INFORMATION EXTRACTION USING TEXT AND DEVELOPMENT OF BIOMEDICAL SEARCH ENGINE

M S Janhavi[1] and Mrs. S. Vijayalakshmi[2]

[1]Student, B.Tech Information Technology
[2]Assistant Professor, Department of Information Technology Meenakshi Sundararajan Engineering College Kodambakkam, Chennai, Tamil Nadu, India.

Abstract – *The amount of natural language text that is available in electronic form is increasing every day. It increases the complexity of natural language and makes it very difficult to access the information in that text. Therefore, an important approach called text mining involves the use of natural-language information extraction. Information extraction (IE) simplifies the complexity of unstructured data gathered from various resources and is converted to structured format by identifying keywords to named entities as well as relationships between such entities. The machine is then trained to perform operations according to the problem statement and also the machine is evaluated based on its performance after training. When the final dataset is prepared, a new web application (search engine interface) which allows fast interactive browsing of the biomedical sentences indexed by the system is created as a tool for accessing the data.*

Key Words: **Search engine, text mining, natural language processing, information extraction, named entity recognition, transfer learning, fine tuning**

1. INTRODUCTION

Currently, the amount of biomedical information is fragmented across the literature. To speed up this process in the biomedical domain, the WhiteText project was created to automatically extract this information from text. White Text was designed to extract information of brain regions and statements describing connections between them.

Using this White Text information, search engines, question and answering systems can be built. The benefit of these systems is that biomedical researchers need not search for biomedical information from various resources including any Wikipedia or even the encyclopedias.

These systems will retrieve text from various research contents and as a result it will benefit researches in getting the specific information without any time constraint.

1.1 INFORMATION EXTRACTION

Information Extraction is the task or process of automatically extracting structured information from unstructured data (can be in JSON, XML formats). In most of the cases, this can be achieved by means of Natural Language Processing.

The process in Information extraction is as follows:

Data Collection and Cleansing

The data is collected from the backend in the form of either JSON or XML format. This is the unstructured format where there can be incomplete data, duplicate data etc. The unstructured data is then cleansed and converted to structured format. The cleansing techniques involve removing of extra whitespaces, un-readable characters, alpha-numeric characters, text alignment, etc.

Document Analysis and Augmented Semantics Then, we perform simple statistical operations such as either data visualization or finding the count frequency distribution of words. This step will help us to understand your problem domain.

Document Preprocessing

In this step, the data pre-processing techniques include text segmentation, stemming, lemmatization, syntactic parsing, and outlier removal.

Document Transformation and Feature Engineering

In this step, we know that machine learning algorithms are unable to interpret text in human readable form. Hence the text needs to be converted into numeric format such as vectors. Bag of Words Model, Word Vectors, One hot vector encoding, n- grams formation are such approaches that converts text to numeric format.

Modeling and Evaluation

The model is either built or a pre-trained model is selected based on the approach (unsupervised, semi- supervised or supervised) selected for the problem statement. The evaluation is usually calculated by using the metrics, Accuracy, Precision-Recall etc.

1.2 NATURAL LANGUAGE PROCESSING

Natural Language Processing is an activity in Information Extraction that helps machine understands text or voice.

Various applications have been developed in the fields such as medical research, risk management, customer care, insurance (fraud detection) and contextual advertising.

1.3 NAMED ENTITY RECOGNITION

Named Entity Recognition (NER) is the subtask of information extraction, where it identifies named entities and classifies them into various categories such as person, location, organization etc. It is also known as entity identification, entity extraction etc. NER is an important pre-processing step for a variety of applications such as information retrieval, question answering, machine translation etc. A named entity is a word or phrase that identifies one item from a set of other items that have similar properties (example: location, person etc.)

The task of NER can be broken into two: identifying the boundaries of the named entity, and identifying its type. The figure below explains the working of Named Entity Recognition.

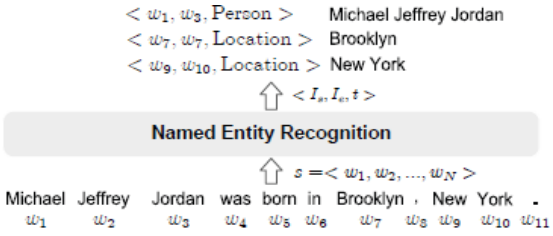


FIG-1: NAMED ENTITY RECOGNITION

In the fig. above, the sentence is split into strings of and the NER model identifies the type of the named entities to be person and location.

Given a sequence of tokens s =<w1, w2,....,wN>, NER is to output a list of tuples <Is,Ie,t>each of which is a named entity mentioned in s and t is the entity type from a predefined category set.

1.4 TRANSFER LEARNING

Deep learning has seen a lot of progress in recent years. The availability of large amounts of data along with increased computation resources has improved this progress. Transfer learning is one of the methods that have helped enhance Deep Learning.

Transfer learning in machine learning (ML) focuses on storing knowledge gained while solving one problem and applying it to a different but related problem.

