Building a Location Based Social Graph in Spark at InMobi

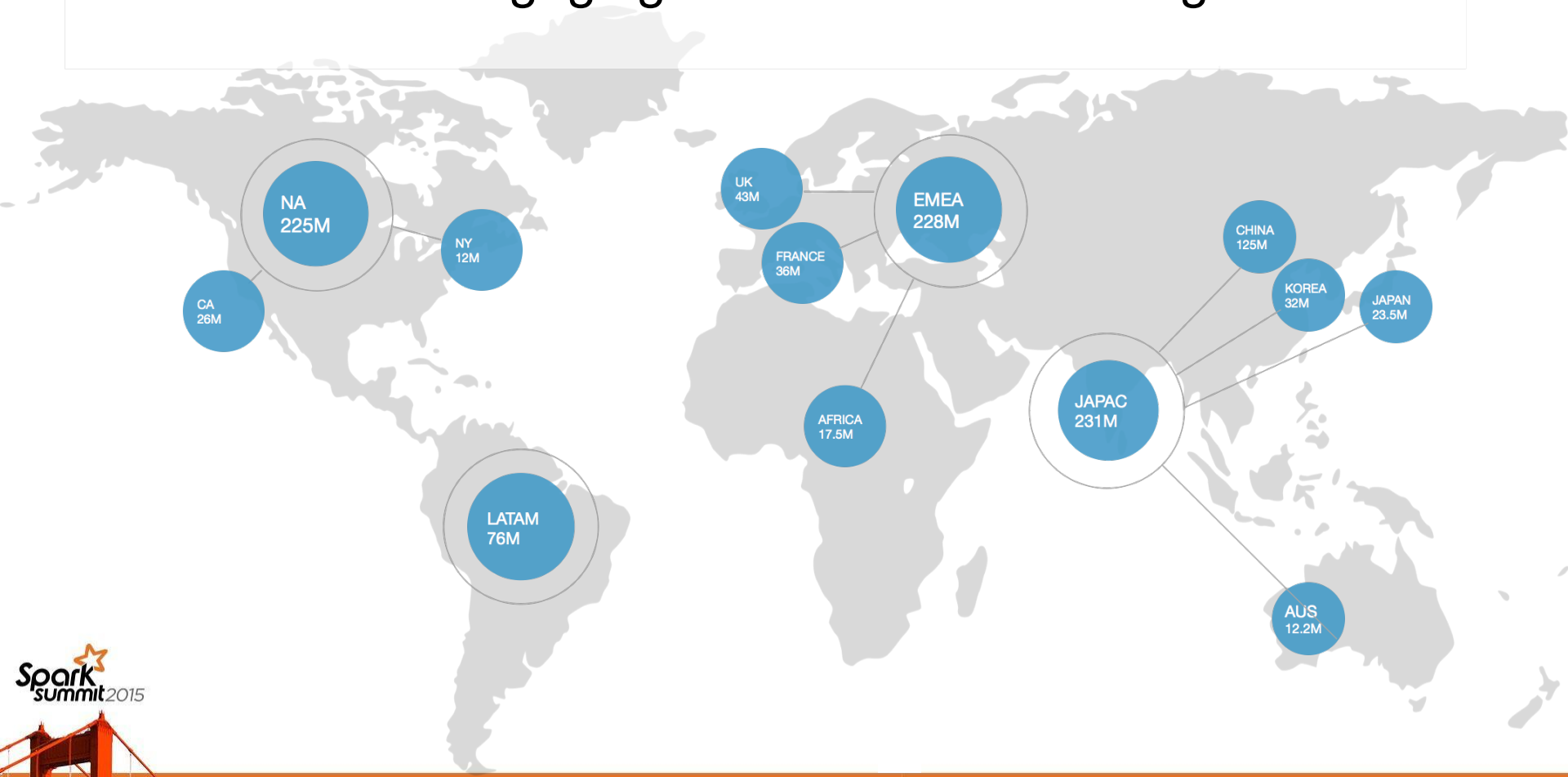
Seinjuti Chatterjee
Ian Anderson

INMOBI™

Talk Agenda

- InMobi background
  - Privacy
  - Location Service
  - Data collection
- Location based social groups
  - Static
  - Dynamic
- Spark at InMobi

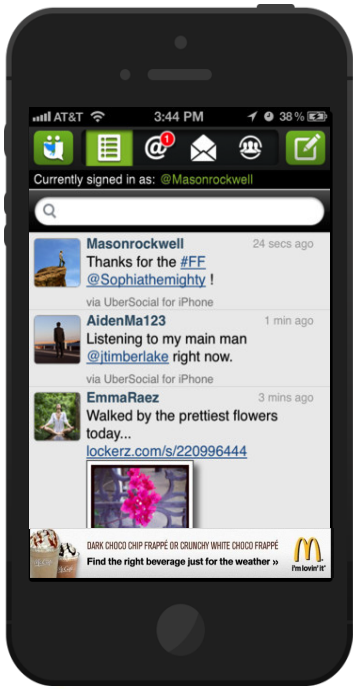
InMobi: Engaging 1bn users across the globe



