

# Online Delivery of Social Media Posts to Appropriate First Responders for Disaster Response

Viyom Mittal  
University of California, Riverside  
vmitt003@ucr.edu

Mohammad Jahanian  
University of California, Riverside  
mjaha001@ucr.edu

K. K. Ramakrishnan  
University of California, Riverside  
kk@cs.ucr.edu

## ABSTRACT

Delivering the right information to the right people in a timely manner can greatly improve outcomes and save lives in emergency response. A communication framework that flexibly and efficiently brings victims, volunteers, and first responders together for timely assistance can be very helpful. With the burden of more frequent and intense disaster situations and first responder resources stretched thin, people increasingly depend on social media for communicating vital information. This paper proposes ONSIDE, a framework for coordination of disaster response leveraging social media, integrating it with Information-Centric dissemination for timely and relevant dissemination. We use a graph-based pub/sub namespace that captures the complex hierarchy of the incident management roles. Regular citizens and volunteers using social media may not know of or have access to the full namespace. Thus, we utilize a social media engine (SME) to identify disaster-related social media posts and then automatically map them to the right name(s) in near-real-time. Using NLP and classification techniques, we direct the posts to appropriate first responder(s) that can help with the posted issue. A major challenge for classifying social media in real-time is the labeling effort for model training. Furthermore, as disasters hit, there may be not enough data points available for labeling, and there may be concept drift in the content of the posts over time. To address these issues, our SME employs stream-based active learning methods, adapting as social media posts come in. Preliminary evaluation results show the proposed solution can be effective.

## CCS CONCEPTS

• **Networks** → **Network protocols**; • **Computing methodologies** → **Natural language processing**.

### ACM Reference Format:

Viyom Mittal, Mohammad Jahanian, and K. K. Ramakrishnan. 2021. Online Delivery of Social Media Posts to Appropriate First Responders for Disaster Response. In *Adjunct Proceedings of the 2021 International Conference on Distributed Computing and Networking (ICDCN '21)*, January 5–8, 2021, Nara, Japan. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3427477.3429272>



This work is licensed under a Creative Commons Attribution International 4.0 License.

ICDCN '21, January 5–8, 2021, Nara, Japan  
© 2021 Copyright held by the owner/author(s).  
ACM ISBN 978-1-4503-8184-0/21/01.  
<https://doi.org/10.1145/3427477.3429272>

## 1 INTRODUCTION

During and in the aftermath of disasters, effective communication between citizens and the incident response team(s) can greatly help with mitigating the impacts of the incident. In disaster management, it is helpful and important to have timely and relevant information delivered to the right person(s). In the recent years, the use of social media by citizens and organizations has also dramatically changed how victims, first responders, and volunteers generate and exchange information, and seek or provide help in a disaster [7]. Often, the interactions of all these different groups are informal and ad-hoc, without a common shared “language” [18]. We seek to develop a communication framework to be used in disaster situations, that organizes these interactions, and help critical incident-related information to quickly reach the most relevant people responding to the incident.

Naming at an information layer [19] can provide a common interface organizing the relationship between how content is created and how it is forwarded towards the right recipients. Information-centric Networking (ICN) uses names as the location-independent entity used at the network layer, enabling both pull- and push-based delivery [2, 19]. In its most general form, an ICN namespace can be a graph. The namespace needs to be carefully managed for a disaster response scenario, providing a robust interface for all participants (victims, volunteers and first responders).

An integral part of our work is enabling the integration of social media into managing the disaster response. It has been shown in many recent major disasters, such as the California Wildfires [13] and the 2019 Hurricane Dorian [4], people (especially common citizens) use their familiar, normal forms of social media communications (Twitter, Facebook, etc.) to send and obtain information during disaster situations. Sometimes, it may be because of the inability to reach the traditional emergency response telephone operator (e.g., 911 in the United States). It should be assumed that most people have little or no knowledge of the notion or structure of an incident namespace required to create a named publication or interest [19]. Thus, it would be quite helpful to allow citizens to use the social media platforms in the way they are used to, i.e., as they do today, and have those social media posts be mapped to the right subset(s) of the namespace, leading to the right first responders and volunteers dealing with that specific task related to the incident.

