Lost Silence: An emergency response early detection service through continuous processing of telecommunication data streams

Qianru Zhou[1], Stephen McLaughlin[1], Alasdair J. G. Gray[2], Shangbin Wu[3], and Chengxiang Wang[1]

[1] School of Engineering & Physical Sciences, Heriot-Watt University, EH14 4AS, UK
[2] Department of Computer Science, Heriot-Watt University, EH14 4AS, UK
[3] Samsung R&D Institute UK, Staines-upon-Thames, TW18 4QE, UK

Abstract. Early detection of significant traumatic events, e.g. a terrorist attack or a ship capsizing, is important to ensure that a prompt emergency response can occur. In the modern world telecommunication systems could play a key role in ensuring a successful emergency response by detecting such incidents through significant changes in calls and access to the networks. In this paper a methodology is illustrated to detect such incidents immediately (with the delay in the order of milliseconds), by processing semantically annotated streams of data in cellular telecommunication systems. In our methodology, live information about the position and status of phones are encoded as RDF streams. We propose an algorithm that processes streams of RDF annotated telecommunication data to detect abnormality. Our approach is exemplified in the context of a passenger cruise ship capsizing. However, the approach is readily translatable to other incidents. Our evaluation results show that with a properly chosen window size, such incidents can be detected efficiently and effectively.

Keywords: telecommunications, event detection, emergency response, C-SPARQL

1 Introduction

At 21:30 on 1st June, 2015, the cruise ship "Eastern Star" traveling on the Yangtze River began to list as a consequence of stormy weather. One minute later, the ship capsized with 458 passengers and crew on board. No distress signal was sent. It was several hours before the emergency services became aware of the tragedy and 442 lives were lost [1]. Although considerable effort has been devoted to the emergency response to natural and man-made disasters, the fatalities and economic cost due to untimely rescue are still significant [2, 3]. In 2016 alone, 225,665 refugees arrived in Europe by sea, and approximately 2,933 lost their lives due to ship capsizing accidents during illegal migration [2]. Most of the capsizes were not noticed until the survivors swam to shore.

When a ship capsizes and is incapable of communication by radio, is there any way to detect the ship capsizing as soon as it happens? In a telecommunication system, the location and status information of a phone are stored in the service providers' databases (the specific name of the databases are visiting location register, VLR, and home location register, HLR). When a cell phone is switched off normally (e.g., by pressing power button, or batteries wearing out), it will register its status as "detached" to the database. However, if a device is forcefully shut down (e.g., dropped into the water, physically damaged, or the battery removed), or it enters a blind spot with no signal coverage, it does not have time to register its status [4]. Within a location update period (this varies from 30 minutes to 1 hour, depending on the service provider), the phone is marked as "unReachable" in the database. After that period, its status will be changed to "detached", which is the same as in the situation of a normal shutdown. These status signals can be used to infer abnormal events. For example, a large number of phones losing signal abnormally at the same time and location can be used as an indication of a significant event, potentially catastrophic. Taking advantage of this fact, we present *lost silence*, an early detection algorithm that can detect abnormalities. The location update process in the telecommunication system generates ularge quantities of data in real time. By annotating this information with RDF/OWL vocabularies, we present real time phone information as streams of RDF encoded linked-data. The pattern is processed with continuous SPARQL (C-SPARQL) queries [5, 6].

This paper is structured as follows. Section 2 presents our illustrative scenario based on the "Eastern Star" shipwreck accident. Section 3 introduces some technical background knowledge of telecommunication networks that are used in our proposal. Section 4 presents the ontology we have adopted in our proposed methodology. Section 5 gives details of the algorithms. The evaluation requirements, results, and discussions based on our scenario are highlighted in Section 6. Related work is presented in Section 7 and our conclusions are given in Section 8.

2   Motivating Scenario

We outline a scenario based on the cruise ship "Eastern Star" capsizing on the Yangtze River, in Jianli, Hubei, China [1]. In addition to the Yangtze River, the city of Jianli consists of two main parts, a densely populated zone – Rongcheng town and a surrounding rural area – Hongcheng village. The detail of population and geographic information is shown in Table. 1.

