Cloud based Social and Sensor Data Fusion

Surender Reddy Yerva
EPFL, Lausanne, Switzerland
surenderreddy.yerva@epfl.ch

Hoyoung Jeung
SAP Research, Brisbane, Australia
hoyoung.jeung@sap.com

Karl Aberer
EPFL, Lausanne, Switzerland
karl.aberer@epfl.ch

***Abstract*—As mobile cloud computing facilitates a wide spectrum of smart applications, the need for fusing various types of data available in the cloud grows rapidly. In particular, social and sensor data lie at the core in such applications, but typically processed separately. This paper explores the potential of fusing social and sensor data in the cloud, presenting a practice—a travel recommendation system that offers the predicted mood information of people on where and when users wish to travel. The system is built upon a conceptual framework that allows to blend the heterogeneous social and sensor data for integrated analysis, extracting weather-dependent people's mood information from Twitter and meteorological sensor data streams. In order to handle massively streaming data, the system employs various cloud-serving systems, such as Hadoop, HBase, and GSN. Using this scalable system, we performed heavy ETL as well as filtering jobs, resulting in 12 million tweets over four months. We then derived a rich set of interesting findings through the data fusion, proving that our approach is effective and scalable, which can serve as an important basis in fusing social and sensor data in the cloud.**

I. Introduction

Mobile phones increasingly become multi-sensor devices, accumulating large volumes of data related to our daily lives. At the same time, mobile phones are also serving as a major channel for recording people's activities at social-networking services in the Internet. These trends obviously raise the potential of collaboratively analyzing sensor and social data in mobile cloud computing—where applications running in the cloud are accessed from thin mobile clients, providing virtually unlimited processing power, and promising cross-device platform compatibility.

The two popular data types, social and sensor data, are in fact mutually compensatory in various data processing and analysis. Participatory sensing, for instance, enables to collect people-sensed data via social network services (e.g., Twitter) over the areas where physical sensors are unavailable. Simultaneously, sensor data is capable of offering precise context information, leading to effective analysis of social data. Obviously, the potential of blending social and sensor data is high; nevertheless, they are typically processed separately in mobile cloud applications, and the potential has not been investigated sufficiently.

In this paper, we explore the possibility of fusing social and sensor data in the cloud, while dealing with massive data streams. To this end, we present a travel recommendation system as a practice of the fusion, which offers the information of people's moods regarding the predicted weather on where and when users wish to travel. The recommendation system gears various components towards effective, large-scale social and sensor data fusion. We summarize the salient features of the system in the sequel.

- First, we propose a conceptual framework that enables to integrate and analyze the heterogeneous social and sensor data. Specifically, the framework first transforms tweets into data points in a *mood space* which consists of 12 subspaces, each of which corresponds to a mood (e.g., happy). We then derive the probability of each mood in the mood space from a large number of tweet data points accumulated over time. The system computes and maintains the mood probability information separately according to day (e.g., Monday), place (e.g., London), and weather (e.g., sunny), which are the major dimensions in query processing.
- Second, we present a scalable fusion system that implements the conceptual framework, extracting the weather-dependent mood information from real-time Twitter and meteorological sensor data. Our travel recommendation system is established upon a combination of several well-known systems typically used for large-scale data store and analysis in the cloud, such as Hadoop [1], HBase [2], and GSN [3]. This allows us to perform ETL jobs as well as analytic processing over massively streaming data.
- Third, we offer in-depth analysis of our data-fusion approach on comprehensive experimental results, obtained from using 12 million tweets as well as meteorological sensor readings collected over four months. The results demonstrate various interesting findings, including the degree of happiness according to a particular weather type, day, and location. Furthermore, we statistically prove that our mood estimation based on the fusion is effective and accurate.

We believe that the approach proposed in this paper can set a firm yard-stone in scalable social and sensor data fusion, serving as an important foundation in further studies towards mobile cloud computing.

The rest of the paper is organized as follows. Section II summarizes the related work. Section III describes in detail the theoretical framework for fusing social and sensor data, while Section IV presents the technical details as well as data collections used in our travel recommendation system. Section V offers experimental analysis on the data fusion, followed by the conclusions in Section VI.

II. Related Work

Social-network services facilitate users to share their ideas, opinions, pictures, videos, news, and other various forms of contents in the Web. Such social data typically contains highly valuable information, aiding a wide range of applications; for example, allowing social scientists to understand human behaviors, companies to figure out their customers' preferences, and news agencies to identify significant news etc. Previously, it was difficult to obtain the rich set of social information, or required large amounts of laborious human efforts like conducting surveys, interacting with the users. With the advent of Web 2.0, all this information is readily available, leading to a variety of interesting research directions. In this section, we summarize three research lines which are closely related to this study.

A. Mood analysis on tweets

One popular research line on social data is to extract and analyze mood information from Twitter messages [4], [5], [6], [7], [8]. In [4], micro-blogs are used for mood analysis, presenting a method for mood association to certain events. Mood analysis in the method is then performed through a Naive Bayes classifier trained on unigram features. Similarly, the authors in [6] extract "Pulse of Nation" that shows the moods of the United States nation from Twitter messages. They present the moods across the country using different cartograms, describing the variation of mood over 24-hour period, as well as the days of a week.

