UNIVERSITY OF CALIFORNIA, SAN DIEGO

PIRC-Net: Twitter-based on demand public health framework for HIV risk estimation

A master thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Computer Science

by

Ajay Mohan

Committee in charge:

Professor Nadir Weibel, Chair
Professor Amarnath Gupta
Professor Lawrence Saul

2017

The Thesis of Ajay Mohan is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____
Chair

University of California, San Diego

2017

DEDICATION

To my beloved **Mom**, **Dad**, **Brother**, and all my **Teachers**. Without you this journey would have been impossible.

EPIGRAPH

As for the search for truth, I know

from my own painful searching, with its

many blind alleys, how hard it is to

take a reliable step, be it ever so small,

towards the understanding of that which is truly significant.

Albert Einstein: The Human Side, New Glipses From His Archives

(1981)

TABLE OF CONTENTS

Signature Page . . . iii

Dedication . . . iv

Epigraph . . . iv

Table of Contents . . . vi

List of Figures . . . viii

List of Tables . . . ix

Acknowledgements . . . x

Abstract of the Thesis . . . xi

Chapter 1 Introduction . . . 1
1.1 Motivation . . . 1
1.2 Scope & Goals . . . 3
1.3 Background & Related Work . . . 4
1.4 UC San Diego contributions:Synopsis . . . 5
1.4.1 Data Collection & Cleaning . . . 6
1.4.2 Evolution of PIRC-Net . . . 9

Chapter 2 Requirements Elicitation . . . 13
2.1 Problem Abstraction . . . 13
2.2 Collaborator Interviews . . . 15
2.2.1 Interview Questionnaire . . . 15
2.2.2 Task Characterization and Abstraction . . . 16
2.3 Feasibility Analysis of the Requirements . . . 17
2.3.1 RQ1: Identify - Risk Behavior exhibiting Events and Specific Venues . . . 17
2.3.2 RQ2: Discover - New population classified in to the clusters . . . 19
2.3.3 RQ3: Discover - Social distance between individuals 19
2.3.4 RQ4: Compare - HIV transmission network with PIRC-Net Risk Network . . . 20

Chapter 3 PIRC-Net Parallel Framework . . . 21
3.1 Architecture Overview . . . 21
3.2 Live Tweet Classifier . . . 21
3.3 Event Extraction and Remodeling . . . 24

3.3.1 Events from Tweets . . . 25
3.3.2 Events from Eventbrite . . . 26
3.3.3 Event Analysis - Ranking Algorithm . . . 26
3.3.4 Vocabulary Adaptation & Classifier Remodeling . 28
3.4 PIRC-Net Dashboard: Machinery . . . 29
3.5 Persistence and Batch Clustering . . . 30
3.6 Monit Daemon - Status Reporting . . . 31

Chapter 4 PIRC-Net Dashboard: A Real-time visual analytics system . . 33
4.1 Coordinated and Multiple Views . . . 34
4.2 Visual Elements and User Interactions . . . 35
4.2.1 Event Calendar and Event Geo Map . . . 36
4.2.2 User TreeMap and Population bar chart . . . 36
4.2.3 Tweet, Bucket Charts & Word Cloud . . . 37

Chapter 5 Benchmarks . . . 38
5.1 Benchmarking Platform . . . 38
5.1.1 Hardware Configuration . . . 38
5.1.2 Software Configuration . . . 40
5.2 Performance and Throughputs . . . 40
5.2.1 Classification Throughput . . . 40
5.2.2 Keyword Matching Performance . . . 41
5.2.3 Peak Load Provisioning . . . 43

Chapter 6 Conclusion . . . 44
6.1 Evaluation Discussion . . . 44
6.2 Future Work . . . 46

Appendix A HIV Vocabulary . . . 49

Bibliography . . . 50

LIST OF FIGURES

Figure 1.1: HIV infection demographics from CDC 2
Figure 1.2: Stages of Data Collection 7
Figure 1.3: SVM RoC Analysis. 9
Figure 1.4: Data Model of the HIV at-risk social network. 10
Figure 1.5: Evolution of the PIRC-Net Framework. 11

Figure 3.1: Architecture of the PIRC-Net Framework 22
Figure 3.2: Stall warning response format. 23
Figure 3.3: MVVM Design Paradigm in PIRC-Net Dashboard. 30

Figure 4.1: PIRC-Net Dashboard: User Interactions 35
Figure 4.2: Pircnet Dashboard Prototype 37

Figure 5.1: Classification Throughput 41
Figure 5.2: Keyword Upper Limit Estimation 42

LIST OF TABLES

Table 1.1: Labelled Tweets Statistics 8

Table 2.1: Interview Questionnaire 16
Table 2.2: Task Characterization and Abstraction 18

Table 5.1: Hardware Configurations 39
Table 5.2: Software Configurations 40

Table A.1: Vocabulary Set 49

ACKNOWLEDGEMENTS

To start off, I am immensely thankful to Prof. Nadir Weibel for being such a wonderful advisor. Thanks for all the brainstorming and discussion sessions, which served as a ray of light while we were trying to look beyond the horizon and beyond the known. Thank you for challenging me and offering different perspectives in approaching a problem. A special thanks to Prof. Amarnath and Prof. Lawrence for being on the committee and lending their immense knowledge to evaluate the project's progress and its future.

I am also very thankful to experts from the AntiViral Research Center, especially Dr. Susan Little for being a vital contributor and being a close associate with us. We gained a lot domain insights courtesy of the interactions with Dr.Susan and her team. To Antoine Chaillon for experimenting with our initial prototypes. Last but not the least, to Ali Sarvghad for all the guidance and support in shaping the visual dashboard. Thank you all for your support, thanks a ton.

ABSTRACT OF THE THESIS

PIRC-Net: Twitter-based on demand public health framework for HIV risk estimation

by

Ajay Mohan

Master of Science in Computer Science

University of California, San Diego, 2017

Professor Nadir Weibel, Chair

Human Immuno-deficieny Virus (HIV) is one of the deadliest known viruses to human-kind which when left untreated can lead to irreparable damages to the immune system. The body's immune system cannot get rid of this virus and there is no medicine invented yet that can completely cure this disease. This though doesn't mean that HIV isn't treatable. Infected People can still lead better lives by detecting the infection early and employing better medical care. Therefore there is an imminent need in identifying HIV at-risk population and providing them with

medical care. This can be achieved by adopting a methodical HIV risk assessment strategy that is fast and efficient.

Contemporary methods of HIV risk assessment have a long turn around time. It relies on static data from national census statistics and other surveys. Though, in some cases this static data it is complemented with local data documented by the HIV clinics about their patient population, it only represents the specific subset of HIV at-risk individuals that are already seen at clinics for treatment or testing. Therefore, lack of early testing in many HIV at risk individuals, hinders the ability to account for many people who are at risk. Hence the HIV population with highest probability of infection transmission continue to remain undiscovered.

Our goal is to use a computational approach, rooted in the analysis of people's online communication, to uncover this population at an earlier stage through a near real-time intervention system to help doctors and researchers to respond to HIV risk effectively. This thesis is focused on the underpinnings of how we can enable researcher's exploration of HIV epidemic and discuss the techniques that we used for it. It also elaborates on the integrated pipeline that is the backbone of this system. It is an amalgamation of components which perform natural language processing, supervised learning and extracts network features, filtering of data publicly available, and confidentially collected from Twitter.

The ultimate goal of the developed infrastructure is to help clinicians to prune demographic information and social connections in the local population. Af-

ter discussing our novel contribution in task characterization and event extraction, we then discuss the integration of various additional components to the existing PIRC-Net infrastructure to support these additional features. Ultimately, our integrated platforms allows clinicians to visualize and identify patterns in online social media communication, providing additional tools to inform targeted interventions.

Chapter 1

Introduction

1.1 Motivation

The first two decades of the 21st century has witnessed a staggering increase in the availability of data through different avenues. This can be attributed to the rise in the ubiquity of devices capable of capturing these data, their reduced form factor, and an increased sharing of personal information through social network applications. Therefore, the availability of these data creates opportunities to design new data-driven solutions to inform people's decision-making capabilities in myriads of different scenarios.

Recent literature trends too evince that this data supported decision making process has found its way application in health care by means of *Digital Epidemiology* [1]. Social media data when harnessed effectively, can provide localized and timely information about diseases and health dynamics in populations around the world. As members of the PIRC-Net research group we dedicate our efforts to make use of this abundance in data to analyze important behavioral dynamics that are emerging through social media and understanding the network character-

istics of the people involved to mitigate one of the perilous known diseases, Human Immunodeficiency Virus(HIV) infections.

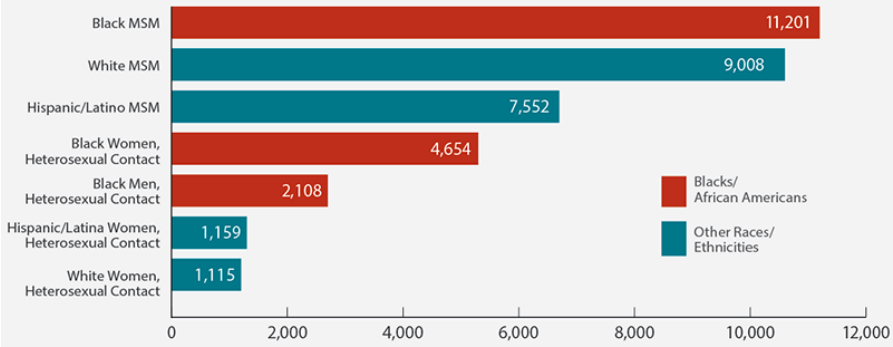

Figure 1.1: HIV infection demographics from CDC

HIV is a major global public health issue. It is an incurable disease which has claimed more than 34 million lives. The Center for Disease Control and Prevention(CDC), reports that there are 50,000 new HIV infections every year and there 35 million people with AIDS worldwide. USA alone has 1.2 million reported people living with AIDS and recorded deaths of up to 660,000 till date. 78% of the new infections in 2010 were Men who have sex with men (MSM). California (along with Florida) had the highest number of HIV diagnoses in 2013 [2]. Figure 1.1 gives an overview of this distribution among different race. With all the statistics, there is not an iota of doubt about the seriousness of this problem. In the following section we outline the key motivating factors for our research work.

