Detecting Dengue Spreading in Sri Lanka based on News Articles

Thesis · August 2019

CITATIONS
0

READS
7

4 authors, including:

Prabhashi Meddegoda
University of Moratuwa
3 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Sampath Deegalla
University of Peradeniya
23 PUBLICATIONS **230** CITATIONS
SEE PROFILE

Detecting Dengue Spreading in Sri Lanka based on News Articles

Nishara Kavindi, Peshali Randika, Prabhashi Meddegoda, Sampath Deegalla
Department of Computer Engineering
Faculty of Engineering
University of Peradeniya
Peradeniya, Sri Lanka
{nishkavindi98, peshalir, prabhashi.mm, deegalla}@gmail.com

***Abstract*—Emerging of the infectious diseases such as Dengue, have become a major challenge for the world. Use of indicator-based surveillance systems is the traditional approach of monitoring diseases, which uses structured data. Use of event-based surveillance systems is the modern approach, where unstructured data such as information from the internet and social media is used.**

In Sri Lanka, there are indicator-based systems established for detecting and monitoring Dengue occurrences. But it still remains a major health problem. Therefore testing and implementing another approach to strengthen the traditional system is important. There are successful event-based systems implemented in other countries. But none of them gives detailed information about Dengue spreading in Sri Lanka.

Our objective is to address this issue through an automated system which query for newly published online news articles and classify them as Dengue-related or not, extract useful information out of Dengue related articles about Dengue outbreaks in Sri Lanka, store them in a database and visualize through a web application.

In this paper we describe data acquisition, classification, data extraction, data storing and the visualization processes of the system.

***Index Terms*—dengue, Sri Lanka, data mining, machine learning, surveillance, event-based, outbreak**

I. Introduction

A. Background

Emerging of the infectious diseases such as Dengue, have become a major challenge for the governments, health officials and have become a public concern. Therefore early detection of an outbreak is very important in order to take relevant actions.

There are many approaches experimented and implemented by different countries for the detecting and continuous monitoring of different kinds of disease occurrences. The traditional approach is Indicator-based surveillance systems.

Indicator-based surveillance systems are the most common and the oldest infectious disease surveillance (monitoring) systems. These systems collect and analyze the structured information. Those data on indicators are reported by the health care providers. The main objective is to find the increased numbers at a particular time period and the location. However the ability of tracking threats quickly is lack in these systems and the data may not be recent as there is a time gap between the event occurrence and the surveillance [1]. These problems have arisen the need of new methods of disease surveillance and outbreak detection in order to strengthen the traditional surveillance systems.

The modern approach is event-based surveillance systems. They collect and analyze unstructured data [2], capturing the event information rapidly. These systems use the information from the reports which are transmitting through different communication media. So the social media news and the health information taken from the internet are the most crucial part of these event-based surveillance systems [1].

Detecting and monitoring diseases by extracting data from news articles published online is one of the methods in the modern approach. At present, many online newspapers are available as data sources which provide useful information to analyze and detect outbreaks early. This approach is proved to be timely and reliable to a great extent.

The main steps of this approach are querying newly published online news articles, classifying the articles (as disease related and not related), extracting data and visualizing data. (Data mining and machine learning algorithms and techniques are used to achieve the task of classification.)

B. Our contribution

Dengue is an infectious disease which is spreading fast and a deadly threat for Sri Lanka. The existing systems for detecting dengue spreading in Sri Lanka are a paper-based system that keeps records of informed Dengue occurrences and a web-based system that keeps track of Dengue incidents in major hospitals in Sri Lanka. Though there are such approaches for the detecting dengue spreading in Sri Lanka, it still remains a major health problem.

Experimenting and establishing another approach for detecting and monitoring Dengue spreading in Sri Lanka is therefore important, in order to strengthen the existing systems. Although the approach of using online news articles for detecting and monitoring of diseases have been established in other countries, there is no such implementation for Sri Lanka. The implementations in other countries do not provide sufficient information about diseases in Sri Lanka.

