Predicting Evacuation Decisions using Representations of Individuals' Pre-Disaster Web Search Behavior

Takahiro Yabe
Lyles School of Civil Engineering
Purdue University, USA
tyabe@purdue.edu

Kota Tsubouchi
Yahoo Japan Corporation
Tokyo, Japan
ktsubouc@yahoo-corp.jp

Toru Shimizu
Yahoo Japan Corporation
Tokyo, Japan
toshimiz@yahoo-corp.jp

Yoshihide Sekimoto
Institute of Industrial Science
University of Tokyo, Japan
sekimoto@iis.u-tokyo.ac.jp

Satish V. Ukkusuri
Lyles School of Civil Engineering
Purdue University, USA
sukkusur@purdue.edu

ABSTRACT

Predicting the evacuation decisions of individuals before the disaster strikes is crucial for planning first response strategies. In addition to the studies on post-disaster analysis of evacuation behavior, there are various works that attempt to predict the evacuation decisions beforehand. Most of these predictive methods, however, require real time location data for calibration, which are becoming much harder to obtain due to the rising privacy concerns. Meanwhile, web search queries of anonymous users have been collected by web companies. Although such data raise less privacy concerns, they have been under-utilized for various applications. In this study, we investigate whether web search data observed prior to the disaster can be used to predict the evacuation decisions. More specifically, we utilize a *session-based query encoder* that learns the representations of each user's web search behavior prior to evacuation. Our proposed approach is empirically tested using web search data collected from users affected by a major flood in Japan. Results are validated using location data collected from mobile phones of the same set of users as ground truth. We show that evacuation decisions can be accurately predicted (84%) using only the users' pre-disaster web search data as input. This study proposes an alternative method for evacuation prediction that does not require highly sensitive location data, which can assist local governments to prepare effective first response strategies.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile devices**; • **Computing methodologies** → **Knowledge representation and reasoning**;

KEYWORDS

web search queries, representation learning, evacuation prediction, mobile phone location data

KDD '19, August 4–8, 2019, Anchorage, AK, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330697

ACM Reference Format:
Takahiro Yabe, Kota Tsubouchi, Toru Shimizu, Yoshihide Sekimoto, and Satish V. Ukkusuri. 2019. Predicting Evacuation Decisions using Representations of Individuals' Pre-Disaster Web Search Behavior. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3292500.3330697

1 INTRODUCTION

Severe disasters such as the Tohoku tsunami (2011) and Hurricanes Harvey, Irma and Maria (2017) have caused mass evacuation activities due to damage on urban infrastructure [37]. Speedy and effective response to natural disasters is of utmost importance to numerous cities around the world, due to the increasing frequency and intensity of large scale hazards [29]. For efficient disaster response, it is crucial to predict evacuation activities before the disaster strikes. Such information can be used by decision makers to prepare effective strategies, such as the optimal allocation of emergency supplies and an efficient spatial distribution of evacuation shelters. Conventionally, surveys and census data have been used as primary sources to analyze evacuation decisions after disasters [16, 21, 22]. More recently, studies have utilized large scale location datasets for post-disaster analysis (e.g. mobile phone GPS [23, 34], call detail records [1, 15], Twitter Geo-tags [30]). Such works have provided insights on evacuation behavior through spatio-temporally detailed analysis. However, such analyses often lack predictive power on what will happen in future disasters.

To overcome such drawback of post-disaster analyses, various studies have proposed online methods to predict near future evacuation mobility [5, 10, 24–26, 28]. Although experiments show the effectiveness of such methods, most of these predictive methods require real time location data for model calibration. Recently, it is becoming increasingly difficult to obtain and use real time location data due to rising privacy concerns [4]. Thus, there is increasing demand for methods that can predict evacuation decisions using alternative data sources which raise less privacy concerns.

In this paper, we attempt to bridge this gap by proposing a method that utilizes web search queries collected from app users to predict evacuation decisions after disasters. Web search queries pose less threats to the users' privacy compared to location information, and are commonly collected by web service companies to improve browsing experiences. Despite this advantage, research

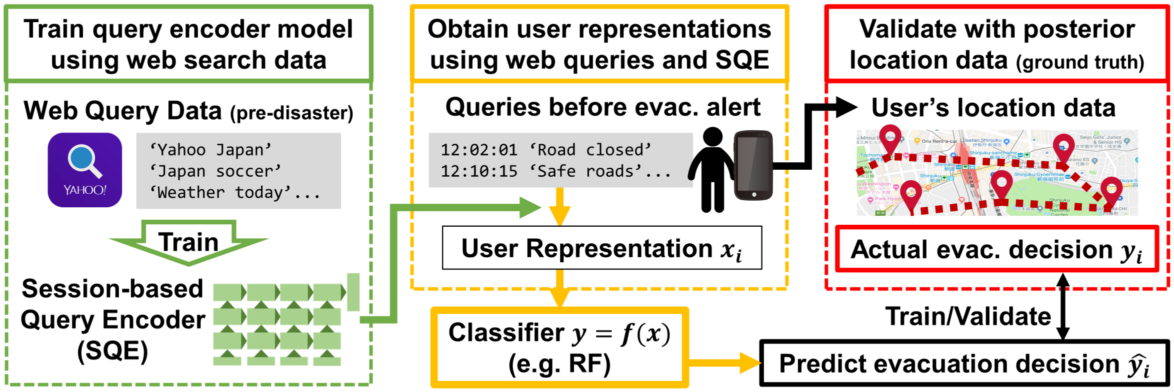
Train query encoder model using web search data
Web Query Data (pre-disaster)
YAHOO!
'Yahoo Japan'
'Japan soccer'
'Weather today'...
Train
Session-based Query Encoder (SQE)
Obtain user representations using web queries and SQE
Queries before evac. alert
12:02:01 'Road closed'
12:10:15 'Safe roads'...
User Representation x_i
Classifier $y = f(x)$ (e.g. RF)
Validate with posterior location data (ground truth)
User's location data
Actual evac. decision y_i
Train/Validate
Predict evacuation decision $\hat{y}_i$

Figure 1: Overall framework of the study. Evacuation decisions are predicted using user representations generated from web search queries observed prior to the disaster, and are validated using location data as ground truth.

on utilizing web search queries for various applications have been under-investigated. Konishi et al., which predicts future congestion in transit stations using transit app queries [14], is the closest study to ours. However, compared to transit queries where users explicitly search names of destination stations, web search queries have a much larger vocabulary set with latent meanings and subtle differences in expressions, making it much harder to utilize for mobility prediction. Thus, to overcome this difficulty, we utilize an LSTM model that learns the representations of queries by exploiting the web search behavior and the underlying search intent of users.

Using web search data and location information (used only as ground truth for validation) of Yahoo Japan App users, we test the predictive performance of our method. Figure 1 shows the overall framework of our study. First, we train our session-based query encoders (SQE) using web search queries collected prior to the disaster. Then, we select users who are located within the disaster region, and learn their representations from query data observed prior to the disaster alert using the SQE. We test the predictive performance of evacuation decisions using the user representations, by validating with actual evacuation decisions observed from location data. The intuition is that, users who decide to evacuate have common web search behavior traits before the disaster alert. Our approach overcomes the aforementioned drawbacks because it does not require location information of the users for prediction. We use real world data (both web search and location information) collected from users affected by the 2018 Japan Floods for experimentation.