1.2 Scope & Goals

Real world assessment of HIV risk is a slow process that relies on static data from national census statistics and other surveys [2]. In some cases these static data is complemented with local data that HIV clinics might have on their patient population, but this only represents the specific subset of HIV at-risk individuals that are already seen at clinics for treatment or testing. However lack of early testing in many HIV at risk individuals, hinders the ability to account for many people who are at risk. Our goal is to reduce the time gap, to uncover HIV at risk individuals, and then build a near real-time intervention system to help doctors and researchers to respond to HIV risk effectively.

Our work is focused on integrating publicly available data that are confidentially collected from Twitter into a unique pipeline. We make use of recent developments in natural language processing, machine learning and social network analysis to filter these data and make them available to clinicians and researchers. The ultimate goal of the developed infrastructure is to help clinicians in characterizing the structure of social networks of local population who may be at risk of acquiring or transmitting HIV infection in the San Diego Area, and to be able to do that in real-time. Our integrated platform allows clinicians to visualize this structure and identify patterns in online social media communication providing an additional tool to inform targeted interventions to reduce HIV risk.

With our work we want to build:

1. A fully integrated on-demand end to end framework of the PIRC-Net infrastructure for risk estimation.

2. A novel way of following at-risk events and venues

3. A dynamic tweet and vocabulary tagging.

4. A visualization Dashboard for experts to use in intervention decision making.

1.3 Background & Related Work

Use of social media to research work in healthcare has been studied for a few years now. A team of 3 researchers headed by Dr. Sean D. Young at UCLA conducted a study on Twitter data to show that social network information could help perform epidemiological analysis [3]. This could further help identify early warning signals for HIV risk. Another study was conducted by researchers at Microsoft Research, Redmond about people who are trying to quit smoking. This study further helped corroborate that people's behavior can be correlated to their social media footprint [4].

Recently, after the World Health Organization announced a Public Health Emergency of International Concern due to an ebola epidemic there was a noticeable activity of signal in twitter feeds. Isaac Chun-Hai Fung and his team from Georgia Southern University studied the data from twitter during this time inter-

val [5]. They were able to detect the spread of misinformation and the specific user reaction after the announcement. The study concluded that social media posts can provide relevant information to public health agencies during emergency responses.

Our work builds on the premise provided by these preceding studies, which provide supporting evidence that proves the efficiency of a prediction system based on social network information. As part of our research we want to realize a live system built to support HIV epidemic prediction and intervention. We would then like this system to be trialled in production with clinicians where the real time performance can then be evaluated to add experimental data points to these preliminary studies.

1.4 UC San Diego contributions:Synopsis

The PIRC-Net researchers are a group of passionate people from the Department of Computer Science and Anti-Viral Research Centre(AVRC), Department of Medicine at UC San Diego. The name PIRC comes from Primary Infection Research Consortium (PIRC) at AVRC who actively research effective ways to combat HIV. The work started with former graduate Students Narendran Thangarajan and Purvi Desai under the guidance of Dr. Nadir Weibel, Dr. Saul Lawrence, Dr. Amarnath Gupta, and Dr. Susan Little working towards a new method of characterizing and identifying HIV at-risk populations.

The effort was focused on the local population of San Diego County and used

publicly available social media data as an indicator for HIV risk. The idea was to apply social network analysis to combine real-time Twitter information extracted and compare this information with the HIV transmission network data obtained from PIRC and evaluate opportunities to target HIV testing and prevention efforts to the communities at the greatest potential risk of HIV acquisition. The following subsections describe about the outcome of this research's early stages.

1.4.1 Data Collection & Cleaning

Twitter's open source Application Programming Interface (API) [6] enables Computer Science researchers to access their publicly available tweet data through an automated program. There are two types of APIs that are commonly used **1) Representational state transfer (REST) APIs** and **2) Streaming APIs**. The Streaming API creates a long-standing connection between the client and the server, and streams the incoming tweets to the subscribing clients. For this research, the streaming APIs were configured to collect all tweets from San Diego location and are save them on to the MongoDB [7]. The unstructured form of the tweets was the reason behind the decision to the use of non sql database in MongoDB. It also supports native map-reduce queries for performing on-demand aggregations at a faster rate.

After data collection, tweets were then filtered to create a corpus of tweets based on the presence of certain HIV transmission risk words in the tweet content 1.2. A 'risk word' is a term correlated with HIV risk behavior identified by domain

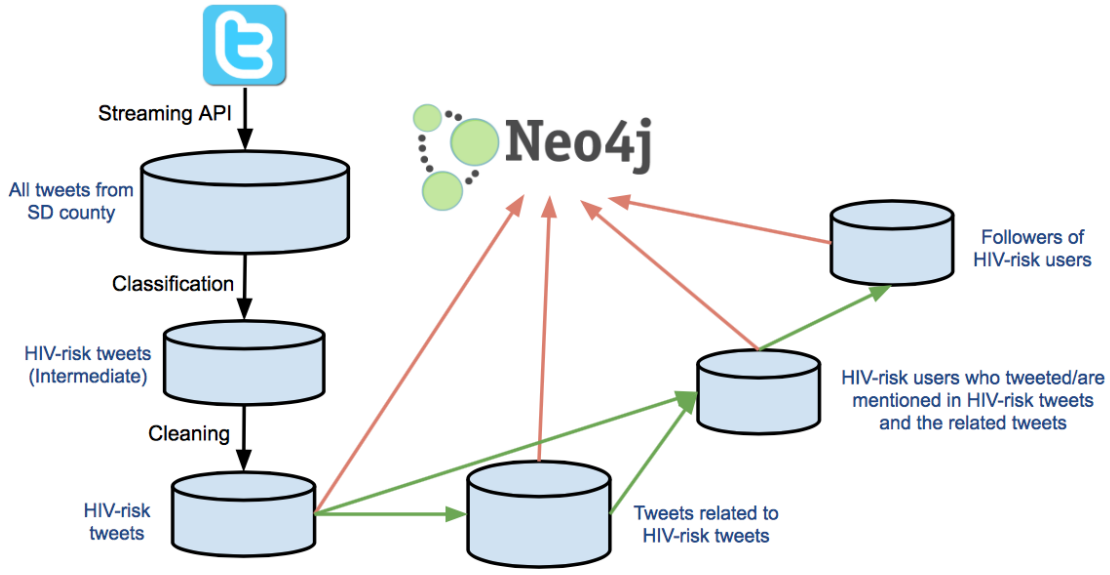

Figure 1.2: Stages of Data Collection: The stages include collection, cleaning of tweet data and extraction of metadata from twitter.

experts and the clinical collaborators on our research team. There risk words were categorized in to 5 risk buckets. A list of these words is listed in Appendix A. The five broad categories of HIV risk words were as follows:**1. Drug Bucket**,**2. Sex Bucket**, **3. Sex Venues Bucket**, **4. Homosexual Terms Bucket**, **5. Sexually Transmitted Infection(STI) Bucket**.

Data collection stage is followed by data filtering stage to remove false positives, and then pulling from Twitter all the related users who either re-tweeted an HIV risk tweet or were mentioned in one. Following which, we perform the Data cleaning process, where we remove some of the words based on an inclusion and exclusion list. For instance, the exclusion list for the keyword 'crack' (which is slang for meth/drug) would include crack me up, crack myself up, crack up, crack

open, crack of dawn. The inclusion list on the other hand would include words that if co-occurred with the keywords under consideration would allow the tweets to pass through this second level of filtering. At the end of this stage around 60% of the noise data was removed from the corpus.

Table 1.1: Labelled Tweets Statistics

Stage	**Positive**	**Negative**
Pilot	171	447
Hackathon	346	1649

This two-stage filtering process although efficient was not practically very usable due to the high number of false positives flagged and therefore there isn't a sizable amount of relevant tweets. Given the volume of the tweets that we were handling and the need for precision in categorization, it was naturally a logical choice to employ Machine Learning tools to build a better filter for the tweets. We initially started of with a pilot labeling activity and then followed it up with a hackathon of labeling tweets from the existing collection. The information of the number labels obtained from these activities are mentioned in table 1.1.

