LITMUS: Landslide Detection by Integrating Multiple Sources

Aibek Musaev
Georgia Institute of Technology
aibek.musaev@gatech.edu

De Wang
Georgia Institute of Technology
wang6@gatech.edu

Calton Pu
Georgia Institute of Technology
calton.pu@cc.gatech.edu

ABSTRACT

Disasters often lead to other kinds of disasters, forming multi-hazards such as landslides, which may be caused by earthquakes, rainfalls, water erosion, among other reasons. Effective detection and management of multi-hazards cannot rely only on one information source. In this paper, we evaluate a landslide detection system LITMUS, which combines multiple physical sensors and social media to handle the inherent varied origins and composition of multi-hazards. LITMUS integrates near real-time data from USGS seismic network, NASA TRMM rainfall network, Twitter, YouTube, and Instagram. The landslide detection process consists of several stages of social media filtering and integration with physical sensor data, with a final ranking of relevance by integrated signal strength. Applying LITMUS to data collected in October 2013, we analyzed and filtered 34.5k tweets, 2.5k video descriptions and 1.6k image captions containing landslide keywords followed by integration with physical sources based on a Bayesian model strategy. It resulted in detection of all 11 landslides reported by USGS and 31 more landslides unreported by USGS. An illustrative example is provided to demonstrate how LITMUS' functionality can be used to determine landslides related to the recent Typhoon Haiyan.

Keywords

LITMUS, landslide detection, multi-source integration, social sensors, physical sensors.

INTRODUCTION

Natural disaster detection and management is a significant and non-trivial problem, which has been studied by many researchers. A conventional approach relies on dedicated physical sensors to detect specific disasters, e.g., using real-time seismometer data for post-earthquake emergency response and early warning by Kanamori (2005). A more recent approach explores the big data from social networks such as Twitter functioning as social sensors, e.g. in the work by Sakaki, Okazaki, and Matsuo (2010). Since physical sensors (e.g., seismometers) are specialized for specific disasters, people have placed high expectations on social sensors. Besides, few physical sensors exist for the detection of multi-hazards such as landslides, which have multiple causes (earthquakes and rainstorms, among others) and happen in a chain of events. However, despite some initial successes, social sensors have met serious limitations due to the big noise in the big data generated by social sensors. For example, Twitter filter for the word "landslide" gets more tweets on the 70's rock song "Landslide"[1] than landslide disasters that involve soil movement.

In this paper, we describe and evaluate a landslide detection system LITMUS, which is based on a multi-source integration approach to the detection of landslides, a representative multi-hazard. LITMUS integrates information from a variety of sensor sources instead of trying to refine the precision and accuracy of event detection in each source. Our sources include both physical sensors (e.g., seismometers for earthquakes and weather satellites for rainstorms) and social sensors (e.g., Twitter and YouTube). Although we still have some technical difficulties with filtering out noise from each social sensor source, LITMUS performs a series of filtering steps for each social sensor, and then adopts geo-tagging to integrate the reported events from all physical and social sensors that refer to the same geo-location. Our evaluation shows that with such integration the system achieves better precision and F-measure in landslide detection when compared to individual social or

[1] http://en.wikipedia.org/wiki/Landslide_(song)

physical sensors.

This paper makes several contributions. The first contribution is the construction of a landslide detection system LITMUS that integrates online feeds from five sources. Two of sources are physical sensors: seismic activity feed provided by USGS and rainfall activity feed provided by NASA TRMM. Three sources are social sensors: Twitter for text information, Instagram for photos, and YouTube for videos. We believe the combination of these relatively independent sources of data enables LITMUS to improve the precision and accuracy of landslide detection. The second contribution is a quantitative evaluation of the system using real world data collected in October 2013. LITMUS detected all 11 landslides reported by USGS as well as 31 more landslides unreported by USGS during this period. The final contribution is an illustrative example of the functionality of the system to determine a list of landslides caused by the recent Typhoon Haiyan, which devastated the Philippines on November 8th.

OVERVIEW OF APPROACH

For better understanding of our landslide detection system LITMUS, we present an overview of the system's data flow in Figure 1.