Another study [5] tries to predict the impact of public mood expressed in Twitter messages on the stock market, investigating the correlation of moods inferred from large-scale twitter feeds with the Dow Jones Industrial Average. They make use of mood tracking tools, namely OpinionFinder (that measures positive vs. negative mood) and Google-Profile of Mood States(GoPMS) that measures mood in terms of 6 dimensions.

The authors of [7] analyze Twitter messages in order to study why certain events resonate well with the population. They assess whether surges of interest in Twitter are associated with heightened emotions, by checking if the average sentiment strength of popular Twitter events is higher than the Twitter average or by assessing whether an important event within a broad topic is associated with increased sentiment strength.

In [8], Twitter data is used as corpus for sentiment analysis and opinion mining, where Twitter becomes a media in which people readily express their opinion. Specifically, the Twitter data is served for training their sentiment classifier, which classifies tweets as expressing positive, negative or neutral sentiment.

B. Social sensing

Given the importance of sensor networks in our everyday activities, some studies [9][10] went ahead and consider the people participating in micro-blogs or social networks as social sensors providing the rich social context, which are hard to infer using physical sensors. For instance, the work in [11] monitors the flows of Twitter messages for quickly detecting an earthquake that occurs in an area where seismic sensors are unavailable. Another study [12] mines the Twitter messages to identify relevant events to given monitoring conditions. Yerva et al. [13] also identify the tweets relevant to ambiguous company entities for its advertising strategies.

C. Social data fusion

CitizenSensing [9] gives a broad overview of the challenges involved in making sense of citizen sensing, which is becoming rampant with ubiquitousness of the mobiles, sensing devices etc. The study introduces the paradigm of Citizen Sensing, enabled by Mobile sensing and Human Computing – humans acting as citizens on the ubiquitous Web, acting as sensors and sharing their observations and view through Web 2.0. Likewise, SocialFusion [14] proposes the use of sensor networks to enable context-aware social applications, analyzing the data generated by the users of the applications. In addition, SocialSensors [10] describes the need for fusing social data with pervasive sensors for better services.

The authors in [15] present heuristic methods for data fusion that combine the user's personal calendar with his social network posts, in order to produce a real-time multi-sensor interpretation of the real-world events. Their study shows that the calendar can be significantly improved as a sensor and indexer of real-world events through data fusion.

III. The Fusion Framework

This section describes three major components of the theoretical framework in our data fusion approach, which are *fusion base, data points*, and *mood probabilities*.

A. Fusion Base: Mood Space

A key goal of this study is to establish a data-fusion approach that collaboratively analyzes both social and sensor data. In particular, we aim to extract people's mood information from social (Twitter) feeds associated with sensor (weather) data. To this end, we propose a data space, called *mood space*, which serves as a conceptual base-ground where social and sensor data can be mapped.

More specifically, we represent the mood of a word (e.g., appearing in a tweet) using the ANEW[16] list, which describes a set of major words frequently appeared in people's conversations as numerical scores. In ANEW, each of such words is scored in three dimensions: *valence*, *arousal* and *dominance*, where the value in each dimension ranges from 1 to 9. Valence is defined by its two poles negative/bad and positive/good, whereas the arousal dimension spans between the two poles sleepy/calm for very low arousal and aroused/excited for very high arousal. Valence and arousal have proven to be the two main dimensions, accounting for most of the variance observed. An additional dimension called dominance is proposed to differentiate subtle emotions like fear and anger (which have similar valence and arousal values).

In this study, we consider the mood space to be defined by valence and arousal metrics, illustrated as Fig. 1. The two dimensional plane VxA:[1,9]x[1,9] is divided into 12 regions, each region maps to a certain mood. For example, a high value of valence and another high value of arousal indicates someone is happy, labeled as "happy" in the figure. Similarly, a lower value for valence and a high value for arousal maps to the mood of "annoying/rage".

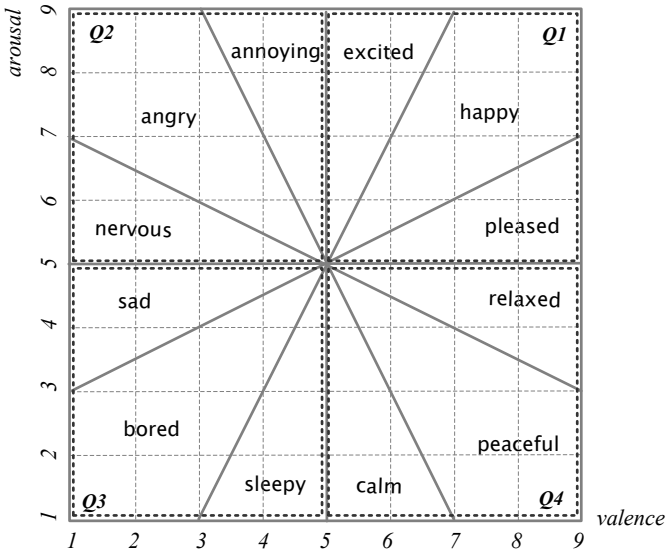


Fig. 1. Illustration of the mood space.