Further with the user connections within the social network alongside analyzing the raw text of the tweets are gathered, we could provide meaningful signal for HIV risk analysis. The user-tweet and user-user connections were modeled in a graph database in the form of nodes and relationships. 1.4 shows the node and edges relationship in our graph database. The user-user connections parameterized

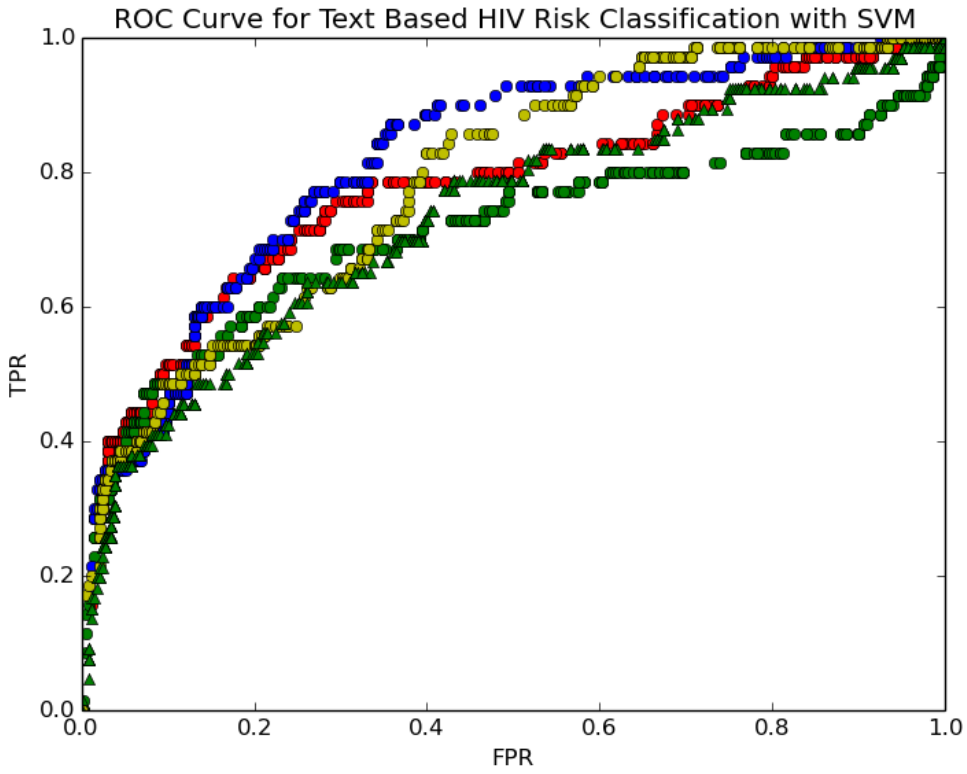

Figure 1.3: SVM RoC Analysis.

by a connection matrix defined by parameters such as mentions, co-located and conversations was also identified to improve the classifier's performance.

1.4.2 Evolution of PIRC-Net

The stand alone classifier instance or the mono-classifier, gathers and filters tweets that are potential candidates for HIV risk analysis. It also gathers the corresponding Twitter users and relevant network based data from Twitter's social graph such as follower/followee connections, retweets, and mentions. There is a

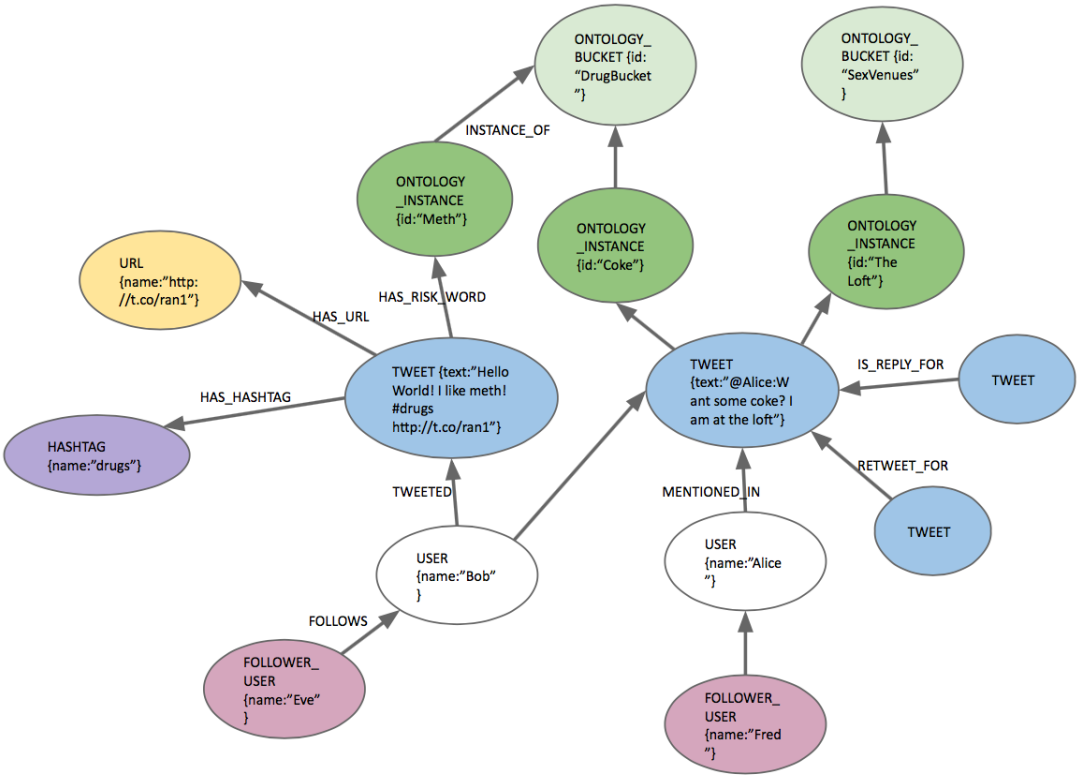

Figure 1.4: **Data Model of the HIV at-risk social network.**The nodes in the figure represent the individual entities and the edges represent their relationship

greater level of detail about the what our PIRC-Net is currently capable of in the subsequent chapters.

Figure 1.5 gives an account of the evolution of the PIRC-Net framework. At the top of the figure we have the basic building block, the instance of the stand-alone classifier with details of all the fundamental stages and to its bottom we have the high level architecture view of our current state of the pipeline.

This thesis explains how the PIRC-Net architecture functions as an exploratory HIV intervention system. In chapter 2, we spell out the details of the requirements elicitation phase that governed our visualization system, chapter 3

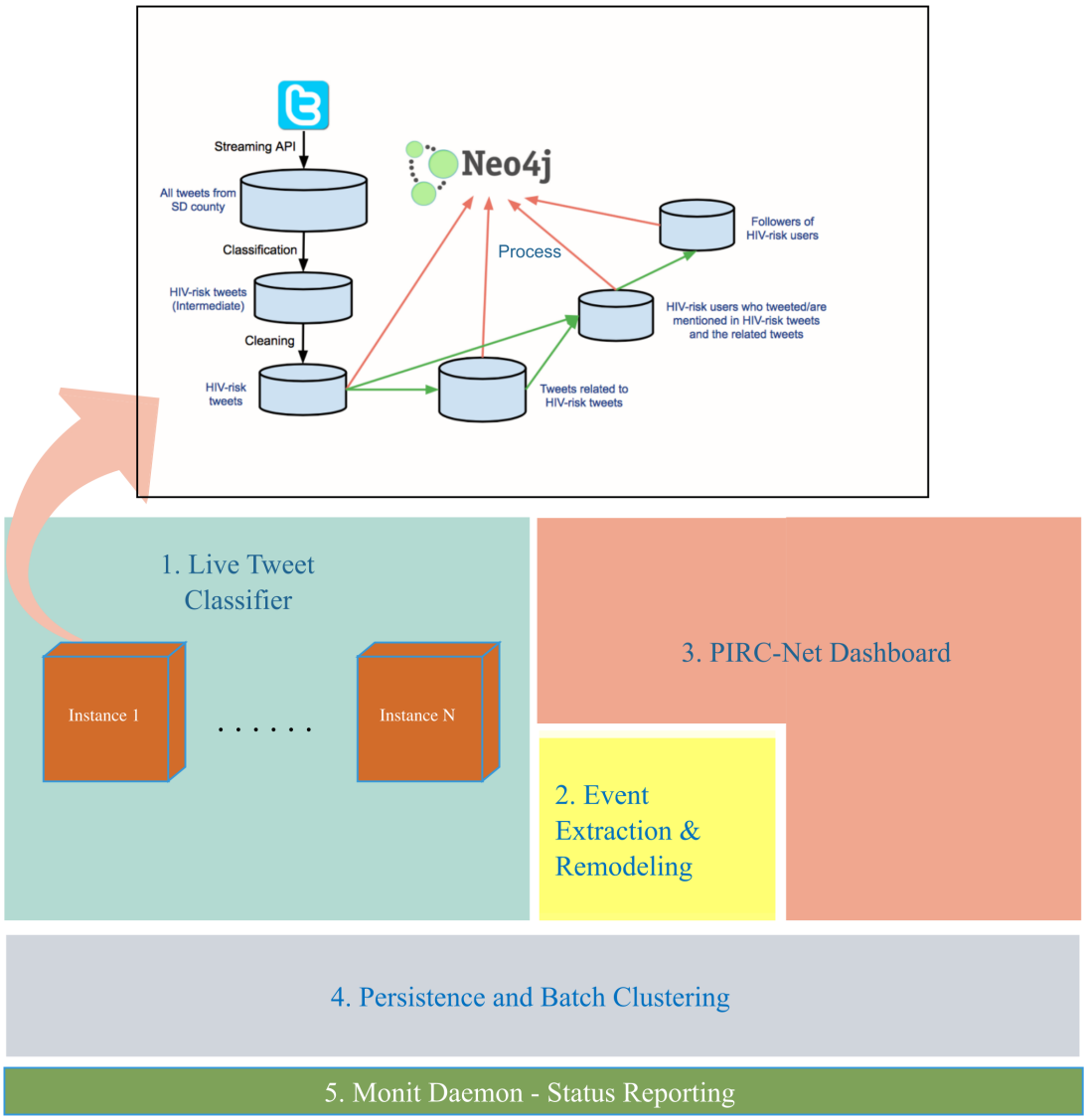

Figure 1.5: Evolution of the PIRC-Net Framework.

focuses on the aspects of transforming the PIRC-Net initial mono-classifier prototype to a fully blown parallel framework for an on-demand classification and visualization. Chapter 4, explains the reasoning behind the selection of specific visual elements in the dashboard used by researchers and how it helps to achieve

our goal of near-real time interventions. Chapter 5 evaluate the design and the performance of the PIRC-Net system. We then conclude with a critical evaluation of the infrastructure, with insights and suggestions for future work in chapter 6.

