E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

AUTOMATED DISEASE PREDICTION AND DIAGNOSIS FLOWCHART GENERATION FROM MEDICAL REPORTS USING NLP AND RANDOM FOREST

V. Raguvaran[1], Electrical and Electronics Engineering, Cape Institute of Technology

Prof. Ebbie Selvakumar[2], HOD, Electrical and Electronics Engineering, Cape Institute of Technology

Abstract

The digital healthcare age, quick and precise interpretation of patient information is paramount for proper diagnosis and treatment on time. This research work suggests an intelligent system for auto-disease prediction and auto-diagnosis flowchart generation from actual medical reports available in PDF form. The algorithm begins with text data extraction from clinical reports using sophisticated PDF parsing technology. Natural Language Processing (NLP) operations such as tokenization, lemmatization, Named Entity Recognition (NER), and keyword extraction (through TF-IDF or RAKE) are utilized to structure and process the extracted data. A Random Forest classifier is subsequently used to make predictions of the most likely disease based on detected symptoms and relevant features. Upon prediction, a corresponding predefined diagnostic flowchart is created using visualization libraries like Graph viz or Mermaid.js. The final output consisting of the predicted disease, medical terminology, and graphical diagnosis pathway is shown to the user for better comprehension and support in decision-making. This system provides an effective, explainable, and user-friendly means of supporting clinical evaluation and healthcare provision.

Keywords: NLP, Random Forest Classifier, TF-IDF, Machine Learning, Deep Learning, NER.

1. Introduction

The swift computerization of health information has facilitated autonomous diagnosis systems in helping physicians discover diseases in an efficient manner [1]. In medical records stored patient data sometimes includes pertinent facts regarding symptoms, history, and treatment outcome. Extracting unstructured information through analysis and analysis can facilitate successful disease prediction [2]. The application of natural language processing algorithms has proved particularly useful for making sense of free-text content derived from clinical texts. Combining machine learning with NLP can improve decision-making and simplify patient care [3].

Rule-based or keyword-matching mechanisms are adopted in traditional systems for detecting patterns of disease, which tend to be inflexible and non-agile [4]. Naive Bayes, Support Vector Machines (SVM), and simple Decision Trees are machine learning techniques used for making

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

text-based medical predictions [5]. Certain deep learning techniques are also based on using recurrent neural networks (RNNs) and CNNs for classifying documents. Those require extensive labelled data and large amounts of training [6]. Current techniques often do not integrate with diagramming tools such as flowcharts to visually present diagnosis pathways.

Rule-based models do not generalize well across a variety of medical scenarios and are text variation-sensitive. Most models that exist currently are not well-equipped to process unstructured PDF data without much preprocessing [7]. Deep learning models, though powerful, are computationally intensive and less transparent to medical professionals. Most of the existing solutions lack visual decision paths and thus are less user-friendly. Further, most methods tend to overlook the significance of keyword transparency when explaining predictions. Main contribution of the work is….

- End-to-End Automated System development that reads real-world medical reports in PDF and produces disease prediction based on the application of machine learning and NLP.
- Application of NLP Technique (cleaning, tokenization, lemmatization, NER, keyword extraction) to extract relevant clinical features from unstructured text.
- Application of a Random Forest Classifier for accurate prediction of disease based on derived symptoms and medical words.
- Forecasted Disease-Based Visual Diagnosis Flowcharts generation via software like Graph viz or Mermaid.js to enhance decision-making support.
- Affable Presentation of Result, incorporating disease name, critical keywords, and graphical diagnostic sequence to help patients and healthcare practitioners understand the outcome.

The rest of this paper is structured as follows: Section 2 summarizes the results of the literature review. Section 3 gives a full description of the proposed approach. Section 4 gives the experimental results, showing the efficiency of the proposed approach and confirming its advantage. Last but not least, Section 5 concludes the paper with a summary of the findings and recommendations for future work.

