Submitted to *Management Science*
manuscript MS-0001-1922.65

Using Social Media to Enhance the Search for Gasoline During a Hurricane Evacuation Event

(Authors' names blinded for peer review)

Panic-buying and shortages of essential commodities is common during early phases of a disaster or an epidemic. This paper has two goals. The first goal is to develop and analyze an optimization model for an efficient search of an essential commodity; this model needs as input data processed from social media posts. The second goal is to establish that the use of social media posts can significantly improve the search efficiency, based on the analysis from a recent hurricane event. Specific contributions in the data processing of social media posts include the development of a classifier that detects shortages and an event localizer that probabilistically infers the location and time of shortage. Specific contributions in the mathematical model development include an integer programming formulation of the resultant search problem on a graph, with the two objective different objective functions: (a) Maximizing probability of finding the commodity (b) Minimizing expected time to find the commodity given the commodity is found. The first model is solved optimally using CPLEX on a linearization of the non-linear model. For the second model, an approximate solution is found using CPLEX on the linearization and a modified branch and bound method is developed. Encouragingly, the modified branch and bound method reduced computational effort by a factor of ten. The methodology is validated using a case study on gasoline search during the Hurricane Irma evacuations. We found that social media posts can predict shortage at gas station for four major cities of Florida accurately with a MAPE of 12%. We also found that addition of social media information to the search process improved the average search time by 41.74%.

Key words: Disaster management, Social sensing, Search theory, Bayesian models

1. Introduction

Shortage of essential commodities is commonly observed when a disaster or epidemic is announced. For instance, shortage of gasoline was observed prior to the landfall of Hurricane Irma in Florida (2017) and after the landfall of Hurricane Sandy in New York/New Jersey (2012). When the COVID-19 pandemic unfolded in the United States in March 2020, a large number of stores ran out of essential commodities such as hand sanitizers and face masks. These shortages can largely be attributed to people's panic-buying behavior. We have developed a search technique for essential commodities in such a circumstance. Our findings show that including social media information enhances the search process significantly. Our case study is based on gasoline shortage prior to a hurricane event (e.g. Hurricane Irma).

Social media is transforming the way people communicate not only in daily lives, but also during disaster events. There is a surge in usage of social media during an emergency in the affected regions. The public use social media to communicate, seek information, raise concerns and express sentiments, and responders use it to plan and communicate important massages to the public (31, 32, 38, 53). As a result, there is a keen interest in employing social media for disaster management. For example, social media has been used to build a mass communication channel, in order to inform large numbers of stakeholders at once (29, 48, 52). Social media can also aid in decision support systems and emergency management processes by utilizing the enormous amounts of real-time data it generates (22, 14). Consequently, multiple social media data analysis techniques have been developed in the context of a disaster, ranging from tools for event detection, prediction and warning; impact assessment; situation awareness; disaster tracking; and response planning (37).

People use social media to ask for help, raise concern, and express emotions during shortages. As an example, during the gasoline shortage in Florida in the onset of Hurricane Irma, the following kinds of tweets were recorded:

"um so is there anywhere in town that still has gas"
"i m wasting gas driving around trying to find it"
"needed gas lines are crazy irmahurricane"
"4 gas stations later and i finally got gas"

The geo-location and timestamps of these tweets provide (probabilistic) information of the spatio-temporal distribution of the shortage. Gas stations near the location of the tweet have a higher probability of being out of gas. Similarly, there is a high probability that the timestamp of tweet is around the time of shortage. A major aim of this paper is to determine if social media data can enhance strategies for people searching essential commodities. Towards this goal we pose three sub-problems:

- **Sub-problem 1–Detecting social media posts related to shortages**: In an earlier paper we have described challenges in detecting social media posts and developed a classification methodology (28). It consists of a SVM classifier that that uses unigrams and latent topics as features to identify tweets about gasoline shortage. We apply this methodology in our paper.

- **Sub-problem 2–Estimating spatio-temporal distribution for shortages**: The spatio-temporal distribution of tweets is not equivalent to the spatio-temporal shortage distribution, due to spatial and temporal lags. We create a methodology to infer the spatial and temporal lags, and thereby infer the location and time of shortages.
- **Sub-problem 3–Optimizing search path**: Information about location and time of shortages is probabilistic. Our search path planning problem is the problem of finding an entity on a graph, given probabilistic information about the entity's location at nodes of the graph.

To solve sub-problem 2, we develop a Bayesian network to model time and spatial lag between observations and postings about shortages. We then calculate the time and location of shortages using inference methods. To solve sub-problem 3, we model the search problem as a mathematical program with the objective of minimizing the expected time to find the commodity. Two solution methods are developed. In the first solution method, we linearize the objective and solve the resultant problem using CPLEX. Our second solution method is a modified branch-and-bound methodology which provides a high quality solution with significantly improved computational efficiency. We illustrate our methods on a case study of the gasoline shortage situation in key Florida cities during Hurricane Irma.

The rest of the paper is organized as follows. Section 2 contains a description of related work. This is followed by a presentation of the details of our data processing, modeling and solution methodology in Section 3. Section 4 presents: (i) numerical comparison of our solution methods, and (ii) a case study based on Hurricane Irma in 2017. Section 5 provides our conclusions and future improvement suggestions.

2. Literature Review

The main contributions of this work are related to geo-location estimation (sub-problem 2) and search theory (sub-problem 3). We therefore confine our literature review to relevant papers in these two topic areas.

2.1. Geo-Location Estimation

Localizing social media posts is imperative to building any application for disaster management. For instance, in a method for responding to tweets about the shortage of supplies like gasoline, location of gas stations without gas must be determined. For search and rescue models using tweets, the location of the SOS message or call for rescue is imperative. In our literature review, we found research for geo-localization of events using social media posts which is related to this problem. Cheng et al. reported that only 0.42% tweets are geo-tagged (16), while Morsattter et al. found that around 3.17% tweets were geo-tagged in their study (36). Even if the tweet is geo-tagged, the location of tweets may not be the location of the event being referred to in the tweet. As a result, there has been a lot of research in localization of tweets and the events they are referring to. We restrict our review to localization of events mentioned from tweets specifically in the context of disasters.

There is a class of methods that tries to infer the location of events by the textual content of the tweets which might contain locations. This is especially applicable for SOS messages for help and rescue which mention the locations of messages. Multiple such tweets were observed during the floods of Hurricane Harvey. There have been studies that have used techniques of "Named Entity Recognition" to recognize locational entities in the tweet content (34, 50). Kumar et al. utilized CNN for this (30). However, in many instances the text of the tweets do not contain location information. Sakaki et al. tried to localize earthquakes using tweets (40). Their method treated each tweet as a sensor for the time and location of the earthquake. They used Kalman and particle filtering methods for spatial inference. For temporal inference they modeled the inter-arrival time between an event and a tweet as an exponential random variable. Singh et al. performed spatial inference in a similar manner by predicting the location of tweets based on historical locations of the users and a Markov model (44). Smith et al. presented a real-time modeling framework to identify areas likely to have flooded (45) using geocoding in tweets combined with simulations from hydrodynamic flooding. Apart from these methods for geo-locating tweets and events pertaining to crisis, there are generic methods for localizing tweets that can be utilized in the context of a disaster (16, 23, 43, 7, 27). All the methods we reviewed have major constraints. Methods that depend on textual content of the posts are not applicable to posts that do not contain any location information (30, 34, 30). Others require the location of the tweets for localizing the event (40, 45). Singh et al. require historical data of the twitter user (44) .

The objective of our application is to identify locations and time of shortage/availability of a commodity like gasoline on the basis of the location and time of the tweet. Localisation method which used named-entity-recognition were not useful for our application as most of the tweets did not have location information in the textual content (34, 50, 30). The method by Sakaki et al. to infer the temporal information of the earthquake event served as the motivation to our methodology (40). They modelled the time between time of the earthquake and the time of the tweet as an exponential random variable. We modeled the time between observation of commodity and a tweet about it as an exponential random variable. Similarly, we also model the distance between observation and tweet as an exponential random variable. Having modeled this, we developed a Bayesian network which inferred the probability of "observed shortage" at different locations at a given time when a tweet about shortage arrived.

2.2. Search Theory

Search problems were originally studied for applications in military operations. They have been used to model problems of locating the enemy/terrorists and missing personnel. It's application has been extended to other areas like astronomy, industries, mineral exploration, disaster management etc. These problems consists of a target and the problems can be classified according to the motives of the target as : (1) One Sided Search, (2) Search Games, and (3) Rendezvous Search. In one sided search problems, the the target is mobile or immobile, has no motives (10, 55, 8, 9, 25). In search games, the target does not want to be found. Search games are studied under two main categories depending on whether the entity is immobile (19, 39, 54) or mobile (24, 18, 17, 33, 5). In rendezvous search, the target wants to be found. In these problems there is a region of search, the player's characteristics and prior agreements, if any, are made about strategy co-ordination (2, 6, 1, 4, 3, 2).

In our search problem, we search for a target/commodity which has the probability of being available at multiple locations with a given probability distribution for each location. The search space is a graph in which The locations of availability are vertices of the graph and the path of a searcher consists of the edges of the graph. This application falls in the category of one sided search as the target is motiveless. In one sided search problems, target is stationary and hidden according to a known/unknown distribution or is mobile and its motion is determined stochastically. In our application, the probability of distribution are determined using information from social media using a Bayesian framework. There are problems in the literature of one sided search which share some common aspects with our application. Earliest problems in this domain were the Linear Search problems (LSP) (10, 55, 8). In these problems, a searcher attempts to locate a randomly placed point x on the real line R according to a known distribution function g(x). The problem was to find the path that minimizes $E[t]$ of finding the target. Beck and Newman modified the LSP by assuming that the probability distribution of the point sought by the searcher on a real line is not known to the searcher (9). In our application, the searcher knows the distribution and tries to minimise expected time. However, the search is on a vertices of a graph and not restricted to a line.

The classical one search models can further be classified into continuous and discrete space and time. Our application is in discrete space and continuous time. In most of these continuous search space problems the search space is divided into grids and the problem is reduced to determining the order of visiting the grids. The grids to be visited are analogous to the vertices of the graph in our application. However, most problems in both continuous and discrete search space have a probability of detection within a search area/grid (15, 46, 56, 35, 42). In our application, if the searcher and the target are at the same position, the searcher will find the object with probability equal to one. The problems can be further classified on the basis of availability of resources. There are problems in the literature in which the searcher has limited time or capacity to move and the objective is to maximise probability of detection in the limited resources (47, 15, 46). In our application for gasoline search, the searcher's capacity to search is limited by the gasoline left in the vehicle. In other applications, the resources are assumed to be unlimited and the objective is to construct a path that covers the entire search space and has a minimal average searching time. For instance, Jotshi and Batta modeled the problem of finding an immobile entity on a graph with a uniformly distribution of finding the entity on the graph's edges while minimising expected search time (25). They expanded the problem to two entities and showed the advantages re-optimisation after updating probabilities (given new information) (26).