Chapter 2

Requirements Elicitation

2.1 Problem Abstraction

The mono-classifier produces a very large number of tweets identified as risky tweets and is accompanied by an equally large list of users. The question then ahead of us was this: "How can we present this data so that it makes sense for clinicians/researchers to take further action?". The PIRC-Net infrastructure exhibits characteristics of a big data application. It handles workloads of varying types of datasets trying to find patterns and correlations to take informed decisions. Our intuitive response would then logically be to make use of visualization techniques to represent this information to the audience, so that it can be used and acted upon. But coming to this conclusion without the support from domain experts is one of the pitfalls in designing. Thus we embarked on this requirements elicitation phase for HIV risk estimation and its results are amongst the novel contributions of our work

We did have a few ideas about the different ways to utilize this data, but the lack of the domain knowledge was an inhibiting factor. Therefore to elicit the

specific requirements of the kind of the system that would be useful, we needed to get the domain experts on board and get them to understand our infrastructure. Design study methodology by Michael Sedlmair et al [8] gives a detailed road map on the path and stages involved in conducting successful requirement elicitations from prospective users. The overview of the different stages is listed below for a quick reference.

1. ***Precondition***:This phase comprises of learn, winnow, and cast which are stages to get the visualization researcher ready.

2. ***Core***: The next phase is discover, design, implement, and deploy stages. In general this phase extends from problem characterization to deployment. We will be discussing about the discover stage in more detail in this chapter to elaborate on our requirement elicitation process.

3. ***Analysis***: Reflect and write are the stages of this phase, which involves a thorough introspection of the outcomes from the earlier two stages.

We had identified our collaborators very early when we interacted with researchers from AVRC, UC San Diego. Their initial contribution was towards defining the labeled datasets of the tweets. We sought their help again to get them involved in the requirement elicitation phase.

2.2 Collaborator Interviews

There were 5 domain experts in total who were involved in the requirement elicitation study. We had two roles amongst the collaborators as defined in design study parlance. One **gatekeeper** and four **front-line analysts**. Each of the individual was interviewed with a specific set of questions to get their views on how to efficiently can the PIRC-Net data be put to use. There was an unequivocal interest expressed in using the data as an additional data point to support some of their decisions. They also opined that this presents us with plethora of new possibilities but also expressed concern to difficulties in measuring the effectiveness of such a system. This is because of the fact that there is no standard method or a benchmark to compare against. This is expected in a new study involving artifacts that weren't available before. The interviewing process took the following course.

2.2.1 Interview Questionnaire

The interviews with the HIV researchers was conducted with the help of a prepared set of questionnaires with a common template adopted to every individual. This ensured that we had the same set of questions whose responses can be evaluated objectively. Each one of them had their own unique way of tackling the questions and the conversation quite often revolved around several tangential but important points. As the interviewers, our role was to bring back the attention to the specific task abstraction being discussed. We also facilitated the domain

expert to delve on a specific topic for a longer time than on others. When conducting these interviews in-person we were able to gather more information than doing them in remotely. The following are the generalized questions that were put forth to each of our interviewees.

Table 2.1: Interview Questionnaire

S.No	**Generalized Category of Questions**
1	What are the specific ways in which the data that we have could help clinicians?
2	What are the questions that you would want to ask about this data?
3	What are information that you would like to see highlighted?
4	How do you evaluate the presence of current HIV related intervention techniques?
5	If the data is validated with real world HIV patients at testing centers what feature do you think will be helpful then?

2.2.2 Task Characterization and Abstraction

The response to the questionnaire was audio recorded and transcribed. These recordings were heard several more times so as to to completely pen all the tasks that the researchers want to carry out. After capturing these tasks as comments and snippets of conversations, they were converted in to an Abstraction statement that is useful to contemplate on the various forms of visualization to be considered.

These task abstractions don't spell out the visualization type or the specific

tool that we should employ to convey the information but captures the specific dimensions that are a mandatory requirement to the end user. The final listing of the major requirements gleaned as a result of this interview process is explained in the table **??**.

2.3 Feasibility Analysis of the Requirements

2.3.1 RQ1: Identify - Risk Behavior exhibiting Events and Specific Venues

The process of identifying specific events which we characterize as risky, their corresponding dates and location venues is not a part of our current pipeline. This adds a new requirement to extract this information and extracting such data points from tweets is one of the key problems of interest in natural language processing. We decided to approach this problem by looking for regular expression patterns in the tweets from twitter handles of interesting venues and translating these to specific event dates. For example, A sentence like, "Let's meet next Monday at 2:00pm" in a tweet can be converted to a specific date. Even if temporal qualifier words like 'next', 'later', 'since'?, 'tonight' etc are used. While this can solve the problem of extracting the event dates, the problem of extracting what the event title and description meta-data is still a unsolved. We worked around this difficulty by deciding to show information about the specific venue that was

Table 2.2: Task Characterization and Abstraction

RQ1	**Comments**	**Abstraction**
RQ1	1. "We need to identify event dates at a night club that has one or two events that are risky" 2. "Find a location repeated a couple of times like a with restaurants and bars.."	**Identify:**Risk Behavior exhibiting Events and Specific Venues
RQ2	1. "Find out probable "incident" infections.. users just new to the cluster" 2. "We need granular data of clusters. They are mostly(men, women, other), of age(20-25), their main substance of abuse is(meth, cocaine, asphalt)" etc.	**Discover:**New population classified in to the clusters
RQ3	1. "I should be able to find out people with whom a given person can physically meet with" 2. "What is the social distance between given two individuals"	**Discover:**Social distance between individuals
RQ4	1. "Growing HIV clusters rate..risky tweets clusters" 2. "Where are they seeking treatment and overlay this information with twitter risk network"	**Compare:**HIV transmission network with PIRC-Net Risk Network.

involved in the tweet.

Additionally, we identified, open source REST based servers of event management portals like eventbrite.com that offered developer APIs to fetch event information. The resulting meta-data had comprehensive list of information that can give insights about a specific event. Hence from these results we concluded that we will be able to meet this requirement.

2.3.2 RQ2: Discover - New population classified in to the clusters

The need to identify new clusters characterized as at-risk is crucial. This is because these are the clusters that have the maximum probability of being 'incident infections???. The reason why identifying incident infections is very important is well documented in literature. The incident infections are people with the highest viral loads and the potential to infect many other people. This requirement can be met by querying the twitter user profile information of new users identified as risk, and one of our components performs this activity as a nightly batch process. As a consequence, we concluded that this requirement is feasible.

2.3.3 RQ3: Discover - Social distance between individuals

Enumerating the connectedness of individuals has two challenges. 1) There is no existing parameter to define how well connected two individuals are for our

specific use cases. 2) For ethical reasons, we shouldn'??t be capable of tagging individuals such that they are traceable to their individual traits and behaviors exhibited online. On further analysis, we were able to prove that these two challenges are solvable. We defined a '??Social distance' parameter that defines connectivity between individuals as a measure of how likely are they to meet each other. This was built on 7 user specific information obtained from a combination of data from the user's personal profile and processing results of our infrastructure. The second challenge was more of an engineering problem than a research problem and solution is to obfuscate the user identity such that it is not traceable to the original user's personal identity through the at-risk behavior.

2.3.4 RQ4: Compare - HIV transmission network with PIRC-Net Risk Network

The HIV transmission network is available with the PIRC cohort, and we generate the twitter risk network. Although the data sources are available, there are two challenges in realizing this requirement. 1) The transmission and risk network share no common characteristics or attributes. They are composed of entirely different people, and there may not be any intersection in the people attribute space. In design study, pursuing direct visualizations when not backed by a task is one of the major pitfalls [9]. Hence we found this requirement not to be feasible and therefore decided not to pursue it.

Chapter 3

PIRC-Net Parallel Framework

3.1 Architecture Overview

The PIRC-Net framework consists of 5 components which orchestrate together as a whole fully blown framework. The are grouped together based on their features and functionalities. 1)Live Tweet Classification, 2)Event Extraction & Remodeling, 3)Visualization Dashboard, 4)Persistence and Batch Clustering, 5) Monit Daemon - Status Reporting.

The rectangular boundaries in the diagram define the extent of each component. The border also means that the particular component can be replaced or updated without being dependent on other components. The next few sections will delve a little longer giving a detailed outlook on every component.

3.2 Live Tweet Classifier

The Live tweet classifier component is the parallel tweet collection and classification module. It is the first stage through which the incoming tweets are collected and assorted. The fundamental building block of this component is the

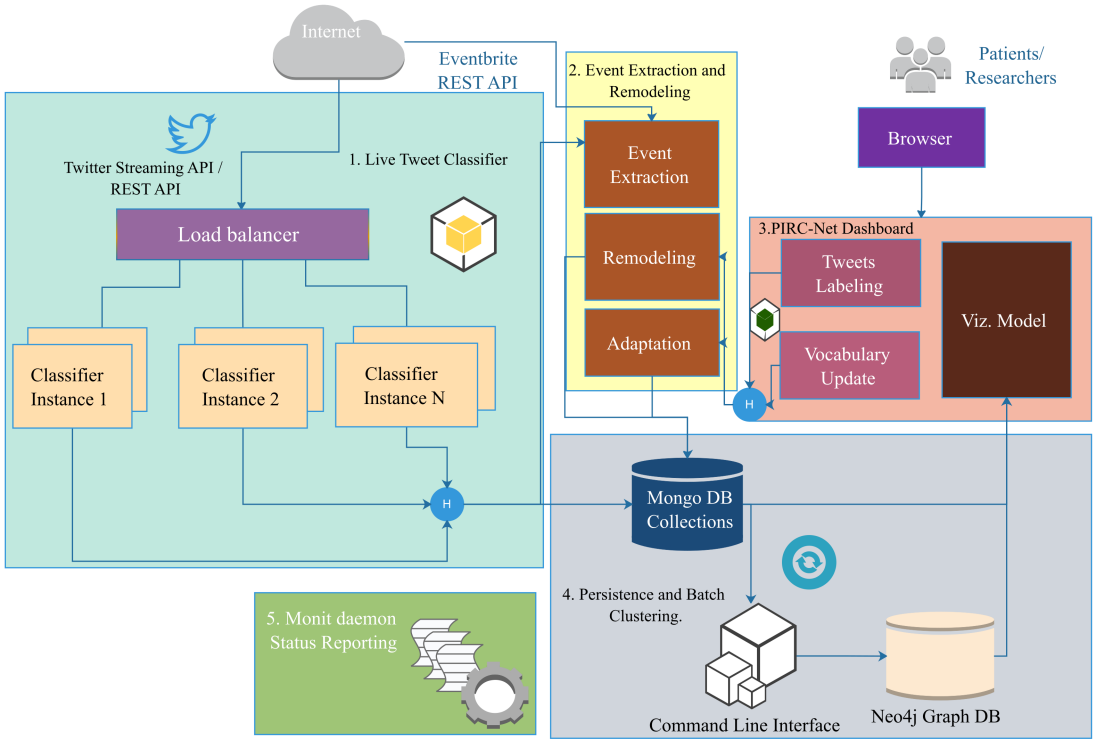

Figure 3.1: Architecture of the PIRC-Net Framework.1)Live Tweet Classifier classifies the incoming tweet, 2) Event Extraction module collects event information from multiple sources, 3) PIRC-Net Dashboard, the visual interface of our system for HIV researchers, 4) Persistence and Batch Clustering keeps the data in usable state, 5) Monit ensures the stability of the entire system.