In a city with a busy telecommunication datacenter with high volumes of phones (approx. 286,000 phones in the city covering an area of 373.520 km^2), we have evaluated the algorithm proposed in this paper against the time taken to detect abnormalities, the accuracy of the detection. In particular, we have investigated the tradeoffs between the time taken in generating linked-data streams for the massive amount of data in telecommunication system, and the accuracy of the abnormalities detected.

Table 1. Geographic data of the city Jianli.

Town	Area (km^2)	Population
Rongcheng Town	85	152, 358
Hongcheng Village	254	133, 544
Water Area	14.520	0

It is possible that some situations would reduce the accuracy of the approach described in this paper. For example, although the telecommunication service providers are trying very hard to achieve full signal coverage, there are some regions, especially in rural areas, that have no or limited signal coverage. These areas are known as blind zones. If a large number of phones enter a blind zone, then a large number of phone signals are lost at the same spot. Each service provider keeps a list of its blind zones. Such false alarms could be avoided by a simple comparison to the list before an alert is sent.

3 Background Technology

3.1 Stream processing

In the last few years, stream processing has gained a prominent attention in the Semantic Web community [5–9]. The data in real telecommunication system, e.g., system log, SNMP polling, tcpdump, and configuration data, etc., are extremely dynamic. The granularity of SNMP, Syslog and tcpdump update period is in terms of a few seconds, one second and microseconds, respectively [10]. Thus, there will be significant advantages to being able to manage rapidly changing systems at the semantic level. The proposed lost silence application is powered by the Continuous SPARQL (C-SPARQL) engine [6], an extension of SPARQL with the ability to query streaming RDF data in real time.

3.2 The location register process in mobile telecommunication networks

To ground our discussion, it is necessary to consider how the telecommunication system maintains records of the location of a phone. In telecommunication networks, phones measure the channel environment and reselect cells every 200ms, and report its location to the cell tower periodically. This period varies from every 30 minutes to every hour, depending on the service providers [11]. When a phone is turned off normally, it will deregister itself to the network system and its status will be marked as "detached". However, if a phone loses its power abnormally, it has no time to deregister. Thus its status in the network system will be "unReachable". After the update period, if the network still does not receive update information from this phone, it will register the phone as "Detached", in the same manner as if it had been shut down normally [4, 11]. Thus, when a phone loses signal abnormally, there is a limited time period to identify whether

it has been normally shut down or not. Typically this period is of the order of 30 minutes to an hour. The location information of a phone is sent and stored in the telecommunication network servers (namely home location register, HLR or visitor location register, VLR) located in the datacenter of the telecommunication system. Normally, there is one datacenter for one service provider per city, managing the data of all the phones in the city. There are usually more than one service provider in a city. In cases when public security is concerned, some information is shared and uploaded to an authorized third party. For example, in America, as enforced by the government, the location information of every 911 call is automatically obtained and shown to the police [12].

3.3   Geo-Pixel

Lost silence needs to identify the spatial location where an abnormally high number of phones lose their signals almost simultaneously. Key to the lost silence algorithm is a uniformed spatial segmentation among the data. We propose the concept of *Geo Pixel* here. It is defined based on the third decimal degree of the latitude and longitude coordination, with the resolution of $0.001° \times 0.001°$, or $100m \times 100m$. Thus, with *Geo Pixel*, phones are aggregated into grid cells based on location. The main task of lost silence algorithm is to detect the *Geo-Pixels* with abnormally high number of lost signal.

