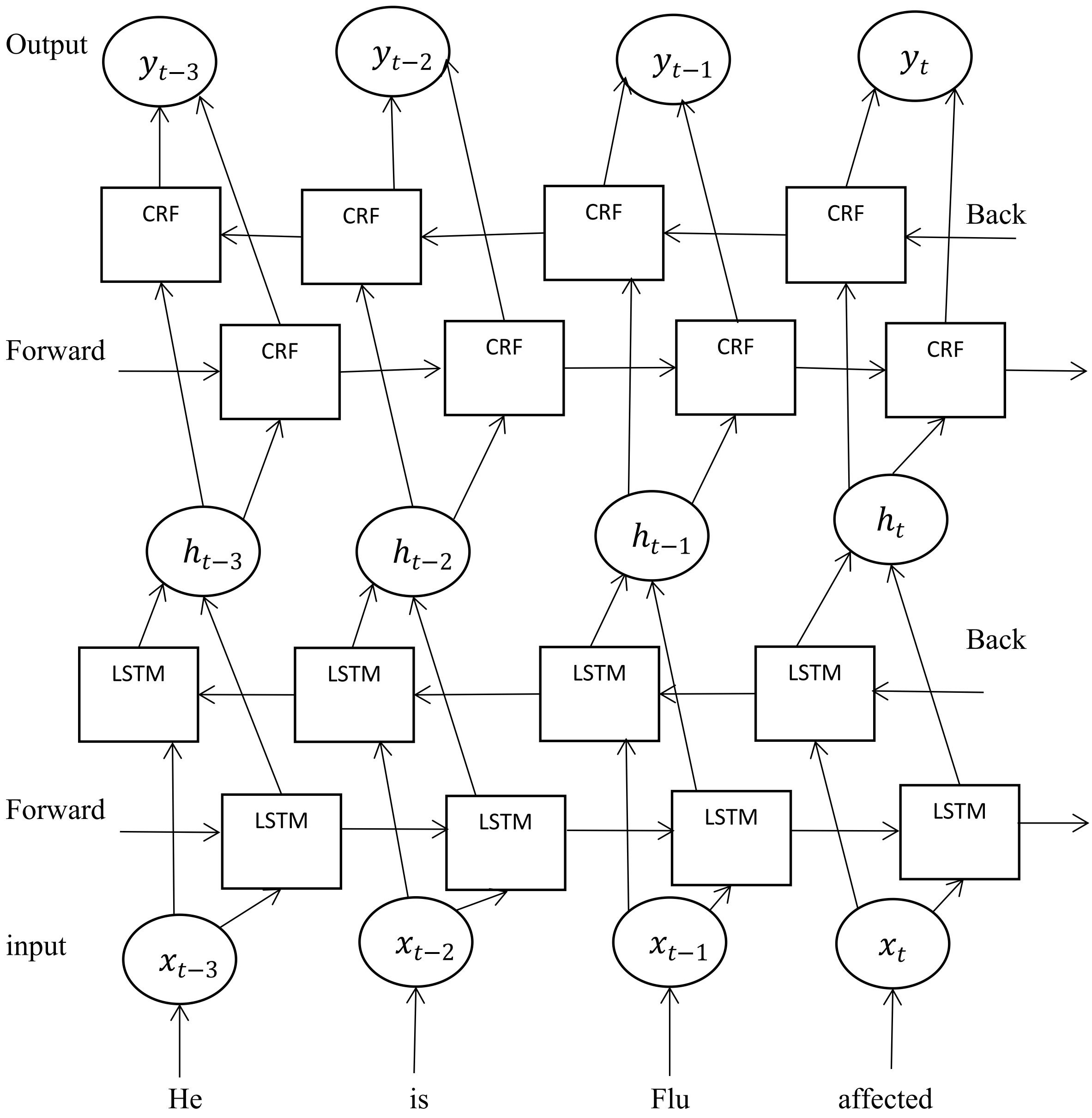

Figure 3.10 Bi-LSTM-CRF Architecture

Chapter 4

Experimental Results and Analysis

4.1 Introduction

The outcome of our proposed model will be discussed in this section. Aside from that, the experimental setup of our research will be explained. Here's a quick rundown of the datasets we used in our research. This chapter will also include a comparison of the results. Graphical representation of our model accuracy and error will be shown here. The carrying out, execution, or practice of a plan, a method, or any concept, idea, model, specification, standard, or policy for accomplishing something is referred to as implementation. As a result, implementation is the activity that must come after any preliminary thinking if something is to happen.