PIRC-Net classifier instance which is the outcome of the previous research work detailed in section 1.4. The classifier themselves are an SVM-based classifier with a tuned hyperparameter obtained from the labeled tweets. The initial prototype classified a small set of tweets until the parameter finalization. After the parameters were optimized, the model was persisted and was then used to classify live incoming tweets. The load balancer driver is the front end of the component which monitors the tweet streams from the twitter servers and segregates the tweets to the processing block. The module, therefore, acts as a live tweet classifier, han-

dling the input streams. The load balancer adds the new incoming tweets to a persistent storage when there is a spike in traffic. Spark Parallelizable context then takes control with the Resilient Distributed Dataset or the RDDs populated by the newly seen tweets. The RDDs are Apache's Spark level abstraction, which is an immutable distributed data store that makes parallel processing efficient and simpler.

Initially, when the live tweets were required to be processed by the classifier, we expected that the additional complexity of processing a tweet would make our client fall back in performance. And when the prototype was first integrated we saw this phenomenon manifest in our pipeline.

```
{
  warning:{
    code:"FALLING_BEHIND",
    message:"Your connection is falling behind and messages are
       being queued for delivery to you. Your queue is now over
       60\% full.",
    percent_full: 60
  }
}
```

Figure 3.2: Stall warning response format

This lag in processing can add up in the long run, and therefore result in the loss of critical tweet data. This solution although, required that we had a good

sense of the tweet traffic and we provision for maximum load

This solution doesn't give us a deterministic way of handling the problem. We then had a second design taking a more conservative approach that is also more deterministic than depending on the cache size. The solution was to use the stall_warning flag to handle additional traffic in the stream through other nodes that get added into the worker pool.

3.3 Event Extraction and Remodeling

This component is another novel contribution to the pipeline. It comprises of three subcomponents 1)Future Event Extraction, 2)Vocabulary Adaptation, and 3)Classifier Remodeling which form the bare bones of our infrastructure's intuitiveness in identifying key events. They also play a vital role in keeping the Live tweet Classifier component updated to the current trends in the web. The key takeaways of the requirement elicitation phase described in chapter 2, is that the domain researchers needed data about the events that are happening around a given community. They wanted this information to serve them during planning sessions of intervention techniques. Therefore the information of risky tweets by themselves was not found to be very useful in the decision-making process, but it is the peripheral data created from the sources that are very significant. Thus we started building the event extraction module, and the first requirement was to identify the data sources. We identified two sources of data

- Timeline Updates from Twitter handles of interest
- Location Specific events extracted using REST APIs from Eventbrite.com portal.

3.3.1 Events from Tweets

The timeline updates of specific Twitter handles which represent the social network footprint of targeted venues are extracted using Twitter REST APIs. We need to note that we are using REST APIs instead of the filter hose streaming APIs because we are interested in all the tweets of the past. We had to keep in mind the rate limiting factors and hence throttle the API requests so as to avoid the connection getting terminated whenever we used the REST APIs. We then iterate through the list of these twitter handles and gather all their pages to go back in the past as configured in our settings or the date of creation of the Twitter account whichever is the earliest. As an initial start point, we collected tweets from until a month ago. The entire list of Twitter handles is published in Appendix A. The tweets that gathered were then added to the database, if they are deemed relevant by the event analysis algorithm. The tweet text and the venue name acts as the input information to the event analysis algorithm. As discussed in the feasibility analysis section, identifying events description from tweets with higher accuracy is a hard research problem. Nonetheless, we work around this restriction by embedding information from the tweet as the events its description.

The hyperlinks and other related information act as useful metadata.

3.3.2 Events from Eventbrite

The other data source for our event collection is Eventbrite, a large online trove of event metadata. The advantage of using REST APIs of Eventbrite is that it has a highly granular list of events, their metadata and a precise list of filters to apply to focus on specific events in very specific geography. This feature makes it very easy to process the event information and complement it with the other information available to us. Our endpoint requests all the future events that are currently in the scheduled state from the Eventbrite repository. We use the Eventbrite wrapper to connect to event search endpoint and retrieve these results. This algorithm executes as a batch process updating the list of events daily. The event analysis algorithm processes the metadata information and adds to our collection of monitored events.

3.3.3 Event Analysis - Ranking Algorithm

The event extraction from the two sources gives us an extensive collection of events. We need to sift through this plethora of events to highlight what is relevant and leave what out the rest. The Ranking algorithm is designed based on our tweet classification algorithm. Thereby using this ranking system we can now prune the list of events to use only the significant subset. The popularity measure of events depends on the following conditions.

- Categorized as Risky by PIRC-Net
- Proximity to a location of interest
- Previous Year Twitter Activity on this day/time

The event title and the description of the events are processed through the workflow which is similar in taste to the classifier algorithm. The first step is to look for key words that form the vocabulary risk set. The next stage we use one of the emotion analysis using a library from the Python Natural Language processing ToolKit(NLTK) called textblob. Textblob comes with a comprehensive set of tools that helps with parts of speech tagging and sentence parsing. We use it in our case to help detect the emotion conveyed in any given text corpora, and this works in harmony with our SVM classifier to categorize the events as risky or non-risky. We have already discussed the way in which the SVM classifier works and so let us understand the working of our emotion analysis. We work with two pieces of information 1)Polarity, 2) Subjectivity.

Polarity values range from [-1.0,1.0], the closer the polarity value is towards the positive end, the more positive is the emotion that is being conveyed by the sentence. Subjectivity is a measure of how objective or subjective is this statement. The value ranges from [0.0, 1.0]. A value of zero means the given sentence is more objective, and one means it is subjective. With the averaged polarity and subjectivity values of a given text sentence we can categorize the event as risky or non-risky and in turn, add them to the collection of events in our persistence

medium.

The reason to use a less restrictive version of the classifier prediction by including the emotion analysis is to take a conservative approach. It is more conservative to add a few additional events to the list to avoid the risk of missing out on some of the relevant events. Additionally, given that this is the first implementation of event extraction feature we decided to allow events of lower score to be rendered at the user interface.

3.3.4 Vocabulary Adaptation & Classifier Remodeling

The list of words that form the risk vocabulary set is listed in Appendix A. Finding the keywords that belong to the list from the tweet and determining if the tweet alludes to a risk behavior is a challenging problem in natural language processing. The reason being that references in human language are often subtle and the words might completely mean something different in another context. In addition to it, certain words are in vogue for a given short span of time, and then they go out of use. These slang words need to be regularly captured to keep abreast with the changing dynamics of the web. Therefore to achieve this, we have a continuous vocabulary update enabled in the background.

Accordingly, the solution that we built is to modify this vocabulary set with new words where these words are obtained from people waiting in the testing centers. This solution is not perfect, where the word list can keep increasing indefinitely. The performance of our infrastructure can be reduced significantly

with increase in the number of words. Therefore we studied the ideal number of words that can be part of the vocabulary, and have this as a theoretical upper bound on the number of words that can be searched at any given time.

3.4 PIRC-Net Dashboard: Machinery

PIRC-Net dashboard is the visual analytics tool that is used by clinicians to perform exploratory data analysis of the HIV risk information collected through our pipeline. A detailed description of the design of the visual elements, their choice, and their utility to analyze the information is deferred until Chapter 4. In this section, we will be looking at the machinery that prepares and serves the data for active exploration. The PIRC-Net dashboard is a web based tool which is built using the Angular.js framework following a Model, View, View-Model(MVVM) design paradigm.

The model is populated by querying a python based REST Server. The reason for selecting a python based server is to ensure that we reuse the components of our framework. The view is built using D3, a javascript visualization library. We chose d3 due to its greater flexibility, the ease of modifying the document object model(DOM) and its seamless browser integration makes it an ideal choice for the front end. The view-model controllers bind and mediate interaction between the model and the view.ie, the REST Server model builders and the D3 client side view. The REST server aggregates the data in batches at periodic intervals and

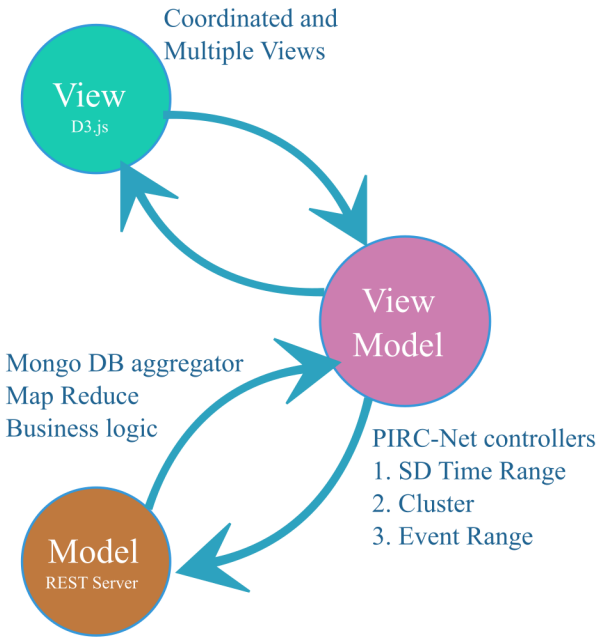

Figure 3.3: MVVM Design Paradigm in PIRC-Net Dashboard: The view comprises of the the visualization libraries used for rendering, the view-model orchestrates the control between the mode and the view, the business logic resides with the model.

saves it in a persistent store. This method is to avoid slow responses when handling huge loads of data at the server. An evaluation of the query response time on an average should be done to measure the responsiveness of the server. This activity is something that we haven't done yet and a good way to quantify the usability.

