Student ID: 201151261

Enriched Symptom Based Disease Classification Using Text Analytics and Web Scraping

Background

In the field of healthcare, patient outcomes often depend upon the timely and accurate diagnosis of diseases. Early detection allows earlier initiation of treatment, leading to improved prognosis and overall patient well-being. However, diagnosis based solely on patient-reported symptoms and clinical assessments, including diagnostic tests, can pose significant challenges for medical professionals. As well as being time consuming, the subjective nature of symptoms, coupled with the complexity of certain medical conditions means the diagnosis process is prone to errors and oversights [1].

Diagnostic errors are the most common type of medical errors worldwide, contributing to adverse patient prognosis, low patient satisfaction and suboptimal quality of care. Factors such as the subjective interpretation of symptoms, varying levels of clinical expertise among practitioners and the strain on healthcare services has exacerbated the risk of diagnostic errors. With the difficulties many healthcare services face globally, limited resources mean physicians are forced to swiftly assess and diagnose patients, compromising their ability to gain a comprehensive understanding of the patients' symptoms. As a result, many rare and uncommon conditions are missed leading to delays in treatment and poor patient outcomes.

Symptoms are subjective indications of diseases and can be difficult to manage, presenting a challenge for healthcare providers to interpret accurately. Acknowledging the pivotal role of symptom science in enhancing patent care, the National Institute of Nursing Research highlighted the importance of advancing our understanding of the symptoms of chronic illnesses to improve the quality of life across diverse patient populations [3].

This outlines a clear need for reliable diagnostic aides to assist medical professionals to assess and manage patients' symptoms, enhancing diagnostic accuracy and optimizing treatment outcomes.

The increased popularity and convenience of Electronic Health Records (EHR's) globally has seen a shift from traditional paper-based record keeping to digitized patient files. These records consist of comprehensive real-world data on an individual's medical profile [6]. The records can contain information on medical histories, family histories, diagnoses and treatments and undertaken procedures. The data in EHR's can be categorized into two main formats; structured data and unstructured data [3]. Structured data accounts for a small proportion of all EHR data and consists of billing information, diagnosis history, prescribed medications, lab test results and medical interventions. The majority of the data, approximately 80% [12], is unstructured free-text. This can include anything from admission documents, discharge summaries, nursing notes and primary physician notes. This unstructured data is also rich in patient specific symptoms and reactions or suitability to prescribed medications. Traditionally, extracting meaningful insights form unstructured data posed a challenge, limiting its use in healthcare research and clinical decision-making.

Developments in text mining techniques now allow for this data to be extracted from EHR's and be used as an extensive and valuable source of symptom and drug interaction data. Leveraging Natural Language Processing (NLP) algorithms and Machine Learning (ML) techniques researchers can now enrich disease classification and detection research, allowing the integration of predictive analysis in healthcare.

Beyond EHR's, there are extensive online resources available, including medical websites, clinical journals, articles and educational texts. These sources offer comprehensive insights into various diseases, their diagnostic criteria, symptomatology and treatment approaches.

Contribution on Knowledge & Importance

An area of investment of the EPSRC is AI and data science for informatics and healthcare technologies to improve healthcare delivery and patient outcomes. One of the high priority goals funding is directed towards is the transformation of health through the delivery of personalised medicine with early disease detection and machine-learning based diagnosis. The proposed project aims to tackle disease detection with the use of alternative models.

Much of the research into automated disease classification and detection is based on Large Language Models (LLM's) such as BERT, MCN-BERT, RoBERTa, GPT and T5 to name a few. MCN-BERT (Medical Concept Normalization) BERT is commonly used when building a medical diagnosis or classification tool. Mentions of medical concepts such as diseases, symptoms, medications, procedures etc. are mapped to standardised codes from controlled medical terminologies such as the ICD, for example. The BERT model has been pre-trained on an extensive corpus of text data. The MCN-BERT model learns to capture the contextual semantics of medical concepts and predict the appropriate normalization. The BERT model is a comprehensive NLP architecture, the enhancement with medical data leads to excellent performance in classification of diseases. For example, an accuracy of 99.58% was achieved by Hassan et al [7] by the MCN-BERT model optimized with AdamP on the symptom2disease dataset [13] used in this study.

Despite their high level of accuracy, these transformer based models are not always practical. Such models demand high computational resources to train to an acceptable level which can be costly. Additionally, since there can be billions of parameters, fine tuning the model can be slow and computationally intensive. Furthermore, such models require large amounts of memory to handle parameters during training and inference [7]. Due to this, these models are reserved for those who have access to these resources to continue research and further development, making them highly inaccessible to those who do not.

Therefore, this research will explore time and computationally effective models and enrich them with text mined data to replace the extensive pre-training the BERT model, and its variants, are subject to. A similar study was conducted by Smith et al [8] using PheNorm to classify COVID-19 infections based on patient reported symptoms. PheNorm is a text mining approach which focuses on clinical research articles to identify relevant medical concepts. This study had a focus towards COVID-19 symptoms and literature published surrounding the novel virus. Other research enriching classification models with text mining techniques have been focused on data from a specific domain of medicine, such as orthopaedics [9], Parkinson's [10] and lung cancer [11]. A general approach has not been widely studied.

