Doctor On The Go!

Parth Sayani
Masters in Computer Science
Department of CIDSE
Arizona State University
psayani@asu.edu

Chirag Jain
Masters in Computer Science
Department of CIDSE
Arizona State University
cnjain@asu.edu

Bineeta Gupta
Doctorate in Computer Science
Department of CIDSE
Arizona State University
bkgupta@asu.edu

Palash Mehta
Masters in Computer Science
Department of CIDSE
Arizona State University
pmehta21@asu.edu

ABSTRACT

We live in a data driven world poised to witness an accelerating rise in its amount each passing minute. The data in its natural form is raw and uninformative. Performing preprocessing to clean the data into a coherent form will only bring one so far. To a novice user, the data still does not reveal the underlying patterns and inferences. This is where data visualization comes into picture. Data visualization is a field of analytics where large and complex data is represented in a graphical manner by using visual elements. These representations are visually appealing and can effectively convey massive amount of information in an easily understandable manner to all range of users. 'Doctor On The Go!' is one such data visualization dashboard which recommends medicines for diseases to the users. This is an intelligent dashboard which permits the user to interact with it using different functionalities. The goal of this dashboard is to give users a seamless experience in identifying the disease they might be suffering from based on symptoms and recommend relevant medicines to them. Further, this is achieved by using visually appealing and interactive elements.

KEYWORDS

Medicine Recommendation, Sentiment Analysis, Disease Detection, Drug Review, Drug Effectiveness, WebMD, Data Dashboard

1 INTRODUCTION

Every year 42,000 people die in USA due to medication errors by doctors as stated in [1]. Our paper describes a dynamic tool built on D3.js which can help solve the problem of recommending the right medicines.

Our tool is to aid the patients in selecting the right medicine for the symptoms the patient is suffering from This paper describe a tool than can be used to find the required medicine for a particular set of symptoms. It uses a two level mapping algorithm for finding out the appropriate medicines. The tool uses percentage matching algorithm to find the top probable diseases for a particular set of symptoms. This paper also describes sentiment analysis on medicine reviews to find the best medicines for a disease. Our easy-to-use dynamic tool effectively gives out the best possible medicine for a particular disease inferred from the selected set of symptoms

Online diagnosis of symptoms is one of the major research topic. There have been a lot of studies going about different ways, a medicine can be recommended to a person who is infected with some disease. This paper uses a two level mapping to map user symptoms to a drug. The tool uses percentage matching to find the probable disease for a particular set of symptoms. The second level consists of sentiment analysis of user reviews given on a medicine. Paper [2] describes a very novel way to infer an overall sentiment about user reviews given on a particular drug. This paper describes a similar approach which uses the bag of words and TF-ID score to get the sentiment (change later). Further a generic review score about the medicine can be obtain by taking the average sentiment score across each drug for each disease.

We also integrated a time-series chart into our tool to give the user an insight about the environmental effect on the probability of a user suffering from that disease in a month. The paper describe an example of influenza which is more prominent during the winters

The easy-to-use dynamic tools is effective in mapping user symptoms to a medicine. It can be used both by a user and a doctor to confirm if the medicines being selected are correct. The dashboard incorporates several design principle to make it more usable and informative.

Section 2 describes the motivation behind the tool followed by Section 3 describes the visual implementation including the components of the dashboard. It also explains about the different interactions between them

2 MOTIVATION

Currently, there is no tool available for the patients to find out the best possible medicine based on the symptoms. The tool is intended to solve the problem of recommending medicines. Doctors can use to the tool to confirm about the disease the user is suffering from based on symptoms. They can also use it to find out the best possible medicine based on the disease the patient is currently suffering from. Patients can have a peace of mind about the drug recommended by the doctor. A lot of times, patients have a very little to no knowledge about their condition. There is a knowledge gap between a doctor and a patient about the patient's condition. This tool can remove that barrier and help the patient gain more knowledge about the probable disease they are suffering from and the best possible medicine for curing that problem.

3 VISUALIZATION IMPLEMENTATION

"Doctor on the go" consists of several visuals helping the users to several questions based on their symptoms. A large amount of data is consolidated and presented in the dashboard for the user to make several inferences. On each click several filters are applied on the back-end to present the information required by the user. The elements in the dashboard have been designed taking to account several design principles. Appropriate colors have been used in different elements. Also, the components are well separated from each other which make it look less cluttered. Also, a consistency is maintained in all the four components with respect to the design. The interactions between them are seamless and make the components appear connected to each other.