Transfer learning can be used if the three following criterion are satisfied. They are, the initial skill, the rate of improvement of skill during the training of the source model is steeper, the converged skill of the trained model is better than it otherwise would be.

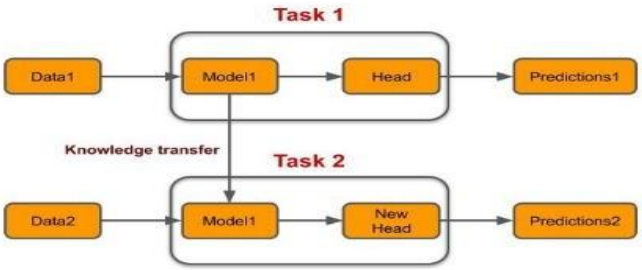


FIG-2: TRANSFER LEARNING FLOW DIAGRAM

1.4.1 FINE TUNING THE MODEL

Fine-tuning can be defined as training a classifier for a different task, by modifying the weights of the model. In other words, it is a process of considering the weights of a pre trained model and use it as either an initialization for an entirely new model trained on the dataset from the same domain (e.g. images, biomedical text) or use for prediction for an entirely different dataset from the same domain. It is used to speed up the training. In fine-tuning, we are not training the entire network. The part that is being trained is not trained from scratch. The parameters that need to be updated is less, the amount of time needed will also be less to acquire the results.

2. RELATED WORK

A. We describe the WhiteText project, and its progress towards automatically extracting statements of neuroanatomical connectivity from text. We review progress to date on the three main steps of the project: recognition of brain region mentions, standardization of brain region mentions to neuroanatomical nomenclature, and connectivity statement extraction. We further describe a new version of our manually curated corpus that adds 2,111 connectivity statements from 1,828 additional abstracts. Cross-validation classification within the new corpus replicates results on our original corpus, recalling 67% of connectivity statements at 51% precision. The resulting merged corpus provides 5,208 connectivity statements that can be used to seed species-specific connectivity matrices and to better train automated techniques. Finally, we present a new web application that allows fast interactive browsing of the over 70,000 sentences indexed by the system, as a tool for accessing the data and assisting in further curation.

B. We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with

just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

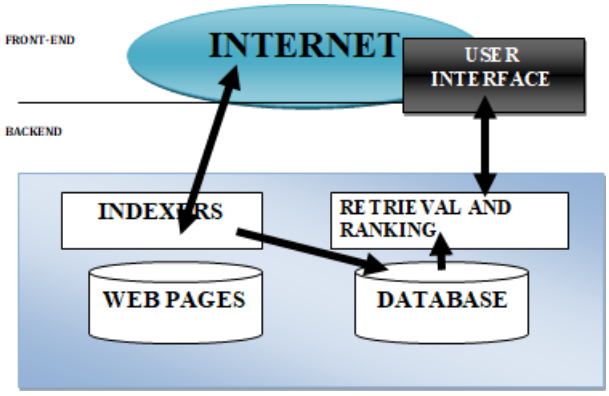
3. PROPOSED SYSTEM

Currently, the amount of biomedical information is fragmented across the literature. To speed up this process in the biomedical domain, the WhiteText project was created to automatically extract this information from text. White Text was designed to extract information of brain regions and statements describing connections between them.

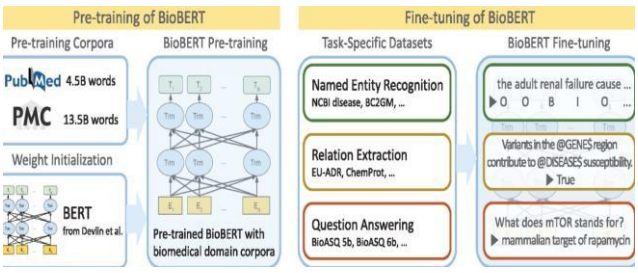
Using this White Text information, search engines, question and answering systems can be built. As a result, White text data which contains information from various resources are fetched, pre-processed and hence using these data the White text search engine is built.

White Text is a corpus of manually annotated brain region mentions. It was created to facilitate text mining of neuroscience literature. The corpus contains 1,377 abstracts with 17,585 brain region annotations. In the project, we take two different datasets from White Text corpus: WhiteTextUnseenEval and WhiteTextNegFixFull. The WhiteTextUnseenEval consists of 2174 sentences with their corresponding Brain Regions. The WhiteTextNegFixFull consists of 4339 sentences.

3.1 Proposed System Architecture Diagram



3.2 BIO-BERT Architecture Diagram



3.3 System Flow Diagram

3.4 Modules and Description

A. Data Collection and Cleansing:

The White text data is collected in the form of XML format. This is the unstructured format where there is incomplete data, duplicate data etc. The unstructured data is then cleansed and converted to structured format which is in the form of .tsv. The cleansed data consists of sentences, its corresponding entities etc.