Further research in this field may contribute to biomedical research by providing insight into underlying mechanisms and manifestations of diseases from patient reported symptoms. Leveraging NLP techniques on diverse healthcare data can identify new patterns, associations and novel insights into disease development, progression and treatment responses. As the domain of medicine becomes more technologically advanced, including genomics, proteomics and clinical phenotypes in health records, discovery of novel disease biomarkers will become more likely, facilitating early detection.

The progress of text mining and the advancement of computationally efficient models for disease classification will also have significant implications for personalised healthcare. By leveraging NLP algorithms on diverse and extensive healthcare datasets, the research holds the potential to revolutionize personalised medicine. Through identification of novel biomarkers and incorporating individual lifestyle factors, genetic profile and treatment responses, personalised healthcare interventions can be tailored to meet the unique needs and preferences for each patient. This can enhance treatment effectiveness as well as minimise adverse effects and repeated appointments.

Student ID: 201151261

Research Hypothesis and Objectives

This study aims to investigate whether computationally inexpensive text classification models enriched with text mining techniques can be developed for disease classification and achieve a comparable accuracy to state-of-the-art LLM, transformer-based models.

Although transformer-based models such as BERT, T5 GPT, and its variants, achieve high accuracy on classification tasks, training and optimization is computationally expensive, requiring additional GPU, which is often not freely available. There are rental options, but there are steep costs associated with them. Additionally, even with more computational power, training and optimization takes a long time, often several days to weeks, meaning experimentation is not easy.

The option to use such models is therefore not open to everyone since many researchers do not have the resources to be able to train a large model for several days without interrupting run-time. An alternative for those with limited computational resources is to develop less taxing models such as convolutional neural networks (CNN's), recurrent neural networks (RNN's), Naïve Bayes, Support Vector Machines, etc. The pre-training from BERT models can be replicated by enriching training data with text mined data specific to the classification task, improving model accuracy. This study will investigate whether computationally inexpensive models can perform as well as state-of-the-art models without the burden of computation and time commitments. A specific method or an ensemble method may achieve comparable performance with sufficient training and optimization.

Much of the recent literature has a focus on working with new extensive transformer-based NLP models since high performance is expected. Development of simpler and computationally cheaper models has ceased in favour of LLM's. There is some literature on enhanced classification of such models on specific data, however a general dataset covering many diseases has not been explored, leaving a gap in the literature.

The objectives of this study cover a systematic exploration of the effectiveness of integrating text mining techniques into the training process of text classification model for disease classification. Firstly, the study aims to establish a baseline performance by evaluating the text classification models using the raw data exclusively, without the addition of additional data. Performance will be evaluated using established performance metrics such as accuracy, precision, recall and the F1 score.

Next, text mining techniques will be integrated, including web scraping to extract supplementary data relevant to disease symptoms, risk factors and treatments. The additional data will be incorporated into the training corpus to enrich the learning process of the classification models. Following training, performance will be evaluated using established performance metrics and compared to previous results. This will discern the influence of text mining techniques on disease classification.

Lastly, the study will cover a comparative analysis between performance of models using raw data and models with raw data enriched with text mining techniques. This evaluation will scrutinize differences in accuracy, precision, recall and F1 score to discern whether the addition of web-scraped data enhances the classification performance.

Pilot Study

Data from the symptom2disease dataset available on Kaggle [13] was used for the piot study. The dataset focuses on the relationship between symptoms and diseases. There are 1200 data points in this dataset covering 24 diseases, each with 50 rows of text descriptions of symptoms. Niyarr Barman curated the dataset by compiling extracted information from various medical sources, including research papers, medical literature, and clinical databases [1]. The range of diseases within the dataset is inclusive of infectious diseases, acute illness and chronic conditions.

Of the 24 diseases, 11 were selected for the pilot study maintaining a diverse range of diseases; acne, arthritis, chicken pox, common cold, dengue, diabetes, jaundice, migraine, pneumonia, psoriasis and urinary tract infection. This subset of diseases covers skin disorders, viral infections, respiratory illnesses, inflammatory conditions and metabolic disorders, providing a comprehensive set of symptoms and health concerns making it ideal for a pilot study. The compiled dataset used in the study has 550 datapoints, with 50 symptom descriptions for each disease. Symptom descriptions are approximately a single sentence.

Text mining methods for web-scraping were employed to collate information on symptoms associated with the diseases in the pilot dataset. SketchEngine was used conduct a web search, altering the filters and parameters for the most relevant data. The search terms used were the '[disease name]', 'symptoms' and '[disease name] symptoms'. The 'size and relevance' filter was altered to 'relevant' to ensure more of the seed words were included in the webpage, reducing the extent of irrelevant data. A maximum of 20 webpages was also specified to ensure the enrichment corpus didn't grow too large. Common websites found by the web search included NHS Inform, NHS England, Web MD, NI direct and Mayo Clinic, which are some of the most popular websites with details on various diseases, common and rare symptoms, possible treatments and advice.