Global Premium Publishers

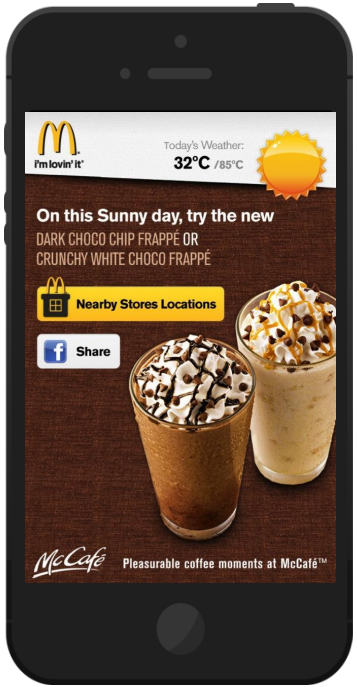
Spark summit 2015

Example Mobile Ads



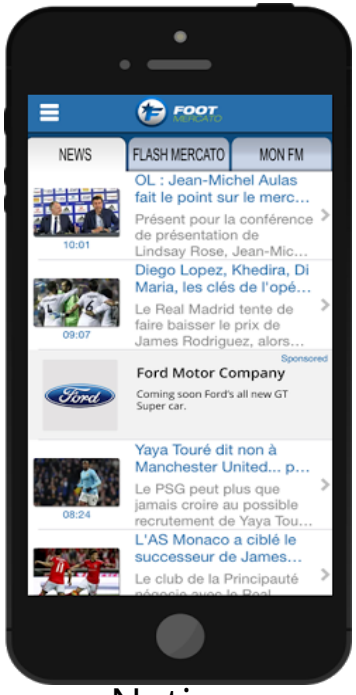
Banner

Video

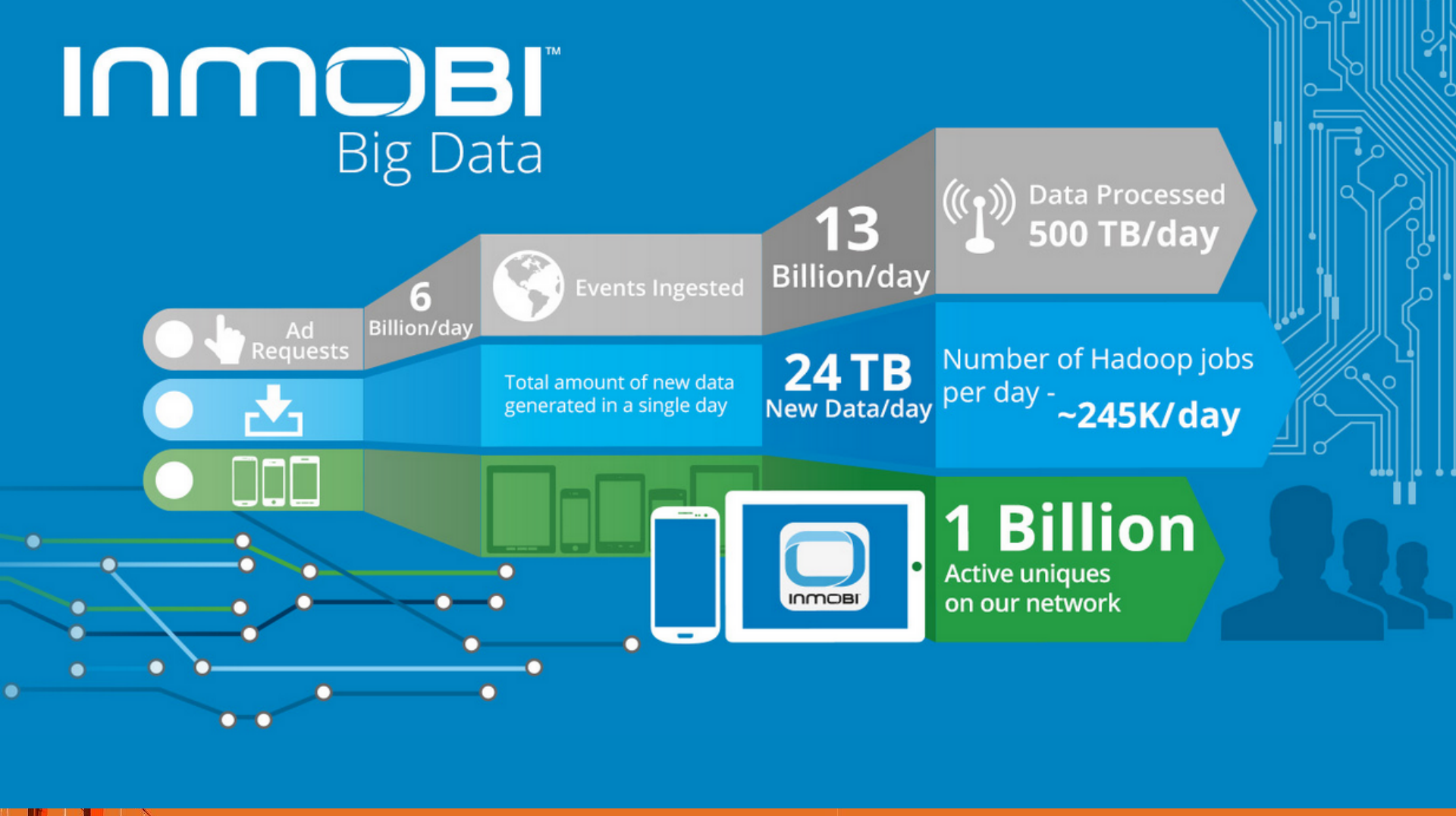


Rich Media

Interstitial



Native


INMOBI™
Big Data
Ad Requests
6 Billion/day
Events Ingested
13 Billion/day
Data Processed 500 TB/day
Total amount of new data generated in a single day
24 TB New Data/day
Number of Hadoop jobs per day - ~245K/day
1 Billion
Active uniques on our network
INMOBI

We are a major contributor to open source projects on big data.

Apache Falcon: Falcon is a data management and process orchestration platform built by InMobi and shared with the larger community as open source through the Apache Software Foundation. Learn more

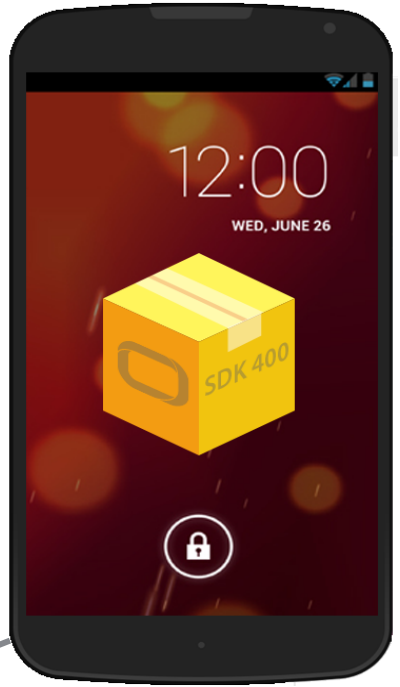
Pintail: Pintail is a client library that provides a streaming view of data as it arrives on HDFS, sourcing it from multiple clusters a.k.a 'tailing' a stream. Learn more

Conduit: Conduit is a system to collect the huge number of events data from online transactions, and make them available as a real-time stream to consumers. Learn more

Apache Lens: Lens provides an Unified Analytics Interface. Lens aims to cut the Data Analytics silos by providing a single view of data across multiple tiered data stores and optimal execution environment for the analytical query. It seamlessly integrates Hadoop with traditional data warehouses to appear like one. Learn more