Our problem for searching essential commodities in emergency shortages fundamentally differs from all the problems previously described. In all these problems, a target has a given probability distribution of being found in the search area. Sum of probability of finding the entity in different sub-areas is 1 (assuming there probability of detection at each location is 1). The goal is to search all the targets in the search area. In our application, we have multiple copies of the target at different locations with a given probability distribution (for each location). The goal is to find the first target at one of the given locations (even if there might be a target in other locations at other location). Sum of the probability of finding the entity at different sub-areas is not 1. In fact, there is a finite probability of not finding the entity in graph (even if the detection probability is 1). We found few problems like this in the literature.

Berman et al. deals with a person who wants to get service from stationary facilities as fast as possible (11). The facilities are prone to disruption (the facility has a constant probability of becoming inactive) and the person is given the a-priori knowledge of the stochastic behavior of each facility. However, in this problem the chance of the facility to fail is equal in all locations while in our case the probabilities could vary for each location. This makes the Berman's problem a specific case of our problem. Our application is also closely related to the problem of sequential testing of multi-component systems where there is a given test cost and probability of functioning for each component (51). This problem determines the right order of testing all of the system's components in order minimise the total testing cost. Our application is a special case of the multi-component system.

The work by Teller et al. is the closest to our application (49). In their work, "*the searcher does not know the object's exact location, but does know the a priori the probability of finding the object at each location. It wishes to build a searching path for reaching the object that starts from a given location and ends when reaching the object (or after searching the entire set with a false result). The objective is to find a searching path which will minimize the average searching time. They consider two scenarios for this problem: one when there is an unknown number of objects on the set and another when there is exactly one object on the set (the sum of probabilities is equal to 1)*". The first scenario closely describes the requirements of our application. However, their model assumes an unlimited amount of search effort available. In our application, the search is limited by the amount of fuel. Therefore, their model does not directly apply. In their work, they minimise the expected search time. However, we show that given when there is limited capacity to search there are other objectives the searcher might want to consider. Minimising the expected time for finding the target given we find the entity and maximising the probability of finding the target. If the number of vertices to be visited in the graph are limited, the three objectives can result in different paths as described in Section 3.3. In our model, we extend this work by building models which minimise expected time for finding the target and maximising the probability of finding the targets for scenario 1 under availability of limited search effort. We also develop efficient solution for these models.

3. Methodology

Figure 1 illustrates our three-stage methodology for the task of going from posts generated on social media to a path of searching gasoline for a given user. In stage 1, we filter out posts related to gasoline by using keywords and regular expressions, and remove space, stopwords, stemwords from the noisy posts. The tweets that remain after this stage are labeled "gasoline-related" posts. Next, we classify the gasoline-related posts into "gasoline-shortage" posts and "non-gasoline-shortage" posts, using a support vector machine classifier that employs unigrams and latent topics as features. In stage 2, an event localiser builds a Bayesian network using the time and locations of the classified posts and the database of gas-station in the region. The time and location of posts are treated as evidence and using Markov Chain Monte Carlo (MCMC) sampling or variations inference, we infer the posterior probability distributions for time and location of shortage. In stage 3, the probability distributions are input into a mixed integer non-linear program that models the problem of searching the gas on a network of gas stations in minimum time given the inferred probability of shortage at each gas-station. A modified branch-and-bound method is used to solve the problem and provide the optimal search path.

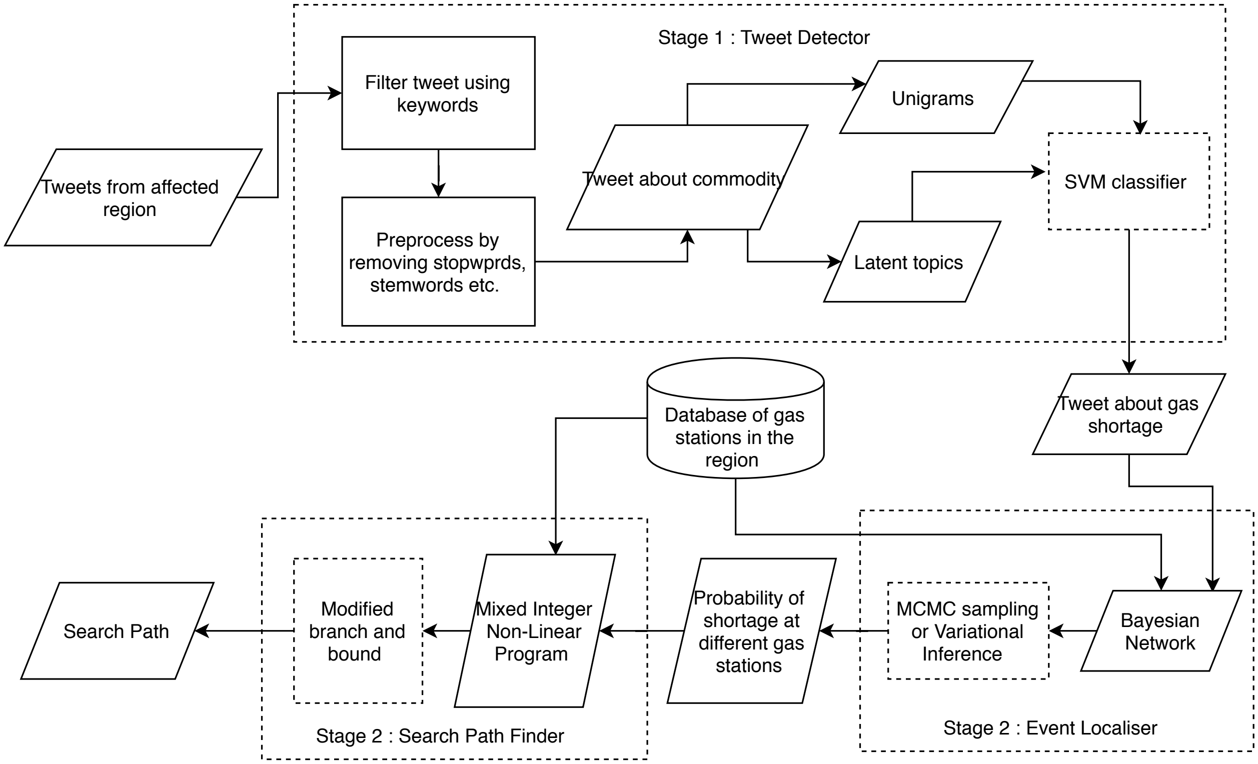

Stage 1 : Tweet Detector
Tweets from affected region
Filter tweet using keywords
Preprocess by removing stopwprds, stemwords etc.
Tweet about commodity
Unigrams
Latent topics
SVM classifier
Tweet about gas shortage
Database of gas stations in the region
Bayesian Network
MCMC sampling or Variational Inference
Stage 2 : Event Localiser
Probability of shortage at different gas stations
Mixed Integer Non-Linear Program
Modified branch and bound
Stage 2 : Search Path Finder
Search Path

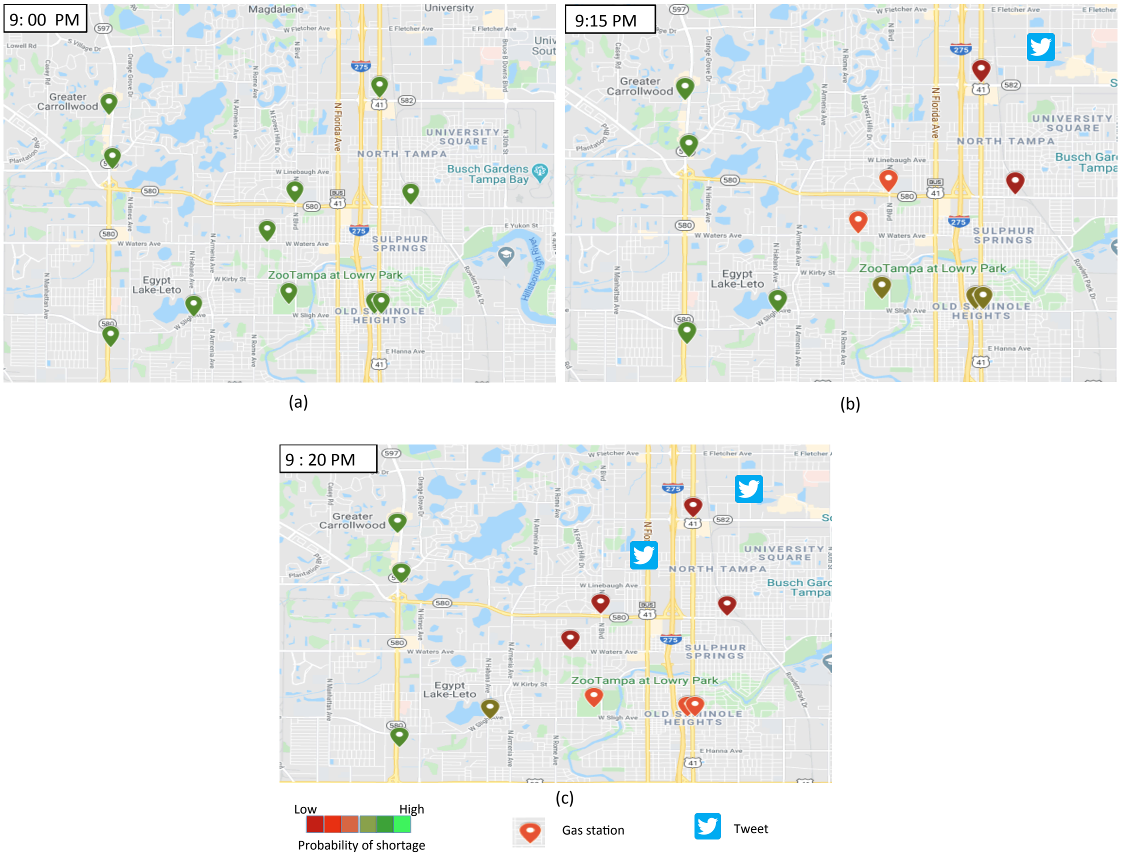
Figure 1 **Schematic of data processing pipeline for the methodology**

3.1. Stage 1: Classification to identify commodity-shortage posts

In Stage 1, first we identify "gasoline-related" posts from the corpus of posts generated in the affected area using keyword search. We preprocess these posts and label them as "gasoline shortage" and "non-gasoline shortage". We use the methodology used in (28) to develop the classifier that can distinguish between these two categories. The methodology includes : (1) Identifying important features i.e. unigrams (using tf-idf scores) and topics (using LDA and CTM), (2) Performing feature selection and model selection to find the most accurate classifier. Refer to (28) for the details of this methodology.