The keyword and terminology search tool in SketchEngine was used to highlight the most relevant keywords and multi-word terms from each corpus. The words extracted by this search tool are more relevant to the diseases and encompass frequently used terms to describe the symptoms in comparison to the raw corpus. Of these keywords and phrases the top 50 of each were collated to be used as enrichment data for the classification task.

Both the labelled disease-symptom data and enrichment data were cleaned in Python and enrichment data was combined with the original dataset. For standardisation, all characters were replaced with lowercase characters and any punctuation was removed.

The text was then tokenized to convert text into lists of tokens. Next stop words were removed to ensure the remaining text data only consisted of relevant words and symptoms. Words such as 'I', 'me', 'have' etc. are commonly used in English language in speech and writing, however for the purpose of the disease classification task they are not needed. Following this, the words were lemmatized to reduce them to their base, normalizing the text and improving text representations. Within the enrichment data, many keywords and multi-word expressions captured the same symptom of the disease, leading to repeated words. To ensure high frequency words do not get dismissed, duplicated words were removed.

For all models, performance metrics such as accuracy, precision, recall and F1 score were calculated on unseen test data (70-30 split), to compare results between models. Accuracy is a reliable metric when classes are balanced. Precision measures the proportion of correct predictions with respect to all positive predictions. Conversely, recall measures performance across a class, considering the proportion of correctly classified instances and incorrectly classified instances. The F1 score is the harmonic mean of the precision and recall, providing a balance between the two metrics.

Precision is favoured over recall for medical diagnosis tasks since the consequences of false positives are high. Incorrect or missed diagnoses can lead to deteriorated health of the individual and delayed or incorrect treatment administration. This works against the goal of improved patient care and outcomes.

The first computationally inexpensive model used for enriched disease classification is multinominal Naïve Bayes'. This classifier is simple and quick to train, making it well suited for text classification. The model is effective for datasets with large feature space, making it suitable for large text datasets. Classification was performed for both the original dataset and the original dataset with additional web scraped data. Padding and vectorising was necessary since sequence lengths differed between training and test instances.

Performance of the original model was poor across all metrics, specifically precision and F1 score with 0.54 and 0.38 respectively. Model performance improved slightly with enriched learning with web-scraped data. Precision and F1 score for this model rose to 0.69 and 0.62 respectively. The confusion matrix for the original model (Fig 1 in appendix) shows low rates of true positives across the diagonal, except for jaundice and psoriasis. The confusion matrix reveals that the model is misclassifying many symptoms as psoriasis. The enriched classification model has a higher proportion of true positives (Fig 2 in appendix), reflected in the increase in precision and F1 score. However, there are still some misclassifications between acne and psoriasis, which are both skin conditions with similar symptoms.

The next model used for classification is the CNN classification model. Although this module is more computationally expensive in comparison to Naïve Bayes, it can still be trained with limited computational resources in a reasonable time. This model does not require any tokenisation or pre-processing since feature engineering is handled internally, making the model suitable for large corpora of text. Finally, CNNs are robust towards differences in sequence length, eliminating the need to pad sequences and allowing for more enrichment data to be used when training, without compromising runtime and model complexity.

Again, both the original data and enriched data were used to train the model and performance was evaluated. The CNN model performed poorer in comparison to Naïve Bayes on the original dataset with precision of 0.37 and F1 score of 0.17. The enriched classification model showed significant improvement, achieving 0.96 precision and F1 score. The confusion matrices for these models (Fig 3 & 4 in appendix) indicate the model is misclassifying many diseases as chicken pox with a low true positive rate for every other disease. True positive rates for the enriched model is much higher, with only a small proportion of misclassifications. The precision rate for this model is much closer to the performance of state-of-the-art transformer based models.

Logistic regression (LR) was the final classification model used to classify the symptom-disease dataset. This model provides more interpretability since users can trace the contribution of each feature to the classification decision. This model is also computationally efficient and can easily handle large large-scale classification tasks, meaning scaling up the dataset will be handled well. Before training, words are embedded using term frequency-inverse document frequency (TF-IDF).

The logistic regression model outperformed the other models on the original dataset with precision and F1 scores 0.86 and 0.79 respectively. The confusion matrix (Fig 5) for this model shows high true positive rates for all diseases except for dengue which was frequently misclassified.

Performance of the enriched LR model surpasses performance of the original model with precision 0.95 and F1 score 0.94. The confusion matrix of the enriched model (Fig 6), indicates that classifications are almost perfect with the model mistaking instances of dengue as chicken pox and diabetes as urinary tract infections. There are some shared symptoms between these diseases causing the model confusion. Incorporation of additional enrichment data may further improve model performance.

Student ID: 201151261

Programme and Methodology

This section outlines the research methodology and work programme for the project. The duration of the research will be 6 months consisting of project setup and planning, data collection and pre-processing, model development and training, evaluation and validation, fine tuning and optimization and finally documentation and reporting.