In this paper, we propose ONSIDE (ONLine Social media delivery in DisastErs), a framework to coordinate disaster response, the many different actors participating in it, using social media, and a name-based pub/sub information dissemination architecture. A key component of ONSIDE is the Social Media Engine (SME), which maps free-form social media posts to the right part of the namespace, thus steering the pub/sub flow. Our SME is able to incrementally select the SMPs with lower confidence and use them

for additional training rather than training on all the incoming SMPs in an online manner. Due to this methodology, the classifier doesn't have to wait for an adequate amount of training data to be collected, as is traditionally needed, before it starts the classification. This approach not only reduces the number of SMPs required for labelling, but also significantly reduces the startup time for the classifier. We evaluate the applicability of our use of social media information using Tweets collected during last year's wildfires in the state of California in the United States. We showed that we were able to achieve same level of performance in terms of accuracy and F1 score, but with the SMEs requiring 30% fewer social media posts for labelling. The contributions of this paper are the following: 1) a system integrating the critical actors in disaster response, *e.g.*, first responders, and social media, in a name-based dissemination model, 2) a social media engine that intelligently and automatically maps free-form social media posts to the right names for publication in a pub/sub framework, 3) a stream-based active learning approach for classification that enables our SME to be effective in an online manner, needing very little trained data to begin with, and 4) demonstration of the effectiveness of our SME through our evaluations with actual, publicly available, fire-related Tweets.

## 2 BACKGROUND AND RELATED WORK

Many works have proposed communication and network technologies for disaster management, as well as pointing out the need for a common language between organizations and citizens and a structured, non-ad-hoc organization of disaster response [18]. In the past decade, large amount of interest has been gone into content-oriented, or information-centric (ICN) [19] design for disaster management communication; this paradigm provides important benefits, such as location-independence, enabling inherent support for timely delivery of content [14]. For better efficiency, especially when it comes to many-to-many dissemination patterns [2], Publish/Subscribe (pub/sub) methods in ICN have been proposed to support efficient, timely dissemination of information to the intended subscribers, in a multicast manner [3, 9]. The challenge with these pub/sub frameworks is the need for users to know the precise names for publishing a critical disaster-related information from the namespace, which they may not know. DiReCT [8] addresses this with an NLP/ML-based mapping of social media posts to the right names. However, it requires a pre-manually labelled fixed training sets. In addition to proposing an adaptive online solution, we also propose a two-level classification approach in ONSIDE.

Social media has been increasingly used for information dissemination in incident response, which has shown to be very beneficial [7, 13]. Server-based Social media extensions and plugins have been developed for help during disasters, providing users with useful information, such as updates and maps [7]. ONSIDE integrates social media with name-based pub/sub, to intelligently guide social media posts to the right and relevant recipients (*e.g.*, first responders), as opposed to the currently unstructured ways of social media post dissemination, *e.g.*, in form of retweets.

Active learning methods typically start with a small initial seed of labeled data, and expands it to a larger labelled trained set, requiring only a subset of it to be manually labelled. Pool-based active

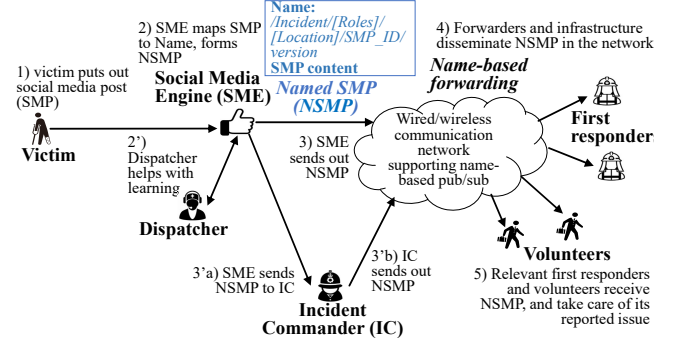


Figure 1: System model

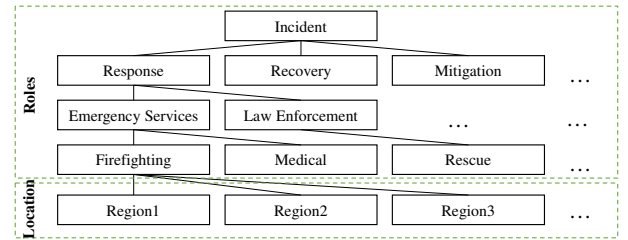


Figure 2: Incident namespace example

learning [6, 20] iteratively determines and picks (the most informative) samples from a large dataset, such as a corpus of tweets, and asks a human labeler to assign a class label to them. After this step, the model gets re-trained, and this whole process continues until a stop criteria is met. The challenge with pool-based active learning, which makes it not suitable for our system, is that it requires the whole tweet corpus to be available; in our real-world scenario, this can be almost infeasible as when a new disaster hits, there is simply not enough relevant data available to form a sufficiently large training set. Additionally, we may see a concept drift in the way people use social media as time goes on, *i.e.*, changes within incoming data warranting a learning algorithm to adapt to the changing data [16]. To address these issues, stream-based active learning methods have been proposed [12, 15]. Stream-based active learning allows data instances, *e.g.*, tweets, to be proposed one by one (or batch by batch) as they are coming in. We adapt a stream-based active learning method, expanding it to a two-level classification, and integrate it with disaster response.

