NLP BASED RETRIEVAL OF MEDICAL INFORMATION FOR DIAGNOSIS OF HUMAN DISEASES

Gunjan Dhole[1], **Nilesh Uke**[2]

[1]*Department of IT, PG Student, Sinhgad College of Engg, Pune, 411041, India*
[2]*Department of IT, H.O.D., IT department, Sinhgad College of Engg, Pune, 411041, India*

Abstract

NLP Based Retrieval of Medical Information is the extraction of medical data from narrative clinical documents. In this paper, we provide the way to diagnose diseases with the help of natural language interpretation and classification techniques. However extraction of medical information is difficult task due to complex symptom names and complex disease names. For diagnosis we will be using two approaches, one is getting disease names with the help of classifiers and another way is using the patterns with the help of NLP for getting the information related to diseases. These both approaches will be applied according to the question type.

Keywords: *NLP, narrative text, extraction, medical information, expert system*

---***---

1. INTRODUCTION

The increasing adoption of electronic health records (EHRs) and the corresponding interest in using these data for quality improvement and research have made it clear that the interpretation of narrative text contained in the records is a critical step. The biomedical literature is another important information source that can benefit from approaches requiring structuring of data contained in narrative text. Different approaches are implemented for extraction of the medical text. Natural language processing approach uses tools like noun entity recognizers, co reference resolution, part of speech taggers and relationship extractors. However medical text is different than normal text as it contains complex terminologies, so medical information needs advance versions of these tools.

1.1 Motivation

Following points which express the need for the automated system:

- Need for Text Processing
- Need of Medical Text Processing
- Need for automated system for diagnosis of disease
- Need for Automated system for doctors

1.2 Why To use NLP for Information Retrieval

Most of the systems are using Rule based systems for extraction of information. In rule based systems the number of rules is limited. Because of limited number of rules the information extraction gets limited. If some new information is required then the rule based system fails to extract information due to lack of rules. In natural language processing, we can extract data from free text form. Enormous amount of data is available in free text form is available today. But it is not properly utilized. The text such as Electronic Health Records has patient data which can be used for many purposes. The Free text such as disease information, its symptom and causes all can be found in the free text. Natural language processing can be used to extract all these and get proper information. It is also found that doctors don’t use their whole knowledge for any disease. The automated system for data extraction can help doctors for proper recognition of diseases.

2. RELATED WORK

Different medical extraction systems like MedLEE, metamap, linguistic string project were proposed [9]. The goal of MedLEE is to extract, structure, and encode clinical information in textual patient reports so that the data can be used by subsequent automated processes. MedLEE was created by Carol Friedman in collaboration with the Department of Biomedical Informatics at Columbia University, the Radiology Department at Columbia University, and the Department of Computer Science at Queens College of CUNY[9]. **MetaMap** is a highly configurable program developed by Dr. Alan Aronson at the National Library of Medicine (NLM) to map biomedical text to the UMLS Metathesaurus or, equivalently, to discover Metathesaurus concepts referred to in text [9]. The Linguistic String Project (LSP) was a sustained research effort (1960-2005) in the computer processing of language based on the linguistic theory of Zellig Harris: linguistic string theory, transformation analysis, and sublanguage grammar [9].

Different tools which are used for natural language processing are NER, pos-taggers, co-ref resolutions and Relationship extractors. As the research in biomedical domain has grown rapidly in recent years, a huge amount of nature language resources have been developed and become a rich knowledge base. The technique of named entity (NE) recognition (NER) is strongly demanded to be applied in biomedical domain. Branimir T. Todorovic and Svetozar R.

Rancic proposed a system for Named Entity Recognition and Classification using Context Hidden Markov Model [14]. Mohamed Hashem proposed A Supervised Named-Entity Extraction System for Medical Text. Andreea Bodnari. Louise Deleger proposed system for Effective Adaptation of a Hidden Markov Model-based Named Entity Recognizer for Biomedical Domain [14].

Jiaping Zheng proposed a system for co reference resolution for the clinical narrative [15]. Wafaa Tawfik, Abdel-moneim proposed a system for Clinical Relationships extraction techniques from patient narratives [7]. The Ontology Development and Information Extraction corpus annotated for co reference relations consists of 7214 co referential makeable, forming 5992 pairs and 1304 chains. Classifiers can be trained with semantic, syntactic, and surface features pruned by feature selection. For the three system components for the resolution of relative pronouns, personal pronouns, and noun phrases. Support vector machines with linear and radial basis function (RBF) kernels, decision trees, and perceptrons can be used for machine learning [10].