The main contributions of this paper are:

- We present a novel approach that utilizes users' web search queries collected prior to the disaster alert to predict their evacuation decisions.
- We clarify that evacuation decisions can be predicted with high accuracy using real world data collected during the 2018 Japan Floods.
- We find that encoding the queries can improve the predictive accuracy of evacuation decisions, by capturing subtle differences in expressions and vocabulary.

The following sections are organized as follows. Section 2 provides details of the web search and location datasets used in this study. We introduce our methodology in Section 3, and present experimental validation results in Section 4. The results are discussed in Section 5, related works are introduced in Section 6, and conclusions are made in Section 7.

2 DATASET

In this study, we utilized web search data and location information collected by Yahoo Japan Corporation[1]. This is a unique dataset that contains both web search data and high granular location information of users, with the same set of IDs. Thus, this dataset provides us a valuable opportunity to explore the predictability of mobility decisions using web search behavior.

2.1 Web Search Data

The "Yahoo Japan app" is a popular smartphone app in Japan, which is a platform that provides various services including web search, shopping, weather information and disaster alerts. Yahoo Japan collects the web search queries of the users to improve the web search quality. Out of all users, a fraction of the users hold a Yahoo Japan user ID, which allows users to personalize and customize their services. We collected the web search data of users that have agreed to provide their data for research purposes. Panel A in Figure 2 shows that the probability distribution of the number of searches per day follows a truncated power law, with mean of around 12.

Figure 3 compares the frequencies of web searches performed by the users inside and outside the disaster affected area (Kurashiki), before, during and after the disaster date. Equal number of users were randomly chosen for both population groups (in and out of Kurashiki). Panel A shows that the total number of searches increase more for users located inside the disaster affected area compared to those who are not. Panels B and C show that in addition to the total number of web searches, users directly affected by the disaster tend to search more about traffic information and flood information compared to users outside the disaster area. These empirical analysis of web search data clarifies our intuition that users' web search behavior are indeed affected by natural disasters,

[1] https://about.yahoo.co.jp/info/en/

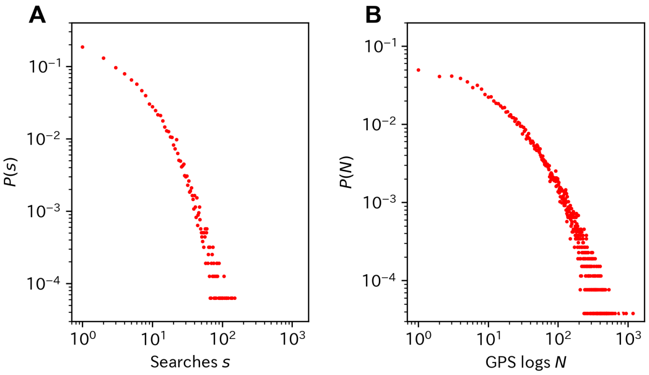

Figure 2: Probability distribution of: A) web search queries per individual on a given normal day B) GPS location data points per user on a given day and.

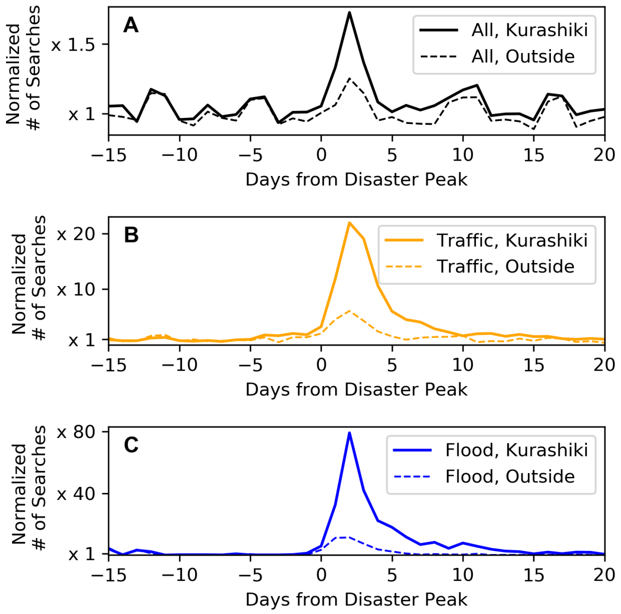

Figure 3: Comparison of web search counts between affected and non-affected users before, during and after the disaster. A) Total searches. B) Traffic related searches. C) Flood related searches.

motivating us to use such data to predict evacuation decisions. In this study, as shown in Figure 1, web query data collected prior to the disaster are used to train the session-based query encoders (SQE), and the users' web search queries prior to the evacuation alert are used as input of the SQE to generate user representations.

2.2 Mobile Phone Location Data

In addition to web search data, the Yahoo Japan app also collects location information of users in order to send only relevant notifications to the users. The users in this study have accepted to provide their location information. The data are anonymized so that individuals cannot be specified, and personal information such as gender, age and occupation are unknown. Each GPS record consists of a user's unique ID (random character string), latitude, longitude, date and time. Panel B of Figure 2 shows that the probability distribution of the number of observations per user per day is heavy tailed, with a mean of around 40 GPS points, allowing us to infer the major staying locations of each user. The GPS data collected by Yahoo Japan Corporation has a sample rate of about 2% of the population, and past studies suggest that this sample rate is enough to grasp the macroscopic urban dynamics [18, 34]. However, privacy concerns on location information are recently increasing, and it is extremely difficult to collect and use such data in real time. Therefore in this study, we assume that real time location data cannot be obtained for prediction, and thus we use only the web query data to predict evacuation decisions. The location information are used in this study as ground truth to validate our predictions.

3 METHODOLOGY

3.1 Preliminaries

*Definition (**Web Search Session**).* A user's web search behavior can be observed as a sequence of web search queries performed by the user. Usually, such continuous sequence of searches within a short time period are performed under a consistent underlying search intent. We define these short sequences of searches as "web search sessions", and utilize the latent relationships between queries within the same session to encode the representations of each query. Section 3.2.1 explains how we obtain web search sessions from the web search dataset.

*Definition (**Session-based Encoding**).* A user's web search session is governed by a consistent underlying search intent, which can be used to infer the semantics of each query. For example, from a user's search session with 4 queries: *"New York"* ⟶ *"New York Metro Station"* ⟶ *"New York sightseeing"* ⟶ *"Times Square"*, we can infer that *New York* is a city (possibly a sightseeing city), and that *Times Square* is one of the sightseeing spots in *New York*, without having prior knowledge on these individual queries. The idea of *Session-based Encoding* is that by modeling a vast number of search sessions, we are able to extract the latent semantics of each query without having any prior knowledge about them. Sections 3.2.2 and 3.2.3 introduce an RNN based method called *Session-based Query Encoding* (SQE) that generates representations of queries $x \in \mathbb{R}^d$ using web search sessions.

*Definition (**Evacuation Decision**).* During the disaster, each user decides whether or not to evacuate from his/her home location to a shelter or other locations. We define an evacuation decision as a binary variable ($y = 1$ if evacuated, $y = 0$ otherwise). The ground truth of this binary variable is estimated using anomaly detection on location information, explained in Section 3.3.1.

*Problem Definition (**Evacuation Decision Prediction**).* Our task is to, for each user i, predict evacuation decision y_i based on his/her user representation x_i generated by the *Session-based Query Encoding* (SQE) model using web search queries observed prior to the disaster alert. We test multiple methods to generate x_i using the user's web search query data, in Section 4.