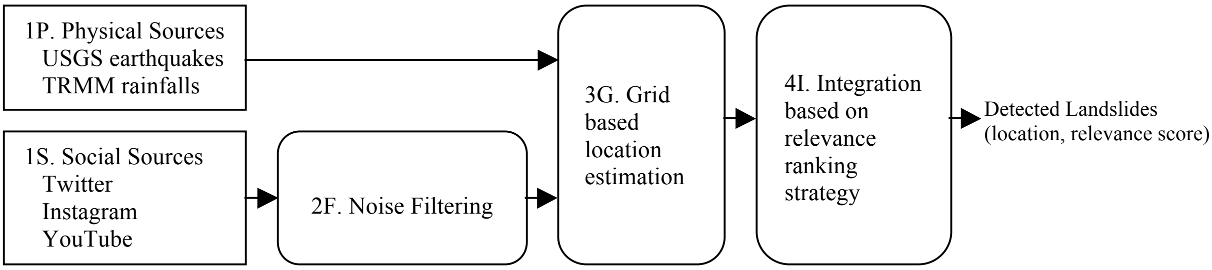

Figure 1. Overview of Data Flow

The system starts with the raw data collection. It periodically downloads data from multiple social and physical sensors. The social sensors supported by LITMUS are popular social network sites, namely Twitter, YouTube, and Instagram. Each of these sensors is among the leading social networks in their respective areas. LITMUS extracts the data from these sensors by applying a search filter based on landslide related keywords. We perform noise filtering in a series of filtering steps, including filtering out items based on stop words and stop phrases, filtering items with accurate geo-tags based on the geo-tagging component, filtering relevant items based on the machine learning classification component, and filtering out items based on a blacklist of URLs – see Figure 2, where "+" indicates an inclusion type of filtering and "-" indicates an exclusion type of filtering. LITMUS also collects data from the physical sensors, namely the seismic activity and the rainfall activity feeds. We support them in our system because these feeds are related to hazards that may cause landslides.

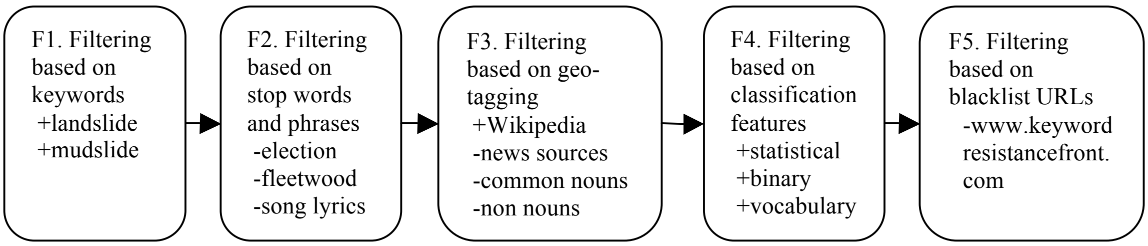

Figure 2. Noise Filtering Steps

In the end, we combine the remaining items from the social sensors with all items from the physical sensors based on the relevance ranking integration strategy. The final output of the system is a list of detected landslides with location information and relevance scores.

1P. Physical Sources Support

LITMUS collects data from several physical sensors. In particular, LITMUS supports a real-time seismic activity feed from the United States Geological Survey (USGS) agency[2]. This feed is updated every minute

[2] http://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php

providing information about earthquakes of various magnitudes. LITMUS downloads the data from USGS on earthquakes of 2.5 magnitude and higher. USGS provides programmatic access via a well-structured GeoJSON format that can be conveniently parsed. It provides time, magnitude, latitude, longitude, name of the place where an earthquake occurred and an event ID.

Another potential cause of landslides is rainfalls, which is why LITMUS also collects data from the Tropical Rainfall Measuring Mission (TRMM) project (http://trmm.gsfc.nasa.gov/). It is a joint project between NASA and the Japan Aerospace Exploration Agency (JAXA), which generates reports based on the satellite data of the areas on the planet that have experienced rainfalls within the past one, three and seven days. The reports are in multiple formats, including a web page on the project's portal, from which LITMUS periodically downloads and parses data about rainfalls.

1S. Social Sources Support

F1. Filtering based on keywords

According to the research on citizen activity done by Palen, Anderson, Mark, Martin, Sicker, Palmer, and Grunwald (2010), social networks have emerged as destinations for collective disaster-related sensemaking. LITMUS uses the data from social networks to help detect landslides as reported by the public. In particular, LITMUS downloads the data from Twitter as an example of a text based social network, YouTube as an example of a video based social network and Instagram as an example of an image based social network. All listed social networks provide programmatic access to their data via search API based on keywords. LITMUS downloads the data from each social network based on "landslide" and "mudslide" keywords.

F2. Filtering out based on stop words and stop phrases

Next, LITMUS performs filtering by excluding social sensor items that contain negative stop words with respect to landslides, such as "fleetwood" or "election". The following is a set of examples from Twitter that represent unrelated to landslides items containing these stop words:

"Landslide by Fleetwood Mac will forever be one of my favorite songs."