.

4.2 Experimental Setup

Framework TensorFlow is a machine learning and high-performance numerical computation framework developed by Google. TensorFlow is a Python library that creates and executes dataflow graphs using C++. It supports a wide range of classification and regression algorithms, as well as deep learning and neural networks in general. In our work, we used some necessary package which provided by TensorFlow. TensorFlow two packages are available tensorflow and tf-nightly both are support in Obuntu and Windows. To perform out program, we selected python programming language. Keras is a high-level neural network library that runs on top of TensorFlow. The 34.18 GB memory is used for completing our task.

Datasets To perform our task, we proposed two datasets. One is CADEC medical disease-related corpus and another is medical twitter dataset. In the twitter dataset, there was about 521 tweets including 1720 sentences and 12426 labeled words. In the CADEC dataset, there was 7771 sentences with 122581 labeled words. For the better training purpose, we added some tweets with CADEC corpus. During prepared a model, we split our dataset 80% and 20% for training and testing.

Learning rate, iteration, loss function The learning rate is a tuning parameter in an optimization algorithm that determines the step size at each iteration as the algorithm moves toward the loss function's minimum. In our task, we used the default learning rate 0.01. We proposed a batch size 512 and number of epochs was 200. We had changed sometimes number of epochs and batch size to better train. The difference between the expected output and the outcome delivered by the machine learning model is quantified by the loss function in a neural network. The loss function graph is shown in figure 4.1 for our model.

4.3 Data Collection Process

In our work, the exiting twitter data is used. Twitter data come from the social media. After getting the tweets, we had performed pre-processing on tweets data. First of all, we had clean some useless character or sign or emoji which are not context in our work. We also used the existing medical corpus named CADEC. We manually tagged the each word of tweets already discussed in Chapter 3. Here some data are given below:

Table 4.1 CADEC Corpus

Document ID	Sentence No	Word	Tag
ARTHROTEC.12	97	I	O
ARTHROTEC.12	97	was	O
ARTHROTEC.12	97	severly	B-Symptom
ARTHROTEC.12	97	fever	I-disease
ARTHROTEC.12	97	and	O
ARTHROTEC.12	97	could	O
ARTHROTEC.12	97	only	O
ARTHROTEC.12	97	walk	O

Table 4.2 Twitter Dataset

Sentence number	Word	Tag
100	Joint	B-disease
100	pain	I-symptom
100	is	O
100	a	O
100	symptom	O
100	of	O
100	Osteoarthristis	B-disease

4.4 Result of Bi-LSTM-CRF Model

In our work, number of class labels are eleven. The labels are: I-Drug, B-Drug, I-Symptom, B-Symptom, I-Disease, B-Disease, O (Outside of medical word), B-ADR, I-ADR (Adverse Disease Reaction), B-Finding, I-Finding. When disease or drug or symptom is found in inside the sentence it then labeled by I-Disease or I-Drug, I-Symptom. When it is found in starting or ending of sentence it then labeled by B-Disease or B-Drug, B-Symptom. In the Bi-LSTM-CRF model additional bidirectional CRF layer is added. So, this model provide better result than existing ME model (Maximum Entropy Model). Same dataset was run on Bi-LSTM-CRF model. It provides accuracy 62.70%.

Confusion Matrix

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. For binary class confusion matrix calculation is easy but slightly difficult for multiple class.

Table 4.3 Confusion Matrix for Binary classification

Predicted Class →

Actual Class ↓

Actual \ Predicted	YES	NO
YES	TP	FN
NO	FP	TN

The precision, recall, F-1 score can be easily calculated by using confusion matrix. It is also said that confusion matrix is model performance matrix. For table 4.3, TP denotes the total number of positive count which model can predicted as positive. FN denotes false negative that means the actual class was positive but model predicted it as a negative. FP denotes false positive that means the actual class was negative but model predicted it as a positive. TN denotes the total number of negative counts which model can predicted as negative.

Precision Precision is defined as the number of true positives over the number of true positives plus the number of false positives.

Precision = $\frac{TP}{TP+FP}$

Recall Recall is defined as the number of true positives over the number of true positives plus the number of false negatives.

Recall = $\frac{TP}{TP+FN}$

F-1 Score The F1-score combines the precision and recall of a classifier into a single metric by taking their harmonic mean. It is primarily used to compare the performance of two classifiers. Suppose that classifier A has a higher recall, and classifier B has higher precision.

