RasteyRishtey: A Social Incentive System to Crowdsource Road Traffic Information In Developing Regions

Rijurekha Sen
School of Information Systems, Singapore Management University
rijurekhasen@smu.edu.sg

Abstract

Intelligent Transport Systems (ITS), used for efficient road traffic management, largely benefits from probe sensing data like GPS traces. But gathering large scale GPS traces is a bottleneck, especially in developing regions, limiting researchers and government organizations from fully exploiting this potentially rich information base. In this paper, we present RasteyRishtey[1], a social networking system, to incentivize smartphone users to gather and share their GPS traces. Our social networking application in the simplest case, aids relatives like parents and spouses to track people, given the latter's consent. In the more complex case, groups of users can organize meet-ups collaboratively, starting from choice of venue to track everyone else until they reach the venue, using the application. Both versions produce crowdsourced GPS traces, that can potentially be a rich source of traffic information. We present the design and implementation of the applications, along with a deployment based user study of the same. We also present some preliminary analysis of the collected GPS traces, that shows certain interesting and intuitive characteristics of road traffic, in the Indian city of Bengaluru.

1 Introduction

Traffic congestion is a recurring problem worldwide. Intelligent Transport Systems (ITS), seek to provide better management of existing infrastructure and better information to commuters. In-vehicle GPS works as a wonderful ITS sensor, able to glean large scale traffic information, if GPS devices are present in a sufficient number of vehicles.

The major source of GPS traces is fleet companies operating taxicabs and transit vehicles in a city. For example, [1] gives taxi-cab commuters the expected travel time and fare, using traces from 15,000 taxicabs, over 21 months, in Singapore. [2] discovers both routes and schedules of transit vehicles from their GPS traces and also predict arrival times, using 3 months of traces from the Chicago Transit Authority. [3, 4, 5] use traces from 33,000 taxicabs in Beijing over 3 months. [3] gives drivers better route choices, relying on the knowledge of experienced cab drivers for optimal routes. [4] comes up with spatial and temporal causes behind anomalies in traffic situations. [5] finds faults in urban planning and design, by simultaneously mining people's travel patterns and commonly occurring urban hotspots.

The transport scenario in a developing country like India is somewhat different. Vehicles with GPS installed still form a negligible portion of the overall traffic. Though taxicabs with GPS, operated by fleet companies, are occasionally used through advance booking in some cities, most people flag down auto-rickshaws for personal movement, for their low cost and high availability. These auto-rickshaws, and also buses, both public and private, and ordinary taxis not owned by any fleet, rarely have GPS. GPS installation and data communication for these mass transit vehicles are being tried in some cities like Mumbai, Delhi and Bengaluru, but progress is slow due to cost, bureaucracy and lack of awareness issues.

On the other hand, the growing multitude of smartphone users in the cities of developing countries, might act as a potential source of crowdsourced GPS traces for traffic analysis. This would also have the advantage of tapping people using different transport modes, as high vehicle heterogeneity is another characteristic of developing region traffic. Buses, private cars, three-wheeler auto-rickshaws, two-wheeler motorbikes all ply on the road, at different speeds (Fig. 2). If GPS traces can be gleaned from smart-phone users in such multi-modality transport mediums, the resulting traffic information will be richer and more realistic than that available from fleet taxicabs alone.

Figure 2. High heterogeneity of vehicles in India

Crowdsourcing GPS traces from smartphone users has enormous challenges. Battery drain on the smartphones and user privacy would be two major concerns. Most impor-

[1] In Hindi, *Rastey* means roads and *Rishtey* means relationships

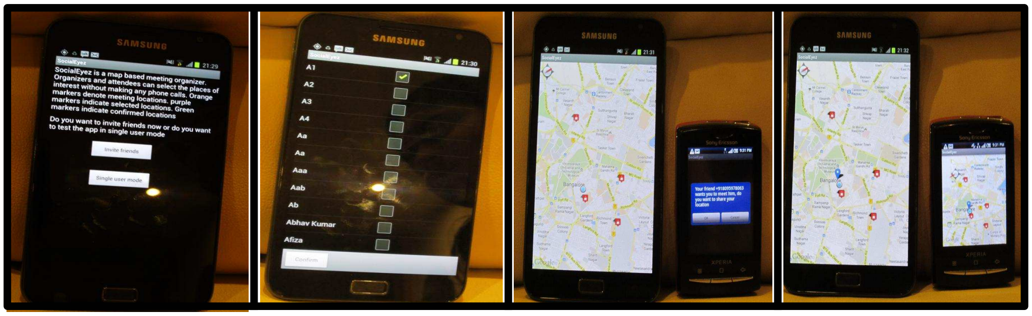

Figure 1. From left to right: (1) Choosing single user or group mode (2) Choosing contact from list (3) SMS received at contact (4) Organizer and contact's current locations and choices of meeting venues from Foursquare, shown on map on both phones