Ddan shen PROPSED A MedPost: a part-of-speech tagger for bioMedical text [13]. Tagger (called MedPost) was developed to meet the need for a high accuracy part-of-speech tagger trained on the MEDLINE corpus. The program currently accepts text for tagging in either native MEDLINE format or XML, both available as save options in PubMed. MEDLINE is a bibliographic database of publications in health sciences, biology and related fields. It currently contains over 12 million records, and nearly 7 million include an abstract [13].

Semantic relations can be extracted with the help of annotation approach which relies on linguistic patterns and domain knowledge which consists of two steps [8]:

(i) Recognition of medical entities and
(ii) Identification of the correct semantic relation between each pair of entities.

The first step is achieved by enhanced use of metamap. The second step relies on linguistic patterns which are built semi-automatically from a corpus selected. According to semantic criteria, evaluation of the treatment relations between a treatment (e.g. medication) and a problem (e.g. disease) can be extracted.

3. PROPOSED WORK

The proposed system consists of two major parts: Document Processing with Natural Language processing and query processing with natural language processing. Both are very important phases of the project. The proposed system consists of 5 modules as shown in the architecture. Also it contains the knowledge base. Knowledge base actually stores all the medical documents:

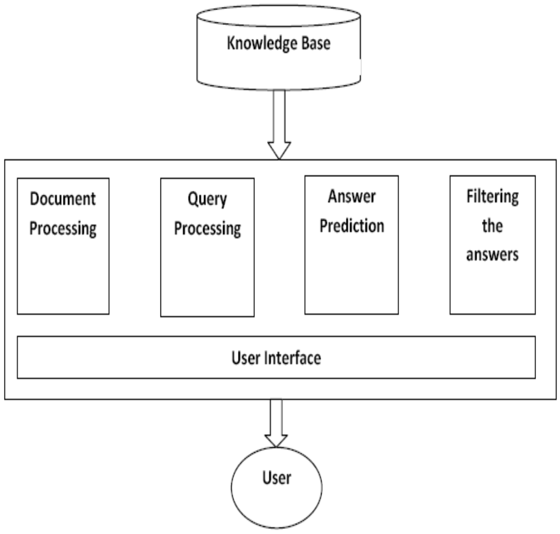

Fig.1 the Architecture of Proposed System

Proposed system contains following modules:

1) Data Extraction
2) Data Processing
3) Query Processing
4) Answer Matching
5) Filtering the Answers

3.1 Data Extraction

This module will manage knowledge Base. It will try to extract some data from Internet. Knowledge base will also consist of different resources which contains biomedical texts regarding different systems. It can have any kind of free text which involves diseases information.

3.2 Data Processing

Document processing involves processing of documents with the help of Natural language processing. It will involve parts such as section splitting, tokenization, relationship extraction etc. At the end of this phase, output will contain the

3.3 Query Processing

Query processing will involve the processing of query with the help of natural language processing. It will extract all the important relationships and keywords from the query

3.4 Answer Prediction

Answer prediction will involve the prediction of answer from the given set of relations and keywords

3.5 Outcome of the Project

The goal of the project is to get the answers of the disease related queries with the best precision and recall.

If user enters set of symptoms then system should be able to answer with the probable disease name.

4. IMPLEMENTATION:

Implementation stages:
Steps which is being performed:
Step 1 : Get the query from the user
Step 2: Get the type of query **-** 'definitive' or 'descriptive'
Step 3 : Process the query with the help of natural language processing.
Step 4 : Get the documents related to the query**.**
Step 5 : Process the top ranked documents with the help of Natural Language processing
Step 6 : Check the type of query if 'definitive' do step 7 or if 'descriptive' do step 8
Step 7: Get the information about the disease
Step 8: Get the disease from described symptoms

4.1 Data Storage and Knowledge Database Management

4.1.1 Knowledge Database Retrieval

Data related to medical domain is extracted from following two resources:

1) Medline Plus
2) Wikipedia

Medline plus consist of documents related to most of the diseases. This data is extracted and a dataset is made from this data. With the Wikipedia, Wikipedia provides an api from which we can extract data if we know the topic name which is their onto Wikipedia. With the help of Wikipedia api tried to extract the list of different diseases data. We are maintaining the corpus with all the disease names.