3.2 Session-based Query Encoding

We design our model so that it can produce query representations that reflect the search intent of the users within a search session. The models are trained by predicting the next query given a query in the search session. This way, the models can learn representations via the consistent underlying search intent of different queries, since the search intent usually stays the same within a session. We name this type of models as "session-based query encoders" (SQE) due to this key characteristic. Further, we build 2 types of SQE for different types of inputs; an SQE for an input with a single query ("*Session-based Single Query Encoder* (SSQE)") that has an LSTM RNN structure, and an SQE for an input with multiple queries ("*Session-based Multiple Query Encoder* (SMQE)"), that has a hierarchical LSTM RNN structure.

3.2.1 Dataset Preparation. First, we extracted web search records from Yahoo Japan's web search service, from randomly determined 75 three-hour blocks between January 2015 and July 2018. Each record has three attributes: a user's unique ID, query text, and the time-stamp. Queries searched by the same user were then grouped into sessions, using two minutes as a threshold for the session timeout. While the standard threshold is 30 minutes, we used a significantly shorter threshold to ensure that the search intent within each session is consistent. Sessions with only one query were discarded, as they are not applicable for next query prediction. Also, sessions with more than ten queries were truncated after the tenth query to prevent learning from exceptionally long sessions. As a result, we obtained 301M sessions containing 804M queries. We put 10,000 sessions aside for validation, 20,000 for testing and the rest for training. We call this the "*query session dataset*". We also constructed the "*query pair dataset*" by extracting consecutive query pairs from sessions. This dataset contains 503M, 16,694, and 33,166 pairs in the training, validation, and test set, respectively.

3.2.2 Session-based Single Query Encoder (SSQE). We used a character-based 3-layer LSTM RNN [8] as the main component of our SSQE model to encode a query, implementing it with LSTM formulation given by Graves [7]. The size of the character embedding is 256, the hidden layer size of LSTM is 1024, and a fully-connected layer producing the final 128-dimension query representation is attached to the last timestep of the top-layer of the LSTM block. The size of the character vocabulary is 6000, which is large enough to include all the Japanese characters for daily use defined in the government guideline. In total, the model has 26M parameters. We trained it using the *query pair dataset*, using cosine similarity as a cost function so that representations of two queries become closer if they belong to an existing pair. More specifically, for a query Q and its next query D in the session, the cosine similarity $R_\Theta(Q, D)$ between queries Q and D is defined as:

$$R_\Theta(Q, D) = \frac{\mathbf{z}_Q^{\mathrm{T}} \mathbf{z}_D}{\|\mathbf{z}_Q\| \|\mathbf{z}_D\|}, \tag{1}$$

where $\boldsymbol{\Theta}$ denotes model parameters in the encoder which generates 128-dimensional representations $\mathbf{z}_Q$ and $\mathbf{z}_D$ for the two queries Q and D, respectively [9, 19].

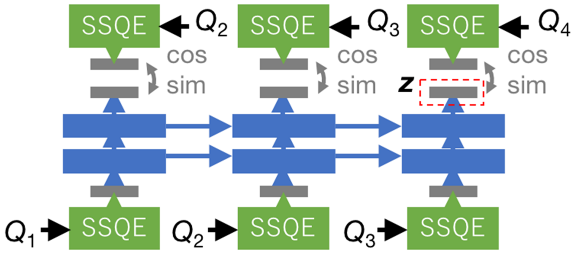

Figure 4: SMQE processing a four-query session in the training phase. The model outputs the representation z of the query session.

To learn representations so that the representations of a query pair comes close to each other, we consider probability $P_\Theta(D|Q)$ for query pair Q and D, where there are five choices $\{D^1, \ldots, D^5\}$:

$$P_\Theta(D_i^k|Q_i) = \frac{\exp(\beta R_\Theta(Q_i, D_i^k))}{\sum_{j=1}^{5} \exp(\beta R_\Theta(Q_i, D_i^j))}, \tag{2}$$

which is obtained by feeding values of $R_\Theta(Q_i, D_i^k)$ with $k = 1, \ldots, 5$ into the softmax function. The index i denotes that the pair (Q_i, D_i^1) is the ith record in a dataset. We set the correct query pair as $k = 1$ (D_i^1), and randomly picked negative samples $\{D_i^2, D_i^3, D_i^4, D_i^5\}$ during training. The inverse temperature coefficient β was set to 10 to make the cross entropy loss large enough. The correct query choice among the five D^k is $k = 1$, so the cross entropy loss l for the ith query pair can be written as

$$l_\Theta(Q_i, D_i^1) = -\log P_\Theta(D_i^1|Q_i). \tag{3}$$

Then, we find the optimal parameters $\hat{\boldsymbol{\Theta}}$ by minimizing the total loss L for all the query pairs in the dataset:

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}}\, L_\Theta = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}} \sum_i l_\Theta(Q_i, D_i^1). \tag{4}$$

3.2.3 Session-based Multiple Query Encoder (SMQE). To encode multiple queries at once, we used a hierarchical architecture [27] in which the SSQE and another 2-layer LSTM RNN were combined to build the SMQE. The hidden layer size of the 2-layer LSTM is 1024, and it is also equipped with a fully-connected layer to produce a 128-dimension representation (z) of a given query sequence. Figure 4 illustrates the SMQE processing a four-query session $\{Q_1, Q_2, Q_3, Q_4\}$ in the training phase. In the inference phase, the vector z obtained at the final time step is used as the representation of the three queries $\{Q_1, Q_2, Q_3\}$. In total, the SMQE model has 47M parameters.

The *query session dataset* is used for training this model. In the training phase, when a sequence of queries is given to the model, the SSQE reads each query, and the 1024-dimension hidden representation at the last time step of the SSQE's top layer of each query is fed to the 2-layer LSTM RNN one by one. The combined model produces a 128-dimensional vector (z) at each time step of the query sequence. The generated representations, which capture the context, are compared with those generated by the SSQE for the next queries, and the closeness of each representation are evaluated by using cosine similarity. In a nutshell, similar to SSQE, SMQE learns representations by next query prediction, but unlike SSQE, it can capture the context of the current query session. This

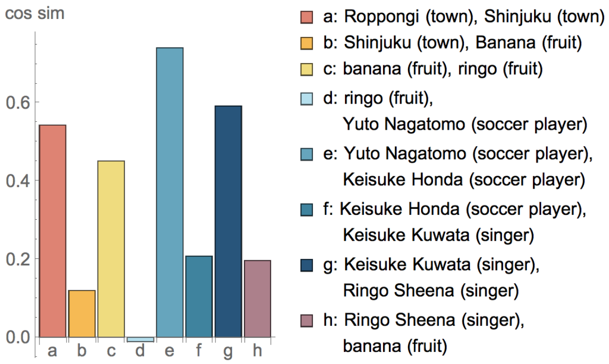

Figure 5: Cosine similarity values for various query pairs.

training process makes it possible for SMQE to generate a contextual representation.

3.2.4 Training. SSQE was trained using Adam [13] for 700M iterations with a batch size 96, over the *query pair dataset.* The accuracy for the aforementioned 5-class classification using the validation set was 93.4%. Using SSQE with the best validation performance as the first stage of the hierarchical model, we trained SMQE using Adam for 500M iterations with the batch size 64, over the *query session dataset.* In this setting, the classification accuracy reached 94.1%, which is higher than that of SSQE. To the best of our knowledge, none of the previous models trained under non-contextual settings have exceeded 0.94 in accuracy for the next query prediction task.