## 3 ARCHITECTURE AND OVERVIEW

An overview of the operational scenario of ONSIDE and its primary actors and architectural components are shown in Fig. 1, with the goal of enabling effective communication and information dissemination during disasters.

Each actor dealing with incident response (*e.g.*, a first responder) has a role, and a task (or tasks) pertaining to specific locations and times. The relations between various roles is captured in a namespace, *e.g.*, the example in Fig. 2. This namespace unifies the interactions between the different actors, and guides the information flow. It is a graph that follows the hierarchical structure as

“/Incident/[Role]/[Location]”, where “[...]” can be any number of name components corresponding. This design is suitable to model the *what* and *where* aspects of content, which are critical aspects of incident information. The name-based pub/sub platform leverages this namespace for efficient dissemination (similar to [8]). First responders and volunteers *subscribe to names*; i.e., prefixes in the namespace, at desired granularity, and associated with their individual task. For example, fireman A dispatched to fight a fire in Region I, subscribes to “/Incident/Response/EmergencyServices/Firefighting/RegionI”. Messages to be sent, are *published to names*, based on who is the intended audience set, following the namespace graph. The fireman A in the above example, will receive all messages published to “/Incident”, “/Incident/Response”, etc. The use of pub/sub is beneficial in ONSIDE as it allows the most relevant information to reach the most relevant first responders.

In a scenario, a civilian victim puts out a *social media post* (SMP), such as a tweet, containing free-form text (a report, update, etc.), and possibly location and time information (step 1 in Fig. 1). The *Social Media Engine* (SME) collects and analyzes the SMPs, checks their incident relevance, and if relevant, maps them to the right part of the namespace, giving the SMPs the right name of form “/Incident/[Role]/[Location]/SMP\_ID/version” (step 2). The suffixes “SMP\_ID” and “version” are additional attributes (as single values) that explain the version and confidence level of the classification of this particular SMP. The SME uses NLP/ML techniques for the classification and mapping to names. It learns tweets through active learning methods, with the help of a dispatcher (step 2’). A dispatcher is similar to a 911 operator: he or she has domain knowledge about the incident, however, needs to do considerably less amount of manual classification compared to a typical 911 operator. More details on ONSIDE’s SME are provided in §4.

The name-enhanced SMP (NSMP) will be sent out to the pub/sub platform. It can be directly sent out by the SME (step 3) or alternately, and depending on the category or criticality, sent first to an incident commander (step 3’a) who will then forward it to the right individual(s) (step 3’b). Name-based pub/sub will forward the NSMPs to the right subscribers, through information-centric name-based forwarding (step 4). Eventually, first responders or volunteers relevant to the task required by the SMP will receive it and deal with it.

While our design assumes the initial interaction between users and social media servers (e.g., Twitter) to be over the Internet, the communications in the pub/sub framework, e.g., between first responders and volunteers, can be either infrastructure-based (with fixed routers), infrastructure-less (with device-to-device data mules), or a hybrid of both. ICN allows for information delivery over such diverse links [14], which is a major benefit of this approach.

## 4 SOCIAL MEDIA ENGINE

The procedures for SME’s learning, prediction, and name mapping are shown in Fig. 3. As shown in Fig. 3, the SME maintains two levels of classifiers trained in a streaming manner using active learning [6, 20], to predict the relevance of the SMP and then map it to the right name(s) in the namespace (e.g., Fig. 2) for pub/sub dissemination. The L1 classifier classifies the incoming Social Media Posts (SMPs), e.g., tweets, as either ‘Relevant’ or ‘Irrelevant’ based