This project addresses this issue by implementing an automated system using this approach; using online news articles

for detecting and monitoring diseases, in order to detect and monitor Dengue occurrences. Through this system, newly published online news articles are extracted and classified as Dengue-related or not. Out of the Dengue-related articles, useful information about Dengue incidents are extracted, stored in a database and visualized using a web application.

II. Related Works

In this section, modern disease surveillance systems and related researches are discussed including their methodologies, status, importance and limitations. Six latest web-based surveillance systems are selected considering their importance.

Event-based surveillance systems collect and analyze unstructured data [3]. GPHIN, Argus, ProMEDmail, HealthMap, EpiSPIDER, BioCaster are the main examples for event-based surveillance systems [1].

GPHIN retrieves articles from subscription-only news aggregators (Factiva and Al Bawaba) every 15 minutes every day 24 hours using well established search queries that are updated regularly [4].

In GPHIN, the related articles are automatically categorized into topics such as animal, human, or plant diseases; biologics; natural disasters; chemical incidents; radiologic incidents; and unsafe products [4]. GPHIN scans, filters the information and categorizes them using Boolean syntax and taxonomy of keywords. GPHIN system contributes to 40% of the WHOs early outbreak warning system [5].

GPHIN consists of both automated and human analysis processes. GPHIN automatically publishes the articles which are considered as highly relevant after the categorization, as well as articles that are obtained manually from the websites that are openly accessible [4]. The articles which fall below the relevancy level are directed to the GPHIN analysts to be checked and to decide whether to dismiss them, issue an alert or to publish them [4]. Other than that, the analysts also identify risks and trends and the links between the events occurred in different regions [4].

GPHIN machine translates English articles into Arabic, Chinese, Farsi, French, Russian, Portuguese, and Spanish, while non-English articles are machine translated into English [4].Here the outputs are edited by the analysts for them to be more comprehensive [4]. Boolean and translingual metadata searches can be used to query the database for the subscribed GPHIN users in order to access the recent published articles and they are notified about any immediate public health concern via email [4].

HealthMap obtains disease outbreak data from online news media such as Google News, expert-curated accounts such as ProMED Mail; a global electronic mailing list that receives and summarizes reports on disease outbreaks [6], Really Simple Syndication (RSS) feeds, multinational surveillance reports such as Euro surveillance and validated official alerts from organizations such as WHO, which is a wide scope of resources and which are freely available [1] [4]. The data are extracted every 1 hour. But HealthMap is automatic in data characterization and it only runs a daily scan of all its alerts [4]. In HealthMap, text mining is used to classify the disease and the location of the outbreak automatically [4].

It facilitates automated querying, filtering, integrating and visualizing unstructured reports on disease outbreaks. The Data Acquisition Engine of HealthMap converts each outbreak report into a standard alert format. The alert contains headline, date, description and info text relevant to the report. Alert is usually obtained from RSS structure. But when it is not available, basic HTML text formatting are done for each input feed. The problem is that it creates many unexpected results.

The classification engine determines the primary locations and diseases of each alert. HealthMap has a dictionary of diseases and locations. Alert is compared with this dictionary and the primary disease and location of the alert is determined [1]. After the characterization, the alerts are generated which are geocoded to the country scale [4].

HealthMap presents each outbreak information in a web application using an interactive geographical map [1]. Initial state of the web application is loaded to the users browser from a server-side cache which is updated every hour [1]. The user is facilitated to adjust the viewing parameter and retrieve the alerts that match with those parameters [1]. It also provides the links to other information sources as well [1].

EpiSPIDER uses the data from Twitter [7], from news sites, from ProMed (curated resources) and extracts data from Central Intelligence Agency (CIA) Factbook, and the United Nations Development Human Development Report Internet sites once the location information are taken from news reports in order to link demographic- and health-specific information to get to know any disease outbreaks in a different context [4]. EpiSPIDER is also automatic in data characterization and it runs sample of its alerts with the help of a person [4].