InMobi SDK and Data Collection

Target behaviours not users



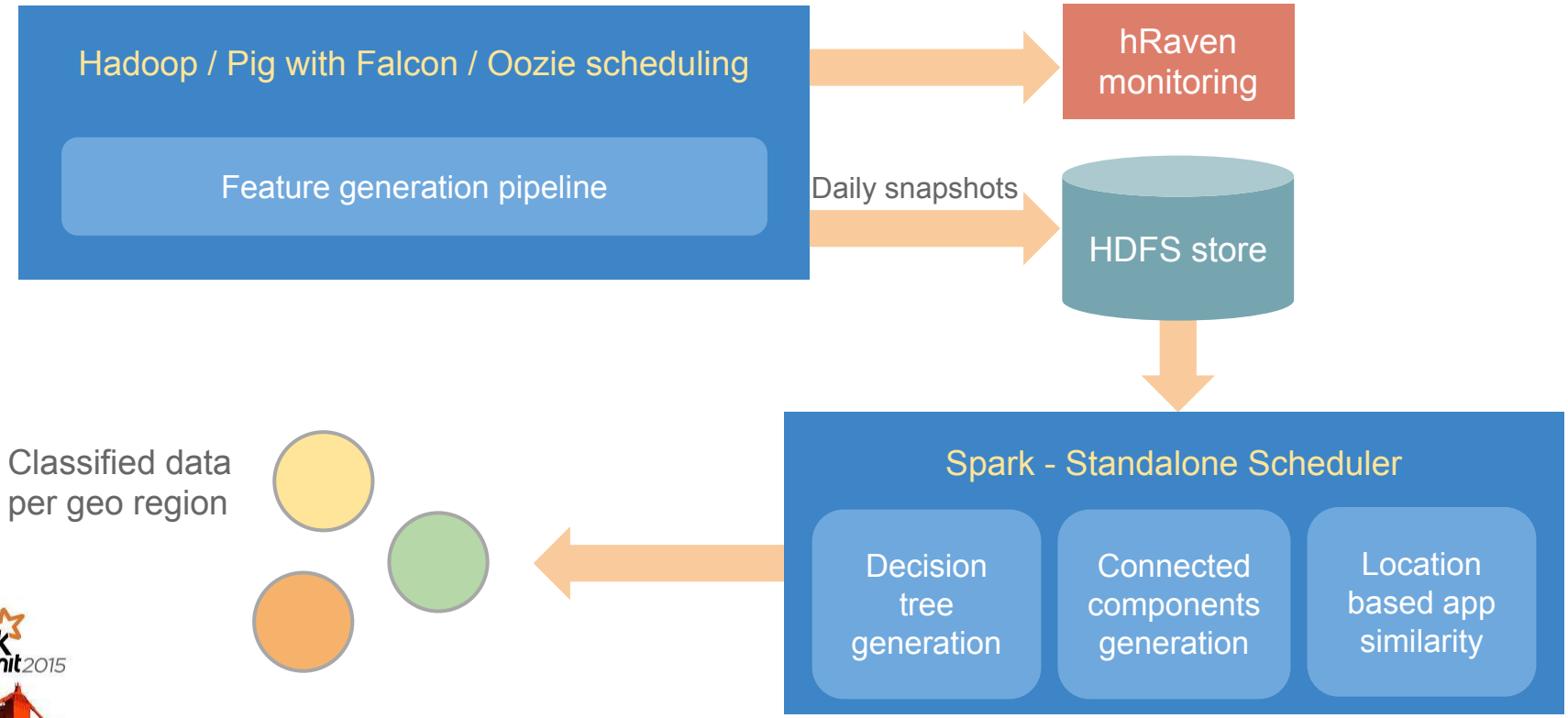
Ad Request

Data Enrichment / Processing

Ad Auction

Serve Ad

Hadoop and Spark Setup



Social Groups

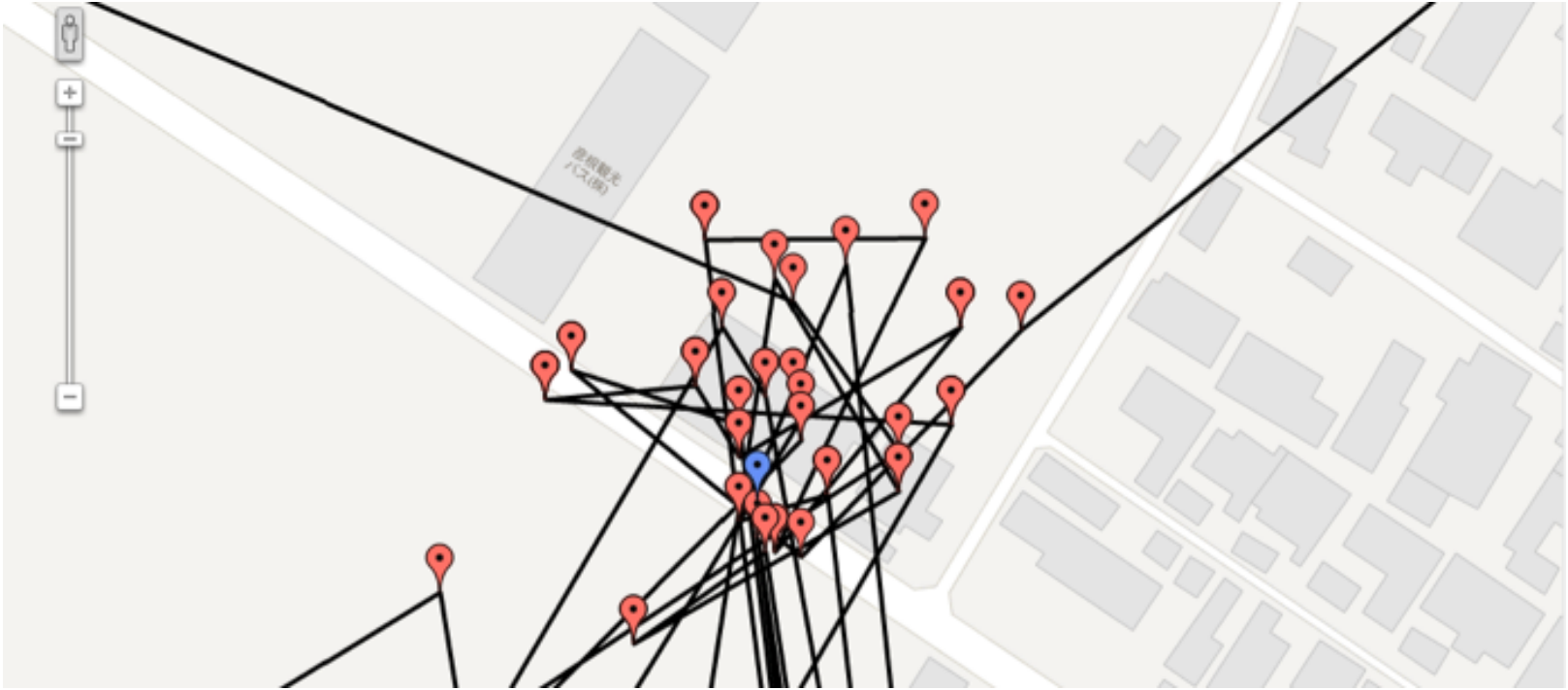
- “A social group within social sciences has been defined as two or more people who interact with one another, share similar characteristics, and collectively have a sense of unity.”

- http://en.wikipedia.org/wiki/Social_group

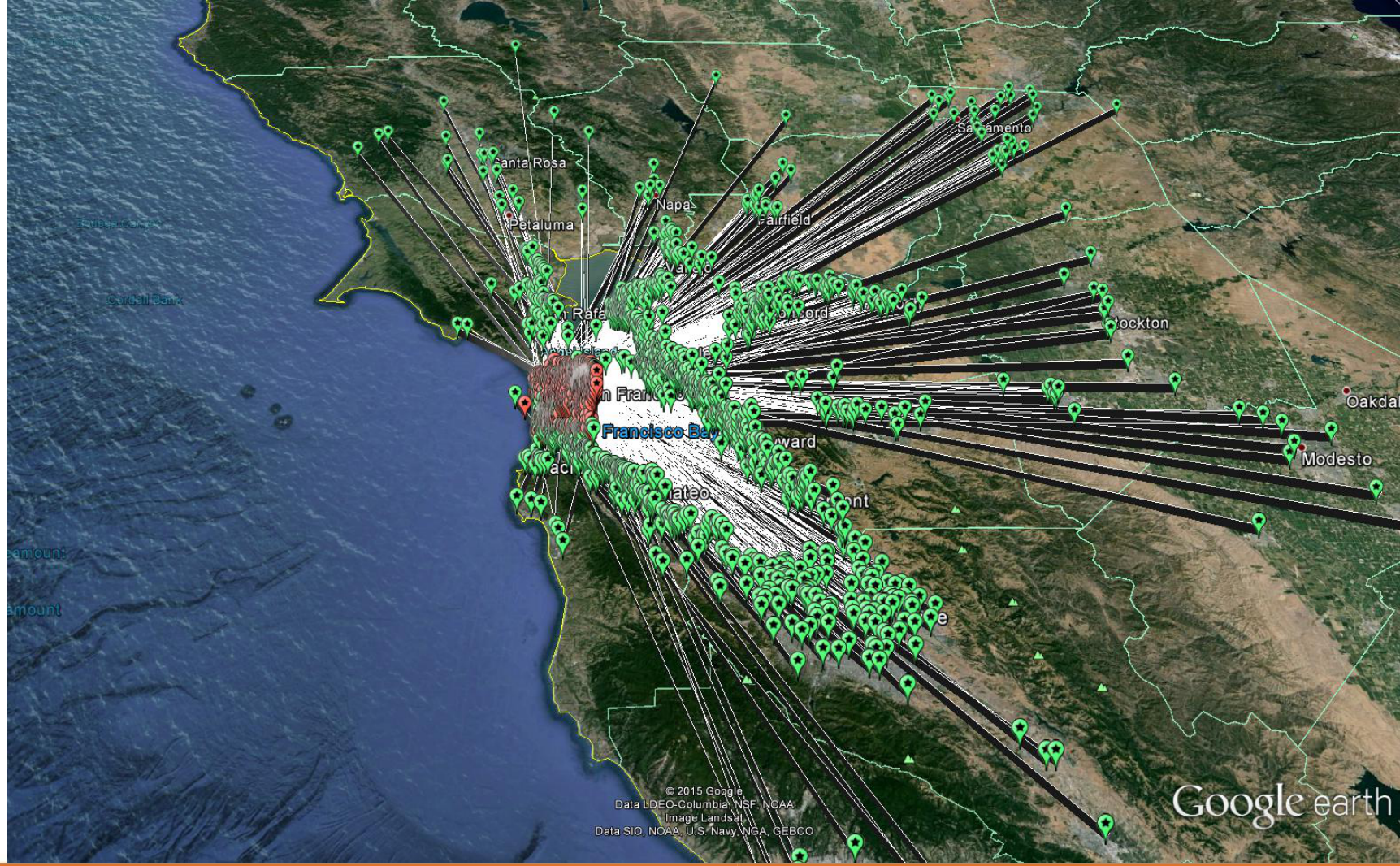
Location-Based Social Groups

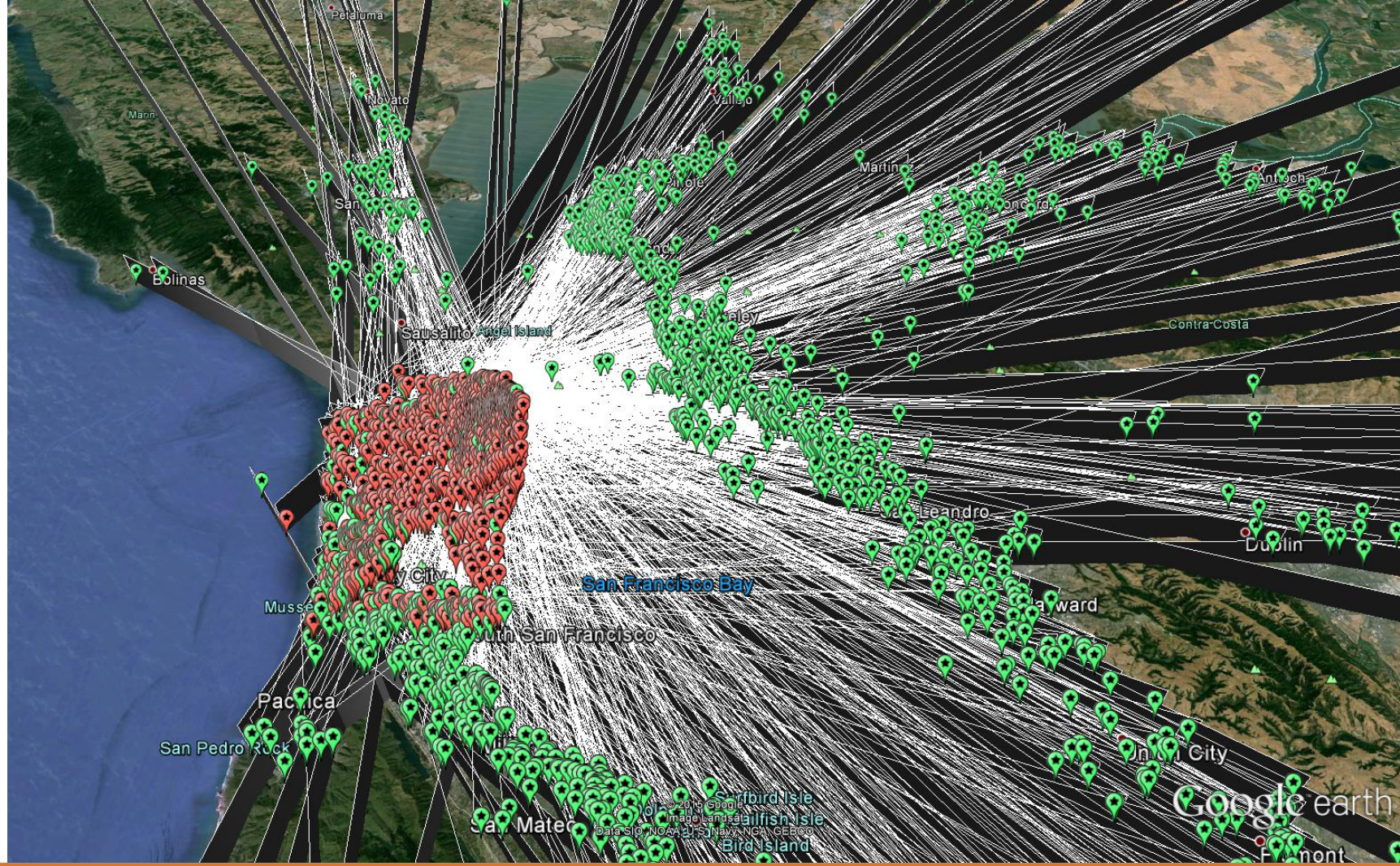
- Focus on location as the group membership criteria
  - Shared behaviours
  - Location can indicate social connection e.g. 49ers fans at Levi's Stadium
- San Francisco example
  - We want to reach a group of people that travel into San Francisco for business purposes
    - Visual demonstration
    - Implementation walkthrough