3.2.5 Examples of Similarity Between Query Words. Figure 5 plots the cosine similarity values between various pairs of query representations. We can confirm that the similarity values are relatively high for query words from the same category (a,c,e,g), but are relatively low for those from different categories (b,d,f). SSQE is able to discriminate similar but semantically different word pairs as well. Bar plot (h) shows that although the singer's name "Ringo Sheena" includes a fruit "ringo (= apple)" and is potentially confusing, the model is able to correctly distinguish between the singer and a fruit.

3.3 Evacuation Detection using Location Data

To validate the performance of our predictive model, we used location information collected from the users affected by the disaster. In reality, as explained in the introduction, location information are becoming more difficult to utilize in real time settings due to privacy concerns. In this study, location information were collected and used only in the validation phase, but not for predicting evacuation.

3.3.1 Anomaly Detection. This section explains the anomaly detection methods used to assign the label indicating evacuation or not ($y_i = \{1, 0\}$) for each user using their location information. In a nutshell, the anomaly detection method assigns an anomaly score to each user based on the deviation of the users' post-disaster mobility patterns from his/her usual (pre-diaster) mobility patterns. Let us denote the learning period as $t_{learn} \in [t_0, t_l)$, pre-disaster period as $t_{pre} \in [t_l, t_d)$, and disaster period as $t_{dis} \in [t_d, t_e]$. t_d is the timing when the evacuation alert was sent out. We estimated the spatial probability density of each user using data observed in t_{learn}, and then we compared anomaly scores of observations in t_{pre} and t_{dis} to assign labels y. If the anomaly score was higher during t_{dis} than t_{pre}, this would indicate that the individual moved irregularly during the disaster (i.e. evacuation).

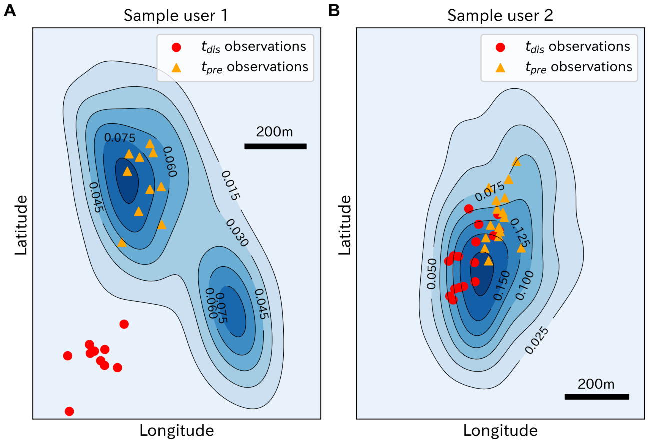

Figure 6: Mobility anomaly detection outputs of 2 sample users. $\hat{p}_i(\mathbf{x}|d,t)$ (blue contours), observations before disaster (orange plots) and during disaster (red plots) are shown. Longitude and latitude values are hidden to protect users' privacy. A) User 1 was observed at unusual locations during the disaster, thus would receive a high anomaly score θ and label $y = 1$ (indicating evacuation). B) User 2 was observed at usual locations during the disaster, thus would receive a low anomaly score θ and label $y = 0$ (indicating no evacuation).

Let us denote the set of observed location data of user i as $O^i = \{o_1^i, o_2^i, \cdots, o_N^i\}$. Each observation is a 4-tuple of $o_n^i = \{i, t_n, x_n, y_n\}$ where t_n, x_n and y_n are the timestamp, longitude and latitude of the n-th observation of user i, respectively. Using location data observed during t_{learn}, we constructed the spatial probability density ($\hat{p}_i(\mathbf{x}|d,h)$) of user i conditional on the day of week and time of day. We assumed that each observation has a 2-D Gaussian probability density [3]. We estimated the spatio probability density $\hat{p}_i(\mathbf{x}|d,h)$ of an observation $\mathbf{x} = (x, y, t)$ of user i conditional on day of week d and time of day h (hour) using multivariate Kernel density estimation with a Gaussian kernel, $K \sim \mathcal{N}(0, \Sigma)$:

$$\hat{p}_i(\mathbf{x}|d,h) = \frac{1}{N(J)(2\pi)^{\frac{3}{2}}|\Sigma|^{\frac{1}{2}}} \sum_{j \in J} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mathbf{x}_j)^T \Sigma^{-1} (\mathbf{x}-\mathbf{x}_j)\right\} \quad (5)$$

where $J = \{j \mid t_j \in t_{learn},\ d(t_j) = d,\ h(t_j) = h\}$ is a set of tags of observations used for density estimation, $N(J)$ is the number of tags in set J, $\Sigma = diag[\sigma_x^2, \sigma_y^2, \sigma_t^2]$, and $d(t)$ and $h(t)$ denote the day of week and hour of timestamp t, respectively. Then, for each user, the mean p_{pre}^i and variance $(s_{pre}^i)^2$ of probability values during the pre-disaster period and the mean probability value during the disaster period p_{dis}^i are:

$$p_{pre}^i = \mathbb{E}[\hat{p}_i(\mathbf{x}|d,h)|\ t \in t_{pre}] \quad (6)$$

$$(s_{pre}^i)^2 = Var[\hat{p}_i(\mathbf{x}|d,h)|\ t \in t_{pre}] \quad (7)$$

$$p_{dis}^i = \mathbb{E}[\hat{p}_i(\mathbf{x}|d,h)|\ t \in t_{dis}] \quad (8)$$

The anomaly score θ_i of individual i was calculated by $\theta_i = \frac{p^i_{pre} - p^i_{dis}}{s^i_{pre}}$. Finally, we labeled users with high anomaly scores $\theta_i > \tilde{\theta}$ as $y_i = 1$ (evacuated), and users with low anomaly scores $|\theta_i| < \tilde{\theta}_l$ as $y_i = 0$ (not evacuated). $\tilde{\theta}$ and $\tilde{\theta}_l$ are threshold parameters used for labelling users. All parameter values are specified in Section 4.1.

3.3.2 Example Outputs. Figure 6 illustrates how the anomaly detection method works using location data of 2 sample users. In each panel, the contour plots show the estimated spatial probability density $\hat{p}_i(\mathbf{x}|d, h)$, orange points indicate observations before the disaster (t_{pre}), and red points indicate the observations during the disaster (t_{dis}). The longitude and latitude values are hidden to protect the users' privacy. The values of Σ that were used for the multivariate kernel density estimation were $\sigma_x = \sigma_y = 100$ (m) and $\sigma_t = 1$ (hour). We can clearly observe that sample user 1 was observed before the disaster in his usual location. However, he was observed at very unlikely locations during the disaster, implying that this user evacuated to a safer location (e.g. evacuation shelter). Thus, this user will receive a high anomaly score θ. In contrast, we can observe that sample user 2 was observed near the usual location even during the disaster, implying that this user did not evacuate. Thus, this user will receive a low anomaly score θ.

4 EXPERIMENTAL VALIDATION

4.1 Experiment Setup

We tested our proposed method using data collected from the 2018 Japan Floods, which resulted in widespread devastating floods and mudflows across Japan from mid-June to mid-July in 2018. More than 220 people were confirmed dead across 15 prefectures, and more than 8 million people were advised or urged to evacuate across 23 prefectures[2]. In particular, Okayama and Hiroshima prefectures experienced severe flooding. Evacuation preparation notifications were sent out on July 6th 10PM, and the evacuation alert was issued on July 7th 1:30AM. The riverbank was overwhelmed by water 5 hours after the evacuation alert, at 6:52AM of July 7th.