3.1 Data Collection and Pre-processing

We needed two types of data

- Mapping of symptoms to disease
- Mapping of disease to respective drugs along with their reviews

[3] gives the symptoms for each disease. [4] gives the drugs and their reviews for each disease. This paper uses these two data sets in the back end to implement the visualization tool. The disease names in the two data sets were slightly inconsistent. This paper uses fuzzy matching as described in [5] to join the two data sets on disease names. Symptoms,diseases and drug reviews obtained from the dataset were pre-processed with the help spellchecker library in python to remove several spelling mistakes and ensure data consistency. Also, the null values were removed in the dataset for consistency. For sentiment analysis, the reviews were pre-processed to remove the words which do not contribute to the sentiment. Further, this paper describes an approach to assign an average sentiment score to each drug in the dataset. Each review was vectored using the "CountVectorizer". Further, it uses the natural language toolkit library and logistic regression to find out the sentiment behind each drug review

Further, for each drug an average sentiment is calculated. This helps in recommending the best medicine to the user based on the disease the person is suffering from.

For the time series chart, year and month level data was rolled up on month level to get the total reviews in a month. This data gives more information about the variation of occurrence of a disease across each month.

3.2 Dynamic Visualization using D3.js

D3.js [6] is a JavaScript library for manipulating document based on data. This library is an effective tool for data visualization tool that uses Hyper Text Markup Language (HTML), Support Vector Graphics (SVG), Cascading Style Sheet (CSS). D3 is compliant with web standards and gives full capabilities of modern browsers. It permits the developers to bind arbitrary data to Document Object Model (DOM), and apply data driven transformation to the document. D3 is very fast, supports large data sets and dynamic behaviours. Our dashboard is backed by D3.js for all the charts, plots and animations. The bubble chart is a non-hierarchical packed circles. Using D3.js, we displayed a bubble chart which shows the medicine based on entered symptoms. The size of each circle is proportional to the number of matched symptoms. D3.js gives the functionality to set the color of the bubbles based on context. It also gives the functionality to enlarge a particular circle when mouse is hovered over it. Further, when clicked on a particular disease bubble, D3.js generates a time series chart of number of reviews of medicine posted over a year. At the same time, it also generate three SVG circles which represent the top three medicines for that disease.

3.3 Dashboard

The dashboard is divided into four parts. Out of these four parts, the first part is search bar with the auto-complete mechanism. The symptoms given by the user as an input are used by bubble chart to predict the top twenty disease the user is likely to have using a percentage matching algorithm. The bubble chart is the second part of the dashboard. The user can interact with the bubble chart using the zoom-in, zoom-out and pan features. On clicking on any circle (disease) a time series graph will be generated for the selected disease and medicines will be recommended for the that disease. The time series chart and the medicine recommendation are the third and the fourth part of the dashboard. There is a flow of data from the search bar (first part) to the bubble chart (second part), and then from the bubble chart (second part) to the time series chart(third part) and medicine recommendation part(fourth part). The bubble chart is the center of the data

flow. The dashboard is interactive as clicking on the bubble chart a time series chart and medicines will be displayed. Another important feature about the dashboard is that it is dynamic in nature. In order to present information at a mere glance dashboards need to be scroll free. According to the article "Common Pitfalls in Dashboard Design" a dashboard is a visual display of the most important information needed to achieve one or more objectives, consolidated and arranged on a single screen so the information can be monitored at a glance [7]. So in order to avoid the problem of scrolling down to view the information there is a single page view in which all the visuals can be seen in a single view by the user, thereby making the dashboard user-friendly. Overall, the dashboard is easy to understand and intuitive in nature.

The following keys points can be noted about the dashboard

- Providing details as per the user's choice.
- No scrolling required to see different components of the dashboard. All the elements are present in the single view of the screen
- Dashboard has responsive design. Elements resize according to the width and the height of the screen. This makes it usable across a wide variety of devices with different screen size
- Smaller learning curve because all the interactions are user friendly