tantly, even if GPS sensing becomes low energy with the current research trends [6, 7, 8, 9], and anonymization schemes are put in use to save user privacy [10, 11, 12, 13], there is still no major incentive for users, to share their location traces. The only incentive is altruism, to improve overall traffic situation, but the users have no guarantee of this.

GoogleTraffic is designed on a similar crowdsourced approach to leverage commuters using GoogleMaps. It gathers people's GPS traces in the GoogleMap background to know their travel delays. But citizens commuting regularly on same routes, for example between home and workplace everyday during peak hours, intuitively do not use GoogleMaps to know directions. This lack of user participation is probably the reason why it does not run for several Indian cities like Kolkata.

This paper aims to provide short-term practical incentives to citizens, to encourage them to share their location traces. It tries to find a sweet spot between travel related smartphone apps like GoogleTraffic, that have high traffic information value, and social networking smartphone apps like Facebook, Twitter and Foursquare, that are potentially attractive to phone users. The combination is a social incentive system called *RasteyRishtey*, that can potentially attract smartphone users with its social merits, and the traffic data obtained in the background can be mined for interesting observations and insights.

RasteyRishtey tries to put the uncertainties of travel delays in a typical city of a developing country and the anxiety it causes to social relations, to some positive use. It is fairly common for a person, to get multiple phone calls from parents or spouse, on the way back home from work, to know the current location and approximate time of arrival. To ease this, our first smartphone application *WeasleyClock*[2], aids individual users to track each other, for a duration of their choice, by mutual consent. For a larger scale of participants, like a group of friends or business partners, the overhead of organizing a meet-up on the fly is enormous, involving multiple one-to-one or one-to-many phone/text conversations. Our second smartphone application *SocialEyez* aids to collaboratively learn current locations of each individual, decide a meeting venue at optimal distance from all, and track each others' progress along their respective routes to the meeting venue.

We present the design and implementation of these applications, along with some analysis of the collected GPS traces from users in Bengaluru, an Indian city. Our analysis shows that even with small scale crowd-sourced GPS traces, interesting and intuitive characteristics of road traffic patterns, like travel time differences between peak and non-peak hours, higher unpredictability of travel times in peak hours, regularly occurring traffic hotspots, can be inferred through appropriate data visualization techniques.

The rest of the paper is organized as follows. The details of the application implementation are presented in Section 2. The deployment based user study is described in Section 3 and the analysis of the collected GPS traces is presented in Section 4. We outline some possible future extensions in Section 5, before concluding the paper in Section 6.

2 WeasleyClock and SocialEyez

RasteyRishtey currently comprises of two android apps - WeasleyClock and SocialEyez, to handle respectively the two use cases of individual tracking and group meet organization, described in the previous section. In this paper, due to space constraints, we give the details of the more complex SocialEyez app. The WeasleyClock app is a restricted version of SocialEyez, involving two people instead of a group and having only the tracking component.

Fig. 1 and Fig. 3 show the sequence of events when two phones interact using SocialEyez. The Samsung Galaxy tab on the left, acts as the organizer, which in Fig. 1(1) gets a prompt to select between (a) single user mode, which is to give a feeling of the app to first time users, and (b) invite friends. When (b) is selected, Fig. 1(2) pops up showing the address book to choose contacts. Choosing one or more contacts, sends an sms to each selected contact. Though sms is cheap in India, we are planning to use the xmpp protocol like WhatsApp [14] in future, to remove the sms costs.

[2] The family of Ron Weasley in the famous Harry Potter series owned a clock, that showed the current locations of each family member, instead of time. Our first use case of tracking close relations during their travel, resembles this clock in spirit, hence the name.