3.5 Persistence and Batch Clustering

Our pipeline uses Mongo DB as the persistence database as it provided ease of storage and access to tweets which were semi-structured documents. Contrastingly achieving this on a relational database would have been harder and costlier

due to many table normalizations. Secondly, our application is a big data application we had to perform several on-demand data querying and processing. And since we also wanted to visualize the data, we decided to take advantages of the aggregators and map-reduce techniques built into the Mongo platform.

The Batch clustering component comprises of a number python scripts which act on the results of the Live tweet Classifier stage. Our design requires that we have information about both the tweet and the authors of the tweets. Since on classification, we are only left with data about the incoming tweets we need to batch process retrieving the particular user's information from the Twitter server using their REST APIs. This information is then fed into a graph database called Neo4j [7]. The graph database will help us to easily establish a relationship between people as a social distance metric. The social distance is a metric which defines how likely are two individuals to meet in the physical world given their behavior on Twitter.

3.6 Monit Daemon - Status Reporting

The infrastructure has now grown very quickly in size and complexity. It has become very hard to monitor the status of every individual cog in this complex machinery.Some of the common issues that we encountered in due course of the research include:

- Intermittent Pipeline down times

- Failure to restart on reboots
- Running out of Disk Space
- Database query hangup
- Status Check on the Tweet Counts

Therefore to improve the overall stability and to reduce the burden of monitoring each component we wanted to integrate a continuous system monitoring daemon. This watchdog should be capable of monitoring all components and their dependents. It should also be capable of taking well-defined actions in case the running state of a process/component changes. We integrated M/Monit, an open-source Unix utility for managing and monitoring POSIX systems into our pipeline. Monit performs proactive maintenance and is capable of taking meaningful causal actions on failure. Some of these measures include restarting or terminating a process, periodically running clean up and batch scripts, attempting process consistency check and failure reporting in case of non-resolvable process states.

This monitor dashboard gives a pretty detailed information about the current status of the pipeline. We configured the monit system's configuration file to include specific scripts which informed the processes that Monit needed to monitor and their causal actions. Therefore by piggybacking on Monit's recovery mechanism, we were able to fix some of the stability issues and improve overall availability.

Chapter 4

PIRC-Net Dashboard: A Real-time visual analytics system

With all the previous chapters describing the aspects of the infrastructure that is responsible for preparing and processing data, we now discuss the presentation facet. The PIRC-Net dashboard is the lone outward facing entity of the entire infrastructure. It is envisioned to be used actively by the HIV researchers and others involved in health care. This dashboard is therefore intended to provide the user with capabilities of performing exploratory data analysis. Because the datasets that power it was obtained from multiple sources, this dashboard should, therefore, help researchers have discussions along different exploration paths by showing different dimensions involved. With these ideologies and the four top requirements elicited in chapter 2, we decided to adopt a visualization technique called Coordinated and Multiple Views [10]. This section elaborates on the design decision, the different coordinated views used and their purpose.

4.1 Coordinated and Multiple Views

The main aim of Coordinated and Multiple views(CMVs) is to allow the user to think about the data from creative perspectives. It provides a highly interactive environment, which the user uses to hypothesize on new possibilities and outcomes from the underlying data while interacting with it. As it is the case most, CMVs are used to allow the user to discover, identify and compare different results with the intention to answer certain questions. This process closely matches the outcomes of the 'Requirement Elicitation' phase, that requires us to enable the users to identify, discover and compare specific characteristics turning us towards CMVs as a first choice visualization technique.

One major challenge with CMVs in our application is that the data we have is vast. Amongst all the stages involved in creating a CMV design data preparation phase occupies most of the time. Therefore it is necessary to handle aggregation of data and mine some of the data beforehand to make the system responsive. Linear search algorithms that apply well with small sets perform badly on larger datasets, and therefore it is required to replace them with better algorithms for efficient rendering.

CMVs also allow the user to directly manipulate or filter elements from the presented visualization, through the technique of linking the brushing. That permits simultaneous display of selected visual components in another connected window or grid. With brushing the user can change the style of the brush or the

area that the brush effects or what happens to the elements when they are selected. Thus we decided to use the approach of linking and brushing elements in multiple coordinated canvases as our primary choice of visual.

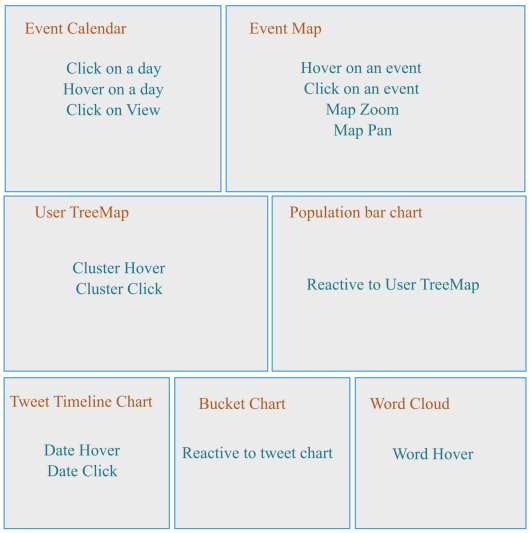

Figure 4.1: PIRC-Net Dashboard: User Interactions, The overall views and their corresponding user interactions

4.2 Visual Elements and User Interactions

The PIRC-Net dashboard aims to provide the clinicians with a seamless interface that enables them to explore the data generated by our infrastructure. The dashboard wears a grid look giving the user incremental information on interactions. The dashboard design has seven different views as follows:1)**Event Calendar**, 2) **Event Geo Map**, 3)**User TreeMap**, 4)**Population bar chart**, 5)**Tweet Timeline Chart**, 6)**Bucket Chart**, 7)**Word Cloud**. We discuss the views in combinations such that we have linked views described together.

4.2.1 Event Calendar and Event Geo Map

The Event Calendar gives a calendar view of the extracted future events from the event data sources(twitter feed and Eventbrite). The events are all given a score based on the ranking algorithm described in Chapter 3. A corresponding color value is used to denote the significance of each event. The user can interact using the mouse by clicking or hovering on a specific date and get the specific events of that day. This interaction is linked to the Event Geo Map view, which brushes on a Mercator map the geographical locations of these specific events. The map also shows a heat distribution of the tweet trends over a selected time frame giving the user a better grasp of three dimensions event's time, location and similar risk activity.

4.2.2 User TreeMap and Population bar chart

The User TreeMap is a cluster view of the Twitter users derived as a result of our batch clustering component. We determine each group's size by the number of connected individuals that represent the cluster information. The bar chart view renders the information about the social distance as described in chapter 3 and the demographic data. These two illustrations help the clinicians to get a sense of the demographical spread of risky users. As an enhancement, the TreeMap can also be modeled to represent a social network by employing an edge node graph. These will allow clinicians to view relationships between Twitter users and their

respective locations and the figure gives a visual example of the look and feel of the dashboard prototype.

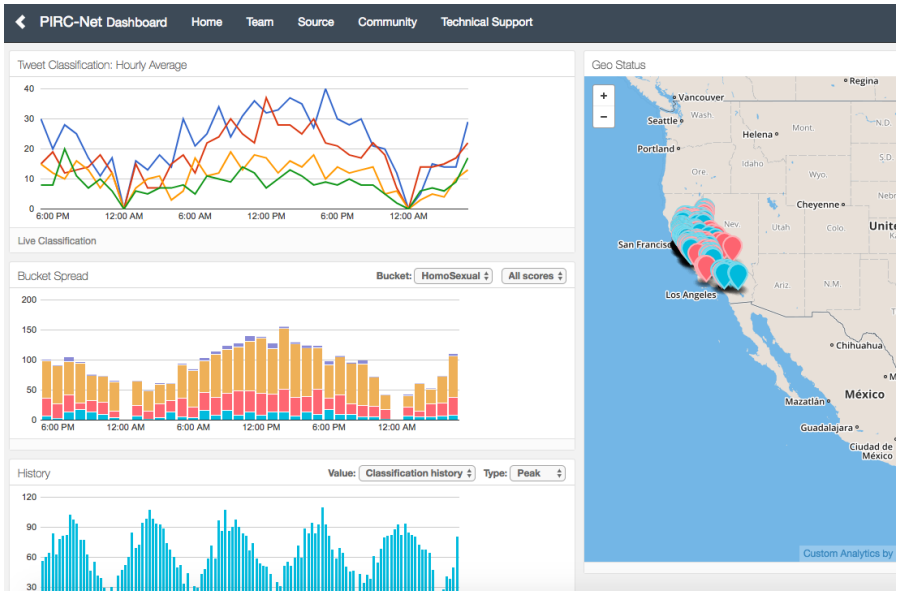

Figure 4.2: PIRC-Net Dashboard Prototype

4.2.3 Tweet, Bucket Charts & Word Cloud

The tweet timeline chart is a bar graph that reveals the daily trend in the at-risk tweets density in a given span of time. This view will enable the clinicians to understand the specific days of peak risky behavior. The bucket charts and the word cloud are linked brushes to the tweet timeline, which get updated on selecting a specific day in the diagram. They help us explore information about which bucket do most of these risk words fall under and the commonly used words that are present in such tweets.