B. Data Points: Tweet Mapping

Given a tweet message, the next step in our fusion approach is to associate a mood label with the message, by computing the valence and arousal scores of the tweet. Specifically, the tweet is decomposed into words, each of which would have a valence and arousal score, then we resort to Naive Bayes setting in order to compute the tweets' overall valence and arousal score which would become a data point in the mood space.

Formally, consider a set of moods M consisting of the 12 moods in the mood space. Given a tweet set $T = SET\{T_i\}$, for each tweet $T_i \in T$, first we try to infer the mood expressed by the tweet by computing the conditional probabilities P($M_k|T_i$) for all $M_k \in M$.

For computing the conditional probabilities P($M_k|T_i$), we resort to Naive Bayes setting. We consider a Tweet T_i as a bag of words, T_i = $set\{wrd_j^i\}$, and we assume each word expresses certain mood (the words which do not express mood will make zero contribution to the final mood). We assume each word independently contributes to the overall mood of the Twitter message.

$$\begin{aligned} P(M_k \mid T_i) &= \frac{P(M_k) * P(T_i \mid M_k)}{P(T_i)} \\ &= \frac{P(M_k) * P(wrd_1^i, \ldots, wrd_n^i \mid M_k)}{P(T_i)} \\ &= C_k \prod_{j=1}^{n} P(wrd_j^i \mid M_k) \end{aligned} \quad (1)$$

Each of the terms in the above product, P($wrd_j^i \mid M_k$), can be interpreted as the amount of contribution a particular word makes towards a mood M_k, which can be learnt based on the training set or one could readily use the weights provided by prior studies like the one in creating the ANEW list [16]. Along with the term weights, we also compute the constant C_k based on the training set. Depending for which mood M_k, the term $P(M_k|T_i)$ is largest, we classify the tweet T_i as expressing that mood.

For example, consider the following tweet T0 : *"Weather here is seasonal, warmish, some rain and sun, green and beautiful"*. This tweet is composed of 12 words, in which four of them are listed in the ANEW set of words. For these four words, we obtain the valence scores of (rain= 5.08; sun=7.55; green=6.18; beautiful= 7.60) and arousal scores of (rain= 3.65; sun=5.04; green=4.28; beautiful= 6.17) by looking up the ANEW list. Finally, applying the above procedure, we get the overall tweet valence and arousal scores as (6.60,4.78) which forms a data point in our mood space and gets a "relaxed" mood label.

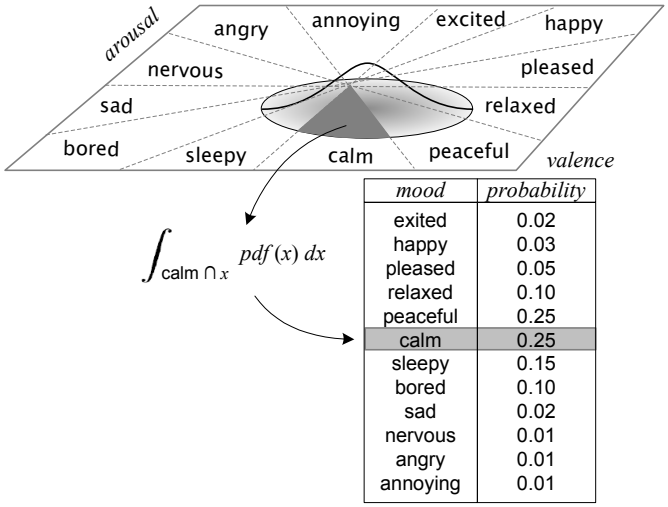
C. Mood Probabilities

Given a set of data pointed in the mood space, derived from raw tweets, we next explain how the fusion framework computes a set of mood probabilities, according to day, location, and weather.

We know that each Tweet T_i carries the information about the location L, the time stamp t and the weather label W. Thanks to the analysis explained above, now the tweet also carries the mood M_i information. Now for each tweet $T_i \in T$ we have a record R : (T_i,L,t,W_j,M_i). Essentially each tweet now maps as a point in the 2D mood space. The complete set of twitter data maps onto the 2D mood space as a distribution of points. For easier querying our next goal is to summarize the distribution of points on the social metric space.

Once we have all the tweet records R's, one can summarize the mood-weather information using p_{ijk} probabilities. The p_{ijk} represents the probability of witnessing mood M_i when the weather is W_j and the day is $D_k \in$ {Monday,. . .,Sunday} i.e., the conditional probability $P(M_i \mid W_j, D_k)$. One can consider different models for computing this p_{ijk} probabilities, ranging from simple model which summarizes all the events so far ignoring the temporal aspects like time, weekday etc., to far more sophisticated models which give more importance to the recent events.