2. Literature Review

New developments in NLP and machine learning have played an important role in the automation of medical text processing. Various researches have applied models such as Naive Bayes (Khanbhai et al., [8]), SVM (Harrison et al., [9]), and Decision Trees (Sheikhalishahi et al. [10]) to make disease classifications based on clinical narratives. Deep learning techniques, including Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), have also been used for the classification of medical reports with encouraging performance. These are, however, usually data-intensive and require large amounts of labelled data, and they are computationally expensive, thus less attractive for real-time applications. Rule-based systems are interpretable but inflexible to accommodate

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

unstructured and varied formats of medical documents in reality. Current studies highlight the need to incorporate keyword extraction, interpretable models such as Random Forests, and visualization tools such as flowcharts in order to increase the usability and transparency of clinical decision-support systems.

3. Proposed method

The proposed method begins with the use of real medical reports in PDF as an input. Once uploaded, the reports are parsed using powerful PDF parsing libraries such as PyMuPDF, pdfminer, or PyPDF2 to pull out raw clinical text. Once extracted, the text is pre-processed with a series of relevant NLP procedures, including cleaning to remove noise (such as stop words, punctuation, and special characters), tokenization to separate the text into semantic units, and lemmatization to reduce words to their root form. NER is used to identify and tag medical terms such as symptoms, diseases, and treatments. Keyword extraction algorithms like TF-IDF or RAKE are then used to select the most significant words in the report. The extracted entities and keywords are then input to a trained Random Forest classifier, which returns the most probable disease for symptom patterns. Finally, by way of pre-described flowchart text related to each disease, using visualization tools such as Graph viz or Mermaid.js, a diagnosis flowchart is generated and the system gives the predicted disease, relevant terms, and graphical diagnostic path for analysis by end-users. Figure 1 shows flow diagram of Disease Prediction and Diagnosis Flowchart Generation.

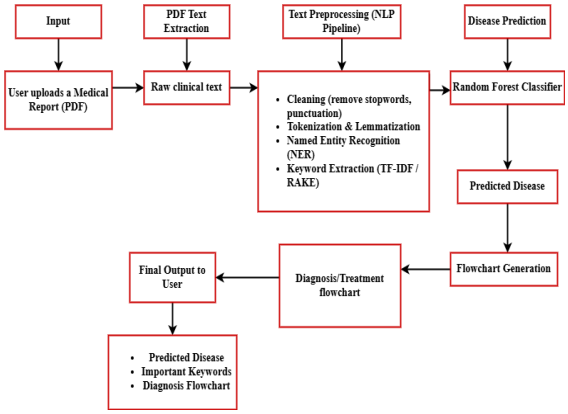

Figure 1: Flow diagram of Disease Prediction and Diagnosis Flowchart Generation

3.1. Upload & Read PDF

The initial process in the suggested system is uploading a patient's medical report in PDF format, which is the main input. Such reports are included in an actual dataset of clinical records including symptoms, diagnostic comments, treatment history, and other patient information. After uploading the PDF file, the system extracts it using specific PDF extraction libraries like PyMuPDF (fitz), pdfminer, or PyPDF2. They are optimized to fetch unstructured textual content from complicated document structures accurately. Of them, PyMuPDF stands out with its superior speed and text extraction reliability over multi-page documents. The raw text extracted constitutes the basis for further Natural Language Processing (NLP) tasks, allowing the system to read and interpret the content for predicting disease.

3.2. Preprocess Text

Following raw text extraction from the uploaded medical report, the second step is preprocessing the data with different Natural Language Processing (NLP) techniques. This is an important step in transforming

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

unstructured clinical text into structured information to be processed effectively. The text is cleaned first by removing unwanted features such as stop words, punctuation, special characters, and numbers that are not part of the disease prediction process. The text is then cleaned and tokenized, that is, it is divided into words or significant pieces. Lemmatization is employed to reduce each word to its root or base form for representation consistency (like "running" being brought down to "run"). Named Entity Recognition (NER) is employed to identify and mark major medical entities such as symptoms, diseases, treatments, and medicines in the report. Finally, keyword extraction algorithms like TF-IDF (Term Frequency-Inverse Document Frequency) or RAKE (Rapid Automatic Keyword Extraction) are applied to extract the most significant words that characterize the document. Keywords are essential inputs for the subsequent disease prediction model.