Identifying POIs



Spark summit 2015


Santa Rosa
Petaluma
Napa
Fairfield
Stockton
Modesto
Oakdale
Cordell Bank
© 2015 Google
Data LDEO-Columbia, NSF, NOAA
Image Landsat
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Google earth



Spark summit 2015

POI Classifier

“Given a geographic location e.g the United States, United Kingdom, India we want to understand the underlying nature of the location, in order to enrich the context of the ad request at the time of origin”

- Types of POI can be **Public Terminal, Hotel or Inn or B&B, Eatery, Sports Center, Community Center, Academic Institution, Retail Store.**

- POI classification helps profile user as a **commuter or a frequent flyer or a university student or frequent business traveller** or avid retail therapist

- We have trained a **Decision Tree Model** on labelled data of POI visitation frequency patterns which gives us upto **70% accuracy** now for predicting a POI type for a location.

- The final classifier was trained on 70K labelled locations and used to label 500K locations and public wifi’s

Feature Ranking using Information Gain

Average merit	Average rank	Attribute
1.331 +/- 0.003	1.0 +/- 0.0	normalized ssid groupsize
0.251 +/- 0.006	2.1 +/- 0.3	avg hour spread
0.241 +/- 0.001	2.9 +/- 0.3	ratio of avg weekend daytime adreq to avg weekend nighttime adreq
0.227 +/- 0.001	4.0 +/- 0.0	ratio of weekend early hours to weekend adreq
0.224 +/- 0.001	5.2 +/- 0.4	ratio of weekend late hours to weekend adreq
0.222 +/- 0.002	5.8 +/- 0.4	ratio of weekend lunch hours to weekend adreq
0.208 +/- 0.002	7.0 +/- 0.0	percentage of devices who visited at least 1 uniq days
0.202 +/- 0.001	8.0 +/- 0.0	ratio of avg weekday daytime adreq to avg weekday nighttime adreq
0.194 +/- 0.001	9.0 +/- 0.0	ratio of avg weekday to weekend adreq
0.184 +/- 0.001	10.0 +/- 0.0	ratio of weekday lunch hours to weekday adreq
0.178 +/- 0.001	11.0 +/- 0.0	ratio_of_weekend_breakfast_hours_to_weekend_adreq
0.172 +/- 0.001	12.0 +/- 0.0	percentage_of_devices_who_visited_atleast_2_uniq_days

Spark Implementation

- **Number of Data Points = 69,465 Num Classes = 7**
- **Test-Train Split = 30-70%**
- **Impurity=”gini”, maxDepth=30, maxBins=32, numTrees=13, featureSubsetStrategy=”auto”**
- **DecisionTree Classifier with holdout set**
- **RandomForest Classifier with holdout set**
- ***More tuning and 10 fold cross validation is required***

Decision Tree Model Results: Spark

Detailed Accuracy by Class

DecisionTreeModel classifier of depth 30 with 22351 nodes

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.54%	0.05%	0.54%	0.54%	0.54%	UNIVERSITY_OR_COLLEGE
0.71%	0.11%	0.72%	0.72%	0.71%	INN_AND_HOTELS
0.70%	0.15%	0.71%	0.71%	0.70%	EATERY
0.29%	0.08%	0.28%	0.28%	0.29%	COMMUNITY_CENTER
0.60%	0.06%	0.60%	0.60%	0.60%	STORE
0.24%	0.05%	0.23%	0.24%	0.24%	SPORTS_AND_FITNESS
0.19%	0.00%	0.20%	0.19%	0.19%	AIRPORT_BUS_RAIL_TERMINAL

Decision Tree Model Results: Spark

Confusion Matrix

a	b	c	d	e	f	g	<- classified as
1100	189	174	375	90	88	4	a = UNIVERSITY_OR_COLLEGE
201	4154	648	304	269	217	22	b = INN_AND_HOTELS
208	633	5047	463	451	364	23	c = EATERY
335	284	389	581	158	226	9	d = COMMUNITY_CENTER
103	246	433	157	1561	94	17	e = STORE
95	228	342	213	74	301	6	f = SPORTS_AND_FITNESS
6	23	27	17	8	4	20	g = AIRPORT_BUS_RAIL_TERMINAL

Random Forest Results: Spark

Detailed Accuracy by Class

TreeEnsembleModel classifier with 13 trees, Test Error = 0.308

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.64%	0.03%	0.73%	0.68%	0.64%	UNIVERSITY_OR_COLLEGE
0.82%	0.09%	0.77%	0.80%	0.82%	INN_AND_HOTELS
0.85%	0.21%	0.68%	0.75%	0.85%	EATERY
0.26%	0.04%	0.42%	0.32%	0.26%	COMMUNITY_CENTER
0.60%	0.03%	0.77%	0.68%	0.60%	STORE
0.18%	0.03%	0.31%	0.23%	0.18%	SPORTS_AND_FITNESS
0.30%	0.00%	0.83%	0.44%	0.30%	AIRPORT_BUS_RAIL_TERMINAL

Random Forest Results: Spark

Confusion Matrix

a	b	c	d	e	f	g	<- classified as
1303	167	251	202	53	52	0	a = UNIVERSITY_OR_COLLEGE
47	4747	721	69	113	97	4	b = INN_AND_HOTELS
51	537	6060	178	141	168	0	c = EATERY
327	259	632	506	109	130	2	d = COMMUNITY_CENTER
37	170	695	94	1587	48	0	e = STORE
23	231	568	151	38	228	0	f = SPORTS_AND_FITNESS
1	21	29	8	11	1	30	g = AIRPORT_BUS_RAIL_TERMINAL

Weka Implementation

Trained a Decision Tree Model in WEKA

- **Scheme:** weka.classifiers.trees.J48 -C 0.25 -M 2
- **Relation:** us.publicPOI.classification
- **Filter:** weka.filters.supervised.instance.SMOTE
- **Instances:** 69465
- **Attributes:** 28
- **Number of Leaves :** 6377
- **Size of the tree :** 12753

Decision Tree Model Results: Weka

Detailed Accuracy by Class

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.671	0.043	0.628	0.671	0.649	0.815	UNIVERSITY_OR_COLLEGE
0.805	0.089	0.776	0.805	0.790	0.873	INN_AND_HOTELS
0.802	0.113	0.788	0.802	0.795	0.859	EATERY
0.333	0.060	0.368	0.333	0.350	0.672	COMMUNITY_CENTER
0.786	0.025	0.820	0.786	0.803	0.887	STORE
0.234	0.040	0.269	0.234	0.250	0.617	SPORTS_AND_FITNESS
0.405	0.001	0.599	0.405	0.484	0.757	AIRPORT_BUS_RAIL_TERMINAL

Decision Tree Model Results: Weka

Confusion Matrix

a	b	c	d	e	f	g	<- classified as
4500	562	478	768	158	226	10	a = UNIVERSITY_OR_COLLEGE
517	15458	1569	765	292	580	31	b = INN_AND_HOTELS
486	1673	19145	1144	497	902	11	c = EATERY
1229	957	1219	2194	315	643	26	d = COMMUNITY_CENTER
169	443	653	363	6849	230	7	e = STORE
257	737	1195	703	219	953	4	f = SPORTS_AND_FITNESS
12	78	47	28	18	12	133	g = AIRPORT_BUS_RAIL_TERMINAL