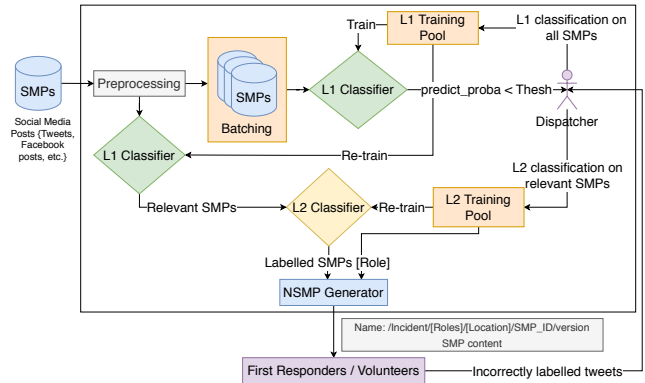


Figure 3: Social Media Engine

on the SMP’s content. All the SMPs classified as ‘Relevant’ are then passed to the L2 classifier which classifies them into various “roles” (Fig. 2) such as ‘Animal Rescue’, ‘Contamination and Pollution Issues’, ‘EMT’, ‘Firefighting’, etc. In the remainder of this section, we describe the different components of SME.

**Preprocessor.** All the incoming SMPs first go through the preprocessing stage (Fig. 3). In this stage, the operations such as removal of special characters, symbols, numbers, emojis, conversion to lowercase, remove URLs and email addresses etc., are performed on the SMPs. This is followed by the stop word removal [10]. This component also utilizes tf-idf [11] for vectorization. A copy of each SMP is passed to the L1 classifier (for fast publication towards the first responders) and another copy is sent to the Batch (for enhancements of the model through the active learning algorithm).

**Batcher.** The SMPs that are sent to the queue are grouped into batches of fixed size, i.e., the *BatchSize*, which is pre-configured in ONSIDE. Once there are enough tweets in the batch, the batch of SMPs is forwarded to the L1 classifier, so that the prediction probabilities are calculated.

**L1 Classifier.** The L1 classifier is a binary classifier to check an SMP’s relevancy, and has two major roles: 1) Directly classify SMPs received from the Preprocessor to minimize delay. Based on the L1 classification, then forward ‘Relevant’ SMPs to the L2 classifier. 2) The batch of SMPs that the L1 classifier receives from the Batcher are used for selecting the SMPs with lower prediction probability for further training. The L1 classifier returns a *prediction probability* of each SMP for both ‘Relevant’ and ‘Irrelevant’ classes. If the SMP has prediction probability for both the classes less than a threshold, *Thresh*, it is collected and sent to the dispatcher for labelling. The L1 classifier gets re-trained every time the L1 Training Pool is updated, which helps the SME’s model adapt to new incoming SMPs and concept drift as the events of the disaster unfold.

**L2 Classifier.** The L2 classifier classifies the relevant SMPs into various disaster-related categories, each associated with a role in the incident namespace (Fig. 2), and hands them to the Named SMP (NSMP) Generator. The L2 classifier also gets re-trained when the L2 Training Pool is updated with new labelled SMPs by the Dispatcher.

**Dispatcher.** An individual (or a set of them) we call a ‘Dispatcher’ helps label the SMPs. The L1 classifier forwards SMPs which have low prediction probability and need manual labelling

**Table 1: L1 class examples**

L1 Class	Tweet Text
Irrelevant	No wonder it smells like campfire in the city today
Irrelevant	Idk what's worse paper cuts or carpet burn. Both are gifts from the devil.
Relevant	Evacuations continue (livestock, too) as #CampFire spreads toward the outskirts of Chico. <a href="https://t.co/NlfphE6xc">https://t.co/NlfphE6xc</a>
Relevant	Well, this is too close to home. This park is walking distance from me. I just saw several fire trucks go by and no ... <a href="https://t.co/QjyQFjNunO">https://t.co/QjyQFjNunO</a>

to the Dispatcher. First responders and volunteers also send the SMPs in case of incorrect name mapping. The Dispatcher provides L1 and L2 class labels to those SMPs and adds them to the L1 and L2 Training Pools respectively. The Dispatcher can also add a new sub-class (categories) to the L2 Training Pool if there are sufficient SMPs belonging to that sub-class. In the beginning when there is insufficient trained data, the Dispatcher receives a large number of SMPs per batch from the L1 classifier. As the classifier is re-trained with additional data in subsequent batches, this number reduces and stabilizes around a fixed value. This results in low variance for the number of SMPs to be labeled, which acts as an indicator that the classifier is sufficiently trained, and the Dispatcher can stop labelling all the new data. The Dispatcher then picks data randomly for labelling. During a stable run, if the Dispatcher receives a large number of SMPs from the first responders, we can again start labelling all the SMPs received from the L1 classifier.