   We made a simplification with regards to *Geo Pixel*. At the equator, one third decimal degree of longitude and latitude both cover about 100 meters. However, when moving toward the pole, the distance represented by one degree of longitude degeneralize to zero, while the latitude stays almost the same, for the distance resolution of longitude (east-west distance) depends on the latitude. For example, the distance between one degree of longitude worth up to 111.320 km when latitude is $0°$, and reduces to only 28.920 km when latitude is $75°$. However, most countries with significant mobile phone using populations lie between $75°$ to $-75°$ latitude. Thus, the spatial resolution of *geo-pixel* varies between $100\ m \times 100\ m$ to $100\ m \times 30\ m$, which does not have any impact on the algorithm except that making the result more accurate for some countries at high latitudes.

4   Ontology adopted in Lost Silence

To be able to transform telecommunication system data into RDF, we adopted an ontology that models the raw data of the system. The ontology which is named TOUCAN Ontology (TOCO) is developed for TOUCAN project[4]. TOCO is proposed to represent the knowledge within a heterogeneous telecommunication system, consisting various technology domains, e.g., mobile network, computer network, optical network, light fidelity (LiFi), and wireless fidelity (WiFi), etc.

---

[4] EPSRC TOUCAN project, No. EP/L020009/1. Website: `http://gow.epsrc.ac.uk/NGBOViewGrant.aspx?GrantRef=EP/L020009/1`
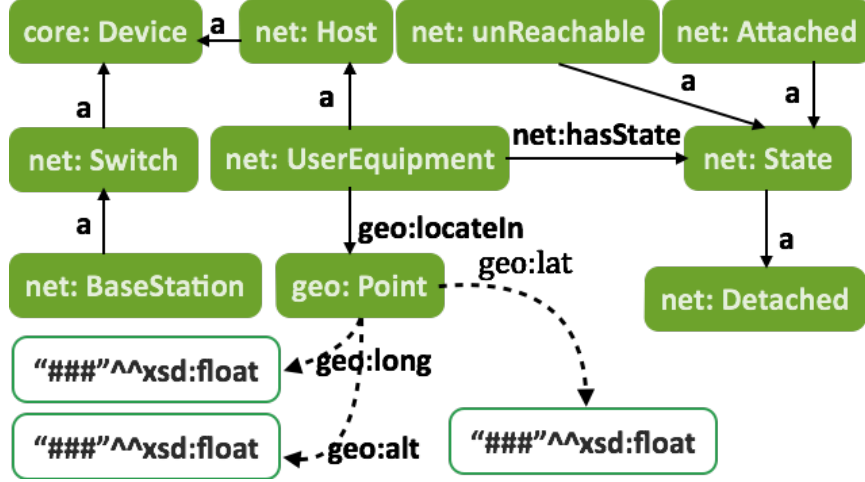
Fig. 1. Classes of the TOUCON Ontology used in the Lost Silence scenario. The solid block denotes an object, while the hollow block denotes a data. The solid lines denote object properties, and dash lines denote datatype properties. The prefixes adopted in lost silence are: *net*: `<http://home.eps.hw.ac.uk/~qz1/ ontologies/wirelessnetwork_networkResource.owl/>`; *core*: `<http://home.eps. hw.ac.uk/~qz1/ontologies/wirelessnetwork.owl/>`; *geo*: `<http://www.w3.org/ 2003/01/geo/wgs84_pos/>`.

TOCO is composed of one core ontology and 5 sub-ontologies, namely, *network resource, service, user, time, location*. A diagram of the portion of the TOCO class hierarchy adopted in lost silence is shown in Fig. 1.

Some important concepts and relations are presented below.

"UserEquipment" - user equipments in mobile communication system, such as phones, tablets, wearable facilities, etc. It has an object property `hasStatus` with the object of class `Status`. It is defined in the network resource sub-ontology.

"Point" - describes the location of phone, extended from `SpatialThing` of the W3C WGS-84 vocabulary[5]. It is defined in the location sub-ontology. It has three datatype properties, namely, `long`, `lat`, and `alt`, describing the longitude, latitude, and altitude of the phone.

"Status" - the status of UE connectivity in communication system. There are three instances of this class, namely, `Attached`, `Detached`, and `unReachable`, which denotes the phone is connected, detached after deregister to the system, and unreachable without deregister to the system, respectively. It is defined in the network resource sub-ontology. It is the range of the property `hasStatus` of `UserEquipment`.