In EpiSPIDER, natural language processing is used, so that the free text content can be converted to a structured set of information that can be stored in a relational database [4].

EpiSPIDER uses other available services for georeferencing the locations parsed from the reports; Yahoo Maps, Google Maps and Geonames [4]. It transforms its structured data into other formats such as RSS, keyhole markup language (KML) and JavaScript object notation (JSON) [4] so that its structured nature can give its best service. It does not have its own user interface and uses open-source software and it also uses conventional formats for reports so that the reports can be integrated with other information sources [4].

Proteus-BIO uses a web crawler to find relevant web pages. It traverses through the web at each night in order to find new relevant web pages. The current system goes through 2 news sites mainly. One is Disease outbreak news of WHO and the other one is ProMed-mail of the International society for infectious diseases. For most news sites this web crawler finds the text body within the web page using HTML markup or specific text tags or other layout indicators [8].

Proteus-BIO uses an extraction engine to analyze and identify the instances of epidemic diseases. It captures the date, type, deaths and the location of the each report. This extraction engine is operated by searching for grammatical patterns in the

text [8]. It uses a database browser to provide the UI for the system. It presents the extracted information in a table and allows the users to select rows and visit the corresponding paragraphs in the corresponding document where the relevant points are highlighted [8].

BioCaster uses RSS feeds as its data source. Each hour a purpose built news aggregator script written in Perl identifies novel links from over 1700 feeds [9].

The documents obtained by BioCaster are put into a cluster queue and automatic classification is done. Here, a Naive Bayes algorithm is used [9]. Named entity recognition (NER) is then performed for the documents for 18 term types based on the BioCaster Ontology (BCO) [10]. Simple rule language (SRL) and Declarative Information Analysis Language (DIAL) [11] are used for the semantic analysis in event extraction in order to get a deep understanding on the impact to the public health from the event [9]. These have the capability to match entity classes, skipwords, string literals, regular expressions, entity types as well as guard lists [9].

Named entity recognition (NER) is then performed for the documents for 18 term types based on the BioCaster Ontology (BCO) [10]. Simple rule language (SRL) and Declarative Information Analysis Language (DIAL) [11] are used for the semantic analysis in event extraction in order to get a deep understanding on the impact to the public health from the event [9]. These have the capability to match entity classes, skipwords, string literals, regular expressions, entity types as well as guard lists [9].

In BioCaster, a plotting of disease-location pairs is presented in a public portal named Global Health Monitor [9]. In order to get an idea of an outbreak in a geographical context, visualization through Google Maps with the facilities of filtering according to pathogen, syndrome or text type is supplied [9]. There, the source evidence can be visited by clicking on the map points which shows the headline along with the links to the scientific databases such as PubMed, HighWire and Google Scholar [9].

Project Argus is also a bio surveillance system which is monitoring emerging biological threats. It uses news media resources in about 40 languages which are available in public. It develops effective Boolean query strings to retrieve information [12].

None of these systems provide detailed information about the spreading of Dengue in Sri Lanka. They either give a very abstract idea or do not give any idea at all. Therefore, these systems cannot be used to solve the problem; to get a detailed idea about Dengue spreading in Sri Lanka.

But through this research, the problem is addressed and solved, so that a system focusing on Dengue in Sri Lanka is tested and implemented.

III. Formulation Process

The main objective of this system is to give detailed and useful information about Dengue spreading in Sri Lanka using the unstructured data available (newspaper articles), as an assistance to the currently available systems. To achieve this task, the Dengue-related newspaper articles should be correctly separated from the other articles published. The system should also give flexible and useful visualization output with useful information.

A. Classification

The objective of using classification in this system is to identify news articles which contain information about Dengue spreading. The classifier classifies each newly published news article and stores content and details of the Dengue-related articles in a database. The system acquires newly published articles through RSS feeds.