3.2. Stage 2: Localisation of commodity shortage in space and time

In Stage 2, the posts identified as "gasoline shortage" posts are used as sensors for location and time of shortage. Figure 2 demonstrates this by determining relative probabilities of shortage at different gas stations in Tampa (11 December 2017) using location of tweets. 56 % of gas stations were out of gas in Tampa at 9 PM (21). In Figure 2(a), at 9 PM, we start with the probability of shortage at each location as zero. At 9:15 PM, a tweet about gas shortage arrives at a location shown in Figure 2(b). Assuming that gas stations near to the location of the tweet are more likely to be out of gasoline, the changes in probabilities are reflected in 2(b). At 9:20 PM another tweet arrives and the probabilities are further updated in 2(c). Similar argument applies for time. A



(a)

(b)

(c)

Figure 2 **Tweets as sensor to estimate probabilities of shortage at different locations and times.**

tweet made at 9 PM is more likely to be about a shortage observed at 9 PM than 8:30 PM.

Bayesian Model : We conceptualise this idea of probabilistic sensing by modeling the arrival of the observations of shortage and the social media posts about shortage in a Bayesian framework. Using the Bayesian framework, we can assign prior probabilities of observation of shortage at different gas stations and update posterior probabilities using the Bayes's Theorem as evidence in the form of location and time of the posts arrive. We formalise this framework by defining the prior distributions for commodity shortage, observation of shortage, distance between observation and post and time between observation and post. Suppose, $i \in (1, 2, 3, ...n)$ denote each of the n locations of gasoline shortage and t denote the time of the day, then the random variables can are formally defined as:

- O_{it} is a Bernoulli random variable for the probability of the observation of shortage at location i at time t
- S_{it} is a Bernoulli random variable for the probability of shortage at location i at time t

- D_p is the random variable for the distance between post p about shortage and the observation of shortage
- T_p is the random variable for the time between post p about shortage and the observation of shortage.

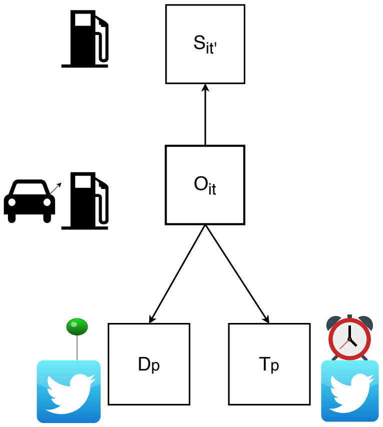


Figure 3 Basic Bayesian network for event localisation

The Bayesian framework is described by the conditional probability distributions (CPD) in equations (a)-(d). Observation of shortage at location i at time t, O_{it} is modeled as a Bernoulli random variable described by equation (a). In Figure 2, we described that the post's distance and time is used to infer the probability of shortage at different locations, O_{it}. We achieve this by modeling the distance (D_p) and time (T_p) of the post as exponential random variables conditioned on O_it. Their respective CPD's are in equation (b) and (c). Given evidence (data) about the variables, D_p and T_p at some time instance t_1 inference about unobserved variable O_{it} can be made for a time instance $t_2 < t_1$. However these two random variables can't be inference about the observation of gas shortage at a future time instances t_3. Suppose, the last tweet about shortage arrived at 9:15 PM from a location 10 minutes away from a gas-station. Then, no inference about observation of shortage at this gas-station can be made beyond 9:05 PM. Therefore, we also modeled the probability for shortage at location i at a future time $t+\delta t$, $S_{it+\delta t}$ given probability of observation at time t, O_{it} in equation (d). This is helpful for inferring shortages at future times. In our model, we assume that the probability of shortage at time $t+\delta t$ is equal to the probability of observation of shortage at time t. Figure 3 is a Bayesian network representing our model with one gas station and one tweet. This network can be expanded to accommodate more stations and tweets as shown in Figure 4. In practice, the Bayesian network to model shortage in our case study are orders of magnitude larger.

$$P_{O_{it}}(o_{it}) \qquad = \begin{cases} \text{p} & \text{for } o_{it} = 1 \\ \text{1-p} & \text{for } o_{it} = 0 \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in V, t \in t' \qquad (a)$$

$$P_{D_p|O}(d_p|o) \qquad = \begin{cases} \mu d \; e^{-\mu d} & \text{for } d > 0, max(o) = 1 \\ 0 & \text{otherwise} \end{cases} \qquad (b)$$

$$P_{T_p|O}(t_p|o) \qquad = \begin{cases} \lambda t \; e^{-\lambda t} & \text{for } t > 0, max(o) = 1 \\ 0 & \text{otherwise} \end{cases} \qquad (c)$$

$$P_{S_{it'+\delta t}|O_{it'}}(s_{it+\delta t}|o_{it}) = \; \{ \text{P}_{O_{it}}(o_{it}) \qquad \forall i \in V, t \in t' \qquad (d)$$

Inference on the Bayesian Network : Given the model, a methodology to infer unobserved variables is required when data about location and time of tweets (evidence) is received i.e. an algorithm to update the posterior probabilities of O_{it}'s and S_{it}'s given the data for the variables T_p's and D_p's. The most common exact inference methods are variable elimination, clique tree propagation or and recursive conditioning. All of these methods have complexity that is exponential in the network's tree-width. Therefore, these methods are not suitable for our case and we employ approximate inference methods. We use Markov Chain Monte Carlo (MCMC) sampling methods and variations inference techniques. The advantage of MCMC sampling methods is that it is asymptotically exact i.e. in the limit, MCMC will exactly approximate the target distribution (13). Variational inference do not have guarantee an exact solution in the limit. The disadvantage of MCMC sampling is that it is computationally expensive while variational inference are much faster (13).

However, there have been recent advances in MCMC sampling methods in the form of Hamiltonaian Monte Carlo (HMC) which have made the posterior sampling much more efficient than previous methods like Gibbs sampling and Metrapolis (12). We used HMC sampling methods in PyMC3 package for most of our inferences. However, for larger network sizes, where MCMC became computationally intractable, we utilised variational inference techniques in PyMC3. PyMC3 by default assigns the No-U-Turn (NUTS) sampler for sampling which is a extension of the HMC samplers (41). HMC's performance are sensitive to the parameter of number of steps and NUTS eliminates the calculation of the number of steps. NUTS is very efficient even for complex models and runs variational inference (i.e. ADVI) to find good starting parameters for the sampler. Commonly used step-methods besides NUTS are Metropolis and Slice. For almost all continuous models, NUTS should be preferred. There are hard-to-sample models for which NUTS will be very slow causing many users to use Metropolis instead.

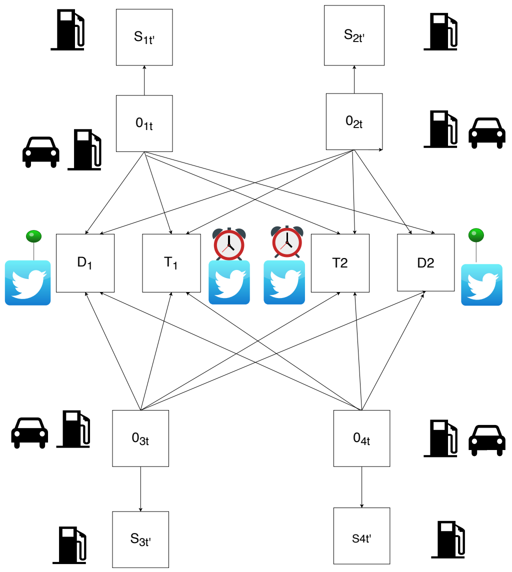
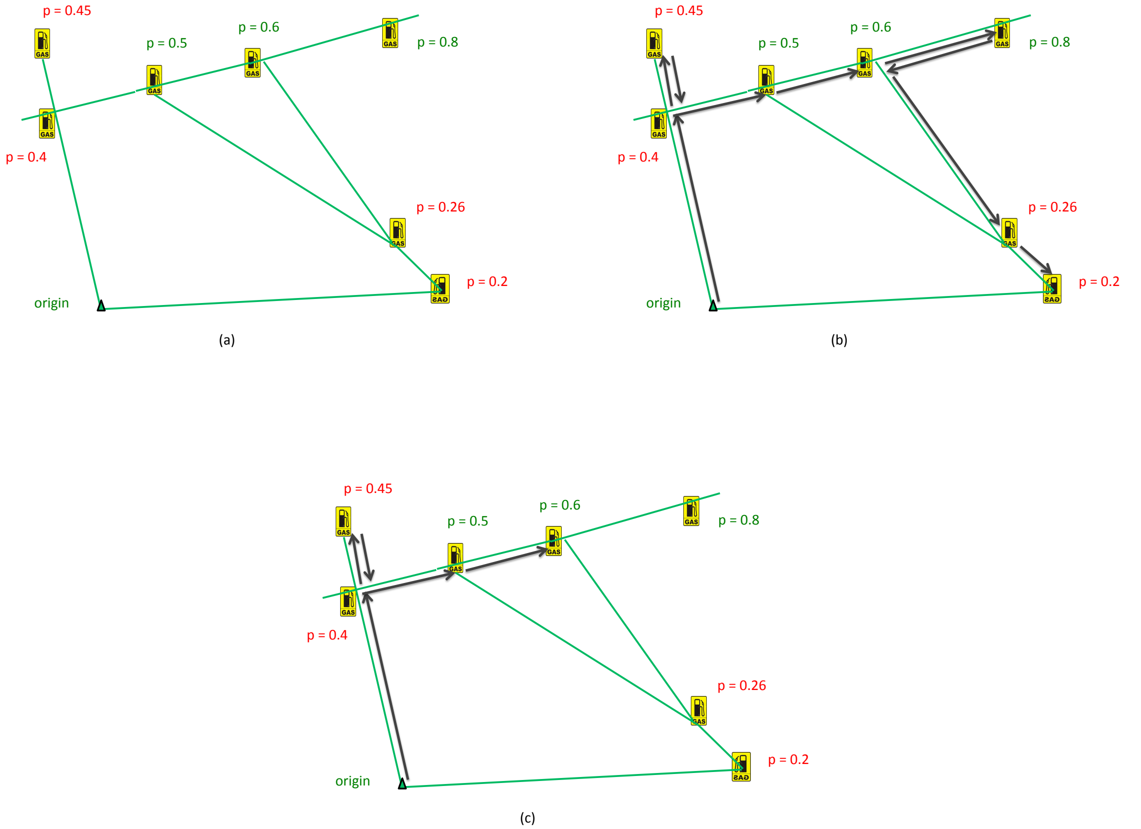


Figure 4 Bayesian network with two tweets and four gas stations



(a)

(b)

(c)

Figure 5 Tweets as sensor to estimate probabilities of shortage at different locations and times.