Month 1: Project setup and planning

To commence the research project, the research objectives and hypotheses will need to be clearly outlined, reviewing relevant and recent literature on classification models. The primary aim of this research is to address the need for a reliable and accessible diagnostic aid that can be deployed on a wide scale across the healthcare industry. This stems from the challenges faced by healthcare professionals, particularly primary care physicians where appointment slots are shortening and less time is spent with the patient. The goal is to develop a diagnostic tool that provides essential information to doctors. Additionally, the insights collected regarding the relationship between symptoms and diseases can support nurses and patients manage their conditions. To achieve this, computationally efficient classification models will be trained using additional data to enhance classification precision. A collection of high performance models can be ensembled with a voting system to ensure higher classification performance close to the performance of computationally expensive transformer-based models, which are majorly inaccessible. A secondary aim of the study is to investigate relationships between diseases and symptoms to be able to provide personalised healthcare options based on genetics, preferences and lifestyle factors, optimising patient outcome.

Month 2: Data Collection and Pre-processing

Data collection and preparation will be the focus of the second month of the research project. Access to an extensive anonymised and unidentifiable medical database will be required by collaborating with government agencies, research repositories and healthcare organisations. Unstructured free-text data will be extracted from the EHR data and filtered for patient reported symptoms with a corresponding diagnosis. Close collaboration with domain experts can verify the accuracy of the extracted data to ensure a reliable classification model. Next, data will need to be collated for enrichment, leveraging advanced text mining techniques, such as SketchEngine, to gather information on diseases, common and rare symptoms, risk factors and treatment options. Extensive use of medical literature published in articles, informative health websites, digitised textbooks and International Classification of Diseases (ICD) codes will be considered to enrich the dataset. Subsequently, the data will be meticulously cleaned and pre-processed to prepare for model training. Techniques such as tokenization, stop word removal and lemmatisation will be employed, followed by appropriate embedding and vectorisation techniques.

Month 3: Model Development and Training

A comprehensive approach will be taken to implement and train a diverse set of computationally efficient classification models. In addition to Naïve Bayes, CNN and Logistic Regression, models such as Random Forest, Support Vector Machines (SVM), Gradient Boosting Machines (GBM), Decision trees, K-Nearest Neighbours (KNN) and Multilayer Perceptron (MLP) will be incorporated into the training process. Furthermore, the additional text mined data will be fed into the training process of each model to enhance their capability to capture complex relationships between diseases and their symptoms. Hyperparameters of each model will be fine-tuned for each model to optimise validation performance. During training, metrics will be closely monitored and tweaked to

identify the models with the highest validation performance. This systematic approach will ensure the development of highly accurate and robust classification models.

Month 4: Evaluation and Validation

To ensure reliability of the classification models, a rigorous evaluation process will be conducted. The performance of each model will be assessed using held back validation data, allowing for unbiased estimation of their performance. Established evaluation metrics including accuracy, precision, recall, F1 scores and confusion matrices will be computed for each model. These metrics will provide crucial insight into each models' classification capabilities and provide a means of performance comparison between models. Furthermore, the findings will be validated through cross-validation techniques and additional testing, enhancing the credibility and generalisability of the results.

Month 5: Fine-Tuning and Optimisation

The penultimate month of the study will focus on fine-tuning. Following extensive evaluation, the selected model, or collection of models, will undergo a meticulous fine-tuning process to optimise performance further. The fine-tuning will be guided by validation results and feedback from the initial evaluation, allowing for targeted adjustment to improve the effectiveness and efficiency of the model or models. Key aspects of the model will be evaluated to ensure computational and time efficiency, thus ensuring accessibility. This includes parameters of the model, the architecture and training strategies, with the aim of enhancing disease classification. The aim is to refine the model to ensure performance is widely accepted and reliable in real-world applications in healthcare settings. Additionally, sensitivity analysis will be conducted to evaluate the robustness of the model to variations in input data or parameters, providing valuable insights into its reliability and generalisability across different scenarios. Through these iterative refinement processes, the aim is to develop a classification model that, not only achieves high performance comparable to state-of-the-art transformer-based models, but also demonstrates resilience and adaptability in handling diverse datasets and real-world challenges.

Month 6: Documentation, Reporting and Dissemination

In the final month of the research project, the methodology, results and conclusions of the study will be formally documented in a comprehensive report or research paper. This will include additional statistical analysis as well as visual representations, such as tables and figures, to summarise and present noteworthy findings and insights. The final report will be peer-reviewed with other domain experts to solicit feedback and improvement suggestions before publication and presentation. Within the report, there will be a reflection on the study process as well as considerations. Future areas of research based on the findings as well as suggestions for improvement will also be proposed.

Student ID: 201151261

Workplan Diagram

The study will only require a single researcher to carry out the tasks as the workplan outlines. Each section of the project will last one month.