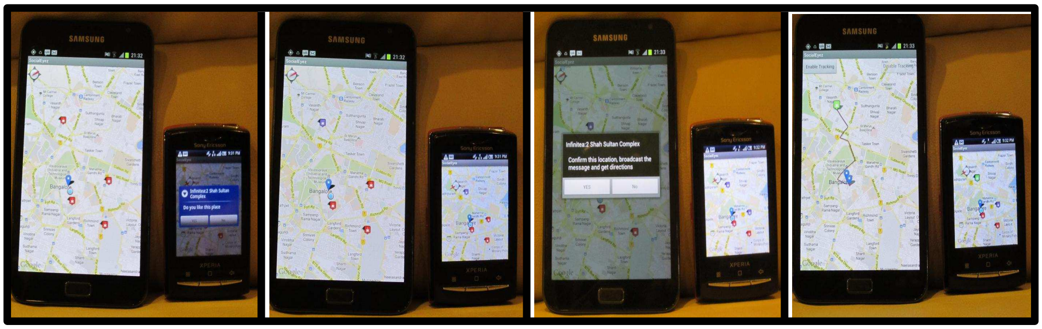

Figure 3. From left to right: (1) Contact prompted to show location preference (2) Preferred location turns purple on both phones (3) Organizer prompted to confirm location (4) Confirmed location turns green on both phones and directions shown

Fig. 1(3) shows the sms coming to the Ericsson phone, on the right. The sms pops up an alert which contains the url to download the app, if the contact does not have it installed currently, and also asks the contact for his consent to participate. If the contact responds positively to the alert, the app triggers on his/her phone, starting with sharing the current location. Until the app running on the contact responds with his/her location, the organizer sees Foursquare locations in his vicinity (Fig. 1(3), Samsung phone). This changes to Foursquare locations near centroid of all participants as and when their locations become known. Fig. 1 and Fig. 3 are taken with both phones at the same location to show their interactivity, hence the Foursquare locations remain same between Fig. 1(3) and Fig. 1(4).

In Fig. 3(1), the contact taps on any of the Foursquare options and is prompted to give his/her preference for that location. If the contact likes it, the location icon turns purple in both phones (Fig. 3(2)). This can go on for all participants, who can give preference for one or more locations. But the ultimate choice of the meeting venue rests with the organizer, who after considering the others' preferences, confirms a particular location as the final venue(Fig. 3(3)). Then the confirmed icon turns green in all phones and directions from current location to destination are shown on request (Fig. 1(4)).

There is a button to enable tracking, which allows everyone to see everyone else's location along their respective routes to the destination. This is shown in Fig. 4 after running the application with the two phones at separate locations, so that the two separate tracks become visible, unlike in Fig. 3(4), where phones being in the same location, the two tracks to destination are overlapping.

SocialEyez consists of an android client application and a node.js server running on Amazon ec2. It uses the current best location finder on the phone (cell-tower for indoor or GPS for outdoor), GoogleMap API to render the locations on a screen, Foursquare API [15] to give options of meeting venues, Leaflet API [16] to know directions and a Parse database [17] for posting and querying locations during tracking. GoogleMap API for directions gives so few intermediate points, that if lines are drawn using only those points, the resulting route does not lie on physical roads. Hence we use the Leaflet API for directions.

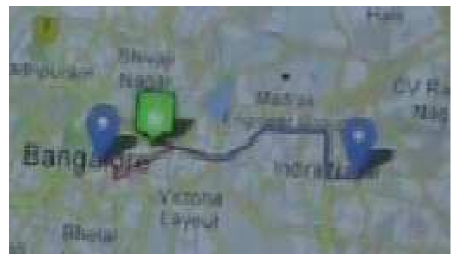

Figure 4. Everyone can see everyone's progress along respective tracks

3 Deployment Experience

The WeasleyClock and SocialEyez apps were distributed to a close circle of 10 friends, who in turn gave it to 7 more people. These 17 users, all based in Bengaluru, India, used the apps over 3 weeks between Jun 25 to Jul 13, 2012. There were 4 female and 13 males in the participants pool. 5 participants used office shuttle to commute from home to work, 7 traveled in motorbikes, 3 traveled in auto-rickshaws and 2 drove their own cars. This was mainly a feedback cycle, where users reported bugs and usability issues, and we provided the necessary fixes. On this limited scale of users, we found WeasleyClock to be more popular and used often on the way back from work. SocialEyez was used only twice by two different groups of people, during the weekend.