According to the simple model, we group all the tweet records corresponding to a particular location L. We observe



mood	*probability*
exited	0.02
happy	0.03
pleased	0.05
relaxed	0.10
peaceful	0.25
calm	0.25
sleepy	0.15
bored	0.10
sad	0.02
nervous	0.01
angry	0.01
annoying	0.01

Fig. 2. An example of mood probability computation.

different weather labels W_j, mood labels M_i and day labels D_k information associated with each of these tweets. Next we compute, p_{ijk} (shown in eqn. 2), as the fraction of tweets expressing certain mood M_i for a particular weather label W_j and the day D_k.

$$p_{ijk} = \frac{\#(\text{tweets with } M_i, W_j \text{ and } D_k)}{\sum_{a=1}^{12} \#(\text{tweets with } M_a, W_j \text{ and } D_k)} \quad (2)$$

If we plot all the tweets satisfying the conditions of having certain weather label W_j and are on certain day D_k, as points on the mood-space, one would expect to see distribution of points over each mood space similar to the one shown in Fig. 2. For a particular day and weather label, the fusion process helps us to obtain the probability distribution over the mood spaces. These probability distributions will be summarized for all days and weather labels combinations and will be used as source of useful input to the travel recommendation system.

IV. The Travel Recommendation System

This section introduces a travel recommender system as a practice of the data-fusion framework described in the previous section. We first offer an overview of how the system works, and then describe each component of the system, as well as data processing.

A. Overview

The intuition behind the system development is to show that the information derived from various, real-time data fusion can enrich recommendations, compared with using solely static, limited-scope reviews posted by experts or other consumers.

In our recommendation system, users provide their travel intentions (place and approximate date of travel), and then the system provides the information of how enjoyable the place would be on the day for travel, in addition to the typical information offered by ordinary travel recommender systems. This recommendation process is comprised of the following steps:

1) A user first offers the details for travel to the system, e.g., going to London next Friday.
2) The system obtains the information of predicted weather on London next Friday, from a real-time weather prediction service (e.g., WeatherUnderground).
3) The system looks up the mood information of people associated London and Friday, which is continuously mined and updated from raw social and sensor data.
4) The system offers the information of how enjoyable the trip to London on next Friday would be, according to the mood probability estimation.

Note that our fusion system is flexible to blend other data sources with the social and sensor data, in order to make the recommendation more meaningful. For example, taking into account the events (e.g., death of a famous person, terrorism) occurring in London would be able to enrich the quality of recommendation. We believe that such an additional data source can be easily fused in the framework of the recommendation system.

B. System Architecture

In order to store and process massively streaming social and sensor data in the cloud, we propose a system established upon a combination of state-of-the-art cloud systems, including Hadoop [17], HBase [2], and GSN (Global Sensor Network) [18]. Fig. 3 shows an overview of the system, which consists of three primary components. In the sequel, we describe in detail each of the components.

- *GSN* is a stream processing engine that supports a flexible integration of data streams. It has been used in a wide range of domains due to its flexibility for distributed querying, filtering, and simple configuration. In our travel recommendation system, GSN serves as a wrapper that receives streaming social as well as sensor data from twitter and weather data sources. GSN provides means to control the rate of data streams, and also allows us to parse and filter incoming data on the fly, before the data are stored in the back-end.
- *Back-End* contains both Hadoop and HBase, serving as a storage-and-computing platform. Hadoop (MapReduce) is a popular framework for data-intensive distributed computing of batch jobs. In particular, it is very useful for "cooking" massive raw data into useful information that is consumed by another storage system. In our system, Hadoop is used to parse continuously streaming tweets as well as weather data delivered in an XML format, based on a cluster that is built on 16 machines. The parsed data are then stored in HBase, which is commonly used as a "Hadoop storage".
- *Front-End* implements a user interface of the recommendation system. Specifically, this component takes user inputs for querying, and delivers the inputs to the back-

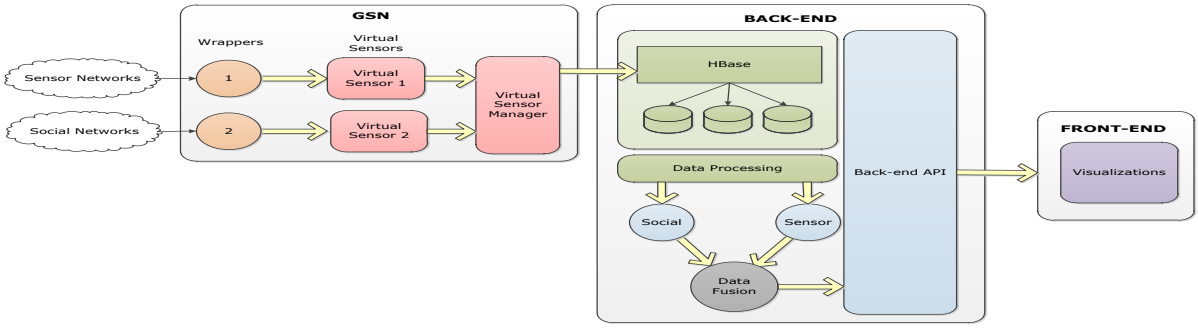


Fig. 3. Architecture of the fusion system.

end. The query results returned from the back-end are then visualized through the front-end.