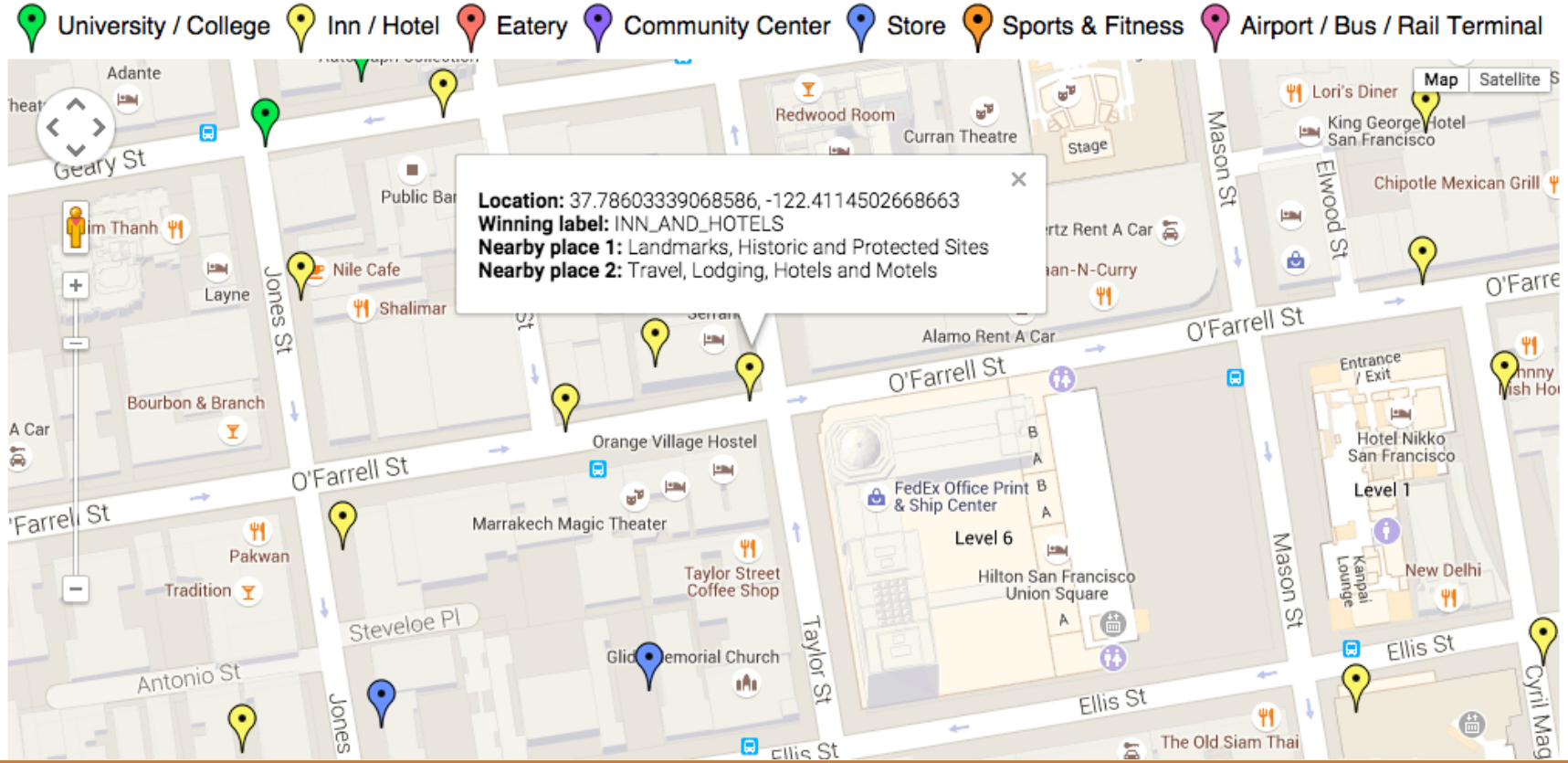
Model built with 10 fold cross validation

Model Results Comparison

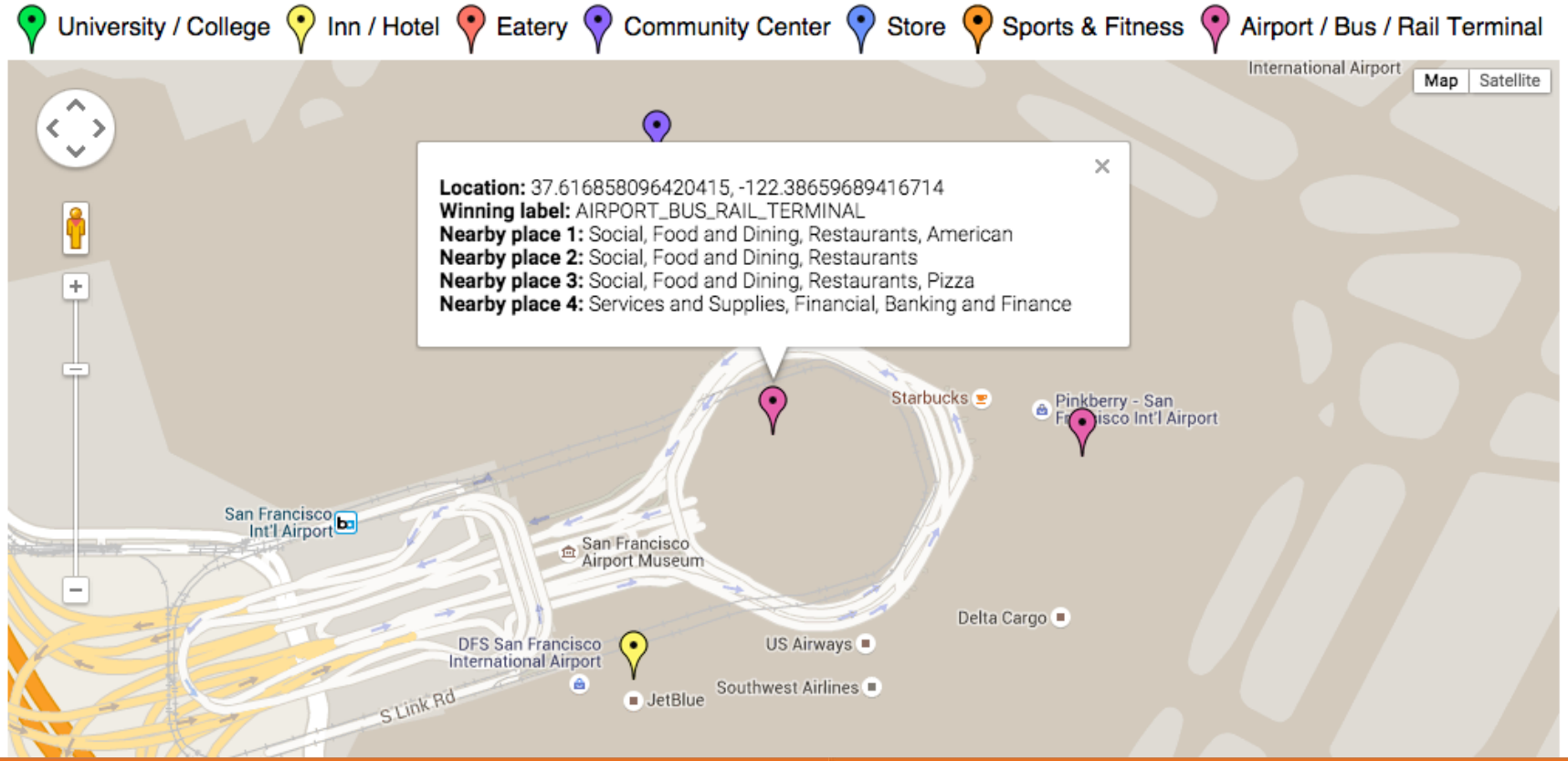
	Decision Tree Spark	Random Forest Spark	Baseline Decision Tree Weka
Input Training Sample Size	70K	70K	70K
Time taken to build model	2 mins	2 mins	35.69 seconds
Resource Usage	Single Node Cluster	Single Node Cluster	Single Node Cluster
Scalability	YES	YES	NO
Parallelism	YES	YES	NO
Accuracy	60%	69%	70%

	Decision Tree Spark	Random Forest Spark	Baseline Decision Tree Weka
Weighted F-measure	0.618	0.676	0.705
Weighted Precision	0.620	0.676	0.702
Weighted Recall	0.616	0.693	0.709
Weighted TPR	0.616	0.693	0.709
Weighted FPR	0.098	0.109	0.079
Test Error Rate	0.384	0.307	0.290
Hotel and Inn F-measure	0.71%	0.79%	0.79%
University/College F-measure	0.54%	0.68%	0.65%
Eatery F-measure	0.70%	0.76%	0.79%
Store F-measure	0.60%	0.76%	0.80%