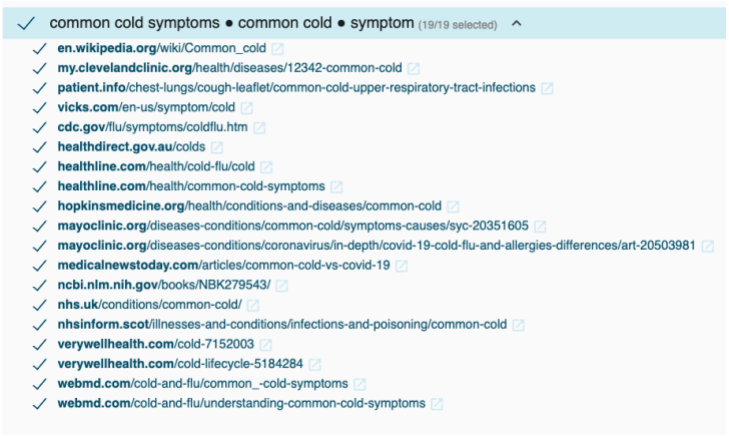
References

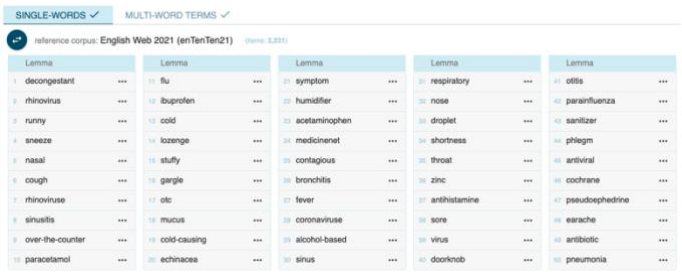
1. Hassan, E., Abd El-Hafeez, T. and Shams, M.Y. (2024). Optimizing classification of diseases through language model analysis of symptoms. Scientific Reports, [online] 14(1), p.1507. doi:https://doi.org/10.1038/s41598-024-51615-5.
2. Dorsey, S.G., Griffioen, M.A., Renn, C.L., Cashion, A.K., Colloca, L., Jackson-Cook, C.K., Gill, J., Henderson, W., Kim, H., Joseph, P.V., Saligan, L., Starkweather, A.R. and Lyon, D. (2019). Working Together to Advance Symptom Science in the Precision Era. Nursing Research, [online] 68(2), pp.86–90. doi:https://doi.org/10.1097/NNR.0000000000000339.
3. Koleck, T.A., Dreisbach, C., Bourne, P.E. and Bakken, S. (2019). Natural language processing of symptoms documented in free-text narratives of electronic health records: a systematic review. Journal of the American Medical Informatics Association: JAMIA, [online] 26(4), pp.364–379. doi:https://doi.org/10.1093/jamia/ocy173.
4. Chen ES, Sarkar IN. Mining the electronic health record for disease knowledge. Methods Mol Biol. 2014;1159:269-86. doi: 10.1007/978-1-4939-0709-0_15. PMID: 24788272.
5. Ross MK, Wei W, Ohno-Machado L. "Big data" and the electronic health record. Yearb Med Inform. 2014 Aug 15;9(1):97-104. doi: 10.15265/IY-2014-0003. PMID: 25123728; PMCID: PMC4287068.
6. Knevel R, Liao KP. From real-world electronic health record data to real-world results using artificial intelligence. Ann Rheum Dis. 2023 Mar;82(3):306-311. doi: 10.1136/ard-2022-222626. Epub 2022 Sep 23. PMID: 36150748; PMCID: PMC9933153.
7. Hassan, E., Abd El-Hafeez, T. & Shams, M.Y. Optimizing classification of diseases through language model analysis of symptoms. *Sci Rep* **14**, 1507 (2024). https://doi.org/10.1038/s41598-024-51615-5
8. Smith JC, Williamson BD, Cronkite DJ, Park D, Whitaker JM, McLemore MF, Osmanski JT, Winter R, Ramaprasan A, Kelley A, Shea M, Wittayanukorn S, Stojanovic D, Zhao Y, Toh S, Johnson KB, Aronoff DM, Carrell DS. Data-driven automated classification algorithms for acute health conditions: applying PheNorm to COVID-19 disease. J Am Med Inform Assoc. 2024 Feb 16;31(3):574-582. doi: 10.1093/jamia/ocad241. PMID: 38109888; PMCID: PMC10873852.
9. P. Ketpupong and K. Piromsopa, "Applying Text Mining for Classifying Disease from Symptoms," *2018 18th International Symposium on Communications and Information Technologies (ISCIT)*, Bangkok, Thailand, 2018, pp. 467-472, doi: 10.1109/ISCIT.2018.8587993.
10. Cinzia Palmirotta, Aresta, S., Battista, P., Tagliente, S., Gianvito Lagravinese, Mongelli, D., Gelao, C., Fiore, P., Castiglioni, I., Minafra, B. and Salvatore, C. (2024). Unveiling the Diagnostic Potential of Linguistic Markers in Identifying Individuals with Parkinson's Disease through Artificial Intelligence: A Systematic Review. Brain Sciences, 14(2), pp.137–137. doi:https://doi.org/10.3390/brainsci14020137.
11. Andrew Houston, Sophie Williams, William Ricketts, Charles Gutteridge, Chris Tackaberry, John Conibear. Automated Derivation of Diagnostic Criteria for Lung Cancer using Natural Language Processing on Electronic Health Records: A pilot study. doi: https://doi.org/10.1101/2024.02.20.24303084
12. SyTrue (2015). Why Unstructured Data Holds the Key to Intelligent Healthcare Systems. [online] hitconsultant.net. Available at: https://hitconsultant.net/2015/03/31/tapping-unstrucured-data-healthcares-biggest-hurdle-realized/.
13. www.kaggle.com. (n.d.). Symptom2Disease. [online] Available at: https://www.kaggle.com/datasets/niyarrbarman/symptom2disease?resource=download
14. Li, I., Pan, J., Goldwasser, J., Verma, N., Wong, W.P., Nuzumlalı, M.Y., Rosand, B., Li, Y., Zhang, M., Chang, D., Taylor, R.A., Krumholz, H.M. and Radev, D. (2022). Neural Natural Language Processing for unstructured data in electronic health records: A review. Computer Science Review, 46, p.100511. doi:https://doi.org/10.1016/j.cosrev.2022.100511.