---

[5] See `http://www.w3.org/2003/01/geo/` for more detail.

For example, to describe the fact that *"a phone lost signal at location (329.860, 246.792)"*, the corresponding triples are:

net:Phone_1   *a*   *net:UserEquipment.*
pos:Point_1   *a*   *pos:Point.*
net:Phone_1   *pos:location*   *pos:Point_1.*
pos:Point_1   *pos:latitude*   *"329.860".*
pos:Point_1   *pos:longitude*   *"246.792".*
net:unReachable   *a*   *net:Status.*
net:Phone_1   *net:hasStatus*   *net:unReachable.*

5   Algorithm

As the data volume from telecommunication system is extremely large, the philosophy of divide-and-conquer is adopted in the algorithm design. In the datacenter of the telecommunication system, for our scenario more than 400 phones lost their signal at the same location within seconds[6]. This abnormality could have been detected if continuous queries were being executed on the streams in real-time. An alert could have been raised much earlier and help sent to those on the ship. In the datacenter of the telecommunication system, every time a phone updates its status, a linked-data event is generated and published as a collection of triples. As the phones in the city keep updating their location information and signal status, a linked-data stream is generated. In order to estimate the volume and velocity of data variation from telecommunication system during accidents, we referred to the literatures. The mobile phones penetration rate is 96.7% in China, [13]. With the *geo-pixel* simplification, the number of geo-pixels and the population density of phones in each town are shown in Table. 2. As a result, in the city of Jianli, there are 29,215 geo-pixels, in which 7,024 of them are in the densely populated zone with on average 21 phones per pixel, 20,991 pixels are in rural areas with on avaerage 6 phones per pixel, and on the river there are 1,200 pixels where the phone density is on average 0 phones per pixel.

Table 2. Phone density to geo-pixel in the city Jianli

Town	Number of *Geo-pixels*	Phone Density
Rongcheng Town	7024	21
Hongcheng Village	209911	6
Water Area	1200	0

The C-SPARQL query string to detect lost phones is shown in Algorithm 1. The query is evaluated every 5 seconds calculating for each geopixel the count of unreachable devices over the last 30 minutes. A result is only returned if the count is above 10. Each stream is denoted as <http://home.eps.hw.ac.uk/~qz1/

---

[6] By personal correspondence.

Algorithm 1 C-SPARQL query string for detecting the lost phones. Each non-empty result in the returned result sets denotes an alert.

REGISTER QUERY StreamingAndExternalStaticRdfGraph AS
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX pos: <http://www.w3.org/2003/01/geo/wgs84_pos/>
PREFIX net: <http://home.eps.hw.ac.uk/~qz1/ontologies/
wirelessnetwork_networkResource.owl/>
PREFIX fn: <http://www.w3.org/2005/xpath-functions#>
SELECT (COUNT(?UE) AS ?counter) ?lat ?long
FROM STREAM <http://home.eps.hw.ac.uk/~qz1/ontologies/wirelessnetwork
_networkResource.owl/stream> [RANGE 30m STEP 5s]
WHERE {
    ?UE net:hasStatus net:unReachable.
    ?UE pos:location ?point.
    ?point pos:lat ?lat;
           pos:long ?long.
    BIND (fn:round(?lat × 1000) as ?roundLat)
    BIND (fn:round(?long × 1000) as ?roundLong)
}
GROUP BY ?roundLat ?roundLong HAVING (?counter ≻10)

Algorithm 2 Stream Generator: Generate RDF streams for each given *geo-pixels*.