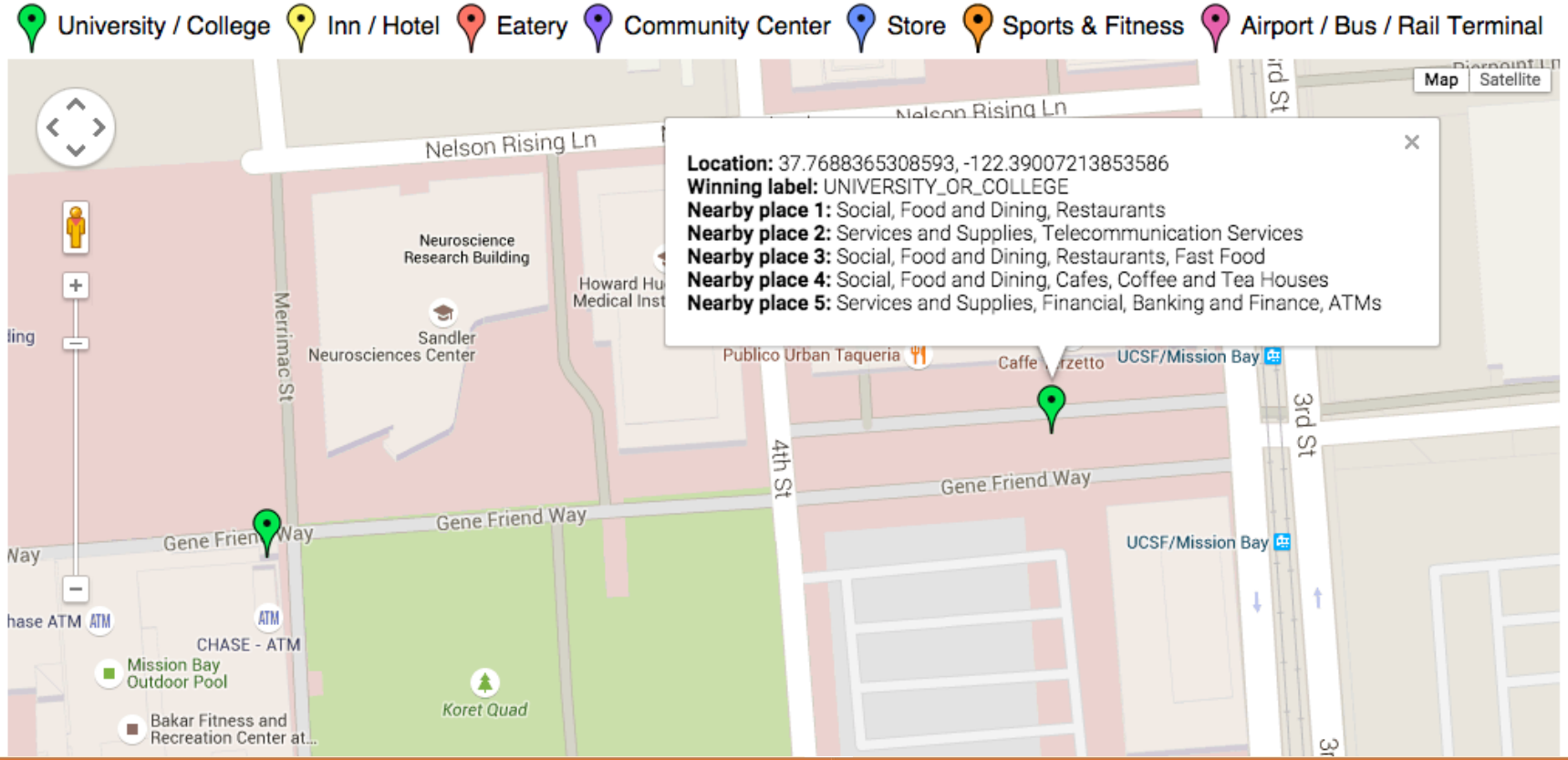
Classifier Visualization: Inn/Hotel



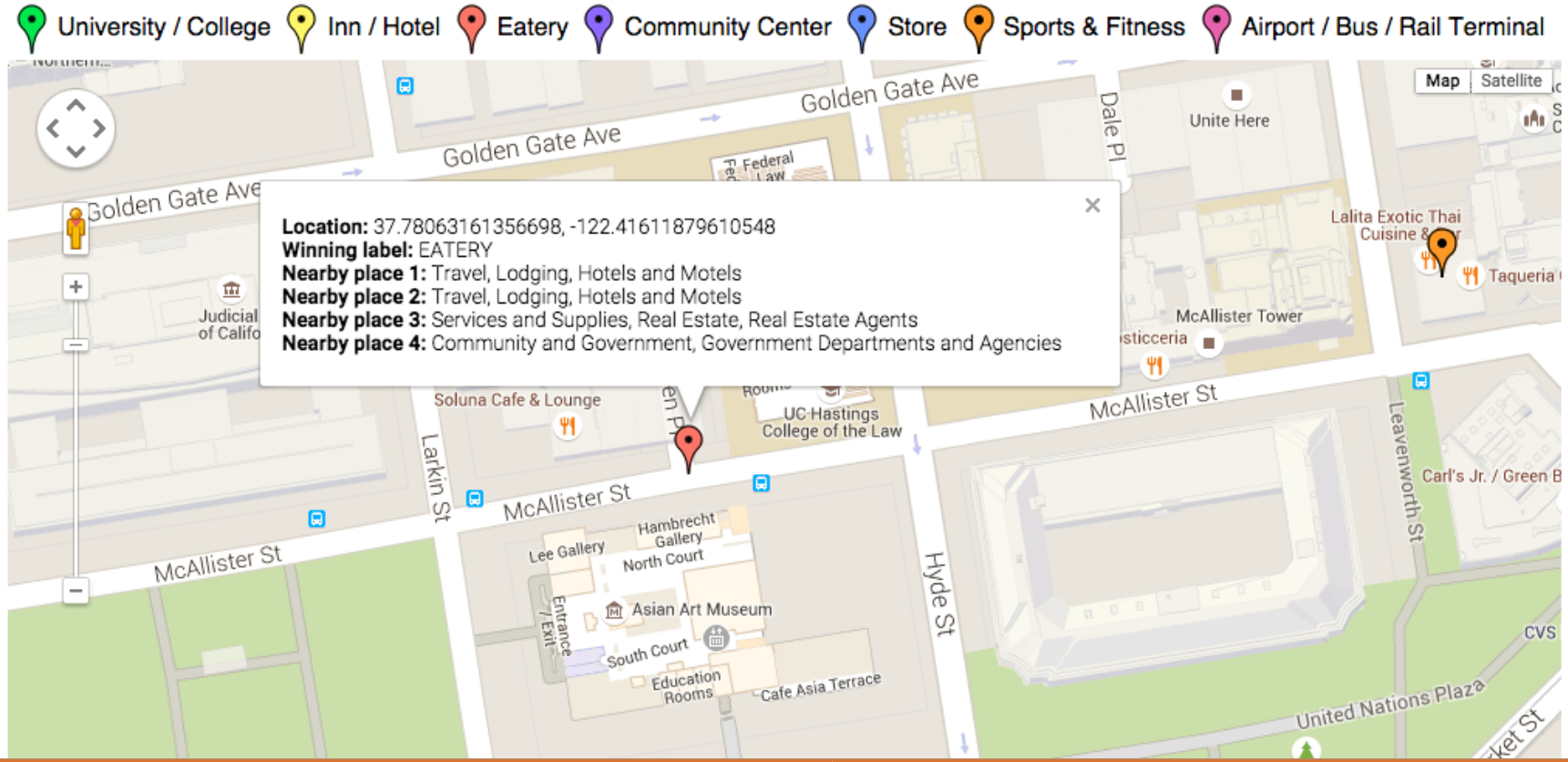
Visualization: Airport



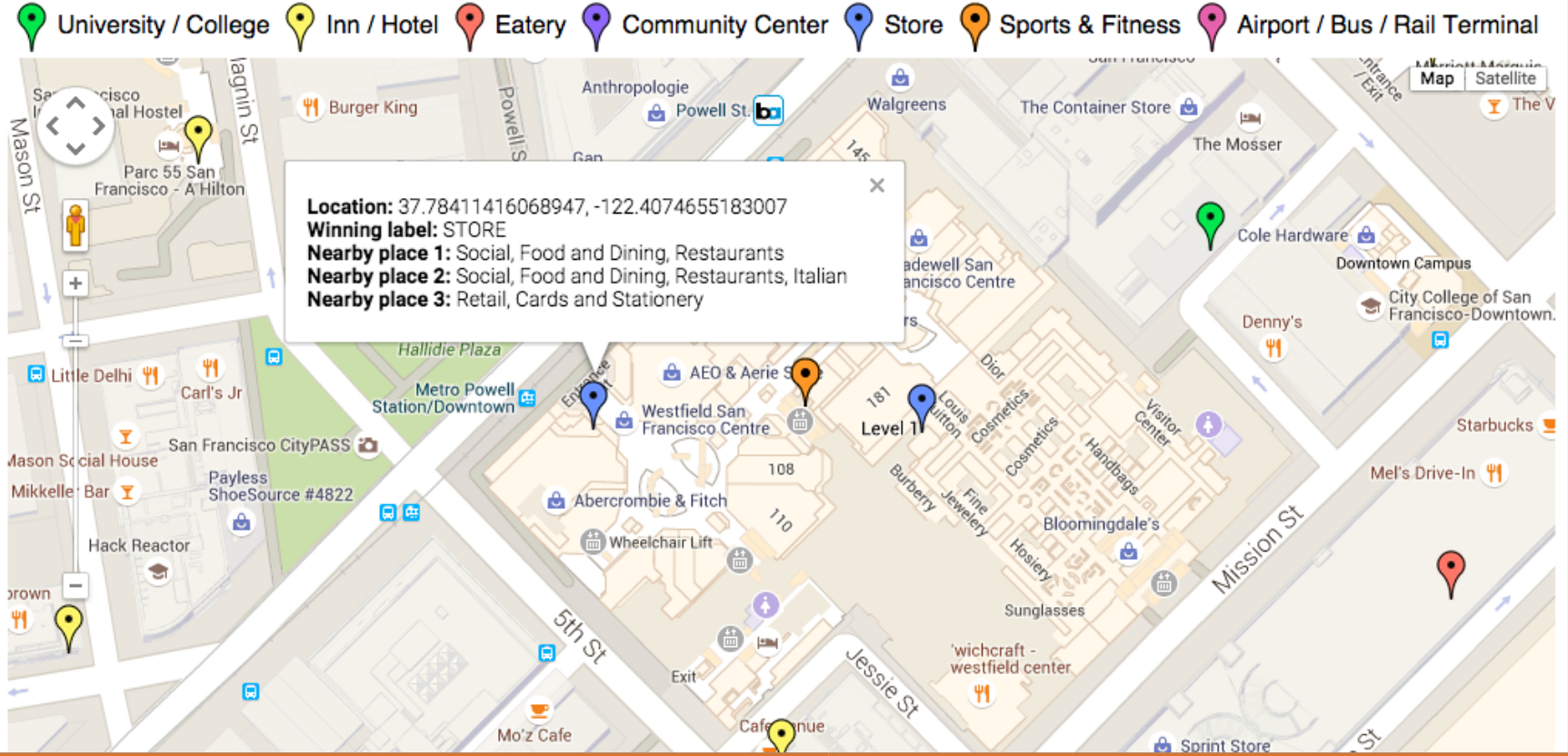
Visualization: University / College



Visualization: Eatery



Visualization: Store



Spark summit 2015

Connected Component

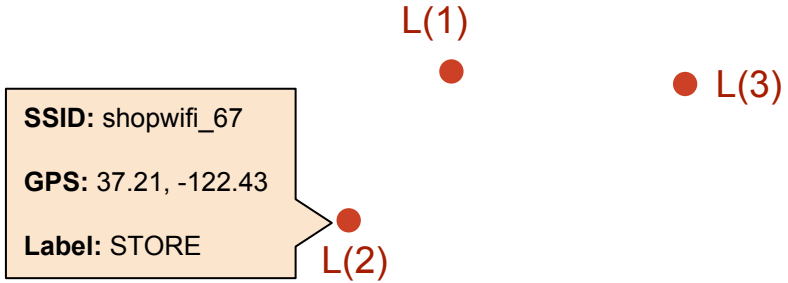
- A connected component is a set of locations which have been frequently co-visited by users over a month.
- Conceptually it is a subgraph of frequent visitation trends which transforms into a profile of the user.
- 5265 connected components generated for 576093 locations in 2 hours where each location has seen on an average 4 devices.