F-1 score = $\frac{2*Precision*Recall}{Precision+Recall}$

Accuracy Accuracy is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

Accuracy = Correct predictions / Total predictions.

Accuracy = $\frac{TP+TN}{TP+FP+TN+FN}$

For our proposed model Bi-LSTM-CRF, the confusion matrix is given below:

Let, some notations for table:

1. I-Drug is denoted by I-Dr
2. B-Drug is denoted by B-Dr
3. I-Symptom is denoted by I-Sym
4. B-Symptom is denoted B-Sym
5. I-ADR is denoted by I-A
6. B-ADR is denoted by B-A
7. I-Finding is denoted by I-Fd
8. B-Finding is denoted B-Fd
9. I-Disease is denoted by I-Ds
10. B-Disease is denoted by B-Ds
11. Outside medical word is denoted by O

Table 4.4 Confusion Matrix of Bi-LSTM-CRF

	B-A	B-Dr	B-Sm	I-A	I-Dr	I-Sm	B-Fd	I-Fd	B-Ds	O	I-Ds
B-A	0	11	24	0	5	0	1	0	0	7	251
B-Dr	0	3	2	0	2	0	0	0	0	3	113
B-Sm	0	3	13	0	3	0	0	0	0	4	122
I-A	0	0	2	7	1	0	0	0	0	1	30
I-Dr	0	1	6	0	3	0	0	0	0	3	64
I-Sm	0	10	35	0	37	0	10	19	0	28	684
B-Fd	0	0	8	0	7	0	2	0	0	7	117
I-Fd	0	0	2	0	2	0	0	0	0	0	40
B-Ds	0	0	3	0	0	0	0	0	0	1	32
O	0	0	6	0	9	0	0	2	0	216	172
I-Ds	0	24	208	0	162	0	25	11	0	130	3957

According to confusion matrix, there are eleven categories of classes. The testing accuracy for our model was about 62.70%. The difference between the expected output and the outcome delivered by the machine learning model is quantified by the loss function in a neural network. The training loss function graph is figure out below:

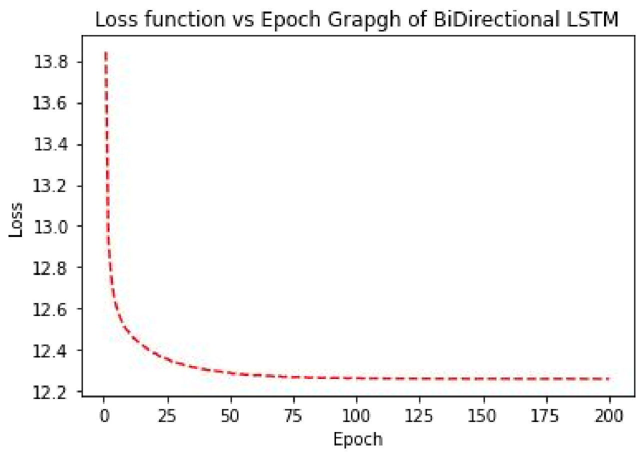

Figure 4.1 Loss Function Graph for Bi-LSTM-CRF

Accuracy = $\frac{3984}{6680} = 0.6270 \; = \; 62.70\%$

Precision, Recall, F-1 Score for ADR (Adverse disease reaction)

Precision = $\frac{7}{7} = 1.00$

Recall = $\frac{7}{333} = 0.02$

F-1 Score = $\frac{2*1.00*0.02}{1.02} = 0.03$

Precision, Recall, F-1 Score for Disease

Precision = $\frac{3957}{5582} = 0.71$

Recall = $\frac{3957}{4877} = 0.81$

F-1 Score = $\frac{2*0.71*0.81}{1.52} = 0.75$

Precision, Recall, F-1 Score for Symptom

Precision = $\frac{13}{309} = 0.04$

Recall = $\frac{13}{968} = 0.01$

F-1 Score = $\frac{2*0.04*0.01}{0.05} = 0.02$

Precision, Recall, F-1 Score for Finding

Precision = $\frac{2}{70} = 0.03$

Recall = $\frac{2}{186} = 0.01$

F-1 Score = $\frac{2*0.03*0.01}{0.04} = 0.02$

Precision, Recall, F-1 Score for Outside

Precision = $\frac{216}{400} = 0.54$

Recall = $\frac{216}{405} = 0.53$

F-1 Score = $\frac{2*0.54*0.53}{1.07} = 0.53$

In our model, the F-1 score of disease and outside medical entities are larger than the other categories.