4.1.1 Parameter Settings. The time parameters defined in Subsection 3.3.1 were set to t_0 = 2018/6/1 12:00AM, t_l = 2018/7/1 12:00AM, t_d = 2018/7/7 1:30AM, and t_e = 2018/7/10 12:00AM. We only used web query data collected before the evacuation alert t_d to generate user representations with our SQE model. Parameters for the anomaly detection algorithm were set to $\sigma_x = \sigma_y = 100$ (m) and $\sigma_t = 1$ (hour). σ_x and σ_y values reflect the spatial error of the location information observations. Past works have shown that average spatial errors of location data obtained from smartphones are typically large (several 100 meters) especially when users are located indoor [18]. Also, σ_t was set to 1 hour to cover sparsely observed time periods (e.g. nighttime). Figure 7 shows the histogram of anomaly scores θ_i of individual users. We observe higher density with high anomaly scores on disaster day (2018/7/7) compared to a usual day (2018/7/4). We chose the value of $\tilde{\theta}$ using results from a survey[3] carried out by the Hiroshima Business and Management School, which found that only 4.6% of the affected people evacuated during the 2018 Japan flood. We set the threshold to $\tilde{\theta} = 4$ so that the percentage of users with anomaly scores exceeding the threshold is close to the reported evacuation percentage. The threshold for detecting "usual" mobility patterns was set to $\tilde{\theta}_l = 1$ because anomaly scores are usually within $|\theta| < 1$ on a usual day (Figure 7). The predictability of evacuation decisions under different parameter values of $\tilde{\theta}$ are tested in Section 4.2.

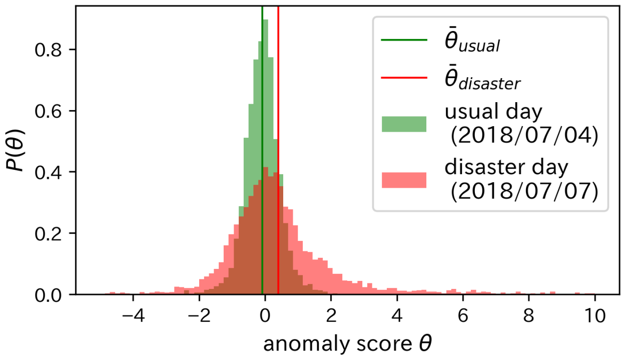

Figure 7: Histogram of anomaly scores on a usual day (green) and the disaster day (red).

Because the problem is a binary classification task ($y_i \in \{0, 1\}$), we used prediction accuracy and area under curve (AUC) as metrics for predictive performances. To prevent overfitting and to ensure that our results are statistically significant, we performed a cross validation with $k = 5$ folds.

4.1.2 Comparative Methods. The following methods for generating user representations were tested.

Input Query Selection. We selected input queries used for feature generation based on 2 criteria. First, we tested selecting only a single query as input for SSQE and selecting multiple (max. 10) queries as input for SMQE. Second, we tested selecting the most recent query (queries) and selecting high-importance queries defined by tf-idf values. Thus, we tested a total of 4 combinations ([Single, Multiple]×[Recent, tf-idf]) for selecting the set of input queries.

Feature Generation. Using the set of input queries, we generated features using SQE (SSQE or SMQE, depending on the number of input queries). To evaluate the predictive performance using the user representations, we compared performances with the case using one-hot encoding of queries (e.g. "cat"=[1,0,...,0], "dog"=[0,1,0,...,0]). Thus, combined with the 4 input query selection methods, we tested a total of 8 combinations ([Single, Multiple]×[Recent, tf-idf]×[One-Hot, SQE]) for feature generation.

Classifier. Because our main contribution is to show how well user representations generated from web search queries can predict evacuation decisions, the classifier we use to make predictions is not the focus of this paper. Thus, we used random forest (RF) which is a standard machine learning model for classification tasks. To evaluate the performances of each feature generation method fairly, we used default parameters for RF, with 100 trees and Gini impurity to measure split quality. For more detail, please see scikit-learn[4].

[2] https://www.cnn.com/2018/07/10/asia/japan-floods-intl/index.html
[3] https://mba.pu-hiroshima.ac.jp/pdf/h30/180901a.pdf (in Japanese)
[4] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Table 1: Prediction accuracy of evacuation decisions using different methods for generating features from pre-disaster web search data.

Method	Input Queries		Feature	**Accuracy**	**AUC**
	# words	Metric	Generation		
1	Single	Recent	One-hot	0.571	0.590
2	Single	Recent	SSQE	0.787	0.812
3	Single	tf-idf	One-hot	0.733	0.600
4	Single	tf-idf	SSQE	0.814	0.833
5	Multiple	Recent	One-Hot	0.689	0.637
6	Multiple	Recent	SMQE	0.756	0.748
7	Multiple	tf-idf	One-Hot	0.759	0.678
8	**Multiple**	**tf-idf**	**SMQE**	**0.840**	**0.837**

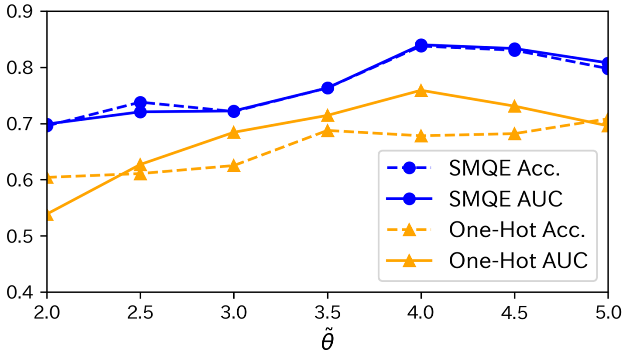

Figure 8: Sensitivity of prediction scores (accuracy, AUC) with respect to different values for $\tilde{\theta}$.

4.2 Results

4.2.1 Prediction Accuracy. Table 1 reports the prediction accuracy of evacuation decisions using different methods for feature generation. SSQE (Session-based Single Query Encoder) was used for single query inputs, and SMQE (Session-based Multiple Query Encoder) was used for multiple query inputs. Out of all 8 combinations of feature generation methods, Method 8, which used SMQE to generate representations of multiple search queries selected by tf-idf scores had the highest prediction accuracy of 84%. The AUC score was also highest for this method, showing the robustness of this result. This result can be interpreted as the following: given the web search queries of Yahoo Japan users from prior to the disaster, we are able to predict whether or not that user will evacuate during the disaster with 84% accuracy.

Most importantly, we can observe that using SMQE to generate features from input queries significantly improves the predictive accuracy over using One-Hot vectors (×110.7% in accuracy and ×123.4% in AUC). This is mainly due to the characteristic of query representations that can capture subtle differences between the queries, as discussed in more detail in subsection 4.2.2. Prediction scores of SMQE were higher than SSQE in all cases (Method 2 vs 6, 4 vs 8), which implies that increasing the number of query words used to generate representations also improved the accuracy of evacuation prediction. It was counter-intuitive to observe that rather than using queries searched by the users just before the disaster (“recent”), using queries with high tf-idf values had higher predictability when other settings were identical (Method 1 vs 3, 2 vs 4, 5 vs 7, 6 vs 8). Intuitively, we may hypothesize that users tend to search about things related to evacuation more as the situation becomes worse (and gets closer to evacuation timing). However, this result implies that our intuition is incorrect, and that selecting the input queries based on importance (e.g. tf-idf) enables higher predictability.