C. Data Processing

The travel recommendation system computes and maintains a set of 2D maps of Weekdays (D_i) x Weather-Labels (W_j). The cells in each map stores the mood probabilities computed by analyzing the data points of tweets mapped to the mood space, as described in Section III. Figures 4(a),4(b) and 4(c) shows the visualization of these 2D maps, where each subfigure corresponds to a distinct weather label. The system manages seven different discs corresponding to seven days in a week (Mon, Tue, ..., Sat, Sun). The outermost disc corresponds to Monday while innermost corresponds to Sunday. Each disc contains the different mood distributions computed through the data fusion process.

The system computes each entry of the discs using a massively parallel computing job. It employs map and reduce jobs in MapReduce (Hadoop) [19] to run the ETL-oriented processing in parallel. Algorithms 1 and 2 offer in detail the operations of mapper and reducer.

The query processing in the recommendation system then uses the 2D mood map discs computed by the mapper and reducer. Specifically, when a user needs to know which would be the mood on a certain day and place, the system queries the WeatherUnderground API to obtain the weather forecast of the input day. At the same time, the system also queries the 2D structures to know the probabilities of mood states for that travel day. As shown, it is straightforward to add another dimension in the fusion.

V. Experiments

A. Data Collection

1) Process: In our current setting, we work with the Twitter social network API for obtaining the *social data*. We collect all the tweets corresponding to *London* location. In order to decide if a tweet is about a particular location, L_i, we use multiple features of the Twitter API. We consider a tweet is about a particular location if the tweet metadata has geo-tag information[1], or if the tweet user is from this place, or if the tweet text contains the location name. With these rules, we manage to obtain an approximate rate of 80-90 tweet messages per minute for the city of London, England.

We consider weather information at a particular location as *sensor data* in our data fusion setting. Specifically, we make use of services provided by WeatherUnderground[2], in order to periodically query the weather (W_j) of a particular

[1]Many smart phones provide this information automatically for the tweets posted using them.

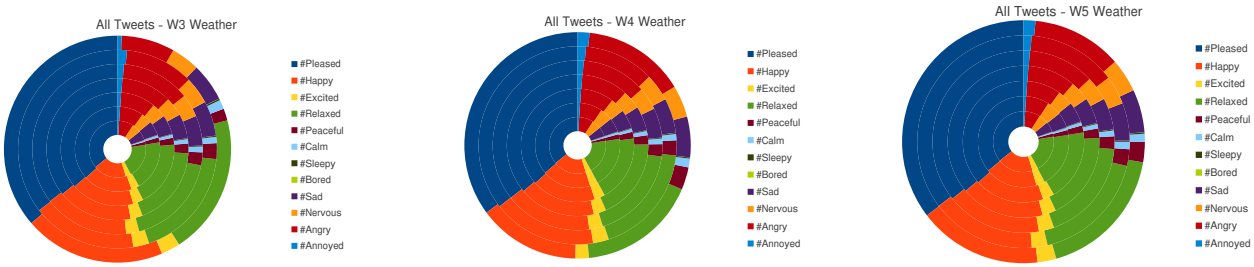
[2]http://www.wunderground.com

Algorithm 1 Mapper

procedure MAP($TweetId, Tweet$)
  ANEW[] /*contains *valence* and *arousal* scores of set of words*/
  $Tweet \rightarrow words[]$ /*decompose Tweet into words*/
  for $word_i \in words[]$ **do**
    if word in ANEW[] **then**
      (val_i,ars_i) $\leftarrow$ ANEW[$word_i$]
    else
      (val_i,ars_i) $\leftarrow$ (0,0)
    end if
  end for
  $tweet\_val \leftarrow \frac{\sum_{val_i \neq 0} val}{num(val_i \neq 0)}$
  $tweet\_ars \leftarrow \frac{\sum_{ars_i \neq 0} ars}{num(ars_i \neq 0)}$
  mood: $M_i \leftarrow$ moodMap2DFn($tweet\_val, tweet\_ars$)
  location: $L \leftarrow$ locationOf($Tweet$)
  time: $t \leftarrow$ timeOf($Tweet$)
  Day: $D_k \leftarrow$ dayOf($Tweet$)
  Emit((L,D_k,t),(M_k))
end procedure

Algorithm 2 Reducer

procedure REDUCER($Key, Value$)
  /*Computes Mood Space Probability Distributions*/
  WeatherMap[] /*contains *weather labels* for different timestamps*/
  (Location:L,Day:D_k,time:t) $\leftarrow$ decompose(Key)
  Weather Label: $W_j \leftarrow$ WeatherMap[t]
  Mood: $M_i \leftarrow$ Value
  increment(locationMoodMap[W_j][D_k][M_i], 1)
  return
end procedure



(a) Mood map for W3 weather (b) Mood map for W4 weather (c) Mood map for W5 weather

Fig. 4. Probability distribution over mood-spaces are shown in (a), (b) and (c) corresponding to weathers W3, W4 and W5 respectively. In each mood map, there are seven discs and each disc corresponding to a weekday. Outer disc corresponds to Monday while the inner most disc corresponds to Sunday.