Chapter 5

Benchmarks

This chapter quantifies the capacity of the various components of PIRC-Net infrastructure through some specific benchmarks. It comprises of workloads that were specifically designed to test out the key performance metrics of the system. We start with an initial description of the benchmarking platform, followed by sections that elaborate on the system specific parameters. Additionally, we also discuss the trade-offs that were considered to achieve maximum efficiency.

5.1 Benchmarking Platform

5.1.1 Hardware Configuration

The hardware configuration of the system is detailed in the table. The underlying CPU is an Intel Xeon dual processor multi-core CPU which consists of 32 cores in total. The number of CPU cores is significant to our discussion as our Apache Spark instance is deployed in a single node cluster. The CPU speed clocks reach a maximum 2.6GHz. The main memory consists of two main memory

Table 5.1: Hardware Configurations

CPU(s)	• Cores: 32 • 2.6GHz • Model: Intel(R) Xeon(R) CPU E5-2640 v3
Main Memory	• Total: 128GiB • Slots: 2 • Banks/slot: 4 • Per bank: 16GiB • Width: 64bits
Cache	• L1: 512KiB • L2: 2MiB • L3: 20MiB
Hard Disk	• Size: 223GiB • Buffered Reads: 276.49MiB/s • Writes: 1.6GiB/s

chips with each of the individual chip containing four blocks of 16GiB of memory. The L1, L2, L3 cache have had a read-write unified capability. The hard disk has 223GiB of storage space. The read and write speeds of the hard disk were

measured by using the benchmarking workloads Matthews et al [11] also listed down after evaluating them with simulated bulk read and writes.

5.1.2 Software Configuration

Table 5.2: Software Configurations

Type	**Value**	**Version**
Platform	Linux-Kernel	3.13.0-32
Parallelization	Apache Spark	2.1.1
Persistence	Mongodb, Neo4J	3.0.14, 2.2.1
Language	Python	2.7
Monitoring	Monit	5.6

We performed these experiments on a server running on a Ubuntu distribution of POSIX Linux. The table lists the other components that are critical to the framework's functioning. It is advisable to use the same set version of the libraries and platforms to get consistent results and backward compatibility.

5.2 Performance and Throughputs

5.2.1 Classification Throughput

The classification throughput is the total number of tweets completely processed by the live tweet classifier component for every second. The stages starting from matching specific keywords in the tweets to inserting the tweet into any one of the collections in the database. Throughput is a critical measure of the efficiency

of the pipeline as this determines the rate at which the system can handle the incoming tweets. We measured the time taken to classify the tweets as compared with the number of instance of the classifier nodes. Figure 5.1 shows the results.

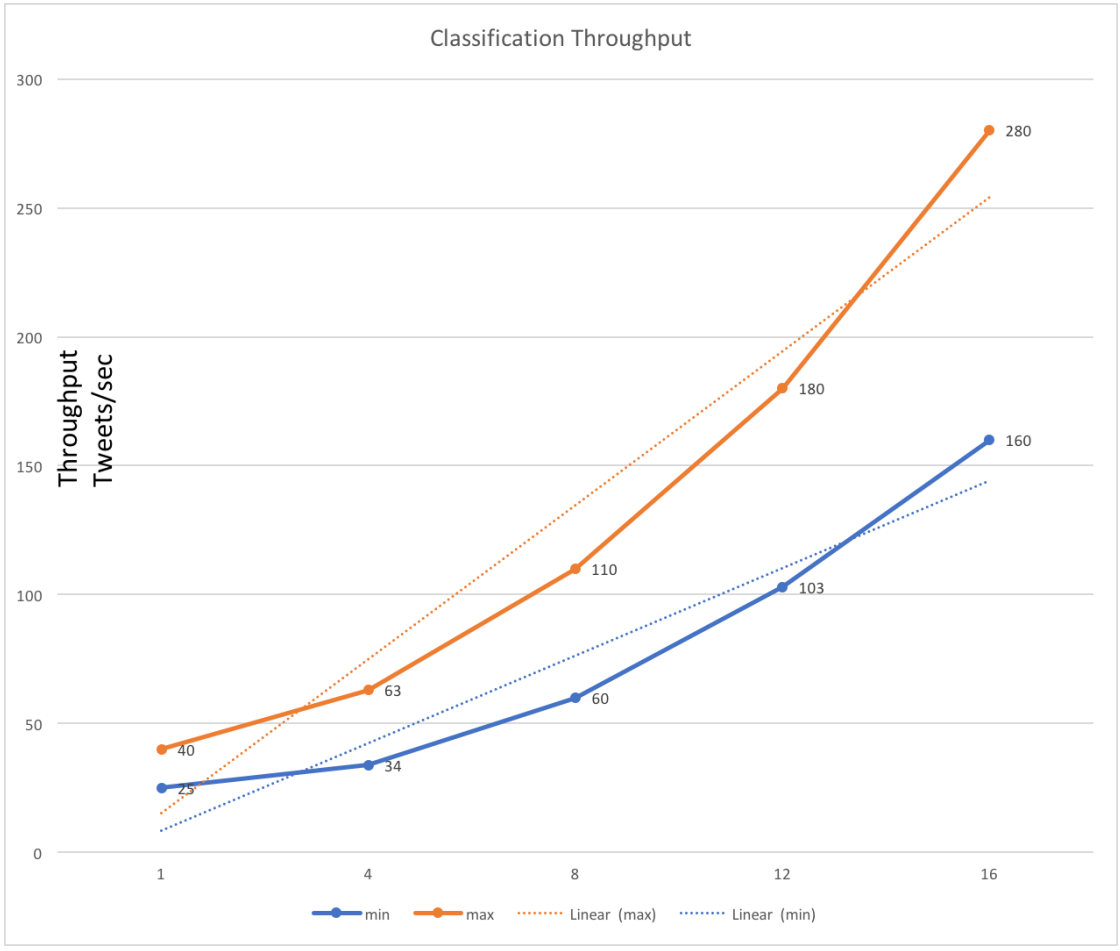

Figure 5.1: Classification Throughput

5.2.2 Keyword Matching Performance

The keywords in our vocabulary set are currently minimal. With the introduction of the vocabulary adaptation back-end support, it is now possible to update

this collection and improve its relevancy. We can now include the frequently used specific web lingos and slang words, in the recent times directly into our group of keywords. There is a potential problem with this approach. The constant increase in the total number of words in our vocabulary set leads to an exponential growth in the running time of the keyword matching algorithm. Therefore the number of words in the set should be delimited by an upper bound on the total number of words used. An expectation on the response time of the Classifier mandates this upper bound. The maximum limit is then used to restrict the number of words inserted into the keyword match vocabulary set.

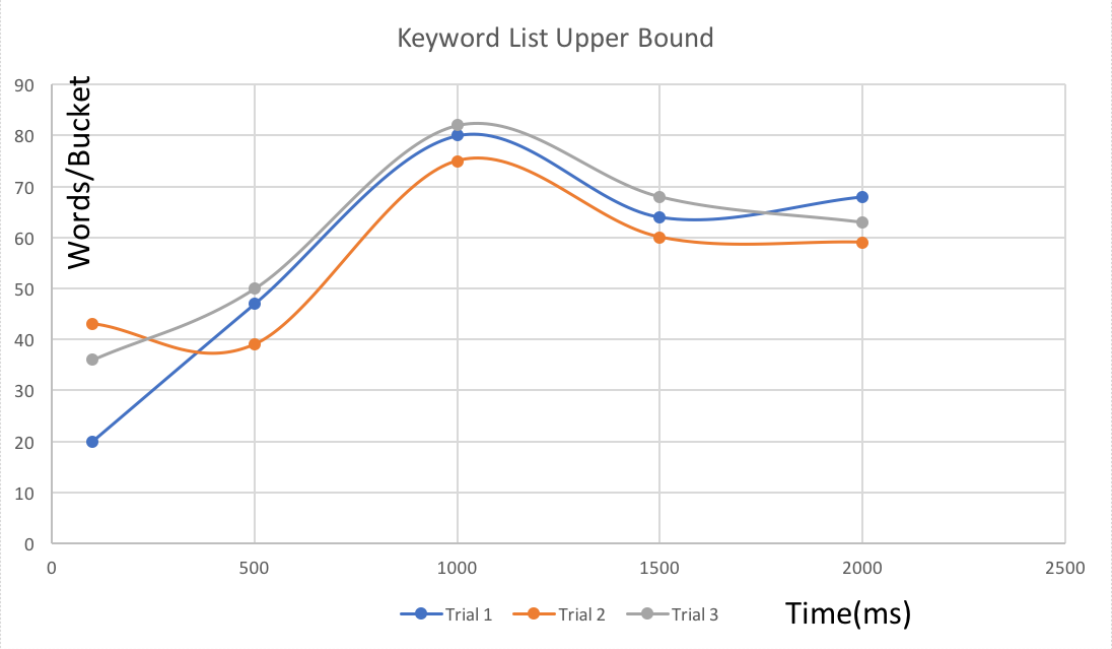

Figure 5.2: Keyword Upper Limit Estimation

5.2.3 Peak Load Provisioning

The tweet density is a measure of the total number of incoming tweets that we receive from the twitter stream. As with all classical network traffic, tweet traffic is also erratic, and therefore the setup needs to be provisioned to handle any sudden spike in the input traffic. Each tweet measures approximately 1KiB in size and therefore in a machine with virtual Memory of 2GiB, there is enough space in the primary memory to cache the incoming tweets before processing them. But, this does not give us a deterministic bound on the number of tweets that can be analyzed.