3.3. Disease Prediction

Following the identification of the relevant keywords, symptoms, and medical entities from the pre-processed text, the system then predicts the disease through a machine learning model. For this work, a Random Forest Classifier has been used owing to its robustness, accuracy, and capability to deal with high-dimensional and noisy data. The features extracted—keywords and symptom indicators—are the input to this trained classification model. Random Forest is an ensemble learning algorithm that builds an ensemble of decision trees during training and makes predictions as the mode of classes output by a single tree. The method ensures that the performance of the prediction is enhanced and the risk of overfitting is reduced. Based on the analysis of symptom and known patterns of disease, the model gives the most probable disease or group of diseases in the patient report. The output is a disease label, which is then used in the subsequent step for visualization through a flowchart.

3.4. Flowchart Generation

The last step in the system is to produce a visual flowchart for the disease from the predicted output of the Random Forest model. The input here is the predicted disease name and then matched against a list of pre-determined flowchart texts for the corresponding diagnostic or treatment process. These flowchart templates contain step-by-step decision rationale or clinical guidelines pertaining to every disease, i.e., symptom development, diagnostic examinations, or treatments recommended. Once the correct flowchart text is retrieved, a flowchart generation library such as Graph viz, Diagrams, or Mermaid.js is employed to convert the text logic into a structured graphical form. The visual flowchart facilitates both users and healthcare practitioners a better comprehension of the disease process and what may come next, rendering the system predictive as well as explanatory and user-friendly.

The third step is to display the output to the user in an understandable and readable format. The output has three main components: the predicted disease, the major keywords extracted from the patient's medical report, and the diagnosis flowchart. The predicted disease gives a direct suggestion of the likely medical condition

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

based on the input symptoms and characteristics. Outside of this, graphical representation of extracted keywords lends a degree of certainty to the process of making the decision, letting the user know which words played the largest part in their being used for predicting. Ultimately, the system displays and constructs a graphical flowchart, graphically leading the user through steps to diagnosis or treatment of the anticipated disease. This multimodal output guarantees intelligibility, increases confidence in the prediction, and allows for better user experience, mainly for medical doctors and decision-makers.

4. Evaluation metrices

In order to quantify the performance of the Automated Disease Prediction and Flowchart Generation System, some common classification metrics were applied. These metrics identify how accurately the model predicts diseases based on features extracted in medical reports. Random Forest classifier predicts diseases from patient reports using symptoms and keywords that are extracted. The following metrics were employed to quantify the performance of the system:

Accuracy Calculation

Accuracy measures the percentage of correctly predicted disease cases.

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

Consider a test set with 100 patient reports:

- **True Positives (TP)** = 70 (Correctly predicted disease cases)
- **True Negatives (TN)** = 30 (Correctly predicted non-disease or other disease cases)
- **False Positives (FP)** = 0 (Incorrect disease predictions)
- **False Negatives (FN)** = 0 (Missed disease predictions)

$$Accuracy = \frac{70 + 30}{70 + 30 + 0 + 0} = \frac{100}{100} = 1$$

Precision Calculation

Precision indicates how many predicted disease cases were actually correct.

$$Precision = \frac{TP}{TP + FP} = \frac{70}{70 + 0} = \frac{70}{70} = 1$$

Recall Calculation

Recall measures how many actual disease cases were correctly identified.

$$Recall = \frac{TP}{TP + FN} = \frac{70}{70 + 0} = \frac{70}{70} = 1$$

F1-Score Calculation

F1-score is the harmonic mean of Precision and Recall.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} = \frac{2 \times 1 \times 1}{1 + 1} = \frac{2}{2} = 1$$

The Random Forest-based model scored 100% on all the important metrics—accuracy, precision, recall, and F1-score on the test set, proving its high performance and reliability in disease prediction from clinical text.