*"Abbott builds on election landslide: TONY Abbott is riding a post-*election *honeymoon high, with nearly half of... http://t.co/P17WAyxud2"*

LITMUS also removes items based on stop phrases that currently contain excerpts from the lyrics of some popular songs that are commonly used in social networks, e.g. the lyrics from the "Landslide" song by Stevie Nicks from Fleetwood Mac: *"...and I saw my reflection in the snow covered hills..."*

F3. Filtering based on geo-tagging

After LITMUS downloads the data from the physical and social sensors, we need to obtain geo locations of the downloaded data. The data from the physical sensors already contains geo coordinates. Unfortunately, the data from the social sensors is usually not geo-tagged since few users disclose their locations. Thus, if an item coming from a social source is not geo-tagged, we need to look for geo terms inside the textual description of the item. An important component included in social sensor items is mentions of place names that refer to locations of landslides. An exact match of words in the textual description of an item is performed against the list of all geo terms. For the list of geo terms we use the approach introduced by Hecht et al., (2011) to locate accurate geo coordinates based on the titles of the geo-tagged Wikipedia articles. However, different types of geo coordinates are supported in the geo-tagged Wikipedia articles. Some of them, like "city" or "country", are more relevant than others, such as "landmark", which often returns irrelevant matches like "houses" or "will".

However, the relevance quality of this algorithm should be improved further. For example, some geo terms may appear valid, such as "Says", which was a municipality in Switzerland, or "Goes", which is a city in Netherlands, however they are also verbs that are commonly used in English texts. That is why prior to applying the geo-tagging algorithm on the downloaded social media data, LITMUS performs pre-filtering of the words inside those items using Part-Of-Speech tagging by excluding non-noun words from consideration.

There are also geo terms like "cliff" or "enterprise" whose type is "city" that are not very helpful for the purpose of landslide location estimation. The algorithm would incorrectly retrieve "cliff" as a geo term from the following YouTube item: *"Driver Survives Insane Cliff Side Crash."* The reason why these words are irrelevant is because they happen to be common nouns, in other words they are used in English texts a lot. To mitigate this issue we use a list of 5000 most frequent words in English based on the Corpus of Contemporary American English (http://www.wordfrequency.info) and exclude those results from the list of geo terms.

Among the supported social sensors, YouTube in particular contains a lot of items where in addition to some valuable information related to landslides, they also contain unrelated information. The following is an illustrative example that follows such pattern:

"After fatal Flash Flood, Mudslide, More Rain Possible for Colorado and other states youtube original. news bloopers, fox news,onion news,funny news bloopers, news failbreaking news,bbc news news reporter news fails cbs news cnn news world news us news uk news syria today syria war syria 2013 syria new,syria news,damascu,syria damascus, syrian army,syrian,syria execution..."

It is clear that "Colorado" is a relevant geo term, whereas "Syria" and "Damascus" are not. In order to take into account patterns like this, we augmented the geo-tagging algorithm as follows: the input text is broken into sentences and for each sentence we find the geo term that is the closest to the landslide keyword. In this example the landslide keyword is "mudslide" and the closest available geo term is "Colorado", hence the geo-tagging algorithm correctly outputs "Colorado".

F4. Filtering based on machine learning classification

The social sources in LITMUS frequently return items that are not relevant to landslides, even though they contain landslide keywords. The following is an example of irrelevant items that use "landslide" as an adjective describing an overwhelming majority of votes or victory: *"We did it! Angel won in Starmometer 100 Most Beautiful Women in the Philippines for 2013! Landslide victory due... http://t.co/2g6ozhJhpj"*

To filter out such items from the social sensors LITMUS employs binary classification, a machine learning technique to automatically label each item as either relevant or irrelevant based on classifier model built from a training set containing labeled items. To prepare a training set we need a list of confirmed landslides. For this purpose we use expert landslide publications. The USGS agency, in addition to earthquakes, also publishes a list of landslide events collected from external reputable news sources, such as Washington Post, China Daily, Japan Times and Weather.com (http://landslides.usgs.gov/recent/). For each event in this list we identify the date of release and geo terms.

To find the social network items related to confirmed landslides within each month, we first filtered the data based on the landslide locations extracted from the confirmed landslides. Then we manually went through each item in the filtered list to make sure that they described corresponding landslides by comparing the contents of the items with the corresponding landslide articles. And whenever there were URLs inside those social items, we looked at them also to make sure that they were referring to the corresponding landslides.

The following is an example of a landslide confirmed by the Latin Times news source, which was published on September 11, 2013[3]: *"Mexico Mudslide 2013: 13 Killed in Veracruz Following Heavy Rains."* The geo terms that LITMUS extracted from this news title are "Mexico" and "Veracruz".

To create a list of unrelated items in the training set, we randomly picked items from each social source and manually went through each item. But this time we had to make sure that the items did not describe landslide events.