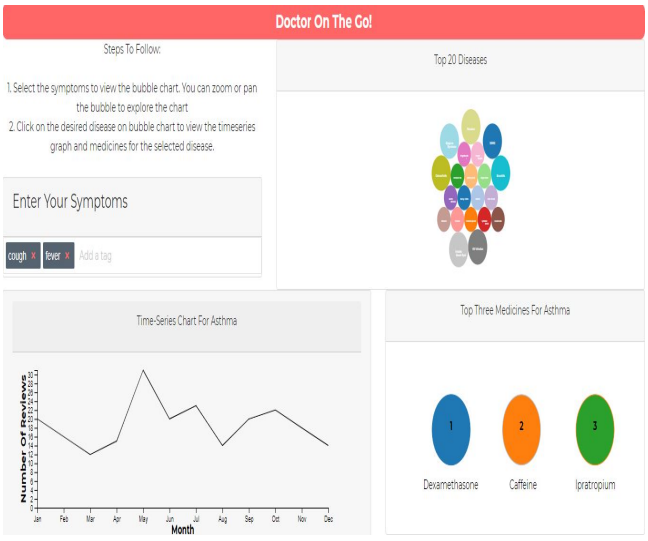

Figure 1: Zoom-in feature in Bubble chart

3.4 Search Bar

As the name suggests, search bar is an entity that permits the users to search or find something they might be interested in. For example, in any web browser, the users type keywords in the search bar and a drop-down suggests relevant web pages that they might be interested in based on the keywords. A similar search bar has been implemented in this project. The search bar forms the first point of interaction in the dashboard. It permits the user to enter the symptoms. The search bar has a blinking cursor which suggests the user to type the symptom. While the user is typing the symptom in the search bar, it suggests relevant symptoms based on character matching. The user can select the symptom from the suggestions which will then display that symptom in the search bar in a highlighted manner. Multiple symptoms can be selected without restriction. The main aim of highlighting is to let the users know which symptoms have been selected so far. So, the user need not remember the selected symptoms but rather concentrate on interacting with the dashboard. It is a natural tendency to make mistakes and the user here can select a wrong symptom. Keeping this in mind, a provision has been provided to delete the selected symptom from the search bar by clicking on 'x' shaped button adjacent to it. Further, the user also cannot select a duplicate symptom. If he does so, the symptom is highlighted in red suggesting that a duplicate symptom has been selected. All the above stated functionalities give the users a seamless and effective experience of interacting with the search bar and selecting the relevant symptoms.

Whenever user adds the symptoms, the bubble chart gets dynamically updated with the probable diseases. This will add to the productivity of the user as it reduces the number of clicks required by the user. This feature is one of the important components of the dashboard.

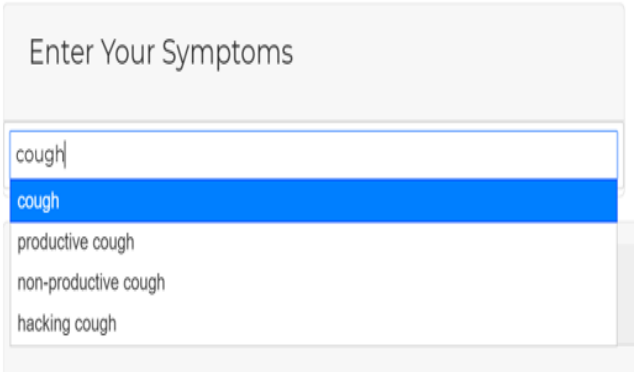

Figure 2: Autocomplete

3.5 Bubble Chart

Data in bubble chart is displayed in bubble-like circles wherein the size of each circle is determined by the value of the data. Larger the size of the circle more is the value of data and smaller the size of the circle lesser is the value of the data