// Input are the longitude and latitude of the city, ratio of phone lost signal, thread sleep time. Output is the RDF stream created
Require: *lat, long, lostRatio, sleepTime*
Ensure: *Stream*
  $keepRunning \Leftarrow 1$
  while keepRunning **do**
    Insert triple (net:Phone, geo:locateIn, geo:Point) to ***Stream***
    Insert triple (geo:Point, geo:latitude, "*lat*"^^xsd:double) to ***Stream***
    Insert triple (geo:Point, geo:longitude, "*long*"^^xsd:double) to ***Stream***
    With a ratio of ***lostRatio***:
        Insert triple (net:Phone, net:hasStatus, net:unReunReachableachable) to ***Stream***
    With a ratio of $(1 - $ ***lostRatio***$)$:
        Insert triple (net:Phone, net:hasStatus, net:Attached) to ***Stream***
    Thread sleep for ***sleepTime*** seconds
  end while

Algorithm 3 Thread Pool Scheduler: Schedule the C-SPARQL engine threads.

// Input is the number of *geo-pixels* for city center, rural area, and water zone. Output is the tragedy detection results of the city
Require: *CityNum, RuralNum, WaterNum*
Ensure: *Gstream*
  Instantiate a blocking queue: ***q***
  Instantiate a thread pool: ***pool***
  for each of the ***CityNum / RuralNum / WaterNum*** *geo-pixels* in the **City / Rural area / Water zone do**
      Create an **Stream Generator** as a thread into ***pool***
      Apply C-SPARQL query on the stream
  end for
  Shut down ***pool***

---

ontologies/wirelessnetwork/networkResource.owl/stream> in the query. To avoid the omissions of accidents, we choose the window size the same as the phone's position update period in communication system – 30 minutes. We adopt *thread-per-geo-pixel* architecture to process telecommunication data and generate a RDF stream in one *geo-pixel* with one thread. The threads are maintained by a thread pool. Each thread in the pool has the same execution priority and receives an equal share of CPU time.

The details of the generated streams and thread scheduling process are illustrated in an self-explanatory way in Algorithm 2 and 3 for brevity. Algorithm 2 shows the stream generating process for different areas. The *lostRatio* is adopted to control the ratio of phones that lost signal in the simulated streams. For example, if the *lostRatio* = 0.1, it means 10% of the phones in this stream will lose signal, and if the *lostRatio* = 1, it denotes all the phones in this stream will lose signal. The density of phones is determined by the *sleepTime*. For example, in the rural area Hongcheng Village where the phone density is 6 phones per pixel, as the location update period is 5 *seconds* (= 5000 *milliseconds*), in order to simulate the scenario that there are 6 phones signal update per 5 *seconds*, we have to generate the RDF triples for one phone every $5000 \div 6 \approx 833.333$ *milliseconds*. Thus, the *sleepTime* for rural area is 833 *milliseconds*. Similarly, the *sleepTime* for Rongcheng Town is $5000 \div 21 \approx 238$ *milliseconds*.

6   Evaluation

We run an extensive and exhaustive evaluation based on the tragedy detection algorithm. The evaluation was carried out on a MacBook Air OS X 10.9.5 with 3MB cache running on Intel Core i5 at 1.5 GHz. The system has 128 GB SSD and 4 GB RAM.

6.1   Prepare the Data

A small number of phones will randomly disappear from random pixels along the river. At a randomly chosen time, 424 phones will lose their signal simultaneously

in the geo-pixel (329.863, 246.792). With a query window of 30 minutes, we experimented with query steps of 5, 20, and 30 seconds respectively. For the three step sizes, we ran in total 3 iterations of the algorithm. Our evaluation for the shipwreck scenario focus on two criteria: the time taken to detect abnormalities and the accuracy of the detection, in terms of the cases of fail-to-report.