F5. Filtering based on blacklist URLs

During the analysis of social media items containing URLs, we found out that in several cases the short URLs were expanded into the same web site (http://keywordresistancefront.com) that generated random content with high-value keywords such as "mudslide". Based on this result we created a blacklist of URLs and added a filter to exclude items containing such URLs from consideration.

3G. Grid based location estimation

As a result of the previous stages in the system's data flow shown in Figure 1, LITMUS has a set of relevant and

[3] http://www.latintimes.com/articles/8234/20130911/13-dead-veracruz-mudslides-landslides-mexico-rains.htm#.UjDOD5LFVBk

geo-tagged items from physical and social sensors. Next LITMUS integrates those items by grouping them based on their geo coordinates to determine areas on the planet where landslides might have occurred. For this purpose we propose to represent the surface of the Earth as a grid of cells. Each geo-tagged and relevant to landslide item is mapped to a cell in this grid based on the item's geo coordinates. After all items are mapped to cells in this grid, the items in each non-empty cell are counted per each source. Currently we use a 2.5 minute grid both in latitude and longitude[4], which corresponds to the resolution of the Global Landslide Hazard Distribution[5]. This is the maximum resolution allowed by the system. The actual resolution is driven by the precision of the geo-tagging algorithm described in section F3.

4I. Integration based on relevance ranking strategy

After mapping all items to cells in the grid, we obtain a set of non-empty cells. These cells represent areas on the planet where landslides may have occurred. To tell which cells are more likely to have experienced landslides, we propose a Bayesian model strategy and compare it with two baseline strategies – "OR" and "social AND physical". For "OR" integration strategy, we grant equal weights to all sensors. And we obtain the decision by combining the votes using boolean operation OR among five sensors. For "social AND physical" integration strategy, we use boolean operation OR to combine the votes from social sensors and physical sensors separately first. And then we calculate the combined result by applying boolean operation AND between votes from social and physical sensors. For instance, if the votes from five sources (Twitter, Instagram, YouTube, USGS, and TRMM) are 1,1,0,0, and 0, the "OR" strategy will return 1, but the "social AND physical" strategy will return 0.

The description of the Bayesian model strategy is as follows. Suppose, there is a cell x and ω is the class associated with x, either being true or false. Then, assuming a hidden variable Z for an event to select one source, a probability for a class ω given x, $P(\omega \mid x)$, can be expressed as a marginal probability of a joint probability of Z and ω:

$$P(w|x) = \sum P(w, |Z_i|x) = \sum_i P(w|Z_i, x)P(Z_i|x), \tag{1}$$

where external knowledge $P(Z_i \mid x)$ denotes each source's confidence given x. For instance, if a certain source becomes unavailable, the corresponding $P(Z_i \mid x)$ will be zero. Also one could assign a large probability for the corresponding $P(Z_i \mid x)$ if one source dominates over other sources.

In our experiment, to provide a balance between precision and recall, we use prior F-measure C from the training dataset as the confidence for each source. Keeping the results in the range from 0 to 1, we normalize the values of F-measure into a scale between 0 and 1 first. After taking the number of items N from each source into account, the formula will be further converted into the following format:

$$P(w|x) = \sum C_i \frac{N_i^x}{N_i^x + 1}, \tag{2}$$

where C_i denotes the normalized prior F-measure of source i from historic data (we use August and September data in our experiments). N_i^x denotes the number of items from source i in cell x indicating that a landslide occurred in the area covered by cell x. It should be noted that for Bayesian model strategy we ignore cells with only 1 vote, i.e. where the total count of items in that cell is equal to 1. This is done to reflect the idea of a multi-source integration as opposed to a single source analysis.

IMPLEMENTATION SUMMARY

LITMUS is developed using free and open-source software. It consists of a front-end implemented as a Web application and a back-end, which is the core of the system. The front-end is a live demonstration that runs on Apache web server. It uses Google Maps JavaScript API to render all feeds, including detected landslides, and PHP to access LITMUS' back-end. The back-end is developed in Python, except for binary classification for which we used Weka's library implemented in Java – see Hall, Frank, Holmes, Pfahringer, Reutemann, and Witten (2009). All data from social and physical sensors is stored in MySQL. The data has been collected since August 2013 and takes up 1.7GB on disk. The total number of lines of code is 12k.

[4] http://en.wikipedia.org/wiki/Latitude_and_longitude

[5] http://sedac.ciesin.columbia.edu/data/set/ndh-landslide-hazard-distribution

EXPERIMENTAL EVALUATION

In this section, we present an experimental study using LITMUS. We designed 4 sets of experiments to evaluate its performance. We start by analyzing the effectiveness of the filtering techniques that are employed in social sensors to retrieve landslide relevant items. Next we compare the performance of physical sensors that monitor seismic and rainfall activities as possible causes of landslides. In the third experiment we measure the effectiveness of 3 integration strategies of both social and physical sensors to find the optimal integration strategy. And in the last experiment we compare the overall performance of LITMUS in landslide detection using the chosen strategy versus an authoritative source of landslide events compiled by USGS.