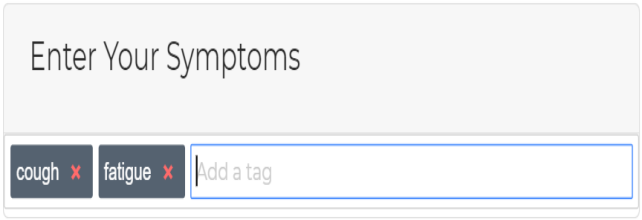

Figure 3: Symptom Highlighting and Deletion

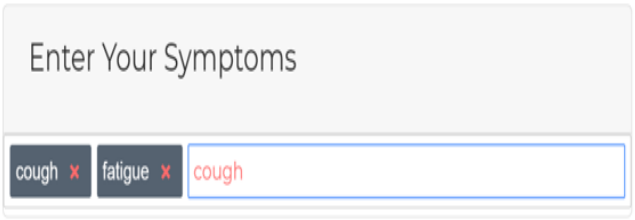

Figure 4: Duplicate Symptom

Figure 5: Bubble Chart for Diseases

[8] . This type of visualization is quite intuitive and visually appealing. In order, to display the top twenty diseases based on the symptoms selected by the user, this project uses the percentage matching algorithm. In the bubble chart mouse-based zoom and pan features are added which are the key features of the chart in Figure 5. Zooming and panning are common and useful for SVG visualizations like bubble charts, maps, etc and a typical viewer expects to zoom in for details, zoom out for context and pan to view the visual element of his interest [9]. Along with D3.js, Hammer.JS is another library that provides panning, zooming, swiping and rotating features [10]. Hammer.JS is mainly used for touch on mobiles. These bubble charts represent the diseases which the user is likely to get based on the symptoms added by the user. The larger the size of the bubble(disease) the more is the probability of disease. This chart is the main of the part of the visualization as the data to time-series chart and the recommendation part is flown from the bubble chart. The radius of the bubble chart gives the user an overview of the probability of the disease the user is likely to suffer thereby making it easier for the user to drill down the diseases he is likely to get. A clear sequential scheme has been used to colour the bubbles such each bubble gets a unique colour. On hovering over any of the circle on the bubble chart the size of the circle increases thereby making it easier for the user to visualise the bubble chart.

Clicking on any bubble, dynamically updates the time line chart and the probable diseases. This gives an insight to the user about how data is flowing from the bubble chart to the time line chart and the medicine chart.

3.6 Time-Series Graph

A time series graph is a data visualization tool that illustrates data points at successive intervals of time. Each data point in the graph corresponds to time and a variable that is being measured. The X-axis or the horizontal axis of the graph represents the time intervals. The Y-axis or the vertical axis represents the variable. Stock price charts of a company are perfect examples of time series graph. This kind of a chart is also called a fever chart as the values are connected in chronological order by a straight line which creates a sequence of peaks and valleys[11]. Such time series graphs are very useful in identifying trends over time which can help reveal very useful information. Our dashboard integrates a similar type of time series chart. The main aim of this chart is to inform the users of the time of the year when the disease will be high. Once the user clicks on the disease bubble, a time series chart is generated. The horizontal axis represents the month of the year. The vertical axis represents the number of reviews of all the medicines for that disease. When there is a rising trend in the graph or when there is a peak, it suggests that the number of reviews for medicines are increasing. One can infer that since the number of reviews are rising, more people bought that medicine. This in turn means that more people are suffering from that disease. Opposite inference holds true when there is a downwards trend or when there is a valley. The user can then interpret the month when the chance of disease is high or low. For example, the number of reviews for medicines for bronchitis is higher in months of October to March. Then there is a downward trend

from April. It is usually caused by a viral infection, and the virus responsible for the common cold can sometimes lead to bronchitis in people whose lung resistance levels are low, such as smokers or asthmatics. In winter months, cold, damp weather can aggravate bronchitis, and may even make it difficult to breathe[12]. October to March are colder months than the rest which supports the interpretation. The axes are labelled for clear understanding. The chart label dynamically changes to the name of the disease to inform the user which disease does the cart pertain to. All these functionalities help the users to effectively identify the disease they might be suffering from.