These users also provided feedback about enhancements, like to have (1) a choice of pre-decided venue in SocialEyez, instead of collaborative choice from FourSquare locations, where people still want to know each others' progress towards destination, for example to see a movie in a group where the movie tickets have already been booked and (2) to have option to schedule the meeting at a later time, instead of immediate meet-up, where the app sets a reminder for the scheduled time and starts the process when the reminder triggers. These enhancements were simple to implement and were added subsequently to the app.

The GPS traces of the users who were tracking each other

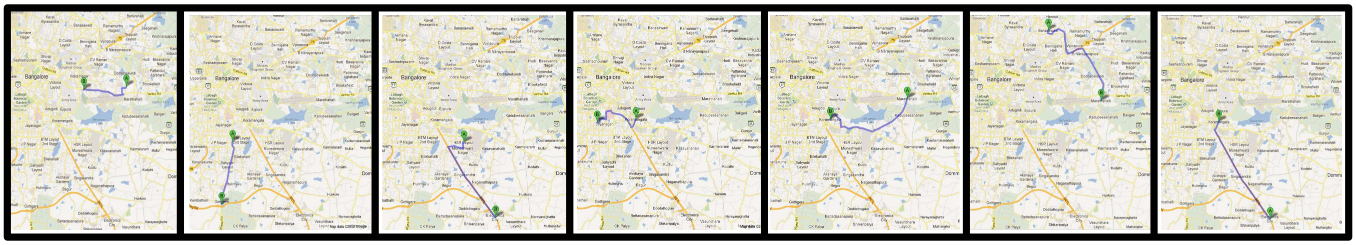

Figure 5. Commonly used segments. (1) Jawaharnagar-Kodihalli, (2) Jayadeva-Gottigere, (3) HSRLayout-ElectronicsCity, (4) Koramangala-Jayanagar, (5) Marthahalli-Koramangala, (6) Banasawadi-Marthahalli, (7) ElectronicsCity-Koramangala

were collected in our Parse database. In the next section, we provide an analysis of these traces. We understand the limited scale of this study and accept that the observations may not be generic, but the goal here is to highlight the usefulness of small-scale crowdsourced traces in finding interesting and intuitive traffic patterns. With more adoption and more data, we believe that our observations will become richer, instead of becoming obsolete.

4 Analysis of Crowdsourced GPS Traces

From the collected GPS traces, we identified some common road segments (Fig. 5), regularly traversed by one or more users. These segments had more traces along them, than other less frequented routes. By plotting the cumulative travel time vs traveled distance, for these segments from the collected traces, we observe some interesting patterns. In each of the subsequent figures, all curves in one figure belong to a single user and each individual curve in a figure is for a different day. The curves are labeled according to their start-times. We next discuss our key observations from these plots.

I: Time difference between peak and non-peak hours - The differences in travel time, for the same traveled distance, is highly apparent for the segments for which we have traces both in peak and in non-peak hours. For example, in Fig. 6, in the 4.5 Km stretch between Jawaharnagar and Kodihalli along old airport road, the travel time is around 600 secs in non-peak hours and above 1000 secs in peak hours. Thus the travel time difference is about 7 mins. This difference is even higher in the 5 Km stretch between Jayadeva junction and Gottigere along Bannerghatta road, as shown in Fig. 7. Here the non-peak travel time is around 600 secs and peak is around 1800 secs, creating a 20 mins delay.

II: Travel time uncertainties in peak hours - More than the increase in travel time at peak hours, a salient feature apparent from the traces is the increase in uncertainties in travel time at peak hours. In non-peak hours, travel times are predictable. For example, in Fig. 8, in the 4 Km stretch between HSR Layout and Electronics City, non-peak travel times has a small variation between 500 and 600 secs, which is little over 1 min.