3.3. Stage 3: Optimising the Commodity Search Strategy

Having calculated the probability of shortage at different locations in Stage 2, in Stage 3, we determine the search path for finding the commodity. Figure 5(a) illustrates the problem. Each gas station has a probability of finding gas. A person starting at origin has to decide the search path (order of visiting gas stations). Figure 5(b) illustrates a feasible solution for the person given there is enough gas to visit all stations. Figure 5(c) is a possible solution given the person does not have enough gas in the tank to visit all stations in the vicinity.

A person searching for gasoline in a shortage scenario could have one of the three objectives. The first possible objective is to maximise the probability of obtaining a commodity. The second possible objective is to minimise the expected time of obtaining gas given that gas is present in at least one gas station. The third possible objective is to minimise the expected time spent in searching. There is a minute difference between the second and third objective. The second objective calculates the expected search time under the assumption that gas is found in one of the gas station while the third objective minimises expected search time without this assumption. This is a good assumption to make when it is certain that all gas stations are not out of gas and the search capacity is limited by the fuel left in the vehicle. Figure 6 consists of a toy example which helps in illustrating that optimal paths are different for the three objective functions. In the example, the person searching for gasoline starts at the origin $V_0 = (0,0)$ and has to visit three gas stations located located at $V_1 = (1,0), V_2 = (0,1), V_3 = (1,1)$. Let the travel-time on the arcs $(0,1),(1,0),(0,2),(2,0),(0,3),(3,0),(1,2),(2,1),(1,3),(3,1),(2,3),(3,2)$ be $t_{01},t_{10},t_{02},t_{20},t_{03},t_{30},t_{12},t_{21},t_{13},t_{31},t_{23},t_{32}$ respectively and be equal to the arc-lengths (arc-lengths are calculated using euclidean distance). Let P_0,P_1,P_2,P_3 are the probability of finding gas on vertices V_0,V_1,V_2,V_3. The probability of finding gasoline at each vertex has been highlighted in green in the figure.

Let's say, the searcher travels a feasible path $[(0,3),(3,1),(1,2)]$. The first objective which is the probability of finding gas can be calculated as the sum of probabilities of all the ways in which gas can be found on this path i.e. probability of finding gas at the first vertex V_3, or the second vertex V_1 or third vertex V_3 which is equal to $P_3 + (1-P_3)P_1 + (1-P_3)(1-P_1)P_2$. The second objective of minimising the expected time of finding gas given gas is found (expected search time given gas is found) can be calculated as the summing over the products of probability of finding gas at the vertices and travel time to those vertices i.e. $P_3t_{03} + (1-P_3)P_1(t_{03}+t_{31}) + (1-P_3)(1-P_1)P_2(t_{03}+t_{31}+t_{12})$. The third objective of minimising the expected time of searching can be calculated by summing over all vertices the products of the probability of travelling an arc in the path and the travel time of that arc i.e. $1 * t_{03} + (1 - P_3)t_{31} + (1 - P_3)(1 - P_1)t_{12}$. The third objective is greater than the second objective by the term $(1 - P_3)(1 - P_1)(1 - P_2)t_{12}$ which is the expected value of time spent on the final arc of search path having not found gas at any vertex. The third objective assumes that gas will be found at one of the vertices. This is good assumption to take if we know that gas is available at at least one of the gas stations.

Table 1 consists of the value of the three objective functions and path length for all possible paths. The first objective i.e. the probability of finding gas is equal for given a set of visited vertices and is independent of the path taken to visit those vertices. It is maximum when all three vertices are visited. Therefore, in any search strategy, if there is enough gas in the tank, all gas stations should be visited to maximise the probability of finding gas. If all gas stations are visited, it is found that the path $[(0, 1), (1, 3), (3, 2)]$ minimises the expected travel time while searching and the path $[(0, 3), (3, 1), (1, 2)]$ minimises the expected time to find gas. Suppose the gas tank of the vehicle only allows a path length less than 3, then probability of finding gas is maximised if vertices 1 and 3 are visited. Path $[(0, 1), (1, 3)]$ minimises the expected travel time. However, expected time to find gas is minimised if vertices 1 and 2 are visited and path $[(0, 1), (1, 2)]$ is followed.

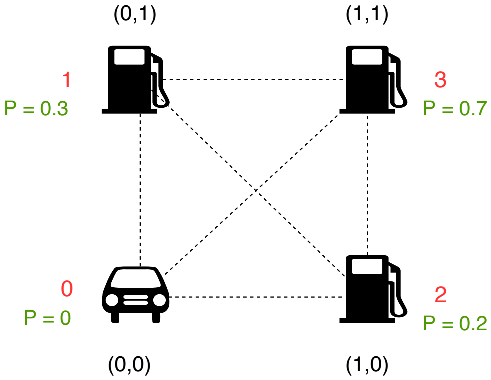


Figure 6 Toy problems for comparing the objective functions

Teller et al. developed the mathematical model and solution method to find a path to minimise expected travel time while searching which they call the call the average search time (49). However, it is clear that the optimal path is different for the objective of minimising expected search given the gas will be found in one of the vertices. In our work, we have developed mathematical models for the first objective of maximising the probability of finding the commodity and the second objective of minimising the time of finding the commodity. Teller et al. also assumed the searcher has unlimited resources and can search the entire graph. Our application of gasoline shortage does

Table 1 **Comparison of the objective functions for different paths on the toy example to determine the optimal path for each objective**

Path	Path Length	Expected Travel Time	Expected Time to Find Target	Probability of Finding Target
[(0, 1), (1, 2), (2, 3)]	3.41	2.547	2.37	0.832
[(0, 1), (1, 3), (3, 2)]	3	1.91	1.69	0.832
[(0, 2), (2, 1), (1, 3)]	3.41	2.688	2.54	0.832
[(0, 2), (2, 3), (3, 1)]	3	2.04	1.85	0.832
[(0, 3), (3, 1), (1, 2)]	3.82	2.0061	1.64	0.832
[(0, 3), (3, 2), (2, 1)]	3.82	2.0484	1.69	0.832
[(0, 1), (1, 2)]	2.41	1.987	1.45	0.44
[(0, 2), (2, 1)]	2.41	2.128	1.77	0.44
[(0, 1), (1, 3)]	2	1.7	1.62	0.79
[(0, 3), (3, 1)]	2.41	1.71	1.52	0.79
[(0, 2), (2, 3)]	2	1.8	1.74	0.76
[(0, 3), (3, 2)]	2.41	1.71	1.49	0.76

not satisfy this assumption and the search is limited by the amount of gas left in the vehicles tank. We included this constraint in our model. In the following sections, we formally define the sets, parameters and the decision variables for the two models.

Mathematical models

Sets

$V \in \{0, 1, 2, ..n\}$ is the set of all vertices in the graph and 0 is the depot

$A \in \{(0,1), (0,2), ..(n,0)\}$ is the set of all arcs available for the path

Parameters

n is the total number of gas stations

p_j is the probability of finding gas at vertex j

d_{ij} is the distance between vertex i and j

t_{ij} is the travel time between vertex i and j

m is the average mileage of the vehicle

f is the amount of fuel left in the vehicle

Decision Variables

$x_{ijk} \in \{0, 1\}$ is 1 if arc (i, j) is the k^{th} arc in the path and 0 otherwise,

where $i, j \in V$ and $k \in \{1, ...n, n+1\}$

Objective

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}})/\prod_{j\in V-\{0\}} p_j \tag{1}$$

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}}) \tag{2}$$

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}} + \sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl})/\prod_{j\in V-\{0\}} p_j \tag{3}$$

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}} + \sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl}) \tag{4}$$

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}} + \sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl}) + M\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} x_{ijk} \tag{5}$$

Two kinds of objective can be modeled. There is a finite probability of not finding the commodity despite searching all the locations. Hence, the first objective could be to maximise the probability of finding the commodity given the commodity is found. Equation (1) describes this objective function. The term $\prod_{j\in V-\{0\}} p_j$ in the denominator of the is the probability of finding gas. It is a constant term and therefore can be removed from the objective function to obtain equation (2). The other objective could be to minimize expected time of finding the commodity given that the commodity is found. Equation (3) is the objective function for expected time of finding the commodity given the commodity is found. Removing the constant term $\prod_{j\in V-\{0\}} p_j$, probability of finding the commodity the objective function in equation (4) is obtained. We need to add the term $M\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} x_{ijk}$ to the objective as shown in equation (5). A large value of M ensures that more than one location is visited.

We have two mathematical models, one for each objective functions. However, the constraints remain same for each of the objective. For brevity, following description contains both the models.

Maximize:

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}}) \tag{6}$$

OR

Minimize:

$$\sum_{k=1}^{n}(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}} + \sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl}) + M\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} x_{ijk} \tag{7}$$

st.

$$\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} d_{ij}x_{ijk} \ \leq f*m \tag{8}$$

$$\sum_{i\in V}\sum_{j\in V i\neq j} d_{ij}x_{ijk} \leq 1 \qquad \forall k\in\{1,2,..n+1\} \tag{9}$$

$$\sum_{j\in V-\{0\} i\neq j} x_{0j1} = 1 \tag{10}$$

$$\sum_{i\in V i\neq j} x_{ijk} \leq x_{ijk+1} \qquad \forall k\in\{1,2,..n\}, j\in V-\{0\} \tag{11}$$

$$\sum_{k=1}^{n+1}\sum_{i\in V i\neq j} x_{ijk} = 1 \qquad \forall j\in V \tag{12}$$

$$\sum_{i\in V-\{0\}} x_{1,0,n+1} = 1 \tag{13}$$

Constraints

Constraint (8) ensures that the length of the path is constrained by the fuel left in the tank. Constraint (9) ensures that only one arc can be the k^{th} arc in the path. Constraint (10) ensures that first arc starts at the origin. Constraint (11) ensures that destination of the k^{th} arc is the origin of the $k+1^{th}$ the arc. Constraint (12) ensures node should be visited only once. Constraint (13) ensures that origin acts as a dummy node for the end of the path.