6.2   Experiment Framework

Fig. 2 shows our workflow and execution environment of the lost silence experiment. Data from the telecommunication system in each *geo-pixel* is converted into RDF streams by the Stream generator. RDF streams are accessed and queries by the C-SPARQL engine thread. All the 29,215 query threads processing 285,902 phone data are scheduled and executed by thread pool.
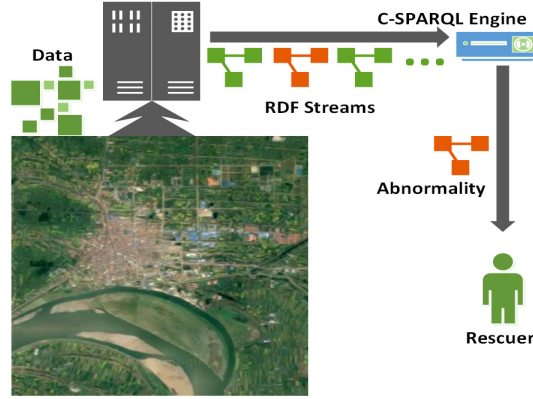

Fig. 2. The Process of disaster early detection in lost silence.

6.3   Evaluation Results

We run the experiment for three times, with the with query steps of 5 seconds, 20 seconds, and 30 seconds, respectively. The simulation results are shown in Fig. 3. The Fig. 3(a) and 3(b) are the simulation result with the query step of 5 seconds, the Fig. 3(c) and 3(d) are with the query step of 20 seconds, and the Fig. 3(e) and 3(f) 30 seconds. In the Fig. 3(a) 3(c), and 3(e) on the left, the horizontal and vertical coordinates indicate the latitude and longitude of the geographical area respectively. The total number of lost phones detected at each *geo-pixel* are shown as circles on the subfigures. The number of lost phones are denoted as the radius and colour of the circles, e.g., the smallest circle in
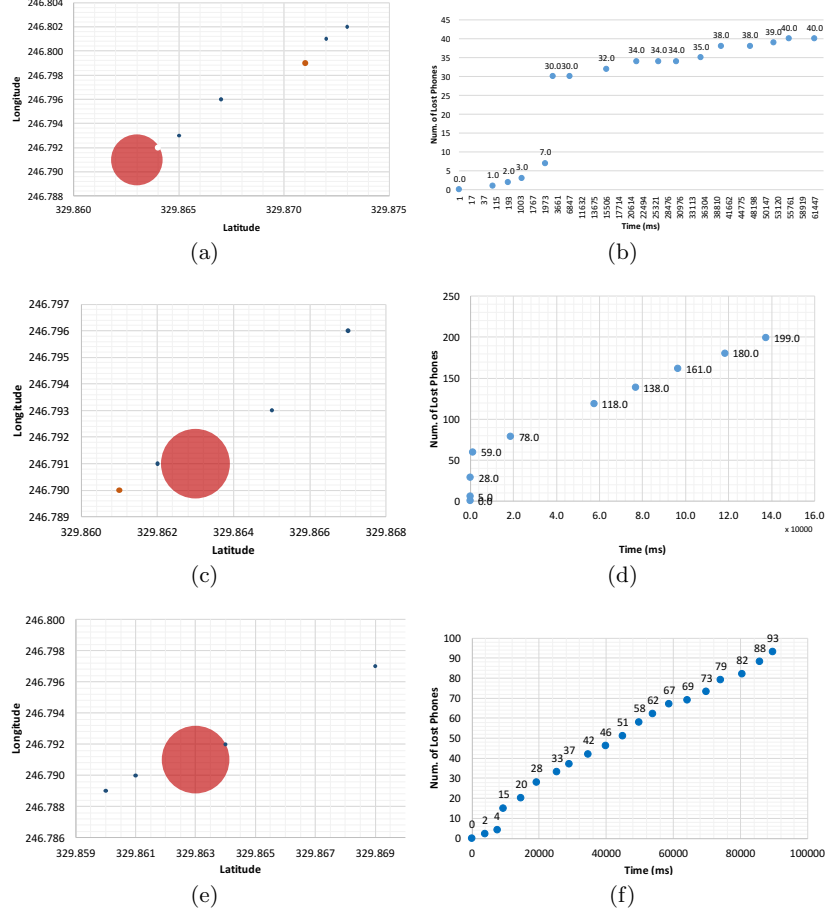
Fig. 3. The total number of lost phones in the shipwreck area and the query results in the *geo-pixel* (329.863, 246.792), where the shipwreck took place, at each query step for the three experiments with the query steps of 5 seconds, 20 seconds, and 30 seconds, respectively. Fig. a) and b) correspond to the experiment with the query step of 5 seconds, Fig. c) and d) correspond to 20 seconds, and Fig. e) and f) 30 seconds. Fig. a), c), and e) illustrate the total number of phones lost signal at the water area of city Jianli at each experiment. The number of phones lost signal scales with the colour and radius of the circle, e.g., the brighter colour and larger radius of a circle denotes a larger number of lost phones. Fig. b), d), and f) show the number of detected lost phones in the *geo-pixel* (329.863, 246.792), at each query step from the beginning of the query.
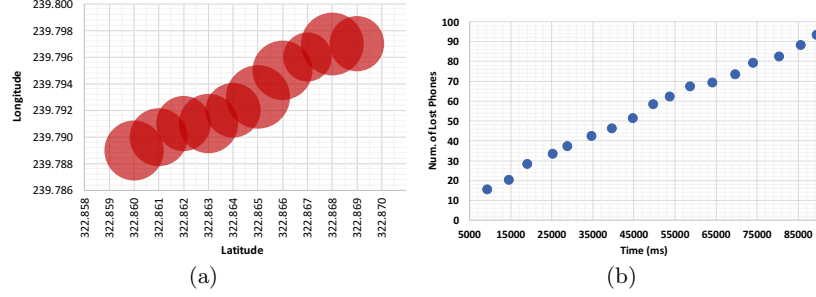
(a)   (b)