4.1.2 Data Storage

Data is stored in MongoDb. MongoDB (from "humongous") is a cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas (MongoDB calls the format BSON), making the integration of data in certain types of applications easier and faster. Released under a combination of the GNU Affero General Public License and the Apache License, MongoDB is free and open-source software.

The data is stored in mongoDb in the following format. It consist of these two formats ->

1) Index database
2) Document Database

Index databases take care of indexing the documents. The documents are indexed with the help of "Positioning Indexing". Positioning Indexing is an index data structure storing a mapping from content, such as words or numbers, to its locations in a database file, or in a document or a set of documents. The purpose of an positioning index is to allow fast full text searches, at a cost of increased processing when a document is added to the database. Positioning indexes are maintained in the separate database into Mongodb

Document database consist of document text with the doc id and term which document is related to.

4.2 Input: User Input Interface:

The user input interface accepts the query from the user. Query can be a sentence explaining about the symptoms or asking information about any disease. The user interface will accept this query and give output as the disease related to these symptoms. And if the query is asking information about the disease it will give the disease information.

The technology used for web development is Flask Framework. Flask is a micro framework for Python based on Werkzeug, Jinja 2. With the help of flask server running onto machine, front end is designed for question answering system.

4.3 Query and Answer Type Analysis:

The Intellect rule engine is a rule engine which can maintain a list of rules. So we are training the system with these set of rules for query and answer type analysis. Now we are reasoning this rule engine with the query which is coming from user and trying to get the type of query.

Two Types:

1) Definitive – Queries asking about information of disease
   Queries like –
   What is lung cancer
   What do you mean by lung cancer
   Meaning of lung cancer
2) Descriptive – Queries describing about the symptoms of the patients
   Queries like –
   I have coughing, coughing up blood, wheezing or shortness of breath. What is my disease.

4.4 Query Processing

It consists of following steps:

- Spell checking
- Capitalisation
- Tokenization
- Pos tagging
- Noun Entity recognition

4.4.1 Spell Check

The words from the query are first checked if they exist into the English dictionary or not. If they do there is no need to check the spelling. If they don't, we will try to find out some suggestions by interchanging the characters, replacing characters and inserting new characters into the word. Now from this list of suggestions the word with minimum edit distance will be chosen.

Input: What is lung cancr?
Output: What is lung cancer?

4.4.2 Capitalisation

Normally if the syntax of the query is rite then it is easier to get the meaning of the query with the help of NLTK. For the correct syntax the words which are important or the words which are proper noun should be capitalised. This is being done with the help of corpus which contains the disease names. If the Ngram from the query matches with the disease name then capitalise the ngram and change the query according to that.

Input: What is lung cancer?
Output: What is Lung Cancer?

4.4.3 Tokenization and Pos Tagging:

These are basic steps in NLP which are needed mostly for every task which is mentioned above like spellcheck and capitalisation. Tokenisation tries to get the individual entity from the query and pos tagging tries to part of speech tag each word with the same tags.

Input:
What is Lung Cancer
Tokenisation:
[‘what’,’is’,’Lung’,’Cancer’]
Pos tagging:
[(‘what’,’WH’),(‘is’,’VBZ’),(‘Lung’,’NNP’),(‘Cancer’,’NNP’)]

4.4.4. Noun Entity Recognition

Noun Entity recognition is done by three ways:

1) With the NLTK noun entity recognition
2) With the help of Chunking different pos tags together
3) By taking nouns in case no noun entity found

Input: What is Lung Cancer
Output: [‘Lung Cancer’]

4.5 Answer Processing Module

Answer processing module takes care of the flow of project. It will check which type of query is coming in. If its “Definitive” it will follow the flow for definitive queries. If its “Descriptive” query, it will follow the flow for descriptive queries. It will take care of sending out final answers to the user.

4.6 Document Processing:

This flow will be followed in case of “Definitive queries”.

4.6.1 Document Retrieval:

The Query processing module will give noun entities as output. With the help of these noun entities documents will be retrieved from the MongoDb database.