B. Visualization

The key objective of visualization is to analyze the spread of Dengue in Sri Lanka in past years over months in different areas. This is a simple analytical approach for the public to identify any Dengue outbreaks. Using this, the public can identify patterns of Dengue occurrences, make comparisons, take decisions and do predictions. The data are visualized in graphs, data tables and in a geographical map. The data needed for the visualization are extracted from the Dengue-related articles selected by the classifier. Extracted data are inserted into a database and the necessary details for the visualization are fetched using queries.

IV. Model Description

The process carried out by the system consists five main steps; data acquisition, classification, information extraction, data storage and visualization. These steps are automatically carried out by using a script that run daily on a server.

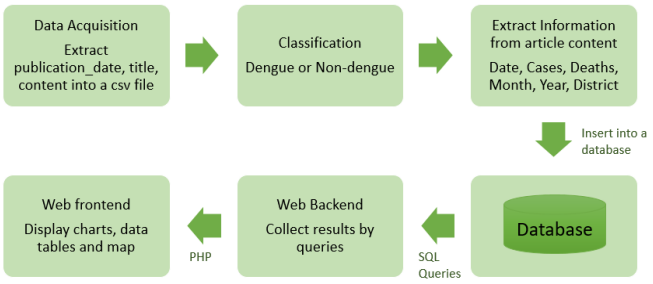

Fig. 1. System Architecture

A. Data acquisition

News articles are published daily in newspaper websites. Those articles are acquired daily, in order to classify them as Dengue-related or not-related, thus we can obtain necessary information from the Dengue-related articles.

To acquire newly published news articles, RSS (Rich Site Summary) feeds of those websites are used. RSS feeds are used by the websites which publish updates frequently, in order to publish them. The updates are published through RSS as a full or summarized text including metadata such as authors name and the date of publication.

For newspaper articles acquisition, a Python script is run daily. Through the Python script, title, publication date, summary and the article link for each article are collected from the RSS feeds. The full text of each article is scraped then by using the article links extracted. The details extracted are stored in csv file format, for the classifier to do the classification for each article as Dengue-related or not.

B. Classification

The newspaper articles acquired daily are classified as either Dengue-related or not, so that Dengue-related articles can be used to extract the useful information needed for the visualization.

In order to accurately classify the articles as Dengue or non-Dengue once an article is published, an accurate classifier is needed to be built. In the process of building the classification algorithm, news articles in newspaper websites are used as the data source. Dengue related articles and non-Dengue articles are extracted in order to prepare the training dataset.

For the article extraction, modern visual web data extraction software called Octoparse is used, which has the facility to extract bulk information from websites. The extracted information is exported in CSV format.

Dengue dataset and non-dengue dataset are combined together to obtain one dataset. Then the dataset is labeled manually as dengue and non-dengue, in order to prepare the dataset required to train and test different classifiers on.

Bag of Words method is used to extract the features out of the prepared dataset and a document-term matrix is created. A document-term matrix is a sparse representation of counts of terms for each document. The matrix is then transformed to a normalized representation. Parameters for feature extraction are tuned to eliminate unnecessary features, in order to improve accuracy. In text classification, feature extraction is needed for the classifiers to understand the dataset.

Different classification models are trained and are tested to obtain the accuracy values. The parameters for the models are tuned, feature selection is done for the models and accuracy is calculated each time.

Accuracy is calculated using 10-fold Cross Validation method. This method is the widely used method and is experimented to be more accurate. In 10-fold Cross Validation method, the whole dataset is split into 10 equal folds. One fold is used as the testing set, while the rest is used as training set and the testing accuracy is calculated. This step is repeated 10 times, using a different fold for testing each time. Finally the mean of the test accuracy values is calculated and obtained as the accuracy.

∗ Classification model selection

Popular algorithms for classification are trained and tested to find the best algorithm out of them. Classification algorithms used are,

- Multinomial Naive Bayes
- Logistic Regression
- Support vector machine (SVM)
- Stochastic Gradient Descent (SGD)
- Random Forest Classifier
- K-Neighbors Classifier
- Decision Tree Classifier

Fig. 2. Behavior of accuracy of Classifiers

The highest accuracy value is given by Random Forest classifier as shown in the Fig. 2. Therefore it is chosen as the classification algorithm of the system.