**L1 and L2 Training Pools.** The L1 and L2 Training Pools contain the L1- and L2- classified SMPs (as trained model used for prediction) respectively. These pools are updated by the Dispatcher to re-train the respective classifiers. Along with this, the L2 Training Pool also transfers the new SMPs it receives from the Dispatcher to the NSMP Generator component for the SMP-to-name mapping.

**NSMP Generator.** The Named SMP (NSMP) Generator uses the L2 prediction of the SMP to map it to the correct Role, and the Location extracted from the SMP as features to convert into a Named SMP with the format as “/Incident/[Role]/[Location]/SMP\_ID/version”. The ‘SMP\_ID’ can be any value uniquely identifying an SMP (e.g., tweet ID) for duplicate detection purposes at the first responder end device. The ‘version’ is a counter value, starting from 0, showing the version of this SMP’s publication and the confidence level in its assignment. A version value of  $n$  means that this is the  $n$ th duplicate of this SMP’s name assignment and publication. NSMP Generator maintains the version values of recently published SMPs as history. Upon a new name assignment different from the prior assignment, it publishes the SMP with an increased version value. The version value helps the recipients determine the order of duplicate SMPs, prioritizing the most recent one for their tasks.

## 5 EVALUATION

We have implemented ONSIDE’s SME and used tweets (as SMPs) collected from the region impacted by the Camp Fire [17] (in Northern California, 2019), similar to [8]. We picked tweets sent throughout November 9, 2018, the peak intensity day. We picked ~1,700 tweets that contain disaster-related terms (such as ‘CampFire’, ‘Fire’, ‘help’,

**Table 2: L2 class examples (corresponding to roles)**

L2 Class	Tweet Text
Animal	#CampFire Animal Evacuations At Erickson Veterinary Hospital
Contamination	Smoke plume blowing from Chico area #campfireparadise <a href="https://t.co/8HYNvY9f9J">https://t.co/8HYNvY9f9J</a>
EMT	Now my eyes are starting to burn. Everyone be very careful, try and stay safe. Going to put wet towel over my eyes.
Fire	We have the #CampFire in #ButteCo and this in LA!
News	Paradise Lost: The Camp Fire Has Reportedly Burned an Entire Town to the Ground <a href="https://t.co/CnN7lysEX4">https://t.co/CnN7lysEX4</a> via @EARTH3R
Other	Praying for everyone affected by these fires
Shelter	Evacuation of Paradise Intermediate School #CAMPFIRE <a href="https://t.co/qyJf1RHFZP">https://t.co/qyJf1RHFZP</a>
Structure	Just found out the house my dad’s lived in my whole life (up until a couple years ago) has also burned up in the ...
Volunteer	We are willing to take in anyone displaced by the Camp Fire. DM me if you need a place to stay.

**Table 3: Accuracy of L1**

Thresh/BatchSize	10	20	25	50	75	100
<b>0.50</b>	0.36	0.36	0.64	0.64	0.64	0.64
<b>0.60</b>	0.79	0.78	0.77	0.78	0.76	0.76
<b>0.70</b>	0.80	0.79	0.80	0.79	0.79	0.78
<b>0.80</b>	0.80	0.81	0.79	0.80	0.80	0.80
<b>0.90</b>	0.80	0.80	0.80	0.80	0.81	0.80
<b>1.00</b>	0.80	0.80	0.80	0.80	0.80	0.80

**Table 4: Recall of relevant class of L1**

Thresh/BatchSize	10	20	25	50	75	100
<b>0.50</b>	0.00	0.00	1.00	1.00	1.00	1.00
<b>0.60</b>	0.89	0.86	0.96	0.93	0.94	0.96
<b>0.70</b>	0.91	0.88	0.89	0.94	0.93	0.95
<b>0.80</b>	0.90	0.90	0.89	0.94	0.89	0.90
<b>0.90</b>	0.90	0.90	0.90	0.89	0.90	0.89
<b>1.00</b>	0.90	0.90	0.90	0.90	0.90	0.90