Solution Methods

Linearize the models and solve using CPLEX : The objective functions are non-linear for both the objectives making the formulation a non-linear integer program. Since both the problems are contains product terms and the decision variables as exponents, we linearized the objective functions using the logarithm function. Minimizing equation (6) is equivalent to minimizing equation (14) and maximising equation (7) is equivalent to minimizing equation (15). Using the properties of the logarithm functions equation (14) and (15) can be exactly reduced to equation (16) and (17). Using equation (16) model as objective for maximising probability is exactly solved using CPLEX to get the exact soltuion. Equation (17) is still non-linear. Converting the logarithm term $\log(\sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\}i\neq j} t_{ij}x_{ijl})$ in equation (17) to $\sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\}i\neq j} t_{ij}x_{ijl}$ in equation (18), the linear model obtained closely approximates the non-linear model for minimising expected time. This approximated model can now be solved via CPLEX and provide an approximate solution to the original problem. For maximising the probability, the linearized model is solved using CPLEX to get an exact solution. For minimising expected search time, the linearized provides a approximate solution using CPLEX. Therefore, we developed a branch-and-bound method for minimising the expected search time which we have described in the following subsection.

$$\sum_{k=1}^{n}(\log(\prod_{i\in V}\prod_{j\in V-\{0\}i\neq j} p_j^{x_{ijk}} \prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\}i\neq j} (1-p_j)^{x_{ijm}})) \tag{14}$$

$$\sum_{k=1}^{n}(\log(\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j} p_j^{x_{ijk}}\prod_{m=1}^{k-1}\prod_{i\in V}\prod_{j\in V-\{0\} i\neq j}(1-p_j)^{x_{ijm}}+\sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl}))+M\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} x_{ijk} \tag{15}$$

$$\sum_{k=1}^{n}(\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} x_{ijk}\log p_j+\sum_{m=1}^{k-1}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} x_{ijm}\log(1-p_j)x_{ijm})) \tag{16}$$

$$\sum_{k=1}^{n}(\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} x_{ijk}\log p_j\sum_{m=1}^{k-1}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} x_{ijm}\log(1-p_j)+\log(\sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl}))+M\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} x_{ijk} \tag{17}$$

$$\sum_{k=1}^{n}(\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} x_{ijk}\log p_j\sum_{m=1}^{k-1}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} x_{ijm}\log(1-p_j)+(\sum_{l=1}^{k}\sum_{i\in V}\sum_{j\in V-\{0\} i\neq j} t_{ij}x_{ijl}))+M\sum_{k=1}^{n}\sum_{i\in V}\sum_{j\in V i\neq j} x_{ijk} \tag{18}$$

A Modified Branch and Bound Method with Clustering of Vertices : Teller et al. developed a branch and bound methodology for the problem of minimising expected travel time while searching (49). We modified their methodology to adapt to the constraints and objective function of our model. Additionally, we introduce the idea of clustering of vertices in the search network which makes the algorithm computationally efficient and converge faster.

Clustering of Vertices: The branch and bound technique of Teller et al. iteratively constructed search path. The root node of the tree is a search path containing the starting vertex (depot) and all the other vertices as candidates. As the tree branches, more vertices were added to the search path. When the candidate list is empty (and the tree is at depth $|V|$), a full search path has been constructed. We found in a lot of real world examples the vertices added to the search path in the subsequent nodes of the branch and bound tree were adjacent to each other on the network and formed a cluster (according to euclidean distance). We illustrate this behavior using the example in Figure 7.

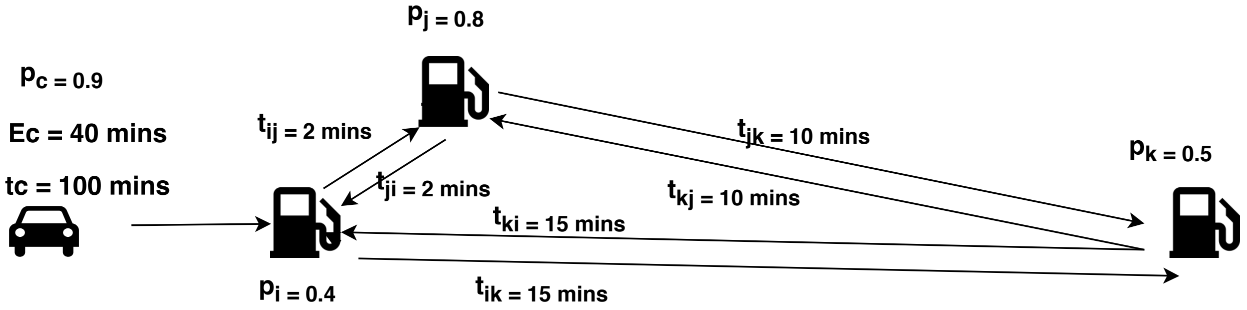


Figure 7 Example to show the clustering property in gas stations

Let v_i v_j, v_k be final three vertices on the search path of a vehicle. Let t_{ij}, t_{jk}, t_{ik}, t_{ji}, t_{kj} and t_{ki} be the travel times between them. Let p_i, p_j, p_k be the probability of finding the commodity at the respective vertices. Say, vertex v_i is added to a search path (visited) at some n^{th} node of branch and bound i.e. it is the n^{th} vertex in the search path. Let p_c be the probability of vertex i being the n^{th} vertex in the search path given by the product of probabilities of not finding gas in previous $n-1$ vertices. Let t_{mi} the travel time between the previous vertex m and i, t_c be the travel time upto vertex v_i, and E_c be the expected search time vertex v_i. To determine the order of visit for the final two vertices v_j and v_k, the expected search time for two search path can be compared i.e. vertex v_j will be the $(n+1)^th$ visited before v_k if the expected time to find gas for the path for this order is less than the expected value if v_k is visited first. Therefore, Equation (19) states the necessary condition for v_j to be visited before v_k which can be simplified to Equation (20). If $t_{jk} = t_{jk}$, then Equation (20) can be further simplified to Equation (21).

$$\begin{aligned}E_c + p_c(1-p_i)p_j(t_c+t_{ij}) + p_c(1-p_i)(1-p_j)p_k(t_c+t_{ij}+t_{jk}) \le E_c + p_c(1-p_i)p_k(t_c+t_{ik})\\ + p_c(1-p_i)(1-p_k)p_j(t_c+t_{ik}+t_{kj}) \quad (19)\end{aligned}$$

$$t_{ij}(p_k+p_j-p_jp_k) + t_{jk}p_k(1-p_j) < t_{ik}(p_j+p_k-p_jp_k) + t_{kj}p_j(1-p_k) \quad (20)$$

$$t_{ij}(p_k+p_j-p_jp_k) + p_kt_{jk} < t_{ik}(p_j+p_k-p_jp_k) + p_jt_{jk} \quad (21)$$

A careful examination of the equations shows that $p_j > p_k$, $t_{ij} < t_{ik}$ and $t_{jk} \le t_{kj}$ are favorable conditions for v_j being visited before. For instance, in the example shown in Figure 7 that satisfies all these conditions, the expected time to find gas for path ending in sub-path $[(v_i,v_j)(v_j,v_k)]$ is 41.40 minutes. On the other hand, the path ending in sub-path $[(v_i,v_k)(v_k,v_j)]$ has expected time to find gas is equal to 46.21 minutes. If this network is expanded to add more gas stations which are further away from v_j and have smaller probabilities than p_j as shown in the figure, v_j will remain to be the $(n+1)^{th}$ in the search path. Also, more vertices can be added near v_j with high probabilities and they will be visited before v_k and vertices further away. These vertices form a the "cluster" as shown in red in Figure 8. In our case study, we found multiple examples of such clusters. These gas stations were near to each other and probabilities of finding gas were high in these clusters and highly correlated amongst them. This could be attributed to our Bayesian network. The distance between the tweets and gas station are similar for adjacent gas stations and the D_p is one the evidence for inferring probability of shortage at a station. This makes the probabilities at adjacent vertices highly correlated. However, we note that the probabilities are conditionally independent given distances from tweets. Correlated shortage probabilities and proximity causes this clustering behavior. Also, it is highly likely that probability of availability of gasoline is correlated in clusters as gas tankers in such emergencies would re-fuel all gas stations it can reach within proximity.

We exploited this property to find clusters of vertices that satisfied this property. To find these clusters we used the DBSCAN algorithm to first cluster according to distance. Then we filtered out clusters which

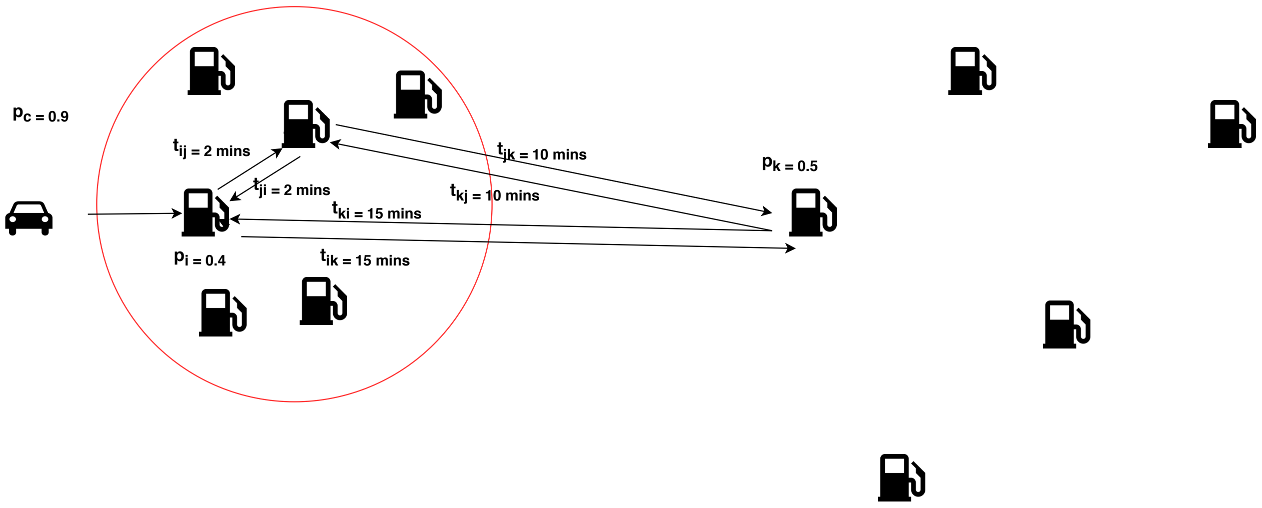


Figure 8 **Example of clustering in gas stations**

satisfied the property in Equation (20) applied to pairs of vertices with vertices outside the cluster. After clustering the vertices we perform a multi level branch and bound. In the first level an iteration of branch and bound is performed in which the branching is performed on these clusters i.e. clusters (represented by one of the vertex in the cluster) are added to a search path (search path of clusters). After having found the order of clusters to be visited, we perform one iteration of branch and bound for each cluster in the next level to identify the path within a cluster. we combine the results from all iterations to get a the optimal search path. Following are the other features of the branch and bound which have were developed in by Teller et al. (49):

Branching: The branching technique iteratively constructs search paths. The root of the tree is a search path containing the starting vertex/clusters and all the other vertices/clusters as candidates. As the tree branches, we add more vertices/clusters to the search path. When the candidate list is empty (and the tree is at depth $|V|$ or equal to the number of clusters), we have constructed a full search path.