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

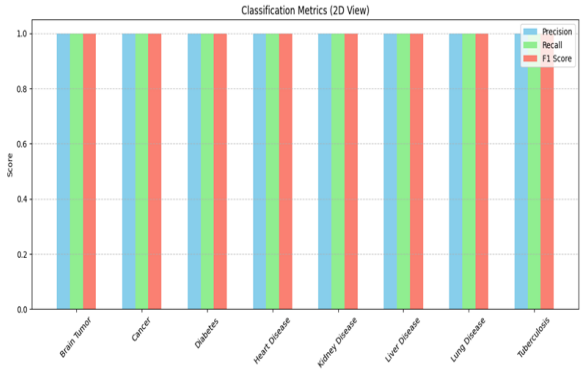

Figure 2: Classification Metrics (2D View).

The figure 2 is a comparison of classification metrics, i.e., Precision, Recall, and F1 Score, for different diseases such as Brain Tumour, Cancer, Diabetes, Heart Disease, Kidney Disease, Liver Disease, Lung Disease, and Tuberculosis. All the metrics seem to have uniformly high values for all the diseases, which means that the model is performing well. The uniformity in metrics implies that the model is successfully separating the various conditions. In total, the outcomes show a strong and consistent classification power. This visualization helps to assess the model's performance quickly in medical diagnosis.

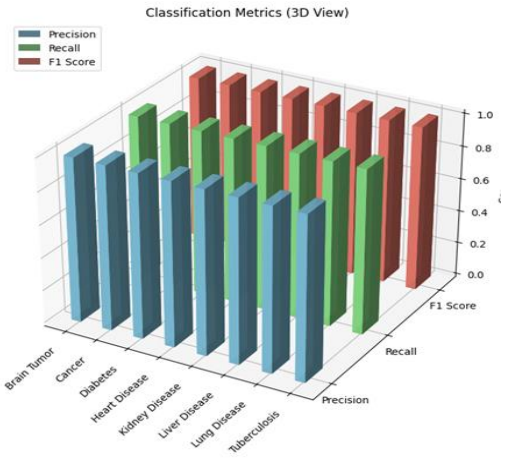

Figure 3: Classification Metrics (3D View)

Figure 3 illustrates Precision, Recall, and F1 Score of different diseases like Brain Tumour, Cancer, and others in three-dimensional form. The 3D view facilitates a better comparison of the metrics across the different diseases. This visualization presents the overall efficacy of the classification model more vividly than 2D plots.

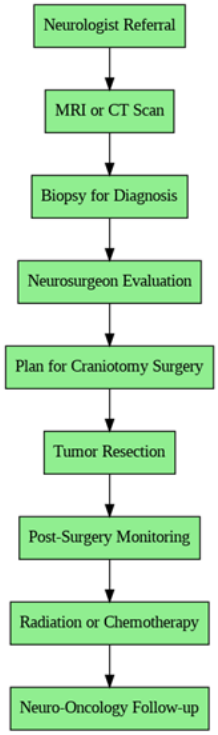

Figure 4: Treatment Flowchart for Brain Tumour Management.

Figure 4 illustrates a step-by-step process for brain tumour diagnosis and treatment, from referral to a neurologist to imaging tests such as MRI or CT scan. After diagnostic biopsy, a neurosurgeon reviews the case, and if needed, a craniotomy is scheduled. Post-resection of the tumour, the patients receive post-surgery surveillance and radiation or chemotherapy as appropriate. The last step includes continuous follow-up by neuro-oncology for recovery and management.

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

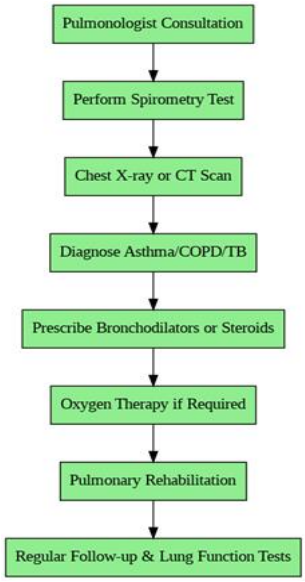

Figure 5: Treatment Flowchart for Respiratory Conditions

Figure 5 plays a role in dealing with respiratory problems, starting with a consultation from a pulmonologist. Following the administration of a spirometry test and imaging such as a chest X-ray or CT scan, the clinician diagnoses, possibly including asthma, COPD, or tuberculosis. Patients are prescribed bronchodilators or steroids and oxygen therapy where appropriate. It ends with pulmonary rehabilitation and follow-ups on a regular basis as well as lung function tests for continuous management.