W5	Clear	W3	Overcast Drizzle
W4	Scattered Clouds Partly Cloudy Mostly Cloudy	W2	Snow Fog
W3	Showers Rain Haze Rain	W1	Thunderstorm Thunderstorms and Rain Thunderstorms and Snow

TABLE I
CATEGORIES OF DIFFERENT WEATHER LABELS

location L_i. WeatherUnderground is a service that provides real-time weather information from nearly 32,000 weather stations around the world. The API provides wide variety of weather information like wind speed, wind direction, pressure, weather label etc. We are mainly concerned with the weather label. Some examples of the weather labels are *drizzle*, *rain*, *clear*, *thunderstorm* etc. We categorize the weather labels into 5 sets $\{W1, W2, W3, W4, W5\}$, as shown in Table I, according to pleasantness. As we move from W1 to W5 the pleasantness of the weather increases. We collect weather information of *London*, once every 30 minutes under the assumption that weather stays same over this period.

We collect both the *social* and *sensor* data by deploying corresponding virtual sensors in the GSN. These social and weather virtual sensors contain all the rules,filtering conditions and rate controlling parameters for collecting the data needed for the fusion process. One can easily include other locations into our travel recommendation system just through adding corresponding virtual sensors to GSN.

The weather data and twitter social data collected using the GSN framework is stored HBase back-end deployed on a cluster of 16 machines. The extraction of metrics from social data, and the fusion process of social and sensor data is done through the use of various configurable Hadoop (Map-Reduce) jobs.

2) Statistics/Datasets Sizes: We summarize the amounts of social and sensor data we collected over a period of 100 days for one particular location "*London*". The emotion expressed in a tweet might be related to different factors (weather, stock market influence, personal, work, product, event etc.). In order to focus on tweets related to weather we identify subset of the collected twitter dataset that is actually related to weather, and we refer to this subset as *weather-related* dataset. Given a tweet, we decide if a tweet is weather-related tweet using a set of weather related keywords. We summarize our results and observations corresponding to both the complete twitter dataset and weather-related twitter dataset. In Table II we summarize the statistics of the sizes of the datasets.

	Twitter Data	**Weather Data**
Complete Dataset		
Duration	28-April-2011 to 10-August-2011	
Number of Entries	12 Million	6600
Weather Related Entries	500000	6600
Training Dataset		
Duration	28-April-2011 to 20-June-2011	
Number of Entries	6.5 Million	3800
Weather Related Entries	300000	3800
Test Dataset		
Duration	21-June-2011 to 10-August-2011	
Number of Entries	5.5 Million	2800
Weather Related Entries	200000	2800

TABLE II
DATA COLLECTION CHARACTERIZATION

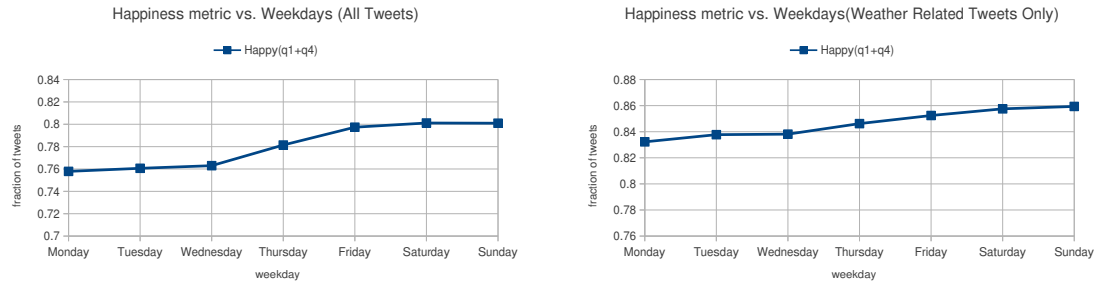
Table III shows summarized view of the number of tweets we observed after binning them according to the weather labels shown in Table I. In the table we observe 0 tweets for weather labels W1 & W2, as there were no thunderstorms or snow during the time window (April-August) in which we collected our tweets. Table IV shows a uniform distribution of number of tweets collected on different weekdays.