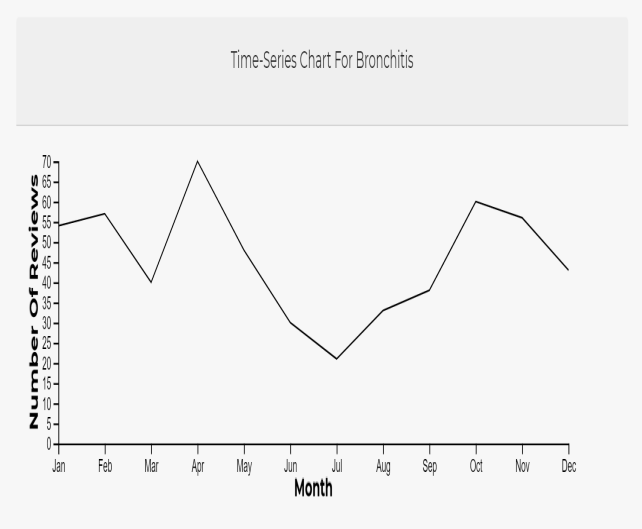

Figure 6: Time-Series Chart For Bronchitis

3.7 Medicines Recommendation

Based on the disease selected by the user top three medicines for that disease will be displayed to the user. The algorithm behind this visualization is performing a sentiment analysis on the different medicines reviews submitted for the selected disease. In this project a novel sentiment analysis model is designed in which sentiment scores for the reviews is assigned based on the positive and the negative polarity of words. More information about the model can be obtained from Section 4.1. In case of a tie in the sentiment of two medicines then the model takes the average of the usefulness of the ratings. It may happen that a medicine has many reviews, so for this to calculate the overall score of the medicine, the average of all the sentiments is taken, in order to find the average sentiment of the medicine for the selected disease. The medicines are arranged in the increasing order of their average sentiment scores, meaning the first medicine will have the highest average sentiment score, the second medicine will have the second highest average sentiment score, while the third medicine will have the third average highest sentiment score. Meanwhile, there are certain diseases which had positive scores for only for two medicines. In that case we only display the top two medicines This project gives recommendation for medicines belonging to the same family, for instance, for influenza the project recommends Acetaminophen/phenyltoloxamine, Acetaminophen/chlorpheniramine and Tamiflu as the top medicines. The top two medicines belong to the same family of medicines, so the model is designed in such a way that it shows the best medicines even though they belong to the same family, thereby giving a better accuracy to the model. Circles are used for representing the medicines as they create an interesting and intuitive visualization for the top three medicines. All the circles are at an equal distance from each other and have the same radius. The dashboard is responsive in nature that reacts to the size of user's screen. Having a responsive dashboard helps in optimising the user's browsing experience by creating a flexible and responsive web page [13].

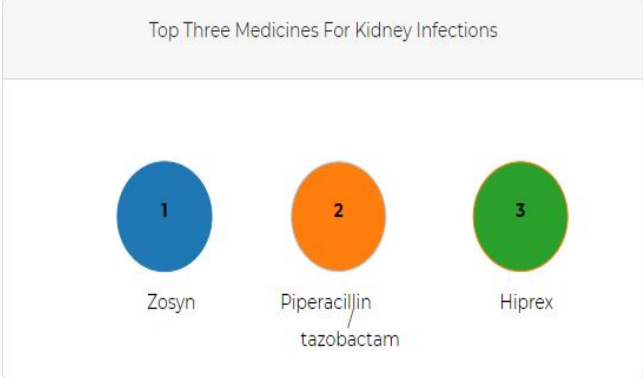

Figure 7: Top Three Medicines

4 METHODOLOGY

The aim of our Intelligent Visual Analytics System is to help patients and other users to answer research questions efficiently. For each section in the dashboard that we have designed, there are some research questions that we tried to solve and use the knowledge of visualizations to solve those questions. Below, we have showed research questions and how our visualization approach helped to solve it.

4.1 Search Section

How can a patient define their symptoms objectively to get the right disease?

This section was for user's problems of finding a friendly interface. Sometimes it so happens that user's don't know

how to define their symptoms clearly. They might type up a symptom to just get an error back of symptom not found or the models incorrectly taking the symptom as some other symptom and showing incorrect model. We have over 150,000 unique symptoms in our dataset. We have provided an auto-complete option to user to define their symptoms clearly. They can enter as many symptoms as they want to get precise results. We also prevent users to repeat any symptom providing a very definitive patient's requirement. This section didn't had a specific type of visualization but this search feature is helpful to patients.

4.2 Bubble Chart

How to know the most probable disease that a patient has based on the symptoms they entered?

How to know if there is possibility of other diseases as well based on the entered symptoms?