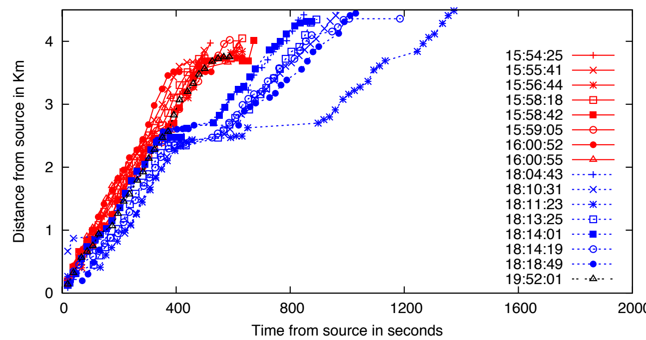

Figure 6. Longer travel times in peak hours on old airport road, between Jawaharnagar and Kodihalli

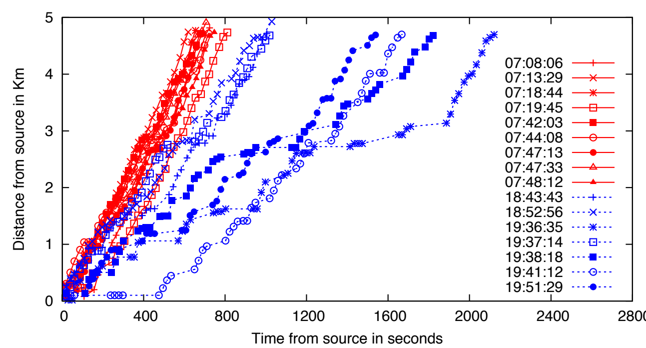

Figure 7. Longer travel times in peak hours on Bannerghatta road, between Jayadeva Junction and Gottigere

On the other hand, as shown in Fig. 9, in the short 3 Km stretch between Koramangala and Jayanagar, the travel time can vary between 1200 secs and 1800 secs in peak hours, which is about 10 mins variation. The longer the road stretch is, the higher the uncertainty in peak travel time. In the 7.5 Km stretch between Marthahalli and Koramangala, as shown in Fig. 10, the peak travel times can vary between 1800 and 3600 secs, an uncertainty of 30 mins!

If peak hours always meant a longer delay over non-peak hours, by a specific time, then people's schedules could be made predictable. But the uncertainties, along with the longer delays, make schedules arbitrary and planning difficult. And the uncertainties will add up the longer a route is.

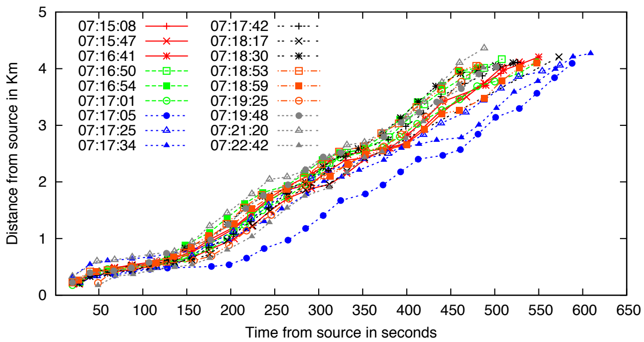

Figure 8. Predictable travel time at non-peak hours on HSR Layout-Electronics City route

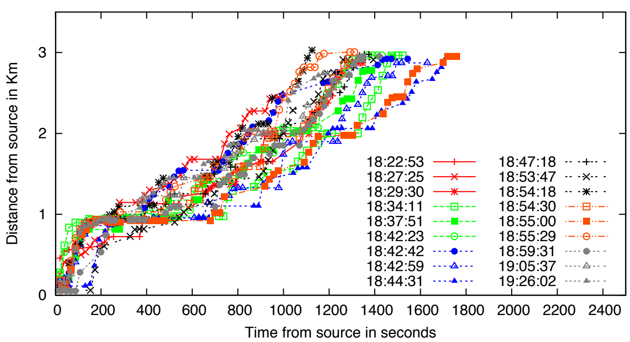

Figure 9. Unpredictable travel times at peak hours on the short Koramangala-Jayanagar route