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting [13].

There are 3 parameters in that classifier which can be tuned to improve the predictive power of the model.

1) max_features: Maximum number of features random forest is allowed to try in individual tree. When the value of this parameter increases, it improves the performance of the model. We chose *max_features=None* which is using all the features that make sense in every tree.

2) n_estimators: Number of trees we build in the random forest The higher the number of trees, better performance can be obtained. It makes the prediction stronger and more stable. We chose a high value (*n_estimators=500*) as such our processor can handle.

3) min_samples_leaf: Minimum number of samples required to be at a leaf . When this value gets smaller, it makes more prone to capture the noise in train data.We chose *min_samples_leaf =1*.

The contents of each article that are extracted and stored in CSV file are transformed into a normalized document-term matrix representation and are fed into the chosen classifier, to separate the Dengue-related articles from others. The title, publication date, content, summary and the link for each of the separated Dengue-related article are then stored in a database table for future reference.

C. Information Extraction

The contents of the Dengue-related articles are used to extract useful data that are needed for the visualization. Publication date, number of Dengue cases, number of deaths due to Dengue, time periods (years, months, time periods, dates) for the incidents mentioned and places (districts, provinces, other locations) related to the Dengue cases/deaths are the data that are needed to be extracted. The extraction is done using

natural language processing, mainly using regular expressions. By using regular expressions, pattern matching is done for each article and all the numbers/words that match each of the regular expressions are extracted separately. We also use template matching to extract the information out of different patterns/styles of writing.

D. Database

The database is designed according to standard relational database normalization principles.Once the classifier identifies an article as a dengue article, the system stores that in a MYSQL database table called articles. After the Dengue-related articles are stored in the database, the script for extracting data from the articles is run. Then those data will be included in another table. All the locations mentioned in articles and their coordinates will be stored in another table called locations for the purpose of visualization.

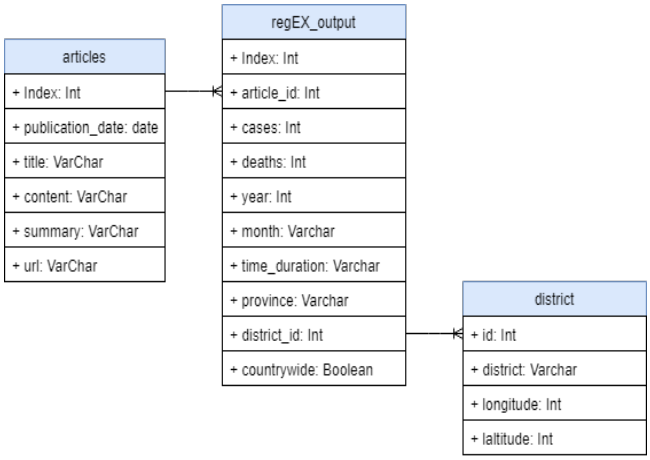

Fig. 3. UML Diagram for database

E. Visualization

Visualization is done by using a web application. The database is queried as needed to obtain the necessary data and those data are represented using graphs and a map.

1) Graphs: Graphic representation is an effective way to present the patterns of the spread of a disease. Several highcharts are used here in the representation in order to give the best out of the collected information. The users can analyze the patterns of the Dengue cases and Dengue-related deaths against years, months of the year and districts. The graphs plotted against months and districts also allow the users to customize them to view them against different years. The data for the charts are queried and fetched from the database tables where the extracted data from the articles are stored. The public can use these graphs to analyze the patterns of Dengue occurrences, compare the occurrences against years, months and locations. They will also be able to make predictions based on the patterns and make necessary decisions to be cautious from possible future Dengue outbreaks. Health professionals can take necessary actions to avoid any Dengue outbreak predicted to be occurred in the future.