**Table 5: # manual labelling for L1**

Thresh/BatchSize	10	20	25	50	75	100
<b>0.50</b>	10	20	25	50	75	100
<b>0.60</b>	428	394	278	281	153	305
<b>0.70</b>	596	637	626	568	598	567
<b>0.80</b>	832	819	834	831	816	848
<b>0.90</b>	1024	1031	1045	1032	1040	1044
<b>1.00</b>	1190	1190	1190	1190	1190	1190

etc.). For L1 classification, we manually labeled those tweets as either ‘Relevant’ or ‘Irrelevant’. For L2 classification, we labeled them based on the following categories (each associated with a role in the namespace, inspired by the FEMA reports [5]): ‘Animal’



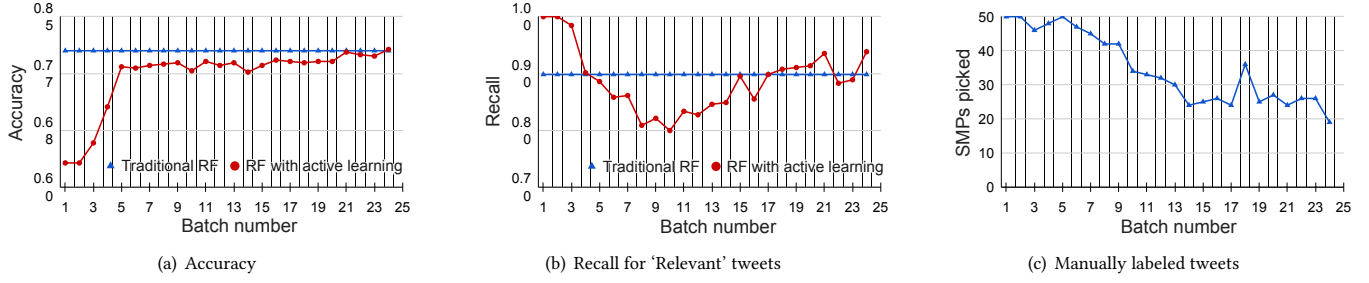


Figure 4: Metrics for comparison of L1 classifier with active learning against the traditional L1 classifier

(Animal Rescue), 'Contamination' (Contamination and Pollution Issues), 'EMT', 'Fire' (Firefighting), 'News', 'Other' (Disaster-relevant, but not classified in the detailed categories), 'Shelter' (Shelter and Evacuations), 'Structure' (Structure Damages) and 'Volunteer' (Volunteering). Table 1 and 2 show examples of tweets falling in various classes for L1 and L2 classification, respectively. There were tweets related to classes such as 'Road' (Road Blockage), 'Shortage' (Shortage of Foods and Resources), 'Debris' (Debris and Ash). But, since there were very few tweets in these classes in our dataset, we merged them to the 'Other' class.

We compared the performance of ONSIDE's Social Media Engine (SME) across different active learning options as well as with a baseline using traditional (non-active) learning, such in [8]. For all the experiments, we used the the last 30% of data for testing which represent the tweets collected after 7pm, for a fair comparison. We used a Random Forest classifier (RF) [11] for both L1 and L2 classification. For a more robust configuration of learning, we tune the hyperparameters using the Parfit tool [1] with the same dataset. In a real world scenario, the hyperparameter tuning may be done a priori on a previously available dataset. In case of RF classifier, the *prediction probability* which is used to compare against the *Thresh*, is the mean predicted class probabilities of the trees in the forest.

An important aspect of the SME's procedure with active learning is to have a reasonable value for the *BatchSize* and *Thresh*. For this purpose we used the approach discussed in [6] where the accuracy and samples used for training were calculated for different combinations of *Thresh* and *BatchSize*. For the L1 classifier, along with comparing the accuracy and the total samples that were sent out to the Dispatcher for labeling, similar to [6], the recall for the 'Relevant' class is also taken into account. A high recall value is required so that there are fewer false negatives. This ensures that the L1 classifier classifies a majority of 'Relevant' tweets correctly and forwards them for L2 classification.

Tables 3–5 show the accuracy and recall for the 'Relevant' class and total samples used for training the L1 classifier. In Table 5, for the model with *Thresh* value of 0.5, no tweet was selected for labeling after the tweets from the first batch, given the binary nature of the L1 classifier. The different values in Tables 3–5 can be compared to find the best combination of *Thresh* and *BatchSize* to get maximum accuracy and recall for 'Relevant' class while minimizing the value of samples picked for training. With *Thresh* as 0.8 and *BatchSize* as 50, the model uses only 834 labeled tweets and is able to achieve almost same accuracy and better recall than

the case where all the 1190 tweets were used for training (*i.e.*, the traditional learning approach).