Lower Bound: Let E denote the expected time to find the commodity. Let v_c be the vertex/cluster added to the path in the considered node. Let v_f represent the vertex/cluster that was added to path in the predecessor (father) node. Let E_f be the E calculated in the predecessor node. Let t_{fc} represent the traveling time required between v_f and v_c. Let p_f and p_c represent the probabilities of finding the commodity in v_f and v_c. Let CP_f and CP_c denote the cumulative probability of not finding the object in the path constructed until the predecessor node and until the current node respectively. The cumulative probability of not finding the object until reaching the current node is $CP_c = CP_f(1 - p_c)$ and The lower bound at the current node, E_c is given by $E_c = E_f + CP_f p_c t_{fc}$

Upper Bound: Run the greedy heuristic to obtain an initial upper bound. This bound is mainly used for fathoming branches in higher tree levels. The greedy heuristic has been described in section below. For each node, calculate E of the path constructed until the current node, and use a greedy algorithm to complete the path with the remaining vertices. This bound may be tighter than the previous bound in branches that start with different vertices than those found using the greedy algorithm.

Greedy Heuristic

In the greedy heuristic the arcs (i,j) are added in the search path in increasing order of the ratio $r_{ij} = t_{ij}/p_j$ till the time it does not exhaust its fuel.

Algorithm 1 Heurestic to minimize expected time to find gasoline

```
$D = f * m$
$r_{ij} = t_{ij}/p_j$
Initialise $A$ = Set of $(i,j)$
Initialise $Path = \{\}$ (Empty set)
Initialise $i = 0$
while $n(Path) \neq n$ do
    Find smallest $(i,j)$ from $A$
    if $D > d_{ij}$ then
        Append $P$ with $(i,j)$
        $D = D - d_{ij}$
        $i = i + 1$
    end if
    if $D < d_{ij}$ then
        break
    end if
end while
```

4. Case Study

In this section, we describe the application of our methodology to find the search path for people looking for gasoline in Florida during the Hurricane Irma onset in September 2017. While the landfall of Irma in Florida happened on the September 10, 2017, the shortage of gasoline in Florida was observed in multiple cities in the period September 6-15, 2017 (i.e. during onset and beyond landfall). We accessed more than 1 million tweets from Florida during this period and the details of the tweet data. In stage 1, we detected gasoline-shortage tweets using the methodology described by Khare et al. (28). In stage 2, we use the classified tweets to localise the gasoline-shortage event using the Bayesian network and variational inference algorithm described in Section 3.2. The data about the percentage of gas stations out of gasoline on these dates in all major cities of Florida was obtained from Gasbuddy.com (20). We use this data as ground truth about shortage to validate our event localizer methodology. Finally, in stage 3, we find the search path for people looking for gasoline in these 4 major cities. We conducted experiments simulating arrival of gasoline searchers at various times of the day from September 9-15, 2017 and measured the probability and expected time of finding gasoline.

Table 2 Summary statistics of tweet data

Summary Statistic	Values
Number of Tweets Collected	1,048,575
Number of Unique Twitter Users	111,801
Period of Data Collection	6th Sept 2017- 15th Sept 2017
Date of Irma Landfall in Florida	9th Sept 2017
Number of tweets prior to Irma landfall in Florida	456,530
Number of tweets during Irma in Florida	151,792
Number of tweets post Irma in Florida	440,253
Number of Gas Shortage Related Tweets Before Landfall	2,805

4.1. Detection of gasoline-shortage tweets

In stage 1, we filtered out gasoline-related tweets from the corpus of 1 million tweets. For this, we combined keyword search in hashtags and tweets and filtered down to 4070 relevant gasoline-related tweet. The hashtags and words were found using regular expressions and included *gasoline, gas, gasinmiami, gaspricefixing, gasstation, gasservice, gastateparks, gasshortage, gasoil, gastation, gaswaste, nogas, outofgas, findgas.* For tweet classification, we labeled the 4070 gasoline-related tweets. Out of the 4070 gasoline-related tweets, 2594 were gasoline-shortage tweets. For classification, we extracted important unigrams on the basis of "term frequency". We filtered out 937 unigrams with "sum of term frequency set" to 5 in the dtm (threshold). These were the candidate features for our SVM classifier. Figure 9 is a wordcloud of the top 50 unigrams in the tweet corpus by term frequency.

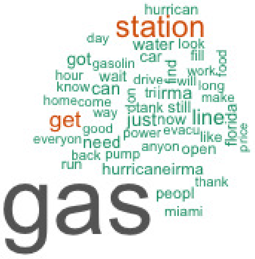


Figure 9 Word cloud of most frequent words

We also used "topics" to be used as features. For this step we determined the number of topics and the best topic model amongst the four models (LDA 1, LDA 2, LDA 3, CTM). Table 3 shows five abstract topics found using LDA 3 along with the top five words in each of the topics. Topic 1 is about people tweeting that they cannot find gasoline due to Irma. On the other hand, Topic 2 is about people tweeting that gas stations are closed and they need gas. Topic 3 is about no gasoline being there in Miami. Topic 4 is about waiting in line for gasoline because of Irma. Lastly, Topic 5 is about high gasoline prices. We had 937 unigrams and 12 topics as candidate features for the classifier. For model selection, we trained multiple models with different sets of features (on the training data) and measure the F1-score on the hold out test set. The best F1 score was achieved for 5 words and 12 topics (out of the 12 candidate topics) of 0.877. Khare et al. can be referred for further details of the results in stage 1 (28).

Table 3 Top 5 topics identified using the topic modeling techniques

LDA 3				
Topic 1	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**
gas	station	gas	gas	gas
cannot	gas	no	station	price
find	need	station	wait	high
know	hurricaneirma	line	line	got
irma	close	miami	irma	irma

4.2. Determining probability of gasoline shortage at gasoline stations

In stage 2, we find the probability of gasoline shortage at different gasoline stations using the methodology described in Section 3.2. The Gasbuddy app reported the number of gas stations out of gasoline in multiple cities of Florida when shortage was observed. Table 4 contains its reports for September 9, 2017.

Table 4 Percentage of gas stations out of gas on Sept 9, 2017 in Florida as reported by Gasbuddy (20)

	9 Sep, 9:05 AM	9 Sep, 1:15 PM	9 Sep, 4:45 PM	9 Sep, 10:05 PM
West Palm Beach	53%	56%	55%	56%
Miami/Fort Lauderdale	65%	66%	66%	65%
Gainesville	55%	48%	50%	57%
Tampa/St Pete	44%	44%	49%	53%
Orlando	34%	35%	34%	39%
Jacksonville	29%	27%	30%	36%
Tallahassee	46%	45%	45%	50%
Fort Myers-Naples	-	61%	53%	49%
Panama City	-	16%	17%	18%

To validate our methodology, we computed the probabilities of gasoline shortage in all the gas stations in 4 cities of Florida for the times Gasbuddy reported the shortages. To compute the probabilities, we build a Bayesian network for for all the gas stations and all the tweets received in the city. The priors for the observation of shortage variables $O'_{it}s$ are initiated with parameter $p = 1$ at time $t = 0$ (12 AM, September 9). O_{it} and $S_{it+\delta t}$ are update for all stations i at times t and $t + \delta t$ as tweets arrive at some time t'. Sampling methods and variational inference in the PyMC3 package as described in Section 3.2.2 are used to calculate the posterior. The resultant posterior probabilities for selected gas stations are detailed in Table 5. The times chosen are the times of reporting on the Gasbuddy app. It must be noted that the same process could be used to calculate the posterior probabilities for any other time of the day. In Table 5, one can see that for all gas stations in different cities the probability of shortage increased as the day progressed. This result is in line with what was observed on the ground. Table 4 shows that the percentage of gas stations out of gas increased as the day progressed. Given this information (assuming shortage increases uniformly in the city) the probability of shortage at each gas station should increase.

Table 5 Probability of gas shortage at selected gas stations on September 9 as calculated by event localizer

	190 SW 8th St, Miami	**5701 Memo. Hwy, Tampa**	**4138 W Oak Rd, Orlando**	**10044 Atl Blvd, Jacksoville**
Sept 9, 9:05 AM	0.53	0.5	0.57	0.51
Sept 9, 1:15 PM	0.55	0.55	0.61	0.51
Sept 9, 4:45 PM	0.76	0.57	0.63	0.51
Sept 9, 10:05 PM	0.85	0.61	0.66	0.53

We applied a threshold on the posterior probability of shortage gasoline shortage to predict which gas stations are out of gas. We found that using the threshold of 0.5 (i.e. all gas stations with posterior probability greater than 0.5 were classified as stations without gas) we achieved a MAPE of 12 percent. Figure 10 compares our prediction using this threshold with the reports by Gasbuddy. We clearly observe the accuracy increases as the day progresses in all cities. We observe that in each city we underestimate the ground truth in the mornings and overestimate in the nights. This can be attributed tweeting behavior of people. In all cities we saw that frequency and volume of of tweets were much much higher in the might times than in the morning times. Also for cities like Jacksonville the tweets observed were much less compared to other cities. This could be because Jacksonville was not in the path of hurricane, far away from landfall and also shortage observed was less. This led to underestimation of shortage using our model in Jacksonville at all times.