	Weather Labels				
#Tweets	W1	W2	W3	W4	W5
All	0	0	1836460	5201270	5108473
Weather-Related	0	0	124048	211000	198645

TABLE III
TWEETS DISTRIBUTION W.R.T. WEATHER LABELS

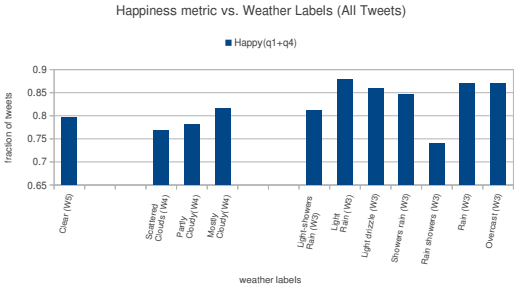
B. Observations

In this subsection we discuss the different observations we made regarding the mood metrics, and the correlation w.r.t. weekdays and weather labels. Some of the questions we asked were: *what is the happiness trend with respect to the weekdays? and what is the trend w.r.t to different weather*

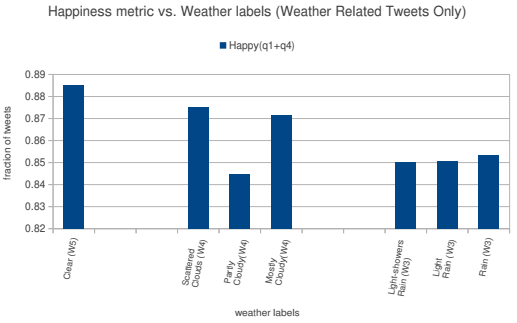


(a) All Tweets

(b) Weather Related Tweets Only



(c) All Tweets



(d) Weather Related Tweets Only

Fig. 5. Fraction of tweets expressing happiness mood (all tweets in Q1 & Q4 quadrant). (a) shows happiness metric for the complete tweets dataset, while (b) corresponds to weather related tweets only. In both cases we observe the tweets on the weekends tend to appear more happier than the tweets on the weekdays. (c) shows happiness metric of all tweets w.r.t. different weather labels, while (d) concerns only weather related tweets. Only in the later case we see people are more happier on sunnier days.

#Tweets	All	Weather-Related
Mon	2197788	83585
Tue	2532976	86485
Wed	1519632	70625
Thu	1585580	80324
Fri	1655613	74684
Sat	1413140	68048
Sun	1241474	69942

TABLE IV
TWEETS DISTRIBUTION W.R.T. WEEKDAYS

conditions? Even though our mood space is divided into 12 different mood spaces, for answering the above questions, we simplify our problem by considering all the moods in the quadrants Q1 and Q4 (valence > 5) as *happy* and the moods in quadrants Q2 and Q3 (valence ≤ 5) will be termed as *sad*. Figures 5(a) and 5(b) show the fraction of *happy* tweets we observed on different weekdays $\{Mon, \ldots, Sun\}$ irrespective of the weather conditions. Figure 5(a) represents the trend when the complete twitter data is taken into consideration, while fig. 5(b) corresponds only to weather related tweets. In either case, we observe that people in general are happier on the weekends $\{Fri, Sat, Sun\}$ compared to the weekdays $\{Mon, Tue, Wed, Thu\}$. Also we observe $Monday$ has the least fraction of *happy* tweets.

Next we tried to see any similar trends with respect to the different weather labels $\{W1, \ldots, W5\}$ irrespective of the weekdays, whose results are shown in Fig. 5(c) and Fig. 5(d). For weather-related tweets, the results shown in Fig. 5(d) we see that people are happiest on sunnier(W5) days, followed by cloudy(W4) days and least happy when it is raining(W3). On the contrary we did not see any clear trend when we consider the complete twitter data, as shown in Fig. 5(c). It may be suggesting that for a place like London, the weather, per se, does not have significant impact on the mood of the public.

C. Recommendation Validation

For travel recommendation system, when a user queries for a particular location in near future, adding to the future weather prediction of the location we would also try to predict the mood levels of the people. It is possible to make the prediction of near future events, based on the past history. One could imagine different prediction models. In our simplistic prediction model we summarize the statistics seen so far and we expect them to be valid for the future events. In order to evaluate the validity of this model, we divided the entire tweet dataset into two time windows, the training window and the test window.

We rely on two accuracy metrics to see if the statistics inferred on the training window (history) are still similar to the statistics observed in the test window (near future events). The first metric we rely on essentially compares two probability distributions, in our case the distributions over mood spaces described in Fig. 1. We have two mood-spaces distributions

corresponding to training and test windows. In order to see if they are from similar distribution we apply Chi-Square Goodness-of-Fit test [20]. We observed values of $\chi^2 = 0.0056$ (Weather Related Tweets) and $\chi^2 = 0.054$ (All Tweets). Such low values of χ^2 suggest that we accept the null hypothesis that both the probability distributions are very similar. Further the distributions are much more similar in the weather-related tweets compared to all-tweets case.

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
W3	1.000	0.667	0.667	0.667	0.667	0.667	0.667
W4	0.667	0.667	0.667	0.667	0.667	0.667	0.667
W5	0.667	0.667	0.667	0.667	0.667	0.667	0.667

TABLE V
MOODS OVERLAP: JACCARD SIMILARITY METRIC WHEN CONSIDERING THE **COMPLETE DATASET**

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
W3	0.66667	1	0.66667	1	1	1	1
W4	0.66667	0.66667	0.66667	1	1	1	1
W5	1	1	1	1	1	1	1

TABLE VI
MOODS OVERLAP: JACCARD SIMILARITY METRIC WHEN CONSIDERING THE **WEATHER-RELATED TWITTER DATASET**

The second test we consider is the percentage of overlap between the top 5 moods predicted by the training window model and the ones we observe during the test window. We use Jaccard Similarity metric to quantify the overlap, which is defined among two sets as the ratio of their intersection size to the union size. The observed overlap metrics are shown in tables Table-V and Table-VI. In both the cases we observe a significant overlap between the predicted moods and observed moods. The mood labels we learn through our fusion process during the training window significantly overlap with the mood labels we observe during the test window.