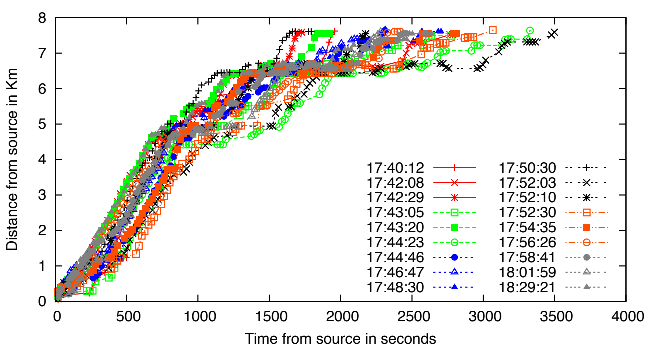

Figure 10. Unpredictable travel times at peak hours on the long Marthahalli-Koramangala route

III: Comparison with GoogleMaps time estimates - The driving directions in GoogleMaps come with a travel time estimate. When we compare the estimates returned by GoogleMap with our empirical travel times, we see a close match for non-peak hours. For example for Jawaharnagar-Kodihalli, the GoogleMap estimate is 10 mins, close to non-peak travel time in Fig. 6. And for Jayadeva-Gottigere, the GoogleMap estimate is 9 mins, again close to non-peak travel time in Fig. 7. But for peak hours, GoogleMap underestimates the travel times. For Koramangala-Jayanagar, the GoogleMap estimate is 15 mins, while the minimum empirical travel time is 20 mins and maximum is 30 mins, as shown in Fig. 9. For Marthahalli-Koramangala, the GoogleMap estimate is 16 mins, while the minimum empirical travel time is 30 mins and maximum is 60 mins! Thus the GoogleMap estimates can definitely benefit from trace driven empirical delay estimates for different times of the day.

IV: Detecting hotspots along a route - The final observation that we make from the traces is presence of lines parallel to the x-axis in the cumulative travel-time vs traveled distance plots. The lines parallel to x-axis indicate time was spent at the same geographical location, indicating that location to be a hotspot. Such hotspots are marked with rectangles in Fig. 11 and Fig. 12. Reverse geocoding the hotspot co-ordinates in GoogleMap, showed all hotspots to be near traffic signals. Thus the length of the curve parallel to x-axis is probably proportional to the length of queue waiting at the signal.

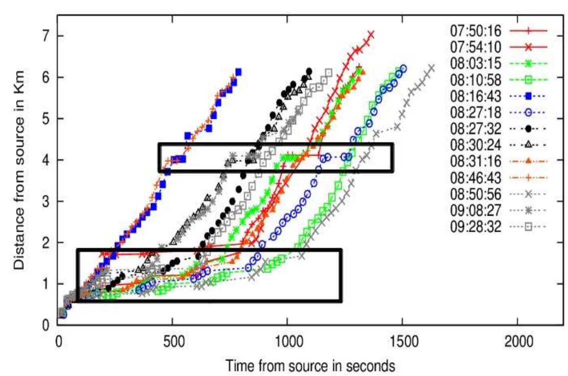

Figure 11. Hotspots on Banasawadi-Marthahalli route

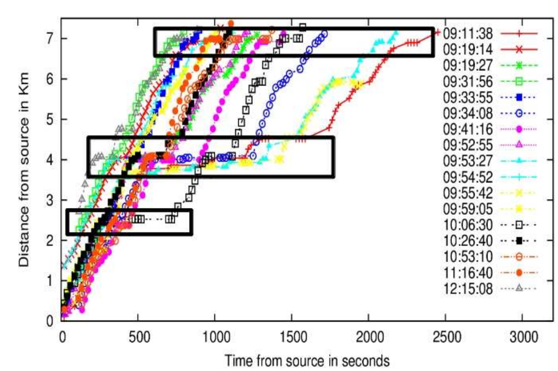

Figure 12. Hotspots on Electronic City-Koramangala route

5 Discussion and Future Work

In this paper, we present some preliminary ideas and implementations for crowdsourcing traffic data through social impetus. The high level idea of providing direct incentives to crowd and do indirect data mining in background, is already in practice today. Targeted advertising through social network data analysis in Facebook and search history analysis in GoogleSearch is a prominent example. We extend this idea to the domain of traffic data analysis, as shown pictorially in Fig. 13.