4.6.2 Document Scoring:

Documents will be scored with the help of TF-IDF scoring and cosine similarity between the query and the documents. tf–idf, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection orcorpus. It is often used as a weighting factor in information retrieval and text mining. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others. Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. The cosine of 0° is 1, and it is less than 1 for any other angle. It is thus a judgement of orientation and not magnitude: two vectors with the same orientation have a Cosine similarity of 1, two vectors at 90° have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1].

4.6.3 Document Processing:

Documents will be processed with the help of textTiling. First they will be split in paragraphs then we will search for patterns which suggest about the information about the disease. We will give it out as an answer.

4.7 Classifiers

Classifiers will be used in case of “descriptive” queries. Descriptive queries basically consist of the symptoms the patient is suffering from. From that we will need to figure out the disease name.

4.7.1 Choosing a Classifier

We tried to choose from the set of classifiers that are normally used for text classification. These are the classifiers which are used for classification normally

1. Linear support vector machines
2. K-neighrest neighbour
3. Bernoulli Naives bayes classifiers
4. MultiNomial Naives bayes classifier

The project document database data is given to all these classifiers and the performances are checked so that best classifier can be chosen for the data. It is observed that Naïve bayes classifiers performs well. Following figure shows this.

4.7.2 Training the Classifiers:

Classifiers will be trained with the help of document database. We will give label and its data to the classifier. And train the classifier and save it off.

4.7.3 Testing the Classifiers

Classifiers will be given the query as a input and it will try to figure out which class (Disease name) it belongs too.

The disease name with the highest probability will be given as output to the user.

Load Balancing

For load balancing gearman is used. Following is the architecture of gearman.

4.8 Implementation Flow

Following figure shows the implementation flow

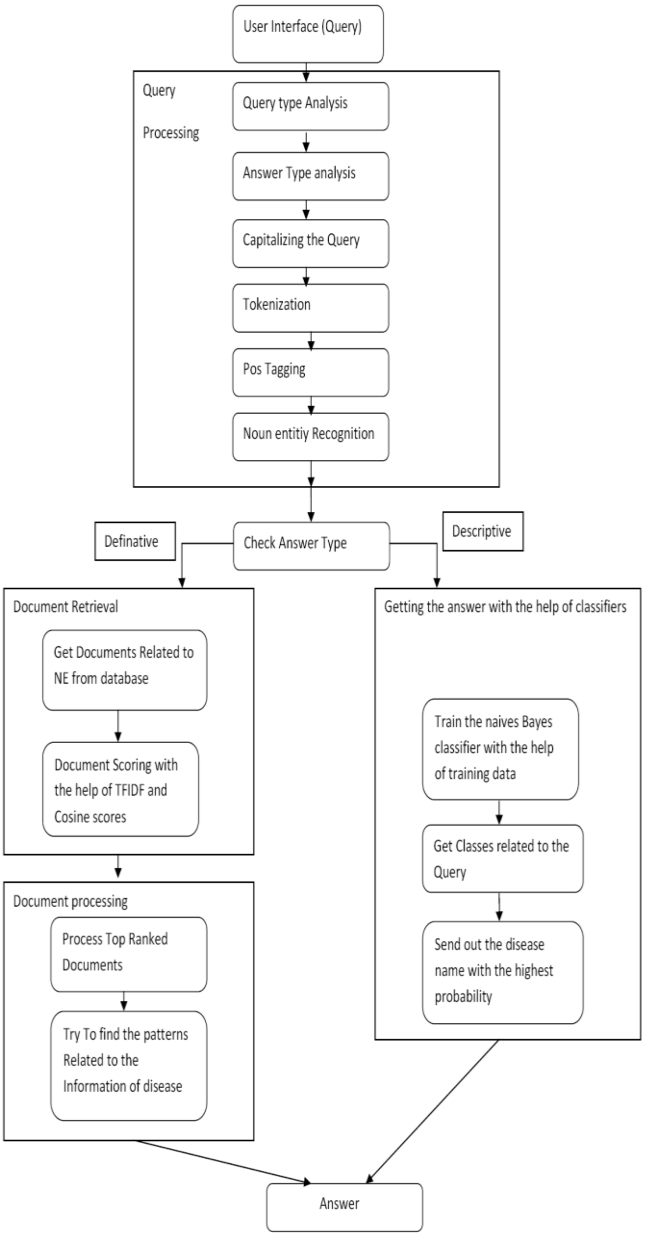
User Interface (Query)
Query Processing
Query type Analysis
Answer Type analysis
Capitalizing the Query
Tokenization
Pos Tagging
Noun entitiy Recognition
Definative
Check Answer Type
Descriptive
Document Retrieval
Get Documents Related to NE from database
Document Scoring with the help of TFIDF and Cosine scores
Document processing
Process Top Ranked Documents
Try To find the patterns Related to the Information of disease
Getting the answer with the help of classifiers
Train the naives Bayes classifier with the help of training data
Get Classes related to the Query
Send out the disease name with the highest probability
Answer