B. Sentence Segmentation

Sentence segmentation is the process of identifying paragraphs and dividing them into individual units or sentences consisting of one or more words.

C. Tokenization

Tokenization is the process of tokenizing or splitting a string, text into a list of tokens.

For example, consider the sentence “God is great! I won a lottery.”

The tokenization of words will give a list containing: ['God',

'is', 'great', '!', 'I', 'won', 'a', 'lottery', '.‘

D. Entity detection and tagging

The IOB format (short for inside, outside, beginning) is a common tagging format for tagging tokens in a chunking task in computational linguistics (ex. named-entity recognition).The B- prefix before a tag indicates that the tag is the beginning of a chunk, and an I- prefix before a tag indicates that the tag is inside a chunk. An O tag indicates that a token belongs to no chunk.

For example, consider the sentence “The cat is going to Los Angeles “and the named entity is “Los Angeles”.

The - O

Cat - OIs - O

Going - Oto - O

Los B-LOC

Angeles I-LOC.

E. Fine tuning the dataset

For the process of fine tuning, the dataset is split into training data (This is the data which your model actually knows both input and output and learn from.), validation data (used to do a frequent evaluation of model, fit on training dataset along with improving involved hyper parameters) and testing data (This is the data which your model actually knows only the input and will predict the output based on the learning.). The pre-processed WhiteTextNegFixFull dataset is split into train and validation data. The pre-processed WhiteTextUnseenEval dataset is taken as test data. The datasets are fed into the pre-trained BIO-BERT model and the results are obtained.

F. Storing the result in the database

The dataset is now added to the database MYSQL.

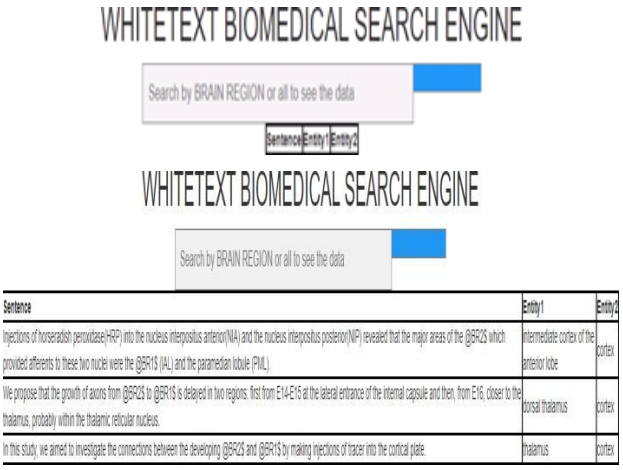
When the user searched for a particular context the data will be retrieved from the database.

The deployment was done using Python Flask.

G. Storing the result in the database

When the user searches for a keyword, the context along with where the keyword appeared appears.

3.5 OUTPUT



4. RESULT ANALYSIS

A. PRECISION RECALL ACCURACY:

Precision measures the ability of a NER system to present only correct entities

Recall measures the ability of a NER system to recognize all entities in a corpus,

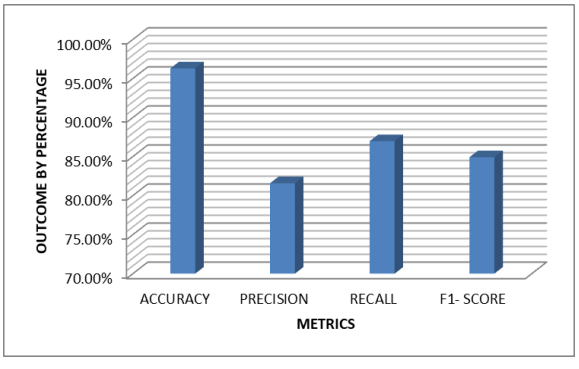
Precision= TP/ (TP+FP) Recall=TP/ (TP+FN)

F-Score= 2*(Precision x Recall/ Precision+Recall)

Table 1: Precision, recall and accuracy

METRICS	OUTCOME IN PERCENTAGE
ACCURACY	96.28%
PRECISION	81.58%
RECALL	86.97%
F1- SCORE	84.91%

The graph below shows the visualization of the metrics.



5. CONCLUSION AND FUTURE ENHANCEMENTS

We introduced the method of fine tuning with White text data using the Deep Learning model, BIO-BERT which yields an accuracy of 96% and precision to be 81.58%. Also, the web application was designed to extract mentions of brain regions and statements describing connections between them.

The project can be further enhanced by developing a question and answering system from where answers related to unanswered queries can be identified. Now, using these model, unsupervised learning algorithms can be used for prediction. An entirely new set of related data can be used for unsupervised learning and thus the model can be further evaluated.

6. REFERENCES

[1] BioBERT: a pre-trained biomedical language representation model for biomedical text mining Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, Jaewoo Kang

[2] A Survey on Recent Advances in Named Entity Recognition from Deep Learning models Vikas Yadav,Steven Bethard