Our fusion process of social and sensor data in the cloud, not only helped us understand the general trends of mood swings with respect to different weekdays and weather labels but also through simple models make accurate predictions. Relying on scalable cloud components, our fusion process can be readily expanded to many more locations with little effort. Through careful tuning of map-reduce jobs our fusion process can handle far more complex prediction models.

VI. CONCLUSIONS

As various smart applications rely on mobile cloud computing, fusing data in the cloud becomes an essential issue. Addressing this concern, we presented a data-fusion approach that blends representative data sources—social and sensor data—commonly managed in mobile cloud applications. Specifically, we explored the potential of the data fusion by proposing a theoretical framework that enables to analyze tweet messages for extracting people's moods depending on day, weather, and location. We implemented the framework as a travel recommendation system that facilitates the fusion process over massively streaming data. The system is established upon several well-known cloud systems, allowing scalable data-fusion processing. We then discussed about various findings obtained from comprehensive experimental results using 12 million tweets as well as meteorological sensor readings, which demonstrate the effectiveness of our proposal.

VII. ACKNOWLEDGEMENTS

This work was partly funded by the Swiss Nano-Tera OpenSense project (Nano-Tera ref. 839 401), NisB project (FP7-ICT-256955) and the European Commission in the PlanetData NoE (contract nr. 257641).

REFERENCES

[1] T. White, *Hadoop: The Definitive Guide*. O'Reilly Media, 2009.
[2] "Hbase," http://hbase.apache.org.
[3] K. Aberer, M. Hauswirth, and A. Salehi, "A middleware for fast and flexible sensor network deployment," in *VLDB*, 2006, pp. 1199–1202.
[4] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller, "Twitinfo: Aggregating and visualizing microblogs for event exploration," 2011.
[5] J. Bollen, H. Mao, and X.-J. Zeng, "Twitter mood predicts the stock market," *ArXiv e-prints*, Oct. 2010.
[6] "Pulse of the nation," http://www.ccs.neu.edu/home/amislove/twittermood/.
[7] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment in twitter events," *J. Am. Soc. Inf. Sci. Technol.*, vol. 62, pp. 406–418, February 2011.
[8] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," *Proceedings of LREC 2010*, 2010.
[9] M. Nagarajan, A. Sheth, and S. Velmurugan, "Citizen sensor data mining, social media analytics and development centric web applications," in *WWW '11*. New York, NY, USA: ACM, 2011, pp. 289–290.
[10] A. Rosi, M. Mamei, F. Zambonelli, S. Dobson, G. Stevenson, and J. Ye, "Social sensors and pervasive services: Approaches and perspectives," in *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, march 2011, pp. 525 –530.
[11] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: real-time event detection by social sensors," ser. WWW '10. New York, NY, USA: ACM, Apr. 2010, pp. 851–860.
[12] J. Weng, Y. Yao, E. Leonardi, and F. Lee, "Event detection in twitter," HP Labs, Tech. Rep., 2011.
[13] S. R. Yerva, Z. Miklós, and K. Aberer, "What have fruits to do with technology? The case of Orange, Blackberry and Apple," in *International Conference on Web Intelligence, Mining and Semantics (WIMS11)*. ACM, 2011, p. 48.
[14] A. Beach, M. Gartrell, X. Xing, R. Han, Q. Lv, S. Mishra, and K. Seada, "Fusing mobile, sensor, and social data to fully enable context-aware computing," in *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*, ser. HotMobile '10. New York, NY, USA: ACM, 2010, pp. 60–65.
[15] T. Lovett, E. O'Neill, J. Irwin, and D. Pollington, "The calendar as a sensor: analysis and improvement using data fusion with social networks and location," in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, ser. Ubicomp '10. New York, NY, USA: ACM, 2010, pp. 3–12.
[16] M. M. Bradley and P. J. Lang, "Affective norms for english words (anew): Instruction manual and affective ratings," Technical Report C-1, The Center for research in Psychophysiology, University of Florida, 1999.
[17] T. White, *Hadoop: The Definitive Guide*, O. Media, Ed. O'Reilly Media, 2009.
[18] K. Aberer, M. Hauswirth, and A. Salehi, "The Global Sensor Networks middleware for efficient and flexible deployment and interconnection of sensor networks," Tech. Rep., 2006, sumitted to ACM/IFIP/USENIX 7th International Middleware Conference.
[19] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters." Berkeley, CA, USA: USENIX Association, 2004.
[20] G. W. Snedecor and W. G. Cochran, *Statistical Methods*, J. L. Klavans and P. Resnik, Eds. Iowa State University Press, 1989, vol. 203, no. 1992.