4.5 Comparison Between Bi- LSTM -CRF model and ME (Maximum Entropy model) Existing Model

In the existing model ME (Maximum Entropy Model), they proposed only twitter dataset for training and testing on that model. But in our study, we added CADEC corpus with the same twitter dataset to train and test our model. In our work, we used bidirectional LSTM-CRF model and this model accuracy was near about 62.70%. But the existing model the accuracy was about 72.31% which was high accuracy rather than our model because they proposed five types of categories but in our work there are eleven number of categories. The maximum entropy model works for times series data with random forest algorithm. ME works with probability of words as input goes with randomness. But Bi-LSTM works better than ME because it works fast due to bidirectional input sequencing. Bi-LSTM-CRF is model with CRF hidden layer that provides more accurate predictions because of discrimination property.

Table 4.5 Comparison between Bi-LSTM-CRF and ME Model

Bi-LSTM-CRF Model	ME (Maximum Entropy Model)
1. Used datasets are twitter dataset and CADEC corpus.	1. Used datasets was BC5CDR-disease and twitter data.
2. Testing accuracy of model was 62.70% due to increasing number of classes than ME model.	2. Testing accuracy was 72.31% with five number of classes.
3. Proposed a deep learning model with faster calculation.	3.Proposed model was probabilistic based.
4. Number of dataset's instances was about 128k.	4. 130k was number of dataset's instances.

Chapter 5

Conclusion

5.1 Summary

In today's world, the most common task is named entity recognition in medicine. The entity may be easily extracted from a vast amount of data using a natural language processing (NLP) model. The work will be challenging if the data pattern differs from the regular data. Twitter data is typically real-time data with a large quantity of noise. Extracting medical entities from Twitter data is a challenging task. With the use of NLP and a machine learning model, we were able to extract medical entities in our study. Two types of datasets used in our research. The CADEC corpus, which records medical events, was one of them. In our work, disease related term was extracted from the twitter data. Disease related term are simply disease, drug, symptoms, treatment and test. We suggested the probabilistic language model Conditional Random Fields (CRF) and the deep learning model Bidirectional LSTM (Bi-LSTM) to extract the medical entity from twitter content. In our work, 50% of twitter data were added with the CADEC corpus to get better accuracy or better extraction of disease related terms. The ME model accuracy was 72.31% and Bi-LSTM-CRF model accuracy was about 62.70%. The more noise data in the twitter also causes the low F-1 score value. Now some comments on my work, twitter data are more noisy rather than normal text data that leads the low accuracy of model. In our work, about 12426 numbers of instances were used and CADEC corpus was used that can not carry total diseases information. When new diseases is discovered so it will not possible to identify the disease from tweets.

5.2 Future Works

- In future, we can be added all medical corpus which will contain all types of medical words not only disease related terms.
- More effective data pattern may be discovered in future which will help to get better accuracy.
- A new learning model will be selected for this kinds of sequential data which will provide better performance rather than Bi-LSTM model.

- In our work, there exits three category of a class. In future the number of categories may be extended.
- In future, entity recognition will not bounded on medical entity from twitter, other entity may have extracted from twitter data.
- Rules based method can be applied in the field of medical named entity recognition in future.

References

1. X. Liu, F. Wei, S. Zhang and M. Zhou, "Named entity recognition for tweets", ACM Transactions on Intelligent Systems and Technology, vol. 4, no. 1, pp. 1-15, 2013. Available: 10.1145/2414425.2414428.

2. D. Wintaka, M. Bijaksana and I. Asror, "Named-Entity Recognition on Indonesian Tweets using Bidirectional LSTM-CRF", Procedia Computer Science, vol. 157, pp. 221-228, 2019. Available: 10.1016/j.procs.2019.08.161.

3. S. Pradhan and N. Elhadad and B. R. South and D. Martinez and L. Christensen and A. Vogel and H. Suominen and W. W. Chapman and G. Savova, Evaluating the state of the art in disorder recognition and normalization of the clinical narrative", Vol. 22(1), pp. 143–154, ISSN 1527-974X.

4. H. Cho and H. Lee, "Biomedical named entity recognition using deep neural networks with contextual information", BMC Bioinformatics, vol. 20, no. 1, 2019. Available: 10.1186/s12859-019-3321-4.X. Liu, F. Wei, S. Zhang and M. Zhou, "Named entity recognition for tweets", ACM Transactions on Intelligent Systems and Technology, vol. 4, no. 1, pp. 1-15, 2013. Available: 10.1145/2414425.2414428.