Retrieval of Landslide Relevant Items from Social Media

Overview of filtering results

As we mentioned earlier, LITMUS performs a series of filtering steps on the data from social sources to retrieve landslide relevant items. Table 1 contains the results of the filtering steps on the data collected during the evaluation period, which is the month of October in 2013. It shows that Twitter has the most number of items and that the geo-tagging component filters out most of items.

Filtering steps / Sources	**F1. Filter based on keywords**	**F2. Filter based on stop words & phrases**	**F3. Filter based on geo-tagging**	**F4. Filter based on classification**	**F5. Filter based on blacklist URLs**	**4I. Integration based on relevance ranking strategy**
Twitter	34508	24898	6107	4630	4624	3861
Instagram	1631	1403	178	13	13	8
YouTube	2534	2221	331	182	182	105

Table 1. Overview of filtering results

Features used in classification

The filtering step F4 employs SVM, which is an algorithm for training a support vector classifier. The training dataset needed by the algorithm consists of social source items in August and September 2013, including 12,328 tweets, 1,266 Instagram images and 3,174 YouTube videos. In classification, we extracted a set of features based on the textual description of items from each social source. In particular, we created 3 groups of features that are applied to each source:

1. Common statistical features: 1) length of the textual description, 2) number of uppercase characters, 3) position of the query term in the textual description divided by number of words, 4) number of lowercase characters, 5) total number of words, 6) maximum word length, 7) minimum word length, 8) average word length, 9) code of the most common character.
2. Binary features: presence of the following elements – 1) at sign, 2) URL, 3) percentage, 4) geo-term, 5) number, 6) hashtag, 7) exclamation mark, 8) question mark, 9) ellipsis, 10) double quotes, 11) colon, 12) '♥' symbol, 13) '♫' symbol.
3. Vocabulary based features: 1) relevant vocabulary score, 2) irrelevant vocabulary score. For these features we collect the lists of words (or vocabularies) based on the training set, which contains items labeled as relevant and irrelevant. For each downloaded item we compute the total count of words that are present in the relevant vocabulary list, which we call a relevant vocabulary score, and also the total count of words that are present in the irrelevant vocabulary list, which we call an irrelevant vocabulary score.

The following is an example Tweet with the corresponding feature values below: *"Philippines - Travel News - Death toll reaches 32 following monsoon rains, landslides and flooding #Philippines #travel #safety #flooding"*

1. 1) 137, 2) 5, 3) 0.588, 4) 132, 5) 17, 6) 11, 7) 2, 8) 6.588, 9) 32.
2. 1) False, 2) False, 3) False, 4) True, 5) True, 6) True, 7) False, 8) False, 9) False, 10) False, 11) False, 12) False, 13) False.
3. 1) 0.185, 2) 0.099.

Performance of social sensors

To evaluate the performance of the social sensors we have used several criteria that are standard in the area of information retrieval, namely precision, recall and F-measure. Precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. F-measure considers both precision and recall and is the harmonic mean of precision and recall.

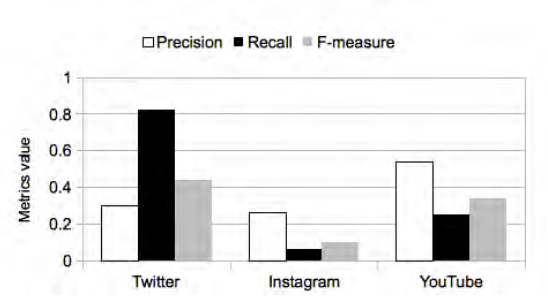

Figure 3. Landslide relevance of social sources

Let us consider the relevance of the social sensors with respect to landslide disaster events based on these criteria. According to the results shown in Figure 3, Twitter has the highest recall as it has the most number of items among all sensors, whereas YouTube has the highest precision. Instagram showed the worst results among these sensors, as most of its images were unrelated to landslide events. Overall, Twitter has the highest F-measure in spite of its low precision, so any improvements in its precision should increase its F-measure even more.