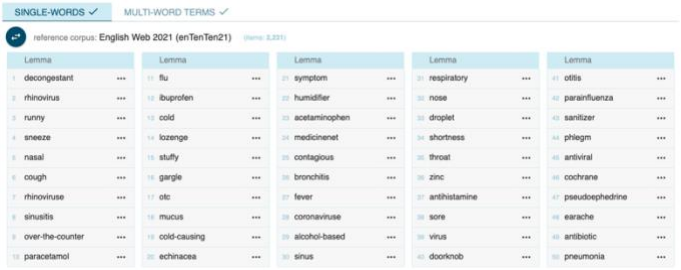
Student ID: 201151261

Appendix

SketchEngine







Python Code

Pre-processing steps applied to text data.

```
# Function for tokenizing each word in a sentence
def tokenize_text(text):
    # Split the text by whitespace to tokenize each word
    words = text.split()
    return words
diseases_df['Symptoms'] = diseases['Symptoms'].apply(tokenize_text)
diseases_df['ws_symptoms'] = diseases['ws_symptoms'].apply(tokenize_text)

stop_words = set(stopwords.words('english'))
# Function to remove stopwords
def remove_stopwords(tokens):
    return [word for word in tokens if word not in stop_words]
diseases_df['Symptoms'] = diseases_df['Symptoms'].apply(remove_stopwords)
diseases_df['ws_symptoms'] = diseases_df['ws_symptoms'].apply(remove_stopwords)

lemmatizer = WordNetLemmatizer()
# Function for lemmatization
def lemmatize_words(tokens):
    return [lemmatizer.lemmatize(word) for word in tokens]
diseases_df['Symptoms'] = diseases_df['Symptoms'].apply(lemmatize_words)
diseases_df['ws_symptoms'] = diseases_df['ws_symptoms'].apply(lemmatize_words)

# Function to remove duplicate words from a list
def remove_duplicates(tokens):
    return list(set(tokens))
# Apply the remove_duplicates function to each list in the 'ws_symptoms' column
diseases_df['ws_symptoms'] = diseases_df['ws_symptoms'].apply(remove_duplicates)

# Encode labels
label_encoder = LabelEncoder()
diseases_df['Disease_Encoded'] = label_encoder.fit_transform(diseases_df['Disease'])
```

Student ID: 201151261

Naïve Bayes model and evaluation metrics

```
[ ] # Initialize Multinomial Naive Bayes classifier
    clf = MultinomialNB()

    # Train the classifier
    clf.fit(X_train_combined_padded, y_train_nb2)
```

```
MultinomialNB()
```

```
# Make predictions on the test set
y_pred_nb2 = clf.predict(X_test_symptoms_padded)

# Evaluate the model
accuracy = clf.score(X_test_symptoms_padded, y_test_nb2)
print("Accuracy:", accuracy)
```

```
Accuracy: 0.6242424242424243
```

```
[ ] # Compute accuracy
    accuracy_nb2 = accuracy_score(y_test_nb2, y_pred_nb2)
    print("Accuracy:", accuracy_nb2)

    # Compute precision
    precision_nb2 = precision_score(y_test_nb2, y_pred_nb2, average='weighted')
    print("Precision:", precision_nb2)

    # Compute recall
    recall_nb2 = recall_score(y_test_nb2, y_pred_nb2, average='weighted')
    print("Recall:", recall_nb2)

    # Compute F1 score
    f1_nb2 = f1_score(y_test_nb2, y_pred_nb2, average='weighted')
    print("F1 Score:", f1_nb2)

    # Compute confusion matrix
    cm = confusion_matrix(y_test_nb2, y_pred_nb2)

    # Plot confusion matrix
    plt.figure(figsize=(7, 6))
    sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=clf.classes_, yti
    plt.xlabel('Predicted labels')
    plt.ylabel('True labels')
    plt.title('Confusion Matrix')
    plt.show()
```

```
Accuracy: 0.6242424242424243
Precision: 0.6899703785788315
Recall: 0.6242424242424243
F1 Score: 0.6240996159651708
```