5. A. Śniegula, A. Poniszewska-Marańda and Ł. Chomątek, "Study of Named Entity Recognition methods in biomedical field", Procedia Computer Science, vol. 160, pp. 260-265, 2019. Available: 10.1016/j.procs.2019.09.466.

6. D. WANG and X. FAN, "Named entity recognition for short text", Journal of Computer Applications, vol. 29, no. 1, pp. 143-145, 2009. Available: 10.3724/sp.j.1087.2009.00143.

7. T. Yang, Y. He and N. Yang, "Named Entity Recognition of Medical Text Based on the Deep Neural Network", Journal of Healthcare Engineering, vol. 2022, pp. 1-10, 2022. Available: 10.1155/2022/3990563.

8. Enrique Alfonseca and Suresh Manandhar. 2002. An unsupervised method for general named entity recognition and automated concept discovery. In Proceedings of the 1st international conference on general WordNet, Mysore, India, pages 34–43.G. Crichton, S. Pyysalo, B. Chiu and A. Korhonen, "A neural network multi-task learning approach to biomedical named entity recognition", BMC Bioinformatics, vol. 18, no. 1, 2017. Available: 10.1186/s12859-017-1776-8.

9. S. Ananiadou, C. Friedman and J. Tsujii, "Introduction: named entity recognition in biomedicine", Journal of Biomedical Informatics, vol. 37, no. 6, pp. 393-395, 2004. Available: 10.1016/j.jbi.2004.08.011.
10. B. Alshaikhdeeb and K. Ahmad, "Biomedical Named Entity Recognition: A Review", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 6, p. 889, 2016. Available: 10.18517/ijaseit.6.6.1367.
11. K. Bae, "An Efficient Named Entity and Topic Word Recognition Method Based on Named Entity Pattern in a Natural Language Interface", The Journal of Korean Institute of Information Technology, vol. 12, no. 1, 2014. Available: 10.14801/kiitr.2014.12.1.121.
12. A. Śniegula, A. Poniszewska-Marańda and Ł. Chomątek, "Study of Named Entity Recognition methods in biomedical field", Procedia Computer Science, vol. 160, pp. 260-265, 2019. Available: 10.1016/j.procs.2019.09.466.
13. K. Lay and A. Cho, "Myanmar Named Entity Recognition with Hidden Markov Model", International Journal of Trend in Scientific Research and Development, vol. -3, no. -4, pp. 1144-1147, 2019. Available: 10.31142/ijtsrd24012.
14. K. Bae, "An Efficient Named Entity and Topic Word Recognition Method Based on Named Entity Pattern in a Natural Language Interface", The Journal of Korean Institute of Information Technology, vol. 12, no. 1, 2014. Available: 10.14801/kiitr.2014.12.1.121.
15. L. Derczynski et al., "Analysis of named entity recognition and linking for tweets", Information Processing & Management, vol. 51, no. 2, pp. 32-49, 2015. Available: 10.1016/j.ipm.2014.10.006.
16. A. B. Abacha and P. Zweigenbaum, "Medical Entity Recognition: A Comparison of Semantic and Statistical Methods", Proceedings of BioNLP 2011 Workshop, BioNLP'11, pp. 56–64, 2011.Nguyen, V., Nguyen, H. and Snasel, V., 2016. Text normalization for named entity recognition in Vietnamese tweets. Computational Social Networks, 3(1).
17. S. Zhao, T. Liu, S. Zhao and F. Wang, "A Neural Multi-Task Learning Framework to Jointly Model Medical Named Entity Recognition and Normalization", Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 817-824, 2019. Available: 10.1609/aaai.v33i01.3301817.

18. G. Crichton, S. Pyysalo, B. Chiu and A. Korhonen, "A neural network multi-task learning approach to biomedical named entity recognition", BMC Bioinformatics, vol. 18, no. 1, 2017. Available: 10.1186/s12859-017-1776-8.
19. C. Wen, T. Chen, X. Jia and J. Zhu, "Medical Named Entity Recognition from Un-labelled Medical Records based on Pre-trained Language Models and Domain Dictionary", 2022.
20. https://data-flair.training/blogs/artificial-neural-networks-for-machine-learning/
21. https://en.wikipedia.org/wiki/Recurrent_neural_network
22. https://www.researchgate.net/figure/The-architecture-of-the-ATT-CNN-BiLSTM-model_fig1_339557594
23. https://www.sciencedirect.com/topics/computer-science/conditional-random-field