Fig 2 Implementation flow

5. CONCLUSION

Lots of research is going on in the field of extraction of medical text with the help of NLP. As medical text is different than normal text, it needs advanced tools as compared to the normal NLP tools. We implemented the system which can give the basic information regarding diseases and also can give the disease information onto the basis of diseases.

REFERENCES

[1] Kyle D. Richardson1, Daniel G. Bobrow1, Cleo Condoravdi1, Richard Waldinger2, Amar Das3, "*English Access to Structured Data*", 2011 Fifth IEEE International Conference on Semantic Computing

[2] Faguo ZHOU Enshen WU, "*The Design of Computer Aided Medical Diagnosis System Based on Maximum Entropy*" 978-1-61284-729-0111 2011 IEEE

[3] Stphane Meystre, Peter J. Haug, "*Natural language processing to extract medical problems from electronic clinical documents: Performance evaluation*", Journal of Biomedical Informatics 39 (2006) 589–599

[4] Stéphane Meystre, Peter J Haug, R. Engelbrecht et al., "*Evaluation of Medical Problem Extraction from Electronic Clinical Documents Using MetaMap Transfer (MMTx) Connecting Medical Informatics and Bio-Informatics*", ENMI, 2005

[5] Lucila Ohno-Machado, Editor-in-chief, Prakash Nadkarni, Kevin Johnson "*Natural language processing: algorithms and tools to extract computable information from EHRs and from the biomedical literature*", amiajnl-2013-002214

[6] Khan Razik, Dhande Mayur , "*To Identify Disease Treatment Relationship in Short Text Using Machine Learning & Natural Language Processing*", Journal of Engineering, Computers & Applied Sciences (JEC&AS), Volume 2, No.4, April 2013

[7] Wafaa Tawfik Abdel-moneim1, Mohamed Hashem , "*Clinical Relationships Extraction Techniques from Patient Narratives*", JCSI International Journal of Computer Science Issues, Vol.10, Issue 1, January 2013

[8] Asma Ben Abacha, Pierre Zweigenbaum, "*Automatic extraction of semantic relations between medical entities: a rule based approach*" From Fourth International Symposium on Semantic Mining in Biomedicine (SMBM)
Hinxton, UK. 25-26 October 2010

[9] Stéphane M. Meystre, MD, MS, Peter J. Haug , "*Comparing Natural Language Processing Tools to Extract Medical Problems from Narrative Text*", MD AMIA 2005 Symposium Proceedings

[10] Jiaping Zheng,1 Wendy W Chapman,2 Timothy A Miller,1 Chen Lin, "*A system for coreference resolution for the clinical narrative*", J Am Med Inform Assoc (2012). doi:10.1136/amiajnl-2011-000599

[11] Romer Rosales, Faisal Farooq, Balaji Krishnapuram, Shipeng Yu, Glenn Fung, "*Automated Identification of Medical Concepts and Assertions in Medical Text Knowledge Solutions*" , AMIA i2b2/VA text mining challenge

[12] D . Nagarani, Avadhanula Karthik, G. Ravi, "*A Machine Learning Approach for Classifying Medical Sentences into Different Classes*", IOSR Journal of Computer Engineering (IOSRJCE) Volume 7, Issue 5 (Nov-Dec. 2012), PP 19-24

[13] L. Smith1, T. Rindflesch2 and W. J. Wilbur, "MedPost: a part-of-speech tagger for bioMedical text", Vol. 20 no. 14 2004, pages 2320–2321, bioinformatics/bth227

[14] Dan Shen Jie Zhang Guodong Zhou, "*Effective Adaptation of a Hidden Markov Model-based Named Entity Recognizer for Biomedical Domain*"