Figure 8 shows the prediction scores of Methods 7 and 8 under various values of the threshold parameter $\tilde{\theta}$. For all $\tilde{\theta}$ values, Method 8 had higher accuracy and AUC compared to Method 7, clarifying the significant improvement of using SMQE over One-Hot encoding. With smaller $\tilde{\theta}$, the confidence of the anomaly detection algorithm would decrease, meaning that we are less certain on whether the user did or did not evacuate. Thus, the general trend that the prediction accuracy decreases as we decrease $\tilde{\theta}$ was expected. However, the prediction accuracy of Method 8 stayed high (AUC = 0.7) even when $\tilde{\theta} = 2.0$, while the AUC of Method 7 dropped to 0.53. This shows that our method is capable of classifying users even when the observed post-disaster mobility patterns were unclear, due to sparse and/or noisy observations.

4.2.2 Analysis of Representations. In addition to showing the high predictability of our proposed method, looking into how the models treated the input queries enables us to understand how the high predictability was achieved. When we compare prediction results between Method 7 and Method 8, queries such as “manga cafe” (a cafe-like space where you can read comics and also stay the night) and “all-night restaurant” (restaurants where you can stay the night) were correctly used to predict evacuation decisions using Method 8, because the SMQE model generated representations close to evacuation shelters and hotels. In addition, because of the accurate representations, Method 8 was able to prevent false predictions compared to Method 7. For example, Method 8 was able to classify specific (local) and sparsely searched location names (“Suna-gawa”, “Kumano-cho”, “Tamashima”) correctly compared to Method 7. Method 7 could not classify such kind of subtle variances of location names, and was limited to encoding names of large districts that were searched frequently (e.g. “Kurashiki City”, “Okayama prefecture”). These examples show how SMQE was able to generate representations of queries so that subtle differences in vocabulary and names of places were captured, which was not possible using one-hot vector representations.

5 DISCUSSION

Our experimental results using real world data confirmed that evacuation decisions are highly predictable by utilizing the users’ web search behavior observed prior to the disaster. This method proposes an alternative method to predicting evacuation behavior, which overcomes the drawbacks of depending on sensitive real time location data.

Now, we discuss future research opportunities that this study enables. First, the data we could use to test and validate our method was limited to one disaster event (2018 Japan floods). Once we start collecting additional data from future disasters, testing the generalizability of our method between different disaster events (future floods) will be our focus of future research. Moreover, testing whether a model pre-trained using data from a given disaster would work on other types of disasters would be an ambitious but very interesting research question to investigate. Even with our limitation on data in this study, we were excited to see that web search behavior can indeed predict evacuation decisions.

Second, in this study, we did not use mobility data as inputs for prediction. This was because we wanted to investigate the predictability using only the web search behaviors, and also assumed the situation where we are not able to use location data at all due to privacy issues. As shown in previous studies, using pre-disaster (usual) mobility patterns is known to improve the accuracy of post-disaster behavior [35]. We are also interested in building a framework that can integrate both pre-disaster physical and web search behavior for post-disaster evacuation prediction.

Thirdly, we found that evacuation decisions can be predicted with high accuracy using just the web search behavior. This motivates us to predict evacuation destinations from web search behavior as well, although it is known that most of the evacuees head to shelters or familiar locations such as houses of family and friends [15]. This exciting extension to our current work would require prior knowledge on where the users typically visit using location data, thus would be an additional topic to our second point.

Finally, we touch upon the industrial applications that will be developed as a product of this research. In contrast to most existing studies, evacuation decisions were predicted based on web search behavior representations generated from data observed *before* the evacuation alert, using web search data which have less concerns on privacy issues compared to location information. This method is feasible in real world settings, by collecting web search data and making predictions in real time, before the disaster alerts are sent out. Using this output, we could support decision making of local governments by providing predictions of how many people will evacuate prior to the disaster. This information could help policy makers to organize their strategies on the allocation of shelters and emergency supplies. Moreover, towards the individual users, we could provide informational support, such as notifications to their smartphones about information of nearby evacuation shelters, road closures, and dangerous areas to avoid.

6 RELATED WORKS

6.1 Human Mobility Analysis during Disasters

Traditionally, surveys have been used to understand evacuation decisions after disasters [16, 21, 22]. Recently, large scale datasets are being utilized to understand the behavior of individuals [2, 6, 36]. Lu et al. analyzed call detail records to investigate the predictability of evacuation destinations after the Haiti earthquake [15]. Song et al. revealed the population decline in various areas after the Great East Japan Earthquake using GPS data [23]. More recent studies showed that evacuation behavior could be monitored after an earthquake by using mobile phone location data collected from evacuees [31–33].

In addition to the literature on post-disaster analysis of evacuation behavior using big data sources, there have been various works that have attempted to predict mobility patterns using real time information. Fan et al. and Sudo et al. use real time location data to predict human mobility dynamics in an online manner [5, 28]. Other works using machine learning models have been proposed for real time prediction [24–26]. More recently, Jiang et al. proposed `DeepUrbanMomentum`, a deep-learning architecture that models the human mobility dynamics [10]. Although these frameworks have been shown to be effective in predicting short term future mobility patterns, they require real time location information, which are becoming harder to obtain due to rising privacy concerns. Thus, we take an alternative approach and attempt to utilize other data sources (web search behavior) to predict evacuation mobility patterns. A closely related work was performed by Konishi et al., which used transit app queries to predict future destinations of people [14]. In their study, input data were names of stations in the transit system. Our study applies a similar idea but with web query data, which do not directly reveal the users' destinations. Web queries contain latent meanings and subtle differences in expressions, which makes our prediction task much more harder.

6.2 Representation Learning of Web Search Behavior

The main component of this work is to understand users' web search behavior using a model that can encode sessions of search queries into representations that reflect their latent meanings and underlying search intent of the user. There have been many studies on such encoders for words and texts. Some of the popular approaches are word embedding methods such as `word2vec` [17], `FastText` [11], and `GloVe` [20]. While these methods provide relatively low-cost, easy-to-use ways to generate word vectors, it is not clear how to combine multiple word representations, for example embedding phrases.

Also, since these methods directly assign vectors to a word (or possibly a query itself when the target domain is search queries), it is difficult to handle the tail part of the query sequence that contains too many words. In our method, we avoid this difficulty by applying LSTM models to the character sequence and by generating the representation compositionally. One of the more closer attempts to develop such encoder is `Query2Vec` [12]. While it has a similarity with our method in the usage of the search query data and the user's web search behavior, our method handles the query's text representation differently. In `Query2Vec`, a dense vector is assigned to each word or query, similar to the word embedding models, thus suffers from the same aforementioned drawbacks. Also, while `Query2Vec` can produce representations of words and queries, it is unable to produce representations of sessions of queries.

7 CONCLUSION

Despite the importance of predicting evacuation activities, most works have used real time location information, which are becoming increasingly difficult to obtain in reality. In this paper, we bridge this gap by proposing a framework that utilizes web search queries of users to predict post-disaster evacuation decisions. Through experiments using real world data, it was found that evacuation decisions are highly predictable by learning the user representations from web search queries. This study opens up a new avenue of research on the prediction of mobility using web query data, and encourages further studies using various similar datasets. Moreover, this study proposes an alternative method for predicting evacuation prior to the disaster, which has huge potential for applications in real world disaster situations.

ACKNOWLEDGEMENTS

T.Y. is partly supported by the Doctoral Fellowship provided by the Purdue Systems Collaboratory. The work of T.Y. and S. V. U. is partly funded by NSF Grant No. 1638311 CRISP Type 2/Collaborative Research: Critical Transitions in the Resilience and Recovery of Interdependent Social and Physical Networks.