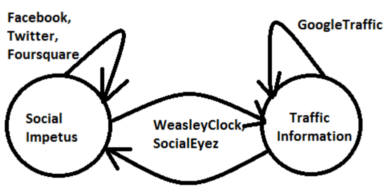

Figure 13. The envisioned interaction in app space

The work and results described here fall under the arrow from "Social Impetus" to "Traffic Information". An interesting avenue to explore is the arrow in the opposite direction. How can we feed back the traffic analysis results into the apps, to make them more valuable to users? For routes and times for which we have travel delays from other users, tracking can be enhanced with realistic arrival time estimates at the destination. Also, choice of meeting venues can be enhanced by considering travel delays of each participant to the destinations, in addition to the current proximity based search.

How else can such limited scale traffic analysis results be used? Can empirical data on travel time differences between peak and non-peak hours be used along with monetary incentive schemes proposed in [18], to encourage more people to shift their commute times? Such data based incentives for commute time shift, has been mentioned recently in [19], which analyzes peak and non-peak density, speed and flux for different Bengaluru roads based on traffic video analysis.

A third extension can be to leverage smartphone accelerometers, to detect vehicle type. As mentioned before, in India, there is high heterogeneity in vehicle type, which in addition to non-lane based traffic movement, results in highly different commute times between say two-wheelers and heavier vehicles like buses. Vehicle classification can thus produce more realistic travel time estimates.

6 Conclusion

This paper presents an application, aimed at incentivizing smartphone users to share their location traces. We also provide some initial observations from small amount of the GPS traces collected in a user study. Our initial prototype and GPS trace analysis show some promise of being useful in future, more so if we can find suitable answers to the above open questions.

7 Acknowledgment

Pradeep Banavara has been instrumental in design and implementation of the android applications. We are grateful to Bhaskaran Raman and William Thies for their valuable inputs on the initial idea. Special thanks to all the participants for their time and enthusiasm and feedback on the application usability and usefulness.

References

[1] R.Balan, N.Khoa, and J.Lingxiao. Real-time trip information service for a large taxi fleet. In *Mobisys*, 2011.

[2] J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson. Easytracker: automatic transit tracking, mapping, and arrival time prediction using smartphones. In *SenSys*, 2011.

[3] Yuan J., Zheng Y., Xie X, and Sun G. T-drive: Enhancing driving directions with taxi drivers intelligence. In *Transactions on Knowledge and Data Engineering (TKDE)*, 2012.

[4] Liu W., Zheng Y., Chawla S., Yuan J., and Xie X. Discovering spatio-temporal causal interactions in traffic data streams. In *SIGKDD*, 2011.

[5] Zheng Y., Liu Y., Yuan J., and Xie X. Urban computing with taxicabs. In *UbiComp*, 2011.

[6] R. Jurdak, P. Corke, D. Dharman, and G. Salagnac. Adaptive gps duty cycling and radio ranging for energy-efficient localization. In *SenSys*, 2010.

[7] J. Liu, B. Priyantha, T. Hart, H. Ramos, A. Loureiro, and Q. Wang. Energy-efficient gps sensing with cloud offloading. In *SenSys*, 2012.

[8] K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao. Energy-accuracy trade-off for continuous mobile device location. In *Mobisys*, 2010.

[9] Z. Zhuang, K. Kim, and J. Singh. Improving energy efficiency of location sensing on smartphones. In *Mobisys*, 2010.

[10] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J. Herrera, A. Bayen, M. Annavaram, and Q. Jacobson. Virtual trip lines for distributed privacy-preserving traffic monitoring. In *Mobisys*, 2008.

[11] C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos. Anonysense: privacy-aware people-centric sensing. In *Mobisys*, 2008.

[12] R. Ganti, N. Pham, Y. Tsai, and T. Abdelzaher. Poolview: stream privacy for grassroots participatory sensing. In *Sensys*, 2008.

[13] H. Ahmadi, N. Pham, R. Ganti, T. Abdelzaher, S. Nath, and J. Han. Privacy-aware regression modeling of participatory sensing data. In *Sensys*, 2010.

[14] `http://www.whatsapp.com/`.

[15] `https://foursquare.com/`.

[16] `http://leaflet.cloudmade.com/`.

[17] `https://parse.com/`.

[18] `http://simula.stanford.edu/Incentive_mechanisms/Projects.html`.

[19] R. Sen, A. Cross, A. Vashishtha, V. N. Padmanabhan, E. Cuttrell, and W. Thies. Accurate speed and density measurement for road traffic in india. In *3rd Annual Symposium on Computing for Development (ACM DEV)*, 2013.