**L1 Classifier Evaluation.** We compare the traditional RF classifier with the RF classifier with active learning. The traditional RF is trained over complete training set (70% of the data). For RF with active learning, we created several batches with the initial training set (70% data). For each batch, only tweets with a *prediction probability* less than *Thresh* are added to the training pool for re-training the classifier. Fig. 4(a) shows the accuracy comparison of traditional RF classifier when trained over all the 1190 tweets with the active learning RF classifier for L1 classification. The accuracy of the active learning approach increases as additional batches are used for training and reaches close to the accuracy of the traditional RF classifier.

Fig. 4(b) presents the recall value for the 'Relevant' class with traditional RF and RF with active learning over the batches for L1 classification. The initial values of recall for the active learning approach is 1. After a few batches, the value of recall decreases as the classifier starts to classify tweets into the two classes. This value then starts to increase again additional tweets are added to the training pool. By the end of all the batches, the model is able to achieve better recall than the traditional approach.

Fig. 4(c) presents the number of tweets per batch which have a lower prediction probability than *Thresh* for both the classes in L1 classification. These tweets were added to the training pool directly as they were already labeled. The initial classifier is not able to provide a good prediction probability and therefore it is lower than the *Thresh* and all the tweets are added to the training pool. As the data in training pool increases, the re-trained classifier is able to predict tweets in the newer batches with higher confidence. In this case, total 834 tweets were used for training which shows that the effort of labeling can be reduced by almost 30% by using the active learning technique.

**L2 Classifier Evaluation.** Since all the Relevant tweets were also labeled with their sub-classes such as 'Fire', 'EMT', 'Contamination', *etc.*, these sub-classes were used for the evaluation of the L2 classifier. Out of the training dataset containing 1190 tweets, 715 number of tweets were 'Relevant' and had L2 classification. These 715 tweets were used for training. Similarly, from the testing set containing 510 tweets, 324 tweets were 'Relevant' and were used as the L2 testing set.

To evaluate the L2 classifier with active learning, we compared it with the traditional model which is trained over all the 715 tweets.

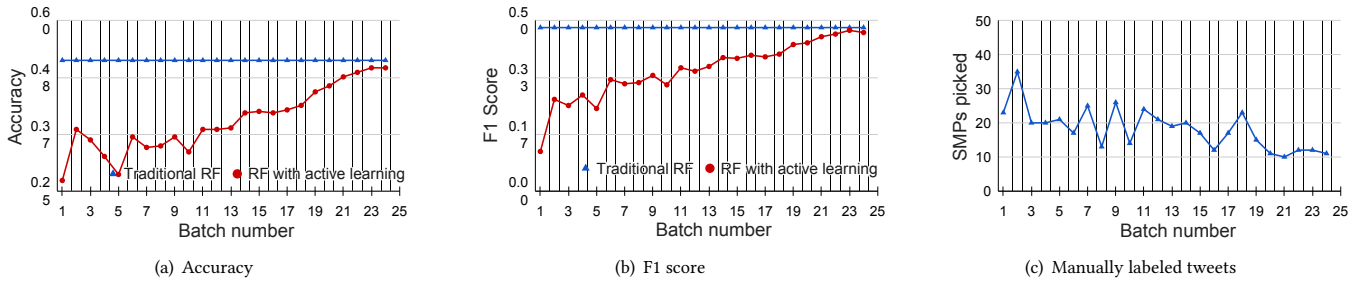


Figure 5: Metrics for comparison of L2 classifier with active learning against the traditional L2 classifier

Figures 5(a) and 5(b) provides the accuracy and F1 score on the test set. In the case of L2 classification, since all the classes must have a good value of both precision and recall, we considered the average F1 score for comparison. The RF classifier with active learning is able to achieve almost the same level of accuracy and F1 score as the traditional model by the end of processing all the batches. Fig. 5(c) shows the number of tweets picked per batch for training. Here we see a trend similar to the L1 classifier where the sample count decreases as the model is trained over more data. It stabilizes after several iterations. Overall, only 438 tweets were used out of the 715 tweets for training. So, the L2 classifier with active learning was able to reduce the data labeling by almost 39% at the cost of slight reduction in accuracy and F1 score (0.015% for both).