REFERENCES

[1] James P Bagrow, Dashun Wang, and Albert-Laszlo Barabasi. 2011. Collective response of human populations to large-scale emergencies. *PloS one* 6, 3 (2011), e17680.

[2] Francesco Calabrese, Giusy Di Lorenzo, Liang Liu, and Carlo Ratti. 2011. Estimating Origin-Destination flows using opportunistically collected mobile phone location data from one million users in Boston Metropolitan Area. *IEEE Pervasive Computing* 10, 4 (2011), 36–44.

[3] Eunjoon Cho, Seth A Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 1082–1090.

[4] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. 2013. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports* 3 (2013), 1376.

[5] Zipei Fan, Xuan Song, Ryosuke Shibasaki, and Ryutaro Adachi. 2015. CityMomentum: an online approach for crowd behavior prediction at a citywide level. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing.* ACM, 559–569.

[6] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. 2008. Understanding individual human mobility patterns. *nature* 453, 7196 (2008), 779.

[7] Alex Graves. 2013. Generating Sequences With Recurrent Neural Networks. *CoRR* abs/1308.0850 (2013).

[8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[9] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management (CIKM '13).* ACM, New York, NY, USA, 2333–2338. https://doi.org/10.1145/2505515.2505665

[10] Renhe Jiang, Xuan Song, Zipei Fan, Tianqi Xia, Quanjun Chen, Satoshi Miyazawa, and Ryosuke Shibasaki. 2018. DeepUrbanMomentum: An Online Deep-Learning System for Short-Term Urban Mobility Prediction.. In *AAAI.*

[11] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. FastText.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651* (2016).

[12] Dongyeop Kang and Inho Kang. 2015. *Query2Vec: Learning Deep Intentions from Heterogeneous Search Logs.* Technical Report. Technical report, Naver Labs.

[13] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). http://arxiv.org/abs/1412.6980

[14] Tatsuya Konishi, Mikiya Maruyama, Kota Tsubouchi, and Masamichi Shimosaka. 2016. CityProphet: city-scale irregularity prediction using transit app logs. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing.* ACM, 752–757.

[15] Xin Lu, Linus Bengtsson, and Petter Holme. 2012. Predictability of population displacement after the 2010 Haiti earthquake. *Proceedings of the National Academy of Sciences* 109, 29 (2012), 11576–11581.

[16] Rodrigo Mesa-Arango, Samiul Hasan, Satish V Ukkusuri, and Pamela Murray-Tuite. 2012. Household-level model for hurricane evacuation destination type choice using hurricane Ivan data. *Natural hazards review* 14, 1 (2012), 11–20.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems.* 3111–3119.

[18] Kentaro Nishi, Kota Tsubouchi, and Masamichi Shimosaka. 2014. Hourly pedestrian population trends estimation using location data from smartphones dealing with temporal and spatial sparsity. In *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.* ACM, 281–290.

[19] Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab K. Ward. 2014. Semantic Modelling with Long-Short-Term Memory for Information Retrieval. *CoRR* abs/1412.6629 (2014). http://arxiv.org/abs/1412.6629

[20] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *In EMNLP.*

[21] Arif Mohaimin Sadri, Satish V Ukkusuri, and Hugh Gladwin. 2017. Modeling joint evacuation decisions in social networks: The case of Hurricane Sandy. *Journal of choice modelling* 25 (2017), 50–60.

[22] Arif Mohaimin Sadri, Satish V Ukkusuri, Seungyoon Lee, Rosalee Clawson, Daniel Aldrich, Megan Sapp Nelson, Justin Seipel, and Daniel Kelly. 2018. The role of social capital, personal networks, and emergency responders in post-disaster recovery and resilience: a study of rural communities in Indiana. *Natural Hazards* 90, 3 (2018), 1377–1406.

[23] Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, Teerayut Horanont, Satoshi Ueyama, and Ryosuke Shibasaki. 2013. Intelligent system for human behavior analysis and reasoning following large-scale disasters. *IEEE Intelligent Systems* 28, 4 (2013), 35–42.

[24] Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, Teerayut Horanont, Satoshi Ueyama, and Ryosuke Shibasaki. 2013. Modeling and probabilistic reasoning of population evacuation during large-scale disaster. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 1231–1239.

[25] Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, and Ryosuke Shibasaki. 2014. Prediction of human emergency behavior and their mobility following large-scale disaster. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 5–14.

[26] Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, Ryosuke Shibasaki, Nicholas Jing Yuan, and Xing Xie. 2017. Prediction and simulation of human mobility following natural disasters. *ACM Transactions on Intelligent Systems and Technology (TIST)* 8, 2 (2017), 29.

[27] Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A Hierarchical Recurrent Encoder-Decoder for Generative Context-Aware Query Suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM '15).* ACM, New York, NY, USA, 553–562. https://doi.org/10.1145/2806416.2806493

[28] Akihito Sudo, Takehiro Kashiyama, Takahiro Yabe, Hiroshi Kanasugi, Xuan Song, Tomoyuki Higuchi, Shin'ya Nakano, Masaya Saito, and Yoshihide Sekimoto. 2016. Particle filter for real-time human mobility prediction following unprecedented disaster. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.* ACM, 5.

[29] WMO UNISDR. 2012. Disaster risk and resilience. *Thematic Think Piece, UN System Task Force on the Post-2015 UN Development Agenda* (2012).

[30] Qi Wang and John E Taylor. 2014. Quantifying human mobility perturbation and resilience in Hurricane Sandy. *PLoS one* 9, 11 (2014), e112608.

[31] Robin Wilson, Elisabeth zu Erbach-Schoenberg, Maximilian Albert, Daniel Power, Simon Tudge, Miguel Gonzalez, Sam Guthrie, Heather Chamberlain, Christopher Brooks, Christopher Hughes, et al. 2016. Rapid and near real-time assessments of population displacement using mobile phone data following disasters: the 2015 Nepal Earthquake. *PLoS currents* 8 (2016).

[32] Takahiro Yabe, Yoshihide Sekimoto, Kota Tsubouchi, and Satoshi Ikemoto. 2019. Cross-comparative analysis of evacuation behavior after earthquakes using mobile phone data. *PLoS one* 14, 2 (2019), e0211375.

[33] Takahiro Yabe, Kota Tsubouchi, Akihito Sudo, and Yoshihide Sekimoto. 2016. Estimating Evacuation Hotspots using GPS data: What happened after the large earthquakes in Kumamoto, Japan. In *Proc. of the 5th International Workshop on Urban Computing.*

[34] Takahiro Yabe, Kota Tsubouchi, Akihito Sudo, and Yoshihide Sekimoto. 2016. A framework for evacuation hotspot detection after large scale disasters using location data from smartphones: case study of Kumamoto earthquake. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.* ACM, 44.

[35] Takahiro Yabe, Kota Tsubouchi, Akihito Sudo, and Yoshihide Sekimoto. 2016. Predicting irregular individual movement following frequent mid-level disasters using location data from smartphones. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.* ACM, 54.

[36] Jing Yuan, Yu Zheng, and Xing Xie. 2012. Discovering regions of different functions in a city using human mobility and POIs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 186–194.

[37] Nam Yi Yun and Masanori Hamada. 2015. Evacuation behavior and fatality rate during the 2011 Tohoku-Oki earthquake and tsunami. *Earthquake Spectra* 31, 3 (2015), 1237–1265.