Examples:

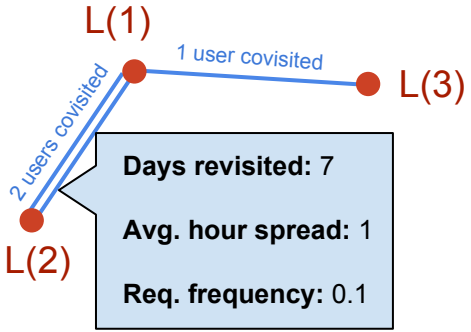
- University students who like eating @BuffaloWing
- Frequent business travellers to SFO who stay at hotels and rent a car

Spark Implementation

1. Create vertices



2. Create edges



3. Run connected component algorithm

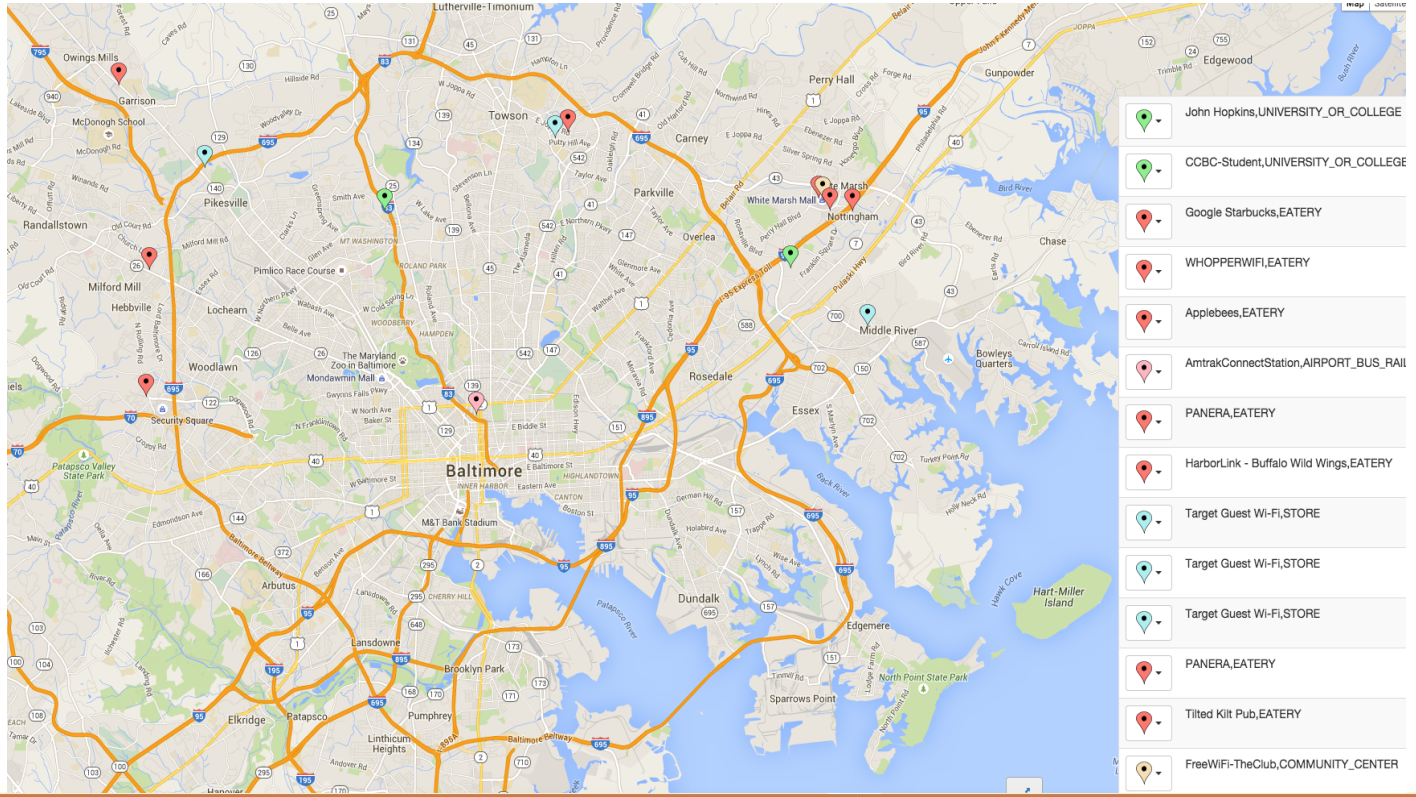
- Connection strength given by number of edges (ie. common users) between L(i) and L(j)
- Cluster locations sorted by component size

4. Rank users

- Rank the users per connected component
- Profile the user with the profile of the connected component