Analysis of Physical Sensors

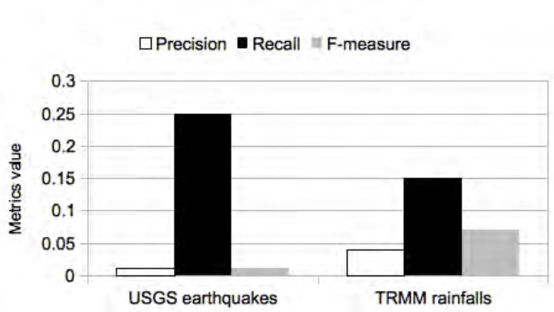

Figure 4. Landslide relevance of physical sources

For our next experiment we compare the relevance of the physical sources with respect to landslide disaster events, namely the seismic and rainfall activities – see Figure 4. LITMUS collected 6,036 seismic activity points provided by USGS and 723 rainfall observations provided by TRMM in October. Due to such gap in the sheer volume of data, the seismic activity sensor shows better recall, but both precision and F-measure are better for rainfalls, which means that in October the influence of rainfalls on landslides was relatively higher.

Multi-Source Integration Strategies

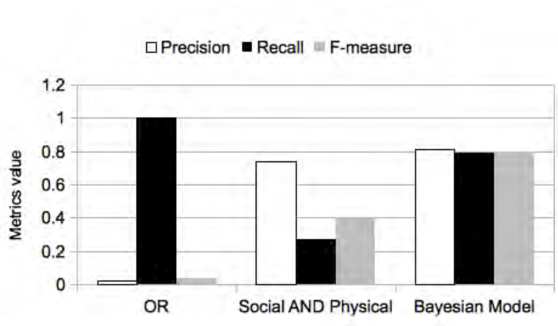

Figure 5. Landslide detection performance of integration strategies

In this experiment we compare the performance of the following relevance ranking strategies with respect to landslide detection: Bayesian model strategy versus two baselines – "OR" and "social AND physical" integration strategies shown in Figure 5. The "OR" strategy expectedly has the highest recall, because it includes all votes from each sensor in its decisions, which is also the reason why it has the lowest precision among all integration strategies. "Social AND physical" strategy produces a much better precision and F-measure, but very low recall. And the Bayesian model produces the best precision and F-measure results and an acceptable value of recall, which is why we select it as the best strategy for landslide detection among these strategies.

System Performance Results

LITMUS scripts run periodically where a period is customizable and currently set to 30 minutes. During each period, LITMUS performs a series of filtering steps F1 through F5 followed by integration step 4I. For each step we provide Latency and Throughput metrics to evaluate the system performance shown in Table 2.

Filtering steps / Metrics	**F1. Filter based on keywords**	**F2. Filter based on stop words & phrases**	**F3. Filter based on geo-tagging**	**F4. Filter based on classifi-cation**	**F5. Filter based on blacklist URLs**	**4I. Integration based on relevance ranking strategy**
Latency (s)	1318.1	13.3	218.2	60.5	670.8	13.7
Throughput (items/s)	11.0	1090.2	35.8	37.9	1.2	462.5

Table 2. Overview of system performance results for a period from 2013-12-12 to 2013-12-19. F5 has the lowest throughput due to the cost of short URL expansion. F1 has low throughput due to pagination and extra delays when downloading the data. F3 and F4 also have low throughput due to the costs of geo-tag search and classification model generation.

Latency is a time interval between the beginning and end of each processing step and Throughput equals the total number of items processed at each step divided by the amount of time to process them.

Landslide Detection Results

LITMUS detected 42 landslide events in October. Of these 42, 11 were reported by USGS – see here[6]. In addition, LITMUS detected 31 landslides not reported by USGS. For each landslide we have performed manual verification by finding other reputable sources that would confirm the detected landslide events. We made sure that both locations and dates of the events were confirmed.

This is an example tweet regarding a landslide event that occurred in Obudu Resort, Nigeria in October, which was not reported by USGS: "*Over 20 People Trapped In Obudu Resort Mudslide http://t.co/aaOUn5465m*" (posted 10/17/2013). The tweet contains a shortened URL that points to a news article on Channels Television website[7], which confirms the location and the date of the landslide event.

It should be noted that 1 event reported by USGS was not an actual disaster report, namely: *"Flash Floods and Debris Flows: How to Manage Nature's Runaway Freight Trains"* (posted 10/30/2013). LITMUS successfully did not detect this report as a landslide event. All of the remaining reported events were successfully detected by the system.

LIVE DEMONSTRATION

We developed a live demonstration of the landslide detection system LITMUS as part of the GRAIT-DM project's web portal (https://grait-dm.gatech.edu/demo-multi-source-integration/). The web portal demonstrates multiple functionalities supported by LITMUS, including live feeds from each social and physical sensor, a separate feed of landslides detected by the system, support for viewing detailed information about each feed, and various user options to analyze results further – see Figure 6.