Supplementary Material

1 DATASET

In this study, we utilized web search data and location information collected by Yahoo Japan Corporation[1]. To protect the users' privacy, we are not able to make both datasets publicly available. For data requests, please contact Dr. Kota Tsubouchi of Yahoo Japan Corporation (ktsubouc@yahoo-corp.jp).

1.1 Web Search Data

Query Session Dataset. Web search records were extracted from randomly determined 75 three-hour blocks between January 2015 and July 2018. Queries searched by the same user were then grouped into sessions, using two minutes as a threshold for the session timeout. Sessions with only one query were discarded, and sessions with more than ten queries were truncated after the tenth query. The dataset contains 301M sessions with 804M queries. 10,000 sessions were used for validation, 20,000 for testing and the rest for training.

Query Pair Dataset. Consecutive query pairs were extracted from sessions to build this dataset. The dataset contains 503M, 16,694, and 33,166 pairs in the training, validation, and test set, respectively.

User Query Dataset. After the users within the disaster area were identified using location data (explained in detail in next subsection), the queries of each user were collected from the Query Session Dataset. These query data were later on, used as inputs to the SQE models to generate feature vectors for evacuation decision prediction.

1.2 Location Data

Users who are located within the disaster affected area were extracted from the dataset using a bounding box (longitude, latitude) of corner points [132.0,34.0] and [134.4,35.0] which cover the disaster affected area. Then, home locations for each user were estimated using the Mean-Shift algorithm with bandwidth parameter of 0.1 degrees, in Python language using the `scikit-learn` package. Users whose home locations were estimated to be inside the bounding box were used for analysis. In total, location data of around 8700 users were used in this study. Note that these procedures are not necessary for real world applications, since location data will not be used for prediction. Location data was only used for assigning ground truth labels ($y = \{0, 1\}$) used for validation.

2 DESCRIPTION OF SQE MODELS

2.1 Model Structure

2 types of SQE were built for different types of inputs; an SQE for an input with a single query ("*Session-based Single Query Encoder* (SSQE)") that has an LSTM RNN structure, and an SQE for an input with multiple queries ("*Session-based Multiple Query Encoder* (SMQE)"), that has a hierarchical LSTM RNN structure.

The SSQE is a character-based 3-layer LSTM RNN with formulation given by Graves [1]. The size of the character embedding is 256, the hidden layer size of LSTM is 1024, and a fully-connected layer producing the final 128-dimension query representation is attached to the last timestep of the top-layer of the LSTM block. The size of the character vocabulary is 6000. In total, the model has 26M parameters.

The SMQE has a hierarchical architecture in which the SSQE and another 2-layer LSTM RNN were combined[3]. The hidden layer size of the 2-layer LSTM is 1024, and it is also equipped with a fully-connected layer to produce a 128-dimension representation ($\mathbf{z}$) of a given query sequence. In total, the SMQE model has 47M parameters.

2.2 Training Procedures

SSQE was trained using the *query pair dataset*, using cosine similarity as a cost function. Adam was used for 700M iterations with a batch size 96[2]. For each query pair, 4 negative sample queries were randomly selected to calculate the cross entropy loss, which is defined by the negative log of the softmax function (equation (2)) in the main manuscript. The inverse temperature coefficient β was set to 10.

SMQE was trained using Adam for 500M iterations with the batch size 64, over the *query session dataset.* SMQE was built using the SSQE with the best validation performance as the first stage of the hierarchical model. In the training phase, when a sequence of queries is given to the model, the SSQE reads each query, and the 1024-dimension hidden representation at the last time step of the SSQE's top layer of each query is fed to the 2-layer LSTM RNN one by one. The combined model produces a 128-dimensional vector ($\mathbf{z}$) at each time step of the query sequence. The generated representations, which capture the context, are compared with those generated by the SSQE for the next queries, and the closeness of each representation are evaluated by using cosine similarity.

3 DESCRIPTION OF EVACUATION DETECTION ALGORITHM

3.1 Model Parameters

The parameter values of the evacuation detection algorithm are shown in Table 1. The actual procedures of detecting evacuation mobility are described in detail in the main manuscript, thus are omitted here. The algorithms were coded using Java and Python language. The codes are uploaded on Takahiro Yabe's Github page (https://github.com/takayabe0505).

[1] https://about.yahoo.co.jp/info/en/

KDD '19, August 4–8, 2019, Anchorage, AK, USA
2019. ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330697

Table 1: Parameter values of evacuation detection algorithm

Parameter	Value
Σ	$diag[100, 100, 1]$
t_0	2018/6/1 12:00AM
t_l	2018/7/1 12:00AM
t_d	2018/7/7 01:30AM
t_e	2018/7/10 12:00AM
θ_l	1
$\tilde{\theta}$	4 (changed in Section 4.2.1)

4 DETAILS OF EXPERIMENTS

4.1 Comparative Methods

A total of 8 combinations ([Single, Multiple]×[Recent, tf-idf]×[One-Hot, SQE]) for feature generation were tested in our experiment. Most of the methods can be implemented in a straight forward manner using the information in the main manuscript.

"Recent". For "[Single]" cases under the "Recent" setting, the word that was searched at the closest timing to the evacuation alert by the user was selected as input queries and fed into the SSQE or the One-Hot encoder. For "[Multiple]" cases under the "Recent" setting, the 10 words that were searched at the closest timing to the evacuation alert by the user was selected as input queries and fed into the SMQE or the One-Hot encoder.

tf-idf. tf-idf values for query words were calculated using the following equation.

$$\texttt{tf-idf}(i,j) = \texttt{tf}(i,j) \cdot \texttt{idf}(i)$$
$$\texttt{tf}(i,j) = \frac{n_{i,j}}{\sum_K n_{k,j}}$$
$$\texttt{idf}(i) = \log \frac{|D|}{|\{d : t_i \in d\}|}$$

where, $n_{i,j}$ is the frequency of word t_i in document d_j, $\sum_K n_{k,j}$ is the sum of frequency of all words in document d_j, $|D|$ is the total number of documents, and $|\{d : t_i \in d\}|$ is the number of documents that contain word t_i. In our case, a document is a bag of query words searched by a single user, and a word is a query word. For "[Single]" cases, the word with the highest tf-idf value of the user was selected as input queries and fed into the SSQE or the One-Hot encoder. For "[Multiple]" cases, the top 10 words with highest tf-idf values were selected as input queries and fed into the SMQE or the One-Hot encoder.

4.2 Classification

Random forest (RF) was used for classification. RF was implemented using scikit-learn [2] using Python language. The parameters of RF are shown in Table 2. To prevent overfitting and to ensure that our results are statistically significant, we performed a cross validation with $k = 5$ folds, and only report statistically significant accuracy scores with standard deviation less than 5%. To produce Figure 8 of the main manuscript, the same procedures were repeated for different values of $\tilde{\theta}$.

Table 2: Parameter values of Random Forest

Parameter	Value
Number of Trees	100
Criterion	Gini impurity
Maximum depth	None
Minimum samples for a split	2
Minimum samples to be a leaf	1
Maximum features to find best split	$\sqrt{128}$
Random state	300
Class weight	None

REFERENCES

[1] Alex Graves. 2013. Generating Sequences With Recurrent Neural Networks. *CoRR* abs/1308.0850 (2013).

[2] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). http://arxiv.org/abs/1412.6980

[3] Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A Hierarchical Recurrent Encoder-Decoder for Generative Context-Aware Query Suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM '15)*. ACM, New York, NY, USA, 553–562. https://doi.org/10.1145/2806416.2806493

[2] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html