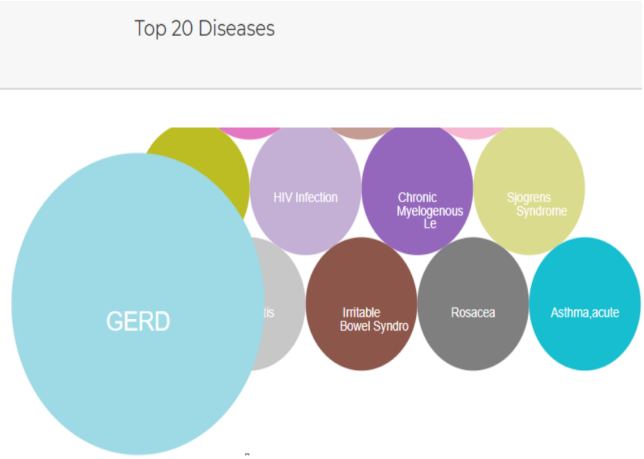

Figure 8: Zoom-in feature in Bubble chart

These are very common but focused questions from a patients end. In our dataset, based on the symptoms, we are showing users the Top 20 diseases that they can potentially have. The bigger the radius of the circle is implying they have more potential of that disease [14]. The bubble chart visualization is helping to solve the above problems by displaying different disease aka bubbles with the radius aka the probability of the disease. The zoom-in and zoom-out features help user to know about the other diseases even if there probability of occurrence is low. Also, by providing the option of moving the bubbles in the pane window makes the user capable to adjust the application as per their screen. Thus, the visualization provides the complete justice to Visual Information Seeking Mantra taught in the class.

4.3 Time-Series chart

The problem we are trying to solve here is subjective but it can be explained by the time-series graph, to some extent. The primary reason we decided to have a time-series was to answer questions such as:

Do we see any trend for this particular disease occurrence in the last 10-years? There may be environmental factors that need to be taken into consideration while analysing the cause of the disease. So, the question we are trying to solve here is basically, if there are any environmental reasons that cause the disease, or if there is any pattern that we can observer and figure out from there? The answer is yes!

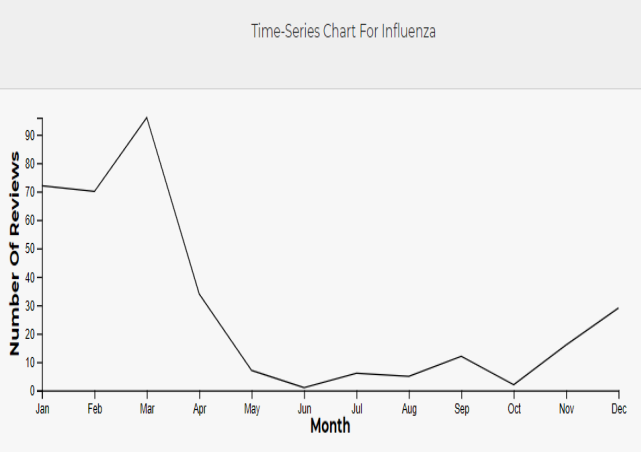

Figure 9: Time-Series chart for Influenza

For example: We have below time-series plot for influenza where you can see from May - October the plot is tampered and the line starts peaking from October on-wards with a peak in March and it starts tampering down again from May on-wards. The reason can be environment- that is the weather. As we know that Influenza is a type of flu, that occurs most commonly in winter season. Hence, we can see that more people are posting reviews about Influenza related medicines during Winter season (that is around October). Thus we can prove here that there are environmental factors that contribute towards the cause of the disease. Similarly if we look at the time-series graph [15] for other diseases, we can try to find a specific pattern to answer the question of the disease cause.

4.4 Medicine Recommendation System

We have a patient who needs flu-related medicine and they went to a medical store to purchase it. Unfortunately the store is out of stock for that specific medicine. Should we ask the patient to find any other shop and go there to buy the medicine? There is a possibility that no other store has this particular medicine in stock.

During this pandemic time of COVID-19, do all the medicines are readily available? No. For this particular section, we wanted to show users the Top 3 medicines for a disease. Why are there so many medicines available for a disease? How do we find the right medicine? What if we want a medicine but that is not available in the medicine store? In that case, can we buy any alternative of that medicine? This visualization which is based on the sentiment analysis model [16] will provide you the answer to all these questions. Our model provides all the medicines even if they belong to the same parent category. So if a patient goes to a medical store and trying to find a specific medicine and it's not present, then user can look up the recommendations of the medicines belonging to the same family.

Another problem that our visualisation tried to solve is clearly communicating to the user which medicine is the top pick even if other medicines are derivative from the same family. We are writing down 1st, 2nd, 3rd and so on to indicate the clear selection.

5 EVALUATION PLAN

Visual analytics is to help people understand a complex thing into simpler approach as it's proven that visual knowledge is more easier to grasp by users. In this application, we used different kind of visualizations to help patients and doctors by recommending different medicines. Although this application uses a wide range of dataset, we do recommend patients to always go to the doctor for emergency and consult them.

We evaluated our research questions and how our visualizations helped in answering those questions. The 'Percentage matching algorithm' we used is efficient (by far as per our testing) to 94 percent. For rest of the 6 percent, we used the count of occurrence of diseases in past 10-years to make the right selections.