Fig. 4. Query result in the pressure test scenario in which massive phones lost signal in multiple *geo-pixels* along the river.

black denotes only one lost phone at that location, the second smallest circle in orange denotes two lost phones, and the largest circle in red is the total 442 lost phones detected where the shipwreck took place. The Fig. 3(b), 3(d), and 3(f) on the right show the number of lost phones detected in the geo-pixel (329.863, 246.792), where the shipwreck took place, at each query step, from the beginning of the experiment. To shade light on the detailed detection process, we show the number of lost phones below 10 as well, although no alert will be send for that. As shown in the Fig. 3(b), 3(d), and 3(f), there are sudden jumps of the number of lost phones in each figures, at the time when the shipwreck happens. As the number of lost phones raised to above 10, the lost silence will send alert of that abnormally rise together with the information of the geo-pixel. As the query step grow from 5 seconds (as shown in Fig. 3(b)) to 30 seconds (as shown in Fig. 3(f)), the number of lost phones increase faster, and the dots intervals are more sparse.

A pressure test is also carried out to simulate an extreme scenario in which 10 ships capsize in different locations on the river simultaneously, as shown in Fig. 4. Each ship carries more than 300 passengers. The duration of the capsizing of one ship is about 3 minutes. Thus, in the 10 *geo-pixels* along the river, there is a massive phone loss incident with the rate of about $300 \div 3 = 100$ phones per minute. As shown in the Fig. 4(a), the abnormality of a large number of lost phones in all of the 10 *geo-pixels* are detected, and alerts are sent successfully. The results at each query step for an random *geo-pixel* is shown in the Fig. 4(b). A sharp increase of the lost phones can be spotted, with the rate about 100 phones per minute. The result demonstrates the efficiency and practicality of the lost silence.

7   Related Work

The utility of data from telecommunication system in geography and social science to improve urban planning has been increasingly investigated [4, 14–17]. In [4], Willessan presented how evidence obtained from mobile system plays a

part in forensic investigation. [14–17] provided a extensive coverage of the smart city applications adopting data from telecommunication system. However, these application mainly focus on mobile positioning only.

Scholte and Rozenkrane [18] have proposed a system to localize and track each ship, and send personalized alert to those that are expected to be in danger. However, this system might lose function when the communication device fails, and cannot detect an accident when the communication device on the ship is not functioning. [19] designed and developed an ontology for emergency notification, such as "a typhoon approaching", but not for detection tragedies already happened.