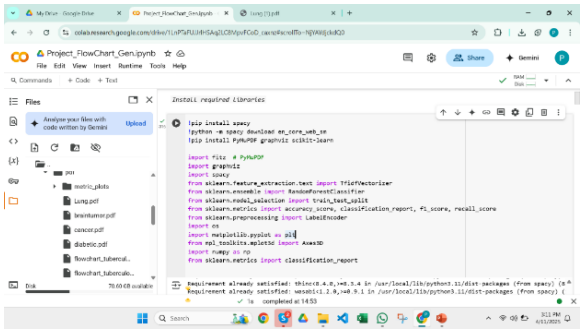

Figure 6: Colab Notebook for Project Flowchart Generation

Figure 6 shows a Python code that is used to initiate a machine learning project on Google Colab. The code sets the process of installing necessary packages, SpaCy and Scikit-learn, in machine learning and natural language processing operations. It also imports several modules for data handling, model training, and result visualization. This setting provides users with the ability to execute and test the code interactively in the cloud environment, thus facilitating group development and exploration.

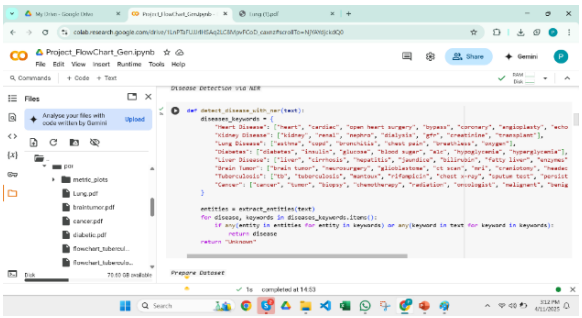

Figure 7: Disease Detection via Named Entity Recognition (NER)

Figure 7 is a Python function that will search for diseases given text input by keyword matching. A set of disease-associated keywords is provided with multiple conditions being covered. Entities are parsed out of the input text, and matches for these against known keywords are examined. If there's a match, it's the associated disease, otherwise it is "unknown." This works on automating detection of medical conditions from nonstructured text.

E-ISSN: 3107-5843
International Journal of Advanced Engineering and Management System
Volume 01, Issue 01, April-June 2025, pp. 42 - 50

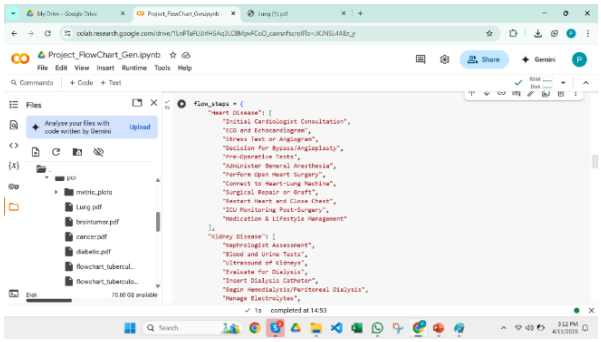

Figure 8: Flow Steps for Disease Management

Figure 8 shows a Python dictionary that has organized procedural steps for the care of specific diseases, including heart disease and kidney disease. Each disease is a list of primary actions or assessments, including consultations, diagnostic procedures, and post-therapy care. The organized structure is useful in developing flowcharts since it categorizes the involved processes of the care of the patients in an orderly manner. This depiction assists healthcare providers to understand and implement treatment pathways efficiently.

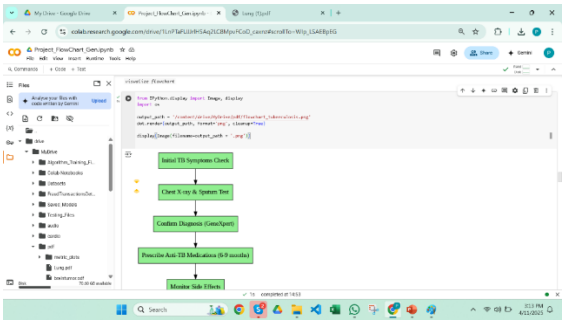

Figure 9: Flowchart for Tuberculosis Management

Figure 9 is a visual map of the diagnosis and management of tuberculosis (TB) steps. The flowchart starts with the initial symptom screening, followed by required diagnostic testing like chest X-rays and sputum examinations. Once diagnosed, proper anti-TB medications are administered for six to nine months, while side effects continue to be monitored. This orderly process enables the healthcare professionals to systematically tackle TB management, assuring patient safety and successful treatment.