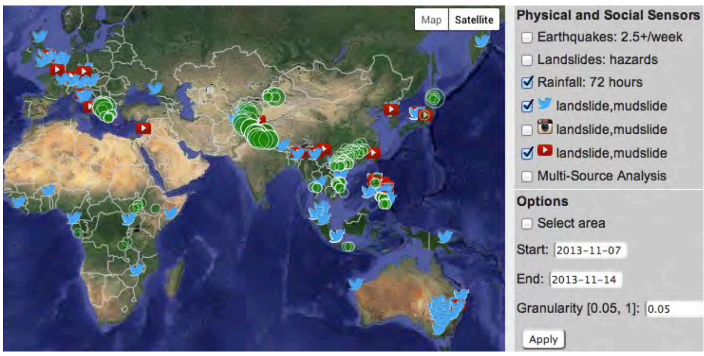

Figure 6. LITMUS live demonstration

[6] http://landslides.usgs.gov/recent/index.php?year=2013&month=Oct

[7] http://www.channelstv.com/home/2013/10/17/over-20-people-trapped-in-obudu-resort-mudslide/?utm_source=dlvr.it&utm_medium=twitter

The data from all feeds is displayed on a Google Map. Each feed can be turned on and off to give a user an ability to view the data from a particular feed or a combination of feeds. Users can also obtain detailed information regarding each feed. For example, if the feed is from the Instagram sensor, then they can view the related images. Similarly, if the feed is from the YouTube sensor then they can view the related videos. Finally, it should be noted that LITMUS has been collecting the data from all sensors since August 2013 only.

Typhoon Haiyan (Yolanda)

As an illustrative example of LITMUS functionality, let us consider the top event identified by the system in November 2013. As of November 14th, the table on the live demonstration page shows that within the last 7 days the cell with the top landslide score is the one for Philippines, which has been devastated by Typhoon Haiyan (known in the Philippines as Typhoon Yolanda) on November 8th. To find out which landslides have been caused by this event during this period we need to use the *Select area* option described above. Using this feature we cover the area of Philippines and recompute the results by applying the changes.

The table now shows 21 locations identified by LITMUS as landslide events in the selected area, including:

Manila: *"100 dead as storm ripped apart buildings and triggered landslides in #Manila http://t.co/6tpLlxyBVy http://t.co/Sw09WyR6KQ"*

Cebu: *"MT Province of Cebu @cebugovph 4m #YolandaPH Another landslide also reported in barangay Buhisan, #Cebu City #hmrd"*

1 result out of 21 was falsely identified as a landslide event, namely:

Antipolo: "*No reported floods, landslides in #Antipolo as of 7:23 p.m. --Dodie Coronado, PIO | @KFMangunay @InqMetro #YolandaPH*"

RELATED WORK

Disaster detection based on social media received a lot of attention in the last several years. Most of previous research studies focused on a single social network. For instance, Guy, Earle, Ostrum, Gruchalla, and Horvath (2010) described Twitter Earthquake Detector (TED) system that infers the level of public interest in a particular earthquake based on Twitter activity to decide which earthquakes to disseminate to the public. Sakaki et al., (2010) investigated the real-time nature of Twitter for detection of earthquakes. Caragea, McNeese, Jaiswal, Traylor, Kim, Miltra, Wu, Tapia, Giles, Jansen, and Yen (2011) compared different classification approaches on the Haiti disaster relief dataset obtained from the Ushahidi project. Starbird, Muzny and Palen (2012) investigated the performance of machine learning techniques in identifying on-the-ground twitterers during mass disruptions. Imran, Elbassuoni, Castillo, Diaz, and Meier (2013) classified unstructured tweets into a set of classes and extracted short self-contained structured information for further analysis. Our disaster detection approach differs in several ways. We propose to integrate data from multiple social sources as opposed to a single social source. We also investigate the detection of multi-hazard disasters, in particular landslides, which can be caused by various hazards, such as earthquakes and torrential rains. That is why our system also integrates data from physical sources, including seismic activities and rainfalls.

Another important aspect of a disaster detection system based on social media is situational awareness. Although most of social networks provide support for users to disclose their locations, e.g. when they send a tweet or share a photo, Cheng, Caverlee, and Lee (2010) showed that less than 0.42% of all tweets actually use this functionality. Vieweg, Hughes, Starbird, and Palen (2010) analyzed microblog posts to identify information that may contribute to enhancing situation awareness. Cheng et al., (2010) proposed and evaluated a probabilistic framework for estimating a Twitter user's city-level location based on the contents of tweets. Hecht and Gergle (2010) proposed to match locations in user profiles against the titles of Wikipedia articles containing geo coordinates. Hecht, Hong, Suh, and Chi (2011) showed that 34% of users did not provide real location information in their Twitter user profiles, and those that did input their locations – mostly specified at a city-level detail. Sultanik and Fink (2012) demonstrated a rapid unsupervised extraction of locations references from tweets using an indexed gazetteer, which is a dictionary that maps places to geographic coordinates. Our system also extracts geo terms from the textual descriptions of data from social media using the Wikipedia articles containing geo coordinates as an indexed gazetteer. We improve the precision of this geo-tagging algorithm based on a number of heuristics to filter out irrelevant matches.