CNN model and evaluation metrics

```
[ ] # Define the CNN-based model architecture
    model = Sequential([
        Embedding(max_words, 128, input_length=max_length),
        Conv1D(128, 5, activation='relu'),  # Convolutional layer with 128 filters and kernel size of 5
        GlobalMaxPooling1D(),
        Dense(64, activation='relu'),
        Dropout(0.5),
        Dense(len(label_encoder.classes_), activation='softmax')
    ])

    # Compile the model
    model.compile(loss='sparse_categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

```
# Train the model
history = model.fit(X_tr_pad, y_train_cnn1, epochs=20, batch_size=32, validation_split=0.1)
```

```
# Calculate accuracy
accuracy_cnn1 = accuracy_score(y_test_orig, y_pred_cnn1)
print('Accuracy:', accuracy_cnn1)

# Calculate precision
precision_cnn1 = precision_score(y_test_orig, y_pred_cnn1, average='weighted')
print('Precision:', precision_cnn1)

# Calculate recall
recall_cnn1 = recall_score(y_test_orig, y_pred_cnn1, average='weighted')
print('Recall:', recall_cnn1)

# Calculate F1-score
f1_cnn1 = f1_score(y_test_orig, y_pred_cnn1, average='weighted')
print('F1-score:', f1_cnn1)

# Compute confusion matrix
cm = confusion_matrix(y_test_orig, y_pred_cnn1)

# Plot confusion matrix
plt.figure(figsize=(7, 6))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=label_encoder.cl
plt.xlabel('Predicted labels')
plt.ylabel('True labels')
plt.title('Confusion Matrix')
plt.show()
```

```
6/6 [==============================] - 0s 3ms/step
Accuracy: 0.9575757575757575
Precision: 0.9609761583445795
Recall: 0.9575757575757575
F1-score: 0.957803709977623
```

Logistic regression model and evaluation metrics

```
# Define the logistic regression model
logistic_model = Pipeline([
    ('tfidf', TfidfVectorizer()),
    ('logistic', LogisticRegression(max_iter=1000))
])

# Train the model
logistic_model.fit(X_train_lr1, y_train_lr1)

# Access class labels
classes = logistic_model.named_steps['logistic'].classes_
```

```
# Make predictions on the test data
y_pred_lr1 = logistic_model.predict(X_test_lr1)

# Evaluate model performance
accuracy_lr1 = accuracy_score(y_test_lr1, y_pred_lr1)
precision_lr1 = precision_score(y_test_lr1, y_pred_lr1, average='weighted')
recall_lr1 = recall_score(y_test_lr1, y_pred_lr1, average='weighted')
f1_lr1 = f1_score(y_test_lr1, y_pred_lr1, average='weighted')

# Print evaluation metrics
print("Accuracy:", accuracy_lr1)
print("Precision:", precision_lr1)
print("Recall:", recall_lr1)
print("F1 Score:", f1_lr1)

# Compute confusion matrix
cm = confusion_matrix(y_test_lr1, y_pred_lr1)

# Plot confusion matrix
plt.figure(figsize=(7, 6))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=logistic_model.na
plt.xlabel('Predicted labels')
plt.ylabel('True labels')
plt.title('Confusion Matrix')
plt.show()
```