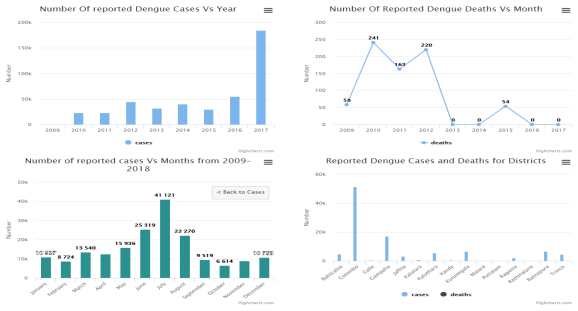

Fig. 4. Graphical Interface for Data

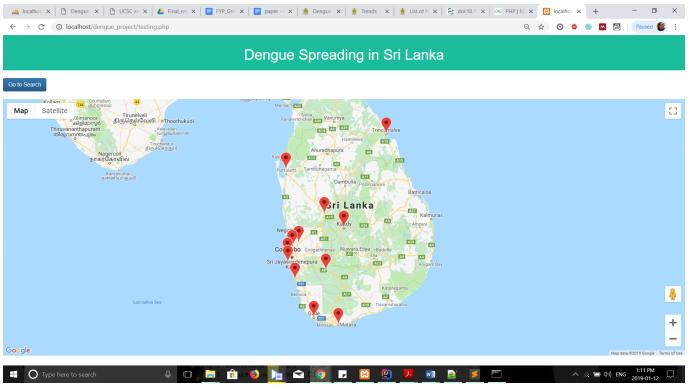

Fig. 5. User Interface for the Map

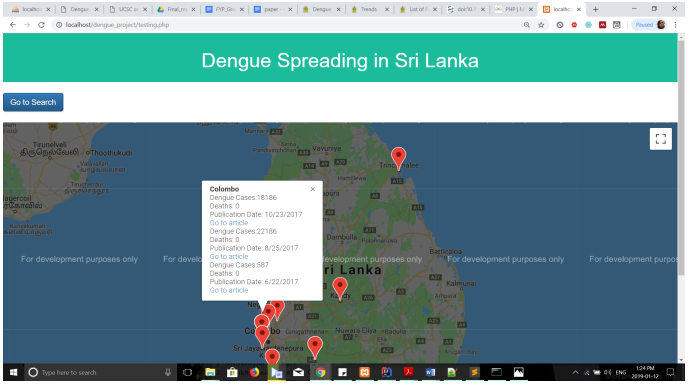

Fig. 6. Location Info-window

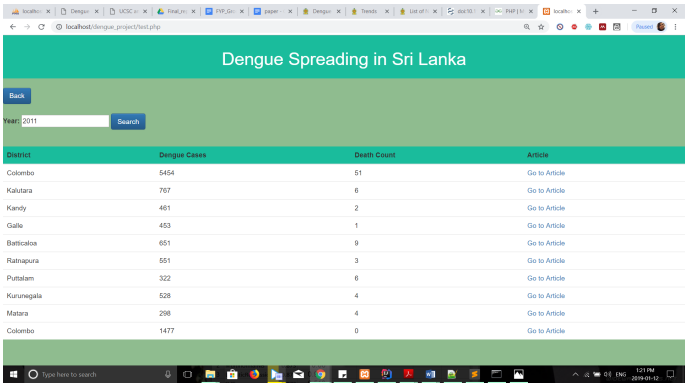

Fig. 7. Search Page for the Map

2) Map Interface: Fig. 5 shows the map interface of the system.It displays the locations of reported Dengue cases using markers. Each marker has an info-window as shown in Fig. 6 which indicates all the cases reported in that location along with the article URL which includes the information. Therefore user can navigate to the article by clicking the URL and read the article for more details. The data needed for the map representation are queried from the database. Fig. 7 shows the search page of the map interface. It gives the facility to search the articles by year.