Therefore we modified our design to listen for 'stall_warnings' which is a flag that will be set if our classifier end point is falling behind the streaming server. When we receive a stall warning is at the endpoint, we initialize a Spark Streaming Context, with an additional set of nodes and handover the tweet collection to be handled by spark instance. This scenario was tested out by generating repeated callbacks to 'on_data' which mimics the tweet being delivered to our load balancer and adding the stall_warning flag.

Chapter 6

Conclusion

PIRC-Net and the research surrounding it has come a long way. It has evolved from a proof of concept to a tool that can now aid to clinicians in planning out evidence-based interventions measures. In our journey to achieve the four goals that we put forth at the beginning of this thesis, we were also able to learn many ways to improve our system and many new perspectives and avenues for such a system. However, progress with the research is far from over. In fact, the understanding we gain of our system's capabilities made us realize that there are still many opportunities to improve our current design and make progress in other avenues unearthed when we gained a better understanding of our system's capabilities. In this final chapter, We present a critical evaluation discussion and suggestions for future work.

6.1 Evaluation Discussion

There are many features offered by the PIRC-Net infrastructure, but there are also a few areas of lapses. The benchmarking results show that the pipeline

performs fairly well for a system that works with such excessive workloads. However, having been developed so that it can take advantage of running in an Apache Spark environment, the pipeline can scale to effectively run in a cluster of nodes instead of running on a single node emulation. Our research work contribution if used by clinicians will become the first system to be adopted in production and we would become the reference system for other newer systems to be evaluated.

These characteristics of our PIRC-Net infrastructure makes it a viable candidate to be adopted widely by the developer and healthcare community. Our pipeline along with the dashboard, can now help clinicians in on-demand decision making for the intervention schemes. The live tweet classifier ensures that real time information is obtained and assorted. The danger though, is that while the hypothesis can be explored in real time, the intervention will have to wait until efforts materialize and outreach programs are planned. These external factors that influence the success of the system and that which need to be addressed for ensuring the success of its adoption.

Additionally, since there is some time difference between the moment when a tweet is identified as risk and when the corresponding user information is retrieved, the data presented in the visual dashboard may remain stale for a small interval of time depending on the system's configuration. This time interval can be improved to become unnoticeably small by running the clustering batch scripts at a greater frequency. We should also be aware of the load that this can cause on the system and hence the decision is tradeoff between consistency and efficiency.

We also see that the classifier currently is limited in effectiveness due do, 1) Insufficient labels for supervised training, 2) Sparseness in connection matrix. The activity of labeling the tweets can only be done by domain experts and that is partly the reason for the lower quantity of labeled tweets. This in turn affects the correctness of the classifier and also the confidence interval. We would partly overcome this problem when the dashboard will integrates the envisioned UI component to let researchers and patients tag tweets as risky and non-risky. Currently though, the system is not yet in place.

Furthermore, the components have a capability to be upgraded and modified without causing disruption to the entire workflow. This an added advantage that will help in the future improvement of specific parts of the infrastructure individually. In summary, the modular design and performance of PIRC-Net offer a great load of features, and promise for the future and to be able to address some of the shortcomings described above.

6.2 Future Work

We foresee the PIRC-Net infrastructure to enable effective intervention and real time response system. Beside the HIV Use Case that drove this work, for health care can now be built upon our system. We also envision that the health care professionals will use the infrastructure for any specific diease of interest which will in turn help us to evaluate the effectiveness of the entire system in production.

As indicated in the previous section, we still believe that there is still potential for novel research from the current state of the system. By consolidating our experiences, we have identified specific problem statements in scope for both Research and Development with PIRC-Net.

Firstly, PIRC-Net can be leveraged and used in scalable information systems. The specific use case can vary but at the crux of it the principle of data collection from social network and processing is going to be similar. One example of such a research system is the envisioned Bionet Studio system that researcher at UC San Diego are exploring to enable a common conduit for network analysis. Given, that PIRC-Net provides a capability of exploratory analysis, Scalability and interoperability meets the requirements of such a system.

Finally, we need to define a way to measure the success of the intervention system. One suggested way to measure validation of PIRC-Net results is to compare it with the real world patient data. There are two approaches in progressing along this path. The first is to enroll more HIV at-risk patients in our study and using the enrollment cohort to validate our system. The other is by measuring ratio of negative testers, which is the ratio of number of patients tested negative to the overall number of patients tested in a given period of time.

In conclusion, we believe that our contribution will provide a building block to detect and mitigate the spread of HIV. The dashboard will let HIV researchers look at the data from different perspectives and inturn take appropriate actions. By detecting relevant information about at-risk communities, We believ this real

time visualization system will add to the repertoire of tools used for HIV combat and that our work will certainly be an additional step toward effective targeted interventions.

Appendix A

HIV Vocabulary

Table A.1: Vocabulary Set

Category	**Words**
Drug	methamphetamine, meth, ice, speed, cocaine, coke, crystal, crank, blow, tina, crack, tweak, tweaker, dope, glass, flailing, speeding, booty bump, booty bumping, plug, plugging, butt rocket, bumping
HomeSexual	homo, queer, gay, homosexual, queen, tink, bear, chicken, chicken hawk, donald duck, girl scout, grimm's fairy, smurf, vampire, wolf
Sex	bareback, barebacking, creampie, raw dogging, basket shopping, cottaging, lucky pierre on the make, ring snatcher, pratt, rumpy-rumpy, yard boy, zipper club, bottom, top, bronco, brown, brownie queen, buggery, bummery, candy maker, chicken dinner, circle jerk, corn-hole, cottaging, daisy chain, glory hole, greek, pearl diver, flipping
SexVenues	aqua day spa, vulcan sauna, the hole, t room, tea room, groovy laidback, baja betty's, brass rail, bourbon street, caliph lounge, cheers, flicks, the loft, ma4, numbers, pecs, redwing, rich's, eagle, sro, urban mo's
STI	full house, chlamydia, clap, gonorrhea, syphilis, syf, sif, syphy, syph, pox, bad blood, vd, dose, gleet, morning drip, running rage, drip, white, racehorse, the clam, clam chowder, gooey stuff

Bibliography

[1] M. Salath, L. Bengtsson, T. J. Bodnar, D. D. Brewer, J. S. Brownstein, C. Buckee, E. M. Campbell, C. Cattuto, S. Khandelwal, P. L. Mabry, and A. Vespignani, "Digital epidemiology," *PLOS Computational Biology*, vol. 8, pp. 1–3, 07 2012.

[2] "Centres for disease control & prevention," Accessed: 2016-03-23.

[3] S. D. Young, C. Rivers, and B. Lewis, "Methods of using real-time social media technologies for detection and remote monitoring of hiv outcomes," pp. 112–115, June 2014.

[4] X. Hu, L. Tang, J. Tang, and H. Liu, "Exploiting social relations for sentiment analysis in microblogging," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, (New York, NY, USA), pp. 537–546, ACM, 2013.

[5] I. C.-H. Fung, K.-W. Fu, C.-H. Chan, B. S. B. Chan, C.-N. Cheung, T. Abraham, and Z. T. H. Tse, "Social media's initial reaction to information and misinformation on ebola, august 2014: Facts and rumors," *Public Health Reports*, vol. 131, no. 3, pp. 461–473, 2016. PMID: 27252566.

[6] "Twitter developer documentation, https://dev.twitter.com/streaming/overview,"

[7] "Mongodb atlas database as a service, https://docs.mongodb.com/,"

[8] Y. Shin, J. H. Hayes, and J. Cleland-Huang, "Guidelines for benchmarking automated software traceability techniques," in *2015 IEEE/ACM 8th International Symposium on Software and Systems Traceability*, pp. 61–67, May 2015.

[9] M. Sedlmair, M. Meyer, and T. Munzner, "Design study methodology: Reflections from the trenches and the stacks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, pp. 2431–2440, Dec. 2012.

[10] J. C. Roberts, "State of the art: Coordinated multiple views in exploratory visualization," in *Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV 2007)*, pp. 61–71, July 2007.

[11] C. Matthews, Y. Coady, and S. Neville, "Quantifying artifacts of virtualization: A framework for mirco-benchmarks," in *2009 International Conference on Advanced Information Networking and Applications Workshops*, pp. 1079–1084, May 2009.

[12] J. Fogarty, R. S. Baker, and S. E. Hudson, "Case studies in the use of roc curve analysis for sensor-based estimates in human computer interaction," in *Proceedings of Graphics Interface 2005*, GI '05, (School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada), pp. 129–136, Canadian Human-Computer Communications Society, 2005.

[13] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, and P. Li, "User-level sentiment analysis incorporating social networks," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, (New York, NY, USA), pp. 1397–1405, ACM, 2011.

[14] N. Weibel, P. Desai, L. Saul, A. Gupta, and S. Little, "Hiv risk on twitter: the ethical dimension of social media evidence-based prevention for vulnerable populations," in *HICSS*, 2017.

[15] J. M. B. Josko and J. E. Ferreira, "Visualization properties for data quality visual assessment: An exploratory case study," *Information Visualization*, vol. 16, pp. 93–112, 2017.

[16] S. Wan and C. Paris, "Understanding public emotional reactions on twitter," 2015.

[17] K. Roberts, M. A. Roach, J. Johnson, J. Guthrie, and S. M. Harabagiu, "Empatweet: Annotating and detecting emotions on twitter," in *LREC*, 2012.

[18] A. Abbasi, A. Hassan, and M. Dhar, "Benchmarking twitter sentiment analysis tools," in *LREC*, 2014.

[19] Y. Shin, J. H. Hayes, and J. Cleland-Huang, "Guidelines for benchmarking automated software traceability techniques," in *2015 IEEE/ACM 8th International Symposium on Software and Systems Traceability*, pp. 61–67, May 2015.

[20] N. Thangarajan, N. Green, A. Gupta, S. Little, and N. Weibel, "Analyzing social media to characterize local hiv at-risk populations," in *Proceedings of the Conference on Wireless Health*, WH '15, (New York, NY, USA), pp. 11:1–11:8, ACM, 2015.