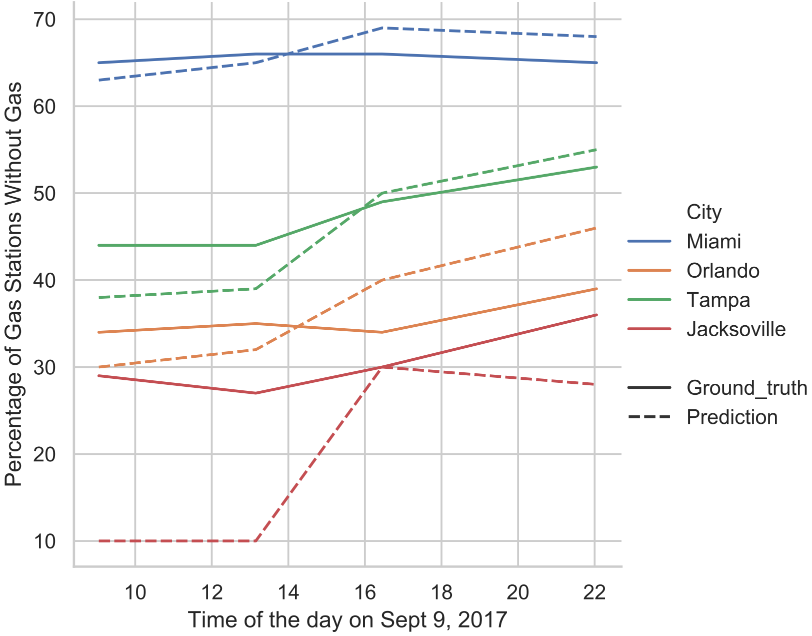


Figure 10 Comparison of predicted gasoline shortage from event localizer and ground truth data from Gasbuddy on September 9, 2017 from 09:00 hours to 24:00 hours

4.3. Determining the search path for evacuates searching for gasoline

In stage 3, having validated the event localizer, we determined the performance of our search path finder. Before testing its performance, we tested the value of of using social media information in the search path planning. For this, we simulated 100 gas stations networks with actual shortages at certain gas stations and tweets reporting shortages at different times and locations. In these simulations, the number of gas stations was a uniform distribution between 10 and 50. Percentage of gas stations out of gas was uniformly distributed between 20 and 80. Say the percentage of gas station out of gas in a case was 60, then shortage at a gas station followed a Bernoulli distribution with $p = 0.6$. Given shortage at a station, a tweet about the shortage at the station follows a Bernoulli distribution with p uniformly distributed between 0 and 30. The distance between the tweet and the station is exponentially distributed with $lambda$ uniformly distributed between 0 and 2 miles. Similarly, time between the tweet and the shortage exponentially distributed with $lambda$ uniformly distributed between 0 and 1 hour. We assumed the shortage was observed for 2 hours and simulated the arrival of a searcher which is uniformly distributed between 0 and 2 hours.

First, we solved the models for building search path without any social media information i.e. without employing the Bayesian inference fixed the probability of shortage at each gas station at all times as 0.5. We solved the model and found a search path that minimised expected time of finding gas for the 100 cases. We calculated the actual time for obtaining gas for the 100 cases using the search path suggested by the model. Next, we performed the same process after calculating the probabilities of shortage using the social media data using our Bayesian Inference methodology. We found that average time of finding gas over the 100 cases was 41.74 % times less in the case we used social media information.

Next, we tested the performance of the models and our solution methodologies using data from the four cities of Florida. We had two models for determine the search path. Model 1 minimises the expected time of finding gas given gas is found. Model 2 maximises the probability of finding gas. To test which is a better strategy for real life scenarios, we devised an experiment to compare the performance of the models. In our experiment on the four cities, we simulated the arrival of 200 different gasoline searchers at random times and at random locations in the city at 9 PM. For each case we calculated the posterior probabilities of gas shortage at all stations within 10 mile radius using the event localiser. Next, we solved the two models using CPLEX. Table 6 shows the comparison between the two models for all the cities. As expected there is a trade off between minimising expected time and maximising probability of finding gasoline. For all cities on an average, model 1 does better in minimising its objective of expected time while model 2 maximises probability better on average. It is also reflected in our experiments that the expected time to find gas in Miami was much higher than other places. The probability of finding gas was lower in Miami. In contrast Jacksonville had the highest probability of finding gas on and smallest least time to find gas on an average. This is in agreement in from the ground truth about shortage observed in the cities and reaction of people on social media like Twitter.

Table 6 Comparison of Model 1 and Model 2 on experiment data

	Model 1 (Minimize Expected Time)		Model 2 (Maximize Probability)	
	Average expected time	Average probability	Average expected time	Average probability
Miami	29 mins	0.53	36 mins	0.71
Tampa	14 mins	0.63	20 mins	0.85
Orlando	10 mins	0.68	13 mins	0.88
Jacksonville	3 mins	0.8	5 mins	0.93

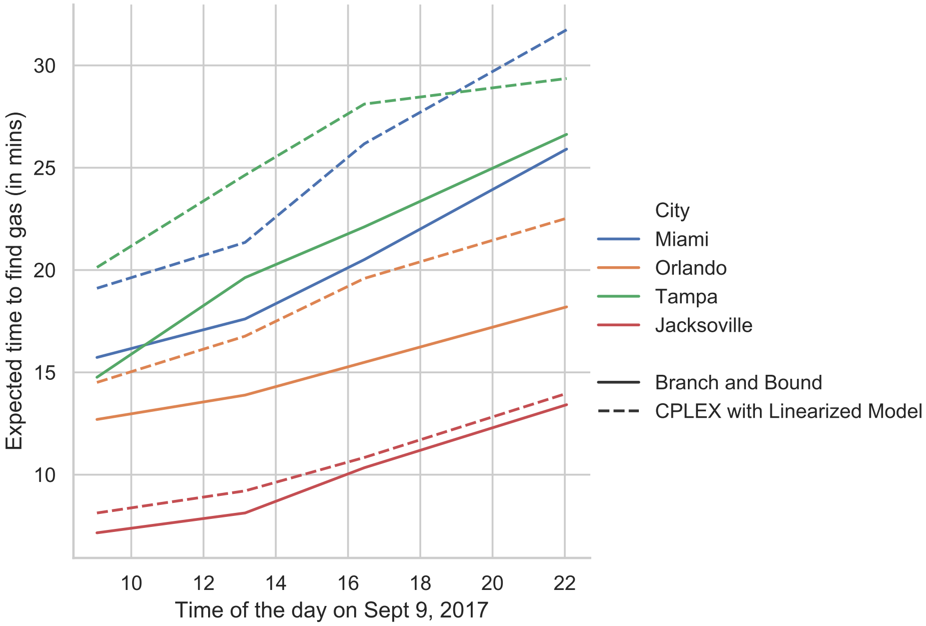


Figure 11 Comparison of Linearized model (CPLEX) and modified branch and bound in minimising expected time to find gas on September 9, 2017 from 09:00 hours to 24:00 hours

We also wanted to compare the performance of the the modified branch and bound with CPLEX solution of the linearized model. Figure 11 compares the performance for (model 1) for the four cities. We found that using CPLEX on the linearized model approximated the solution of branch and bound very well. There was a difference of 6 percent in results from CPLEX of the linearized model and branch bound averaged over all the cities. For cities like Jacksonville where the probability of shortage was low and probability of finding commodity was high, the difference was a low as 2 percent. We also found that the difference in expected time to find gas also varied by the time of the day. During the mornings (before 12 PM) the difference was a low as 4 percent. While at nights the difference was 9 percent. This can be attributed to the fact that the probability of shortage at a a gas station was underestimated during the mornings. the difference in optimal objectives of CPLEX and branch-and-bound was low and The probability of shortage was found to be low,

We also wanted to compare the computational performance of our the branch and bound with our modified branch and bound method with clustering of vertices. Table 7 compares the performance of the two algorithms on experiments conducted in the four cities. From the results it is clearly evident that

Table 7 Comparison of computational times of Branch and Bound with and without clustering of gas stations

City	Avg. No. of Gas Stations in a Search	Avg. No. of Clusters	Avg. No. of Gas Stations in a Cluster	Computational Time (Branch and Bound)	Computational Time (Branch and Bound with Clusters)
Miami	36	8	4	2345.61 sec	738.26 sec
Tampa	25	5	3	1917.38 sec	831.33 sec
Orlando	24	7	3	1889.52 sec	604.17 sec
Jaksonville	19	2	3	1668.13 sec	1345.21 sec

the modified branch and bound leads to significant reduction in computational time. In Miami, which has maximum clustering of gas stations with average of 8 clusters (containing 4 gas stations each) in 36 gas stations, there is 10 fold reduction in computational time. Orlando are Tampa are comparable for the average number of gas stations in a search. However, the number of clusters in Orlando was higher on a average. This is reflected in the modified branch and bound as the average computational time for Orlando is less than of half that of Tampa. Jacksonville where there is hardly any clustering in gas stations, the computational time hardly changes with the modified branch and bound.

5. Conclusion and Future Work

We found that our Bayesian inference model was very effective in predicting the number of gas stations out of gas in four cities of Florida during Hurricane Irma. This makes this methodology good for making probabilistic inferences about the location and time of gas shortages during disasters. A lot of social media data is unreliable for situation awareness because of prevalence of misinformation. In our methodology, the inference is probabilistic and multiple tweets are used as sensors to infer the probabilities of shortage for one gas station. This fusion of inference from multiple tweets (sources of information) makes the conclusions less susceptible to inaccuracies and misinformation. Similar Bayesian methodologies could have far reaching implications in improving situation awareness in other disaster management applications. We also found in our analysis that using social media, we could improve the average time to obtain gasoline by 41.74 %. This shows that social media information is valuable in increasing the effectiveness of our gasoline search models. We believe that our methodology can be expanded to search for other essential commodities during disasters. For instance, the same methodology could be used to search essential items like face masks and hand sanitizers in the shortage caused by Covid 19 pandemic onset in the United States. Furthermore, social media can be utilised in a similar probabilistic framework for improving other decision making models in disaster like search and rescue models.

We developed two models for searching gas. The model which maximises probability of finding gas was optimally solved using CPLEX after linearizng the objective while the model for minimising expected time for finding gas was solved using a modified branch and bound method. As expected the former did better in maximising the probability of finding gas on the case study while the latter did better in minimisng expected time to obtain gas. In a real life scenario, could prioritise their requirements and accordingly choose the model to plan the search path. The modified branch and bound methodology we developed to minimise the expected search time outperformed branch and bound method on our case study. Our method took advantage of the clustering of gas stations (with high probabilities of gas availability) and reduced the search load of the branch and bound algorithm.