8   Conclusions

In this paper we have shown that the system data inside cellular networks can be used to detect accidents with large number of phones lost signal at the same location simultaneously, in our scenario a ship capsizing incident.

In lost silence, user data from all the vendors are concentrated and organized with Geo-Pixel, which is proposed as a geography unit. Continuous RDF processing is adopted to query the data streams in real time. We perform the simulation based on a real life shipwreck incident in China 2015, and discussed the results in different scenarios.

This lost silence service adopts the ontology developed and built for the TOUCAN project. Current ontologies of telecommunication networks generally do not provide information and knowledge inside the system. In our future work, we will intent to extend the lost silence with machine learning approaches and real data from cellular networks.

Acknowledgements

Bibliography

[1] Peace time disasters. 2017. `https://en.wikipedia.org/wiki/List_of_maritime_disasters_in_the_21st_century#Peacetime_disasters`

[2] Missing migrates. 2017. `https://missingmigrants.iom.int`

[3] Shipwreck since 1833. 2017. `http://www.infoplease.com/ipa`

[4] Willassen, S., 2003. Forensics and the gsm mobile telephone system. International Journal of Digital Evidence 2 (1), 1–17.

[5] Valle, E. D., Ceri, S., Harmelen, F. V., Fensel, D., 2009. It's a streaming world! reasoning upon rapidly changing information. IEEE Intelligent Systems 24 (6), 83–89.

[6] Barbieri, D. F., Braga, D., Ceri, S., Valle, E. D., M.Grossniklaus, 2009. C-sparql: Sparql for continuous querying. In: ACM the 18th international conference on World wide web. pp. 1061–1062.

[7] Kolchin, M., Dmitry, M., Sergey, A., 2014. Demonstration: web-based visualisation and monitoring of smart meters using cqels,.

[8] Calbimonte J P, Corcho O, Gray A J G., 2010. Enabling ontology-based access to streaming data sources. In: International Semantic Web Conference. Springer, Berlin, Heidelberg, 2010. pp. 96 –111.

[9] Margara, A., Urbani, J., van Harmelen, F., 2014. Streaming the web: Reasoning over dynamic data.

[10] Gowrishankar, N., Satyanarayana, P. S., 2008. Neural network based traffic prediction for wireless data networks. International Journal of Computational Intelligence Systems 1 (4), 379–389.

[11] 3GPP, 2017. LTE specifications. `http://www.3gpp.org/specifications`

[12] FCC, 2017. Enhanced 911 - wireless services. `https://www.fcc.gov/general/enhanced-9-1-1-wireless-services`

[13] 2017. Ministry of industry and information technology of the people's republic of china. `http://www.miit.gov.cn/`

[14] Steenbruggen, J., Tranos, E., Nijkamp, P., 2015. Data from mobile phone operators: A tool for smarter cities? Telecommunications Policy 39 (3), 335–346.

[15] Antonelli, F., Azzi, M., Balduini, M., 2014. Citysensing:visualising mobile and social data about a city scale event. In: ACM International Working Conference on Advanced Visual Interfaces. pp. 337–338.

[16] Gonzlez, M., Lind, P. G., Herrmann, H. J., 2006. System of mobile agents to model social networks. Physical review letters 96 (8), 088702.

[17] Song, C., Qu, Z., Blumm, N., Barabási, A.-L., 2010. Limits of predictability in human mobility. Science 327 (5968), 1018–1021.

[18] Scholte, K., Rothkrantz, L., 2014. Personal warning system for vessels under bad weather conditions. Pennsylvania State University.

[19] Malizia, A., Astorga-Paliza, F., Onorati, T., Díaz Pérez, P., Aedo, I., 2008. Emergency alerts for all: an ontology based approach to improve accessibility in emergency alerting systems. In: ISCRAM, Washington, DC, USA, May 4-7, 2008.