Similarly, we used different models for sentiment analysis and decided upon SentiStrength Model [17] which ignored the stop words, assigned strongly positive, weekly positive, strongly negative and weekly negative polarity of the words for the reviews posted. Based on the sentiment strength, we decided upon the Top 3 medicines.

We also did code coverage analysis to clean-up our code for any extra variables and efficiency.

6 DISCUSSION AND FUTURE WORK

After we designed the application, we performed multiple testing and found that reviews about cancer, tumours are less compared to common diseases such as flu, stomache etc. This implies that people don't like discussing about diseases that have effected them in critical ways. Where as diseases for which the counter medicines are available, have higher number of reviews posted.

Similarly we noticed that there are more medicines available for certain diseases but as there are not many reviews for them, it's difficult to show those medicines on our application.

As this application derives medicines and analysis from the dataset obtained from Kaggle- University from California, Irvine, it can not be that accurate as the data is collected by scrapping websites originally. Every patient should consult doctor for sure in order to avoid any infections or reactions. In future, we can get dataset from different hospitals as well to be more accurate and precise.

Many times, doctors take into account a lot of other factors in addition to the symptoms to conclude on a particular disease. The tool can be further developed to take more inputs from the user like the past medical history, gender, age, sex to give an exact disease. It also depends on geolocation to get environmental effects. Sometimes a patient's family history also contribute into cause of a disease. For example: Diabetes Type-1 is due to genetic where as Diabetes Type-2 is from eating and environmental habits.

For now, we used a percentage matching algorithm to find the Top 20 Probable diseases but as the some research work [18] , we can further use Random Forest Classification Algorithm to find out the disease the patient is suffering from. There can be improvement in sentiment detection.

7 ACKNOWLEDGMENTS

As part of CSE 578: Data Visualization course, we would like to thank Professor Dr. Sharon Hsiao for teaching this class even during this pandemic time. It was well organised, and the slides were super helpful. We really like dhow the assignments were challenging and unique and encouraged us to learn something new everytime. We would also like to thank our Teaching Assistant, Mohammed Alzaid to answer promptly for our emails and any queries related to assignments. We would also like to thank Arizona State University's School of Computing, Informatics and Decision Systems Engineering for providing this course via zoom to continue us learning and expanding our skill sets.

REFERENCES

[1] Y. Bao and X. Jiang Plant. An intelligent medicine recommender system framework. *2016 IEEE 11th Conference on Industrial*, 50:1283–1388, 2016.

[2] Z. Min. Drugs reviews sentiment analysis using weakly supervised model*. *2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pages 332–336, 2019.

[3] R. Lee and C. Chitnis. Improving health-care systems by disease prediction. *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 726–731, 2018.

[4] Dheeru Dua and Casey Graff. UCI machine learning repository. 2017.

[5] Krishna Kalyanathaya, Akila D., and Suseendran G. A fuzzy approach to approximate string matching for text retrieval in nlp. *Journal of Computational Information Systems*, pages 26–32, 2019.

[6] Mike Bostock. Data-driven documents, March 2020.
[7] Stephen Few. Common pitfalls in dashboard design, February 2006.
[8] Tomomi Imura. Fun with d3.js: Data visualization eye candy with streaming json, May 2010.
[9] Helder da Rocha. Learn d3.js: Create interactive data-driven visualizations for the web with the d3.js library, 2001.
[10] Jeff Huang Michelle A. Borkin Michail Schwab, James Tompkin. Easypz.js: Interaction binding for pan and zoom visualizations. Technical report, Cambridge, MA, USA.
[11] Margaret Rouse. Time series chart, March 2020.
[12] Medmedia Group. Cold and damp weather, May 2017.
[13] Stewart Dymock. 7 business advantages of responsive web design, February 2018.
[14] Edward Tufte. Tufte. data-ink and graphical redesign. in the visual display of quantitative information. 2010, May 2010.
[15] Peter McLachlan. Liverac: interactive visual exploration of system management time-series data. *CHI '08: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1483–1494, 2008.
[16] Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. Sentiment strength detection for the social web 1. 2012.
[17] Liang Wu. Slangsd: building, expanding and using a sentiment dictionary of slang words for short-text sentiment classification, 2018.
[18] R. Lee and C. Chitnis. Improving health-care systems by disease prediction. *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 726–731, 2018.