```
Accuracy: 0.9393939393939394
Precision: 0.9504044974633209
Recall: 0.9393939393939394
F1 Score: 0.9395560822938042
```

Student ID: 201151261

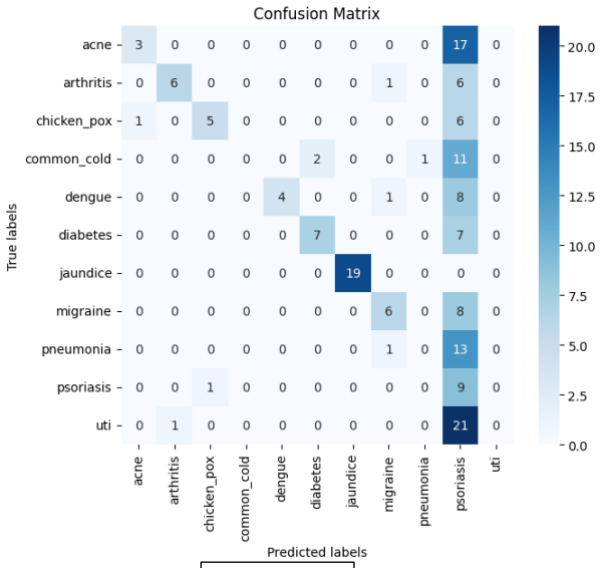


Figure 1

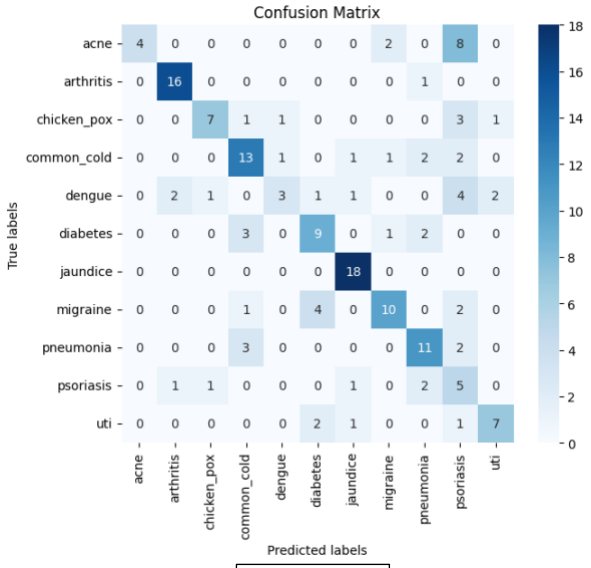


Figure 2

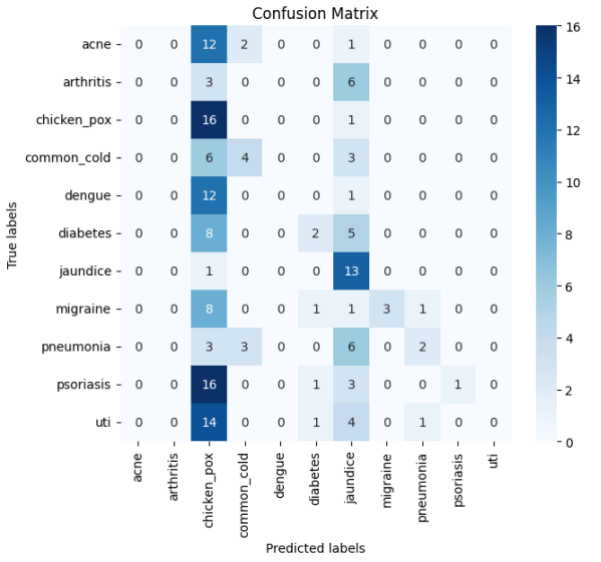


Figure 3

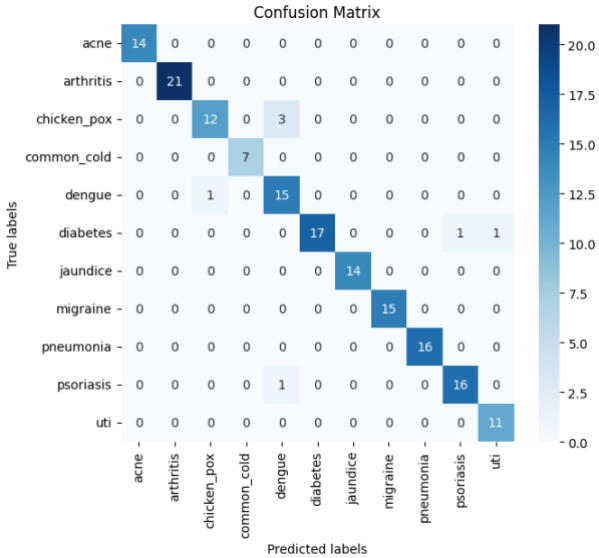


Figure 4

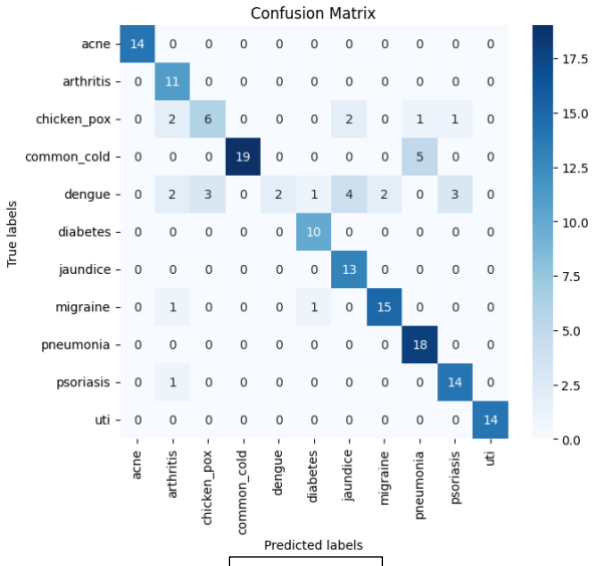


Figure 5

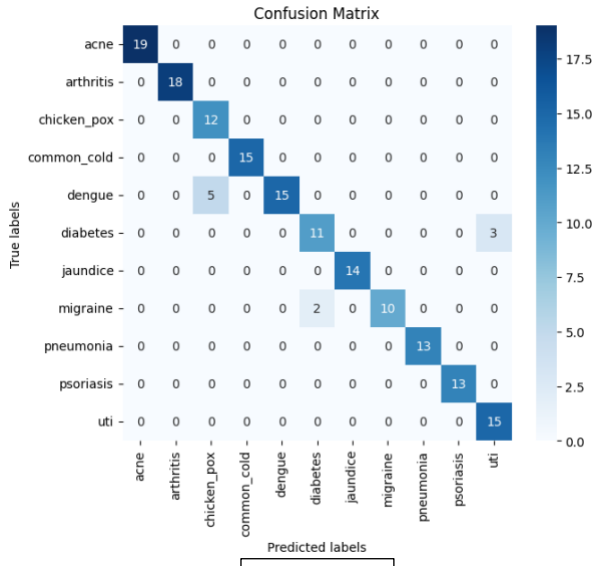


Figure 6