CONCLUSION

In this work we describe the landslide detection system LITMUS, which integrates multiple sources to detect landslides, a representative multi-hazard. In particular, the system integrates social sensors (Twitter, Instagram, and YouTube) and physical sensors (USGS seismometers and TRMM satellite). The data from social sensors is processed by LITMUS in a series of filtering steps, including data collection based on landslide keywords, a filter based on stop words and stop phrases, a smart geo-tagging filter, a machine learning based classification filter, and a filter based on a blacklist of URLs. The remaining data from social sensors as well as all data from physical sensors are combined for the final integration of all sensors to produce a list of detected landslides.

The effectiveness of the system is evaluated using real world data collected in October 2013. The full integration of five sensor sources applying a modified Bayesian integration strategy detected all 11 landslides reported by USGS as well as 31 more landslides unreported by USGS during the evaluation period.

The user functionality of the system as well as its application to Typhoon Haiyan is described in the Live Demonstration section. The landslide detection system LITMUS is online and openly accessible, collecting live data for continued evaluation and improvement of the system, and the reader is encouraged to use the demo.

REFERENCES

1. Caragea C., McNeese N., Jaiswal A., Traylor G., Kim H., Mitra P., Wu D., Tapia A.H., Giles L., Jansen B.J., Yen J. (2011) Classifying Text Messages for the Haiti Earthquake, *Proceedings of ISCRAM '11*, Lisbon, Portugal.
2. Cheng Z., Caverlee J., Lee K. (2010) You Are Where You Tweet: A Content-Based Approach to Geo-locating Twitter Users, *Proceedings of CIKM '10*, Toronto, Canada.
3. Dittrich A., Lucas C. (2013) A step towards real-time analysis of major disaster events based on tweets, *Proceedings of ISCRAM '13*, Baden-Baden, Germany.
4. Guy M., Earle P., Ostrum C., Gruchalla K., Horvath S. (2010) Integration and dissemination of citizen reported and seismically derived earthquake information via social network technologies, *Proceedings of IDA '10*, Tucson, Arizona.
5. Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P., Witten I.H. (2009) The Weka Data Mining Software: An Update, *ACM SIGKDD Explorations Newsletter*, vol. 11, issue 1: 10-18.
6. Hecht B., Gergle D. (2010) On the "Localness" of User-Generated Content, *Proceedings of CSCW '10*, Savannah, GA.
7. Hecht B., Hong L., Suh B., Chi E.H. (2011) Tweets from Justin Bieber's Heart: The Dynamics of the "Location" Field in User Profiles, *Proceedings of CHI '11*, Vancouver, Canada.
8. Imran M., Elbassuoni S.M., Castillo C., Diaz F., Meier P.P. (2013) Extracting Information Nuggets from Disaster-Related Messages in Social Media, *Proceedings of ISCRAM '13*, Baden-Baden, Germany.
9. Kanamori H. (2005) Real-time seismology and earthquake damage mitigation, *Annual Review of Earth and Planetary Sciences*, vol. 33: 195-214.
10. Khairunniza-Bejol S., Petrou M., Ganas A. (2006) Landslide Detection Using a Local Similarity Measure, *Proceedings of the 7th Nordic Signal Processing Symposium*, Reykjavik, Iceland.
11. Palen L., Anderson K.M., Mark G., Martin J., Sicker D., Palmer M., Grunwald D. (2010) A Vision for Technology-Mediated Support for Public Participation & Assistance in Mass Emergencies & Disasters, *Proceedings of ACM-BCS Visions of Computer Science 2010*.
12. Sakaki T., Okazaki M., Matsuo Y. (2010) Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors, *Proceedings of WWW '10*, Raleigh, North Carolina.
13. Starbird K., Muzny G., Palen L. (2012) Learning from the Crowd: Collaborative Filtering Techniques for Identifying On-the-Ground Twitterers during Mass Disruptions, *Proceedings of ISCRAM '12*, Vancouver, Canada.
14. Sultanik E.A., Fink C. (2012) Rapid Geotagging and Disambiguation of Social Media Text via an Indexed Gazetteer, *Proceedings of ISCRAM '12*, Vancouver, Canada.
15. Vieweg S., Hughes A.L., Starbird K., Palen L. (2010) Microblogging During Two Natural Hazards Events: What Twitter May Contribute to Situational Awareness, *Proceedings of CHI '10*, Atlanta, Georgia.