The Bayesian inference methodology we used in our application showed some limitations in the case study. We found that the accuracy of the inference about probability of gas shortages in all cities depended a lot on the time of day. During the mornings there was an underestimation of the probability of shortage and in the evenings there was overestimation. This could be attributed to the tweeting behavior. We found that people tended to tweet more later in the day despite the shortage levels. It could also point to the fact that more people were searching for gas later in day and hence we were getting more information from social media at that time. In the future there is opportunity to overcome this bias in the data. we can either introduce more random variables that introduce the probability of tweet given the time of the day. Other ways to de-bias can also be looked at.

Currently our search path finder methodology optimises the search path for a single searcher. However, in real-life scenarios like shortage of gasoline, there could be multiple searchers and the path taken by one searcher could his/her path affect the chances of finding the commodity for another searcher. In future work, a model for multiple search agents starting at different origins each with the goal of locating the commodity could be formulated. Game theoretic approaches can be explored to model these competitive search problem. The current search path finder is static. We do not update probabilities of shortage at gas stations after visiting a gas station. In principle, after visiting a gas station in the search path and not finding gas, the probabilities can be updated using the Bayesian Network we constructed and a new search path can be found using the updated probabilities. This is a very computationally expensive method if used in conjunction with branch-and-bound or CPLEX. However, it could be combined with a greedy heuristic in a future work for an efficient dynamic method.

References

[1] Steve Alpern. The rendezvous search problem. *SIAM Journal on Control and Optimization*, 33(3):673–683, 1995.

[2] Steve Alpern. Rendezvous search: A personal perspective. *Operations Research*, 50(5):772–795, 2002.

[3] Steve Alpern, VJ Baston, and Skander Essegaier. Rendezvous search on a graph. *Journal of Applied Probability*, 36(1):223–231, 1999.

[4] Steve Alpern and Shmuel Gal. Rendezvous search on the line with distinguishable players. *SIAM Journal on Control and Optimization*, 33(4):1270–1276, 1995.

[5] Steve Alpern and Shmuel Gal. Searching for an agent who may or may not want to be found. *Operations Research*, 50(2):311–323, 2002.

[6] Edward J Anderson and RR Weber. The rendezvous problem on discrete locations. *Journal of Applied Probability*, 27(4):839–851, 1990.

[7] Lars Backstrom, Eric Sun, and Cameron Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. In *Proceedings of the 19th international conference on World wide web*, pages 61–70. ACM, 2010.

[8] Anatole Beck. On the linear search problem. *Israel Journal of Mathematics*, 2(4):221–228, 1964.

[9] Anatole Beck and DJ Newman. Yet more on the linear search problem. *Israel journal of mathematics*, 8(4):419–429, 1970.

[10] Richard Bellman. An optimal search. *Siam Review*, 5(3):274, 1963.

[11] Oded Berman, Eduard Ianovsky, and Dmitry Krass. Optimal search path for service in the presence of disruptions. *Computers & operations research*, 38(11):1562–1571, 2011.

[12] Michael Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[13] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[14] Maged N Kamel Boulos, Bernd Resch, David N Crowley, John G Breslin, Gunho Sohn, Russ Burtner, William A Pike, Eduardo Jezierski, and Kuo-Yu Slayer Chuang. Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, ogc standards and application examples. *International journal of health geographics*, 10(1):67, 2011.

[15] Scott Shorey Brown. Optimal search for a moving target in discrete time and space. *Operations research*, 28(6):1275–1289, 1980.

[16] Zhiyuan Cheng, James Caverlee, and Kyumin Lee. You are where you tweet: a content-based approach to geo-locating twitter users. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 759–768. ACM, 2010.

[17] Joe G Foreman. The princess and the monster on the circle. *Differential Games and Control Theory*, pages 231–240, 1974.

[18] Joe G Foreman. Differential search games with mobile hider. *SIAM Journal on Control and Optimization*, 15(5):841–856, 1977.

[19] Shmuel Gal. A stochastic search game. *SIAM Journal on Applied Mathematics*, 34(1):205–210, 1978.

[20] Gasbuddy. https://business.gasbuddy.com/hurricane-irma-live-updates-fuel-availability-station-outages/, 2017.

[21] Gasbuddy. https://tracker.gasbuddy.com/?q=buffalo,

[22] Mark Gaynor, Margo Seltzer, Steve Moulton, and Jim Freedman. A dynamic, data-driven, decision support system for emergency medical services. In *International conference on computational science*, pages 703–711. Springer, 2005.

[23] Bo Han, Paul Cook, and Timothy Baldwin. A stacking-based approach to twitter user geolocation prediction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 7–12, 2013.

[24] Rufus Isaacs. *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization*. Courier Corporation, 1999.

[25] Arun Jotshi and Rajan Batta. Search for an immobile entity on a network. *European Journal of Operational Research*, 191(2):347–359, 2008.

[26] Arun Jotshi and Rajan Batta. Investigating the benefits of re-optimisation while searching for two immobile entities on a network. *International Journal of Mathematics in Operational Research*, 1(1-2):37–75, 2009.

[27] David Jurgens. That's what friends are for: Inferring location in online social media platforms based on social relationships. In *Seventh International AAAI Conference on Weblogs and Social Media*, 2013.

[28] Abhinav Khare, Qing He, and Rajan Batta. Predicting gasoline shortage during disasters using social media. *OR Spectrum*, pages 1–34, 2019.

[29] Eyun-Jung Ki and Elmie Nekmat. Situational crisis communication and interactivity: Usage and effectiveness of facebook for crisis management by fortune 500 companies. *Computers in Human Behavior*, 35:140–147, 2014.

[30] Abhinav Kumar and Jyoti Prakash Singh. Location reference identification from tweets during emergencies: A deep learning approach. *International journal of disaster risk reduction*, 33:365–375, 2019.

[31] Kenneth A. Lachlan, Patric R. Spence, and Xialing Lin. Expressions of risk awareness and concern through Twitter: On the utility of using the medium as an indication of audience needs. *Computers in Human Behavior*, 35:554–559, 2014.

[32] Brooke Fisher Liu, Julia Daisy Fraustino, and Yan Jin. Social Media Use During Disasters: How Information Form and Source Influence Intended Behavioral Responses. *Communication Research*, 43(5):626–646, 2016.

[33] Nimrod Megiddo, S Louis Hakimi, Michael R Garey, David S Johnson, and Christos H Papadimitriou. The complexity of searching a graph. *Journal of the ACM (JACM)*, 35(1):18–44, 1988.

[34] Stuart E Middleton, Lee Middleton, and Stefano Modafferi. Real-time crisis mapping of natural disasters using social media. *IEEE Intelligent Systems*, 29(2):9–17, 2013.

[35] Michael Morin, Irène Abi-Zeid, Pascal Lang, Luc Lamontagne, and Patrick Maupin. The optimal searcher path problem with a visibility criterion in discrete time and space. In *2009 12th International Conference on Information Fusion*, pages 2217–2224. IEEE, 2009.

[36] Fred Morstatter, Jürgen Pfeffer, Huan Liu, and Kathleen M Carley. Is the sample good enough? comparing data from twitter's streaming api with twitter's firehose. In *Seventh international AAAI conference on weblogs and social media*, 2013.

[37] Tahora H Nazer, Guoliang Xue, Yusheng Ji, and Huan Liu. Intelligent disaster response via social media analysis a survey. *ACM SIGKDD Explorations Newsletter*, 19(1):46–59, 2017.

[38] Panos Panagiotopoulos, Julie Barnett, Alinaghi Ziaee Bigdeli, and Steven Sams. Social media in emergency management: Twitter as a tool for communicating risks to the public. *Technological Forecasting and Social Change*, 111:86–96, 2016.

[39] JH Reijnierse and Jos AM Potters. Search games with immobile hider. *International Journal of Game Theory*, 21(4):385–394, 1993.

[40] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM, 2010.

[41] John Salvatier, Thomas V Wiecki, and Christopher Fonnesbeck. Probabilistic programming in python using pymc3. *PeerJ Computer Science*, 2:e55, 2016.

[42] Hiroyuki Sato and Johannes O Royset. Path optimization for the resource-constrained searcher. *Naval Research Logistics (NRL)*, 57(5):422–440, 2010.

[43] Axel Schulz, Aristotelis Hadjakos, Heiko Paulheim, Johannes Nachtwey, and Max Mühlhäuser. A multi-indicator approach for geolocalization of tweets. In *Icwsm*, pages 573–582, 2013.

[44] Jyoti Prakash Singh, Yogesh K Dwivedi, Nripendra P Rana, Abhinav Kumar, and Kawaljeet Kaur Kapoor. Event classification and location prediction from tweets during disasters. *Annals of Operations Research*, pages 1–21, 2017.

[45] Luke Smith, Qiuhua Liang, Phil James, and Wen Lin. Assessing the utility of social media as a data source for flood risk management using a real-time modelling framework. *Journal of Flood Risk Management*, 10(3):370–380, 2017.

[46] TJ Stewart. Search for a moving target when searcher motion is restricted. *Computers & operations research*, 6(3):129–140, 1979.

[47] Lawrence D Stone. *Theory of optimal search*. Elsevier, 1976.

[48] Václav Stříteskỳ, Adriana Stránská, and Peter Drábik. Crisis communication on facebook. *Studia Commercialia Bratislavensia*, 8(29):103–111, 2015.

[49] Ron Teller, Moshe Zofi, and Moshe Kaspi. Minimizing the average searching time for an object within a graph. *Computational Optimization and Applications*, 74(2):517–545, 2019.

[50] Sayan Unankard, Xue Li, and Mohamed A Sharaf. Emerging event detection in social networks with location sensitivity. *World Wide Web*, 18(5):1393–1417, 2015.

[51] Tonguç Ünlüyurt. Sequential testing of complex systems: a review. *Discrete Applied Mathematics*, 142(1-3):189–205, 2004.

[52] Sonja Utz, Friederike Schultz, and Sandra Glocka. Crisis communication online: How medium, crisis type and emotions affected public reactions in the fukushima daiichi nuclear disaster. *Public Relations Review*, 39(1):40–46, 2013.

[53] Annemijn van Gorp, Nicolai Pogrebnyakov, and Edgar Maldonado. Just Keep Tweeting: Emergency Responder's Social Media Use Before and During Emergencies. *Proceedings of the 23rd European Conference on Information Systems (ECIS 2015)*, pages 1–15, 2015.

[54] Bernhard von Stengel and Ralph Werchner. Complexity of searching an immobile hider in a graph. *Discrete Applied Mathematics*, 78(1-3):235–249, 1997.

[55] Franck Wallace. On the optimal search problem. *SIAM Review*, 7(4):503–512, 1965.

[56] RR Weber. Optimal search for a randomly moving object. *Journal of applied probability*, 23(3):708–717, 1986.