From the results, we observe the effectiveness of ONSIDE, in ensuring that the majority of tweets get correctly detected as disaster-relevant, assigned the right name, and delivered to the right first responders, all with progressively increasing accuracy as more tweets come in. This is especially important given that we assume not having pre-labeled training data, and that the general user does not know anything about the namespace, and we are performing the mapping automatically. The small percentage of tweets that may be inaccurately delivered can be appropriately forwarded manually afterwards. We can further enhance our system with incremental learning (for more efficient re-training) and also re-classifying the previously incorrectly predicted tweets after each re-training, which we are currently working on. Our preliminary results show the good performance of our learning/inference at the social media engine, indicating the effectiveness of ONSIDE in mapping social media posts to the right names, leading to its delivery to the relevant first responders and volunteers.

## 6 CONCLUSION

We proposed ONSIDE, a framework to coordinate disaster response with first responders that receive timely relevant information. ONSIDE bridges free-form generally used social media platforms with a pub/sub dissemination architecture for specific disaster management purposes. It employs a naming schema, and an NLP/ML-based social media engine, that learns latent text patterns for classification in an online manner, using stream-based active learning, and maps social media posts to the right name(s) or role(s) of first responders. Results from our preliminary evaluation show that ONSIDE is effective and efficient in providing the mapping between free-form social media text and pub/sub-based names with reasonably

high and increasing accuracy, requiring little initial training data or manual labeling.

## ACKNOWLEDGMENTS

This work was supported by the US National Science Foundation grant CNS-1818971 and the US Dept. of Commerce, NIST PSIAP award 70NANB17H188. We thank Prof. Amr Magdy of UCR for providing us with the dataset of tweets.

## REFERENCES

- [1] Jason Carpenter. 2020. Parfit. <https://github.com/jmcarpenter2/parfit>.
- [2] Jiachen Chen et al. 2011. COPSS: An efficient content oriented publish/subscribe system. In *ANCS*.
- [3] Jiachen Chen et al. 2016. CNS: Content-oriented Notification Service for Managing Disasters. In *ICN*.
- [4] Ellen Cranley. 2019. 'Our government failed us': Bahamians were left to coordinate rescue efforts on social media after Hurricane Dorian. <https://tinyurl.com/y3kvvmce>.
- [5] FEMA. 2018. State and Federal Partners Respond to the California Wildfires. <https://tinyurl.com/y4o7zmk9>.
- [6] Mohamed Goudjil et al. 2018. A novel active learning method using SVM for text classification. *IJAC* 15, 3 (2018).
- [7] Mohammad Jahanian et al. 2018. The Evolving Nature of Disaster Management in the Internet and Social Media Era. In *LANMAN*.
- [8] Mohammad Jahanian et al. 2019. DiReCT: Disaster Response Coordination with Trusted Volunteers. In *ICT-DM*.
- [9] Mohammad Jahanian, Jiachen Chen, and K. K. Ramakrishnan. 2019. Graph-based Namespaces and Load Sharing for Efficient Information Dissemination in Disasters. In *ICNP*.
- [10] Edward Loper and Steven Bird. 2002. NLTK: the natural language toolkit. *arXiv preprint cs/0205028* (2002).
- [11] F. Pedregosa et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* (2011).
- [12] Daniela Pohl, Abdelhamid Bouchachia, and Hermann Hellwagner. 2018. Batch-based active learning: Application to social media data for crisis management. *Expert Systems with Applications* 93 (2018).
- [13] Mark Saunders. 2018. Social media: California wildfires force thousands to evacuate. <https://tinyurl.com/y2pfolyd>.
- [14] Jan Seedorf et al. 2015. The benefit of information centric networking for enabling communications in disaster scenarios. In *2015 IEEE Globecom Workshops (GC Wkshps)*.
- [15] Jasmina Smilović et al. 2014. Stream-based active learning for sentiment analysis in the financial domain. *Information sciences* 285 (2014).
- [16] Geoffrey I Webb et al. 2016. Characterizing concept drift. *Data Mining and Knowledge Discovery* 30, 4 (2016).
- [17] Wikipedia. [n.d.]. Camp Fire (2018). [https://en.wikipedia.org/wiki/Camp\\_Fire\\_\(2018\)](https://en.wikipedia.org/wiki/Camp_Fire_(2018)).
- [18] L Yan. 2011. A survey on communication networks in emergency warning systems. *Sci. Comput* (2011).
- [19] Lixia Zhang et al. 2014. Named data networking. *SIGCOMM CCR* (2014).
- [20] Ye Zhang, Matthew Lease, and Byron C Wallace. 2017. Active discriminative text representation learning. In *AAAI*.