V. Conclusion And Future Works

In this study we present an event-based approach to detect dengue spreading in Sri Lanka; a system that acquires newly published online Dengue related news articles in Sri Lanka and visualizes the useful information using an web application. Dengue has become a deadly threat for Sri Lanka and therefore testing and implementing another approach to strengthen the traditional system is important. The systems that have used this approach in other countries have proved to be reliable and timely to a great extent. This application is publicly available as a reference for the health professionals and for the general public. But this needs some improvements to become a lot useful for them.

As the future works, improving this system to focus on all the major diseases in Sri Lanka other than Dengue, can be done. And also, improvements can be done by expanding the sources used to extract data about disease occurrences to a broader range, such as other online newspaper websites, news websites, blogs and social media feeds.

References

[1] C. C. Freifeld, K. D. Mandl, B. Y. Reis, and J. S. Brownstein, HealthMap: global infectious disease monitoring through automated classification and visualization of Internet media reports., Journal of the American Medical Informatics Association : JAMIA, vol. 15, no. 2 , pp. 1507, 2008.

[2] E. Velasco, T. Agheneza, K. Denecke, G. Kirchner, and T. Eckmanns, Social media and internetbased data in global systems for public health surveillance: a systematic review., The Milbank quarterly, vol. 92, pp. 7 33, 3 2014.

[3] E. Christaki, New technologies in predicting, preventing and controlling emerging infectious diseases., Virulence, vol. 6, no. 6, pp. 55865 , 2015.

[4] M. Keller, M. Blench, H. Tolentino, C. C. Freifeld, K. D. Mandl, A. Mawudeku, G. Eysenbach, and J. S. Brownstein, Use of unstructured event-based reports for global infectious disease surveillance., Emerging infectious diseases, vol. 15, pp. 68995, 5 2009.

[5] A. Villanes, E. Griffiths, M. Rappa, and C. G. Healey, Dengue Fever Surveillance in India Using Text Mining in Public Media., The American journal of tropical medicine and hygiene, vol. 98, pp. 181191, 1 2018.

[6] L. C. Madoff and J. P. Woodall, The Internet and the Global Monitoring of Emerging Diseases: Lessons from the First 10 Years of ProMEDmail, Archives of Medical Research, vol. 36, pp. 724730, 11 2005.

[7] A. Lyon, M. Nunn, G. Grossel, and M. Burgman, Comparison of Web-Based Biosecurity Intelligence Systems: BioCaster, EpiSPIDER and HealthMap, Transboundary and Emerging Diseases, vol. 59, pp. 223 232, 6 2012.

[8] R. Grishman, S. Huttunen, and R. Yangarber, Information extraction for enhanced access to disease outbreak reports, Journal of Biomedical Informatics, vol. 35, pp. 236246, 8 2002.

[9] N. Collier, S. Doan, A. Kawazoe, R. M. Goodwin, M. Conway, Y. Tateno, Q.- H. Ngo, D. Dien, A. Kawtrakul, K. Takeuchi, M. Shigematsu, and K. Taniguchi, BioCaster: detecting public health rumors with a Web-based text mining system., Bioinformatics (Oxford, England), vol. 24, pp. 2940 1, 12 2008.

[10] A. K. Jain, M. N. Murty, and P. J. Flynn, Data clustering: a review, ACM Computing Surveys, vol. 31, pp. 264323, 9 1999.

[11] R. Feldman and J. Sanger, The Text Mining Handbook, 2006.

[12] M. Torii, L. Yin, T. Nguyen, C. T. Mazumdar, H. Liu, D. M. Hartley, and N. P. Nelson, An exploratory study of a text classification framework for Internet-based surveillance of emerging epidemics., International journal of medical informatics, vol. 80, pp. 5666, 1 2011.

[13] Scikit-learn.org. (2019). 3.2.4.3.1. sklearn.ensemble. RandomForestClassifier scikit-learn 0.20.2 documentation. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble. RandomForestClassifier.html