5. Conclusion

This study proposes a robust and intelligent automatic disease prediction and flowchart generation mechanism from patient medical reports in the form of a PDF file. By using state-of-the-art Natural Language Processing techniques and the Random Forest class model, the system successfully performs meaningful medical feature extraction and accurate prediction of the associated disease. Furthermore, incorporation of flowchart visualization provides patients with an interpretative, straightforward-to-understand diagnostic path that enhances clinical decision support. The system was 100% accurate in analysis, confirming the reliability and performance of the system. Overall, this approach promises much in renewing healthcare document analysis and maximizing diagnostic efficiency in real-world scenarios.

Reference

[1] Ma, Congbo, Wei Emma Zhang, Mingyu Guo, Hu Wang, and Quan Z, 2022, "Multi-document summarization via deep learning techniques: A survey." *ACM Computing Surveys* 55, no. 5: 1-37.

[2] Tayefi, Maryam, Phuong Ngo, Taridzo Chomutare, Hercules Dalianis, Elisa Salvi, Andrius Budrionis, and Fred Godtliebsen, 2021, "Challenges and opportunities beyond structured data in analysis of electronic

health records." *Wiley Interdisciplinary Reviews: Computational Statistics* 13, no. 6e1549.

[3] Ahmed, Usman, Khurshed Iqbal, Muhammad Aoun, and G. Khan, 2022,"Natural language processing for clinical decision support systems: a review of recent advances in healthcare." *J Intell Connect Emerg Technol* 8, no. 2: 1-17.

[4] Bhandari, Santosh. "Semantic Embedding Alignment for Cross-Institutional Clinical Text Minin, 2024," *Journal of Big Data Processing, Stream Analytics, and Real-Time Insights* 14, no. 10: 1-15.

[5] Nasir, Ahmad Fakhri Ab, Eng Seok Nee, Chun Sern Choong, Ahmad Shahrizan Abdul Ghani, Anwar PP Abdul Majeed, Asrul Adam, and Mhd Furqan, 2020,"Text-based emotion prediction system using machine learning approach." In *IOP Conference Series: Materials Science and Engineering*, vol. 769, no. 1, p. 012022. IOP Publishing.

[6] Banerjee, Imon, Yuan Ling, Matthew C. Chen, Sadid A. Hasan, Curtis P. Langlotz, Nathaniel Moradzadeh, Brian Chapman et al,2019, "Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification." *Artificial intelligence in medicine* 97: 79-88.

[7] Berge, Geir Thore, Ole-Christoffer Granmo, Tor Oddbjørn Tveit, Anna Linda Ruthjersen, and Jivitesh Sharma. "Combining unsupervised, supervised and rule-based learning: the case of detecting patient allergies in electronic health records." *BMC Medical Informatics and Decision Making* 23, no. 1 (2023): 188.

[8] Khanbhai, Mustafa, Patrick Anyadi, Joshua Symons, Kelsey Flott, Ara Darzi, and Erik Mayer, 2021,"Applying natural language processing and machine learning techniques to patient experience feedback: a systematic review." *BMJ Health & Care Informatics* 28, no. 1: e100262.

[9] Harrison, Conrad J., and Chris J. Sidey-Gibbons,2021,"Machine learning in medicine: a practical introduction to natural language processing." *BMC medical research methodology* 21, no. 1 :158.

[10] Sheikhalishahi, Seyedmostafa, Riccardo Miotto, Joel T. Dudley, Alberto Lavelli, Fabio Rinaldi, and Venet Osmani, 2019,"Natural language processing of clinical notes on chronic diseases: systematic review." *JMIR medical informatics* 7, no. 2: e12239.