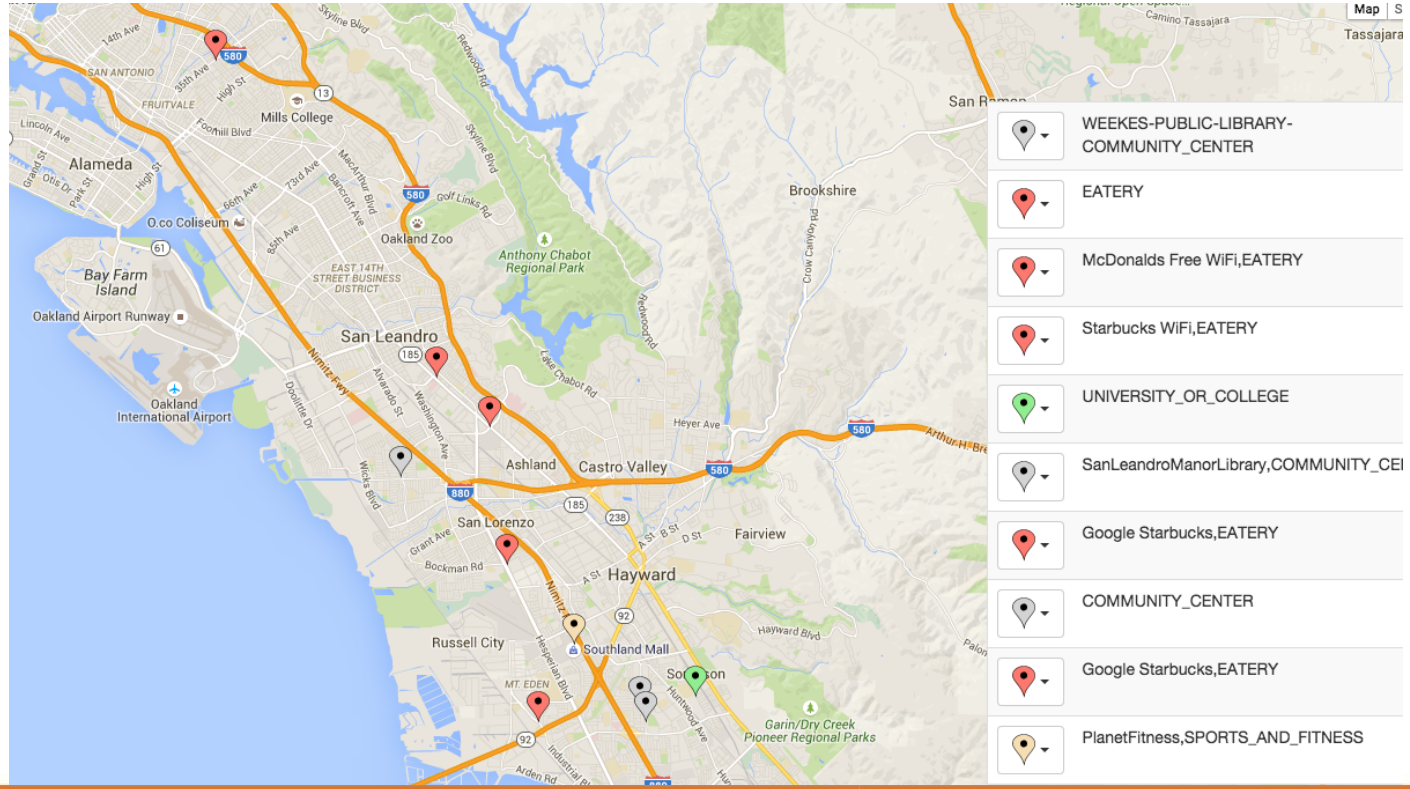
Connected Component

University, Eateries, Target Store



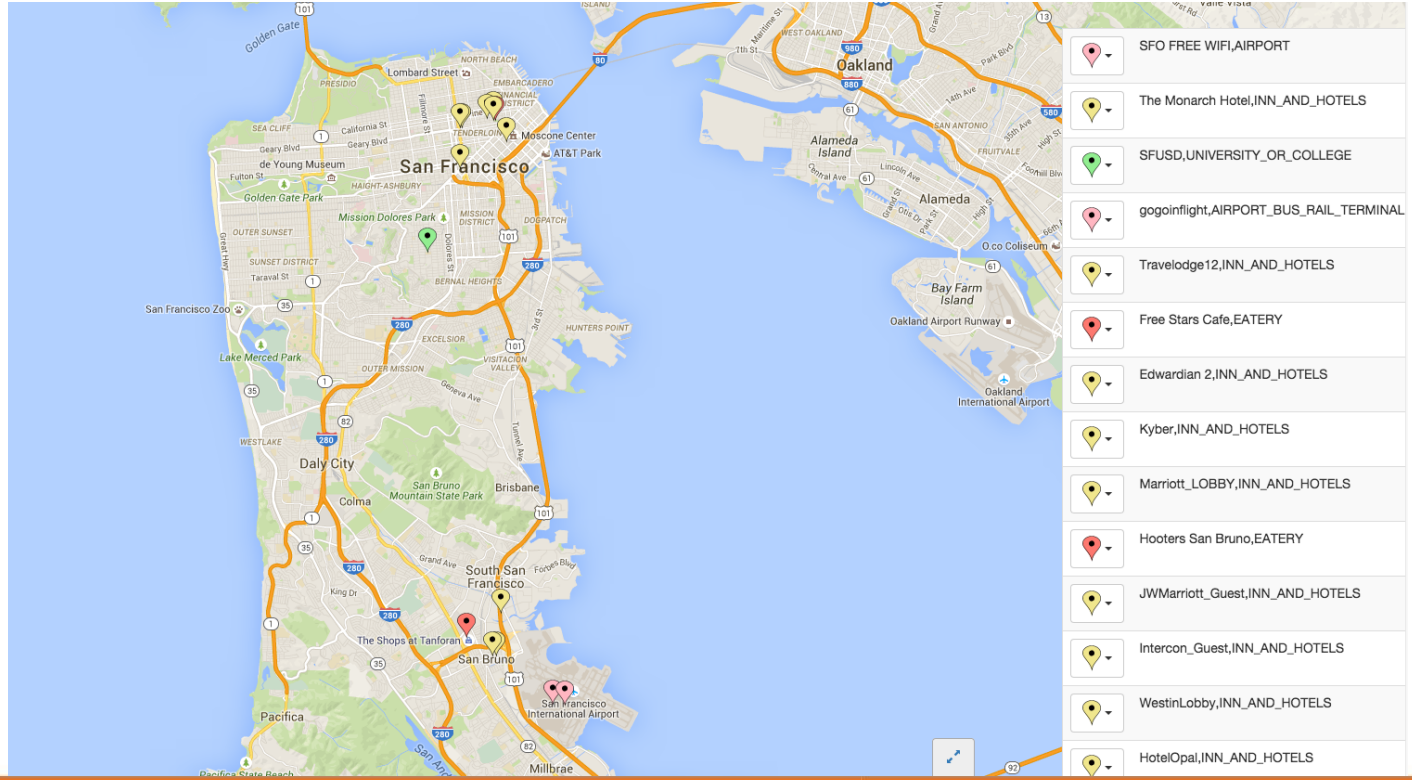
Connected Component

Students, Coffee Shops, Library



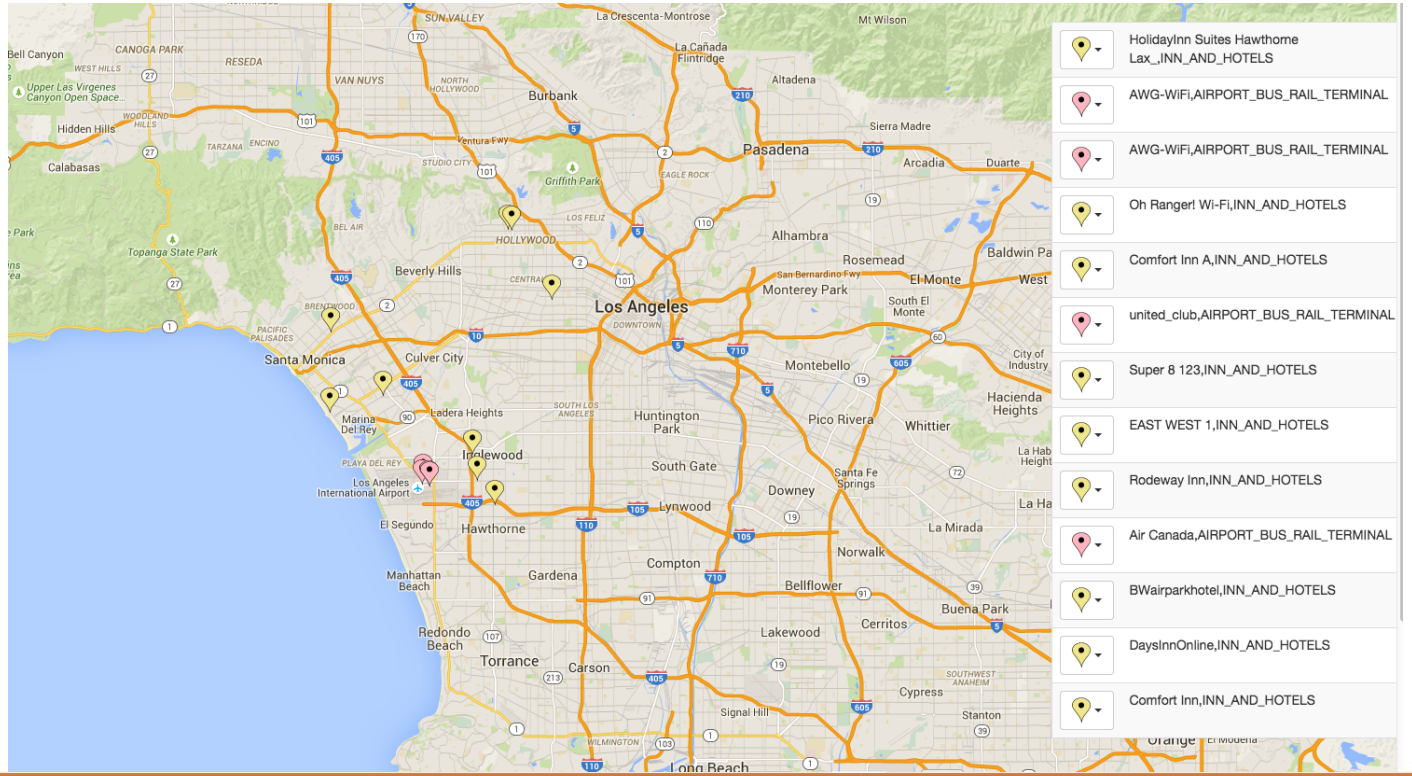
Connected Component

Frequent Business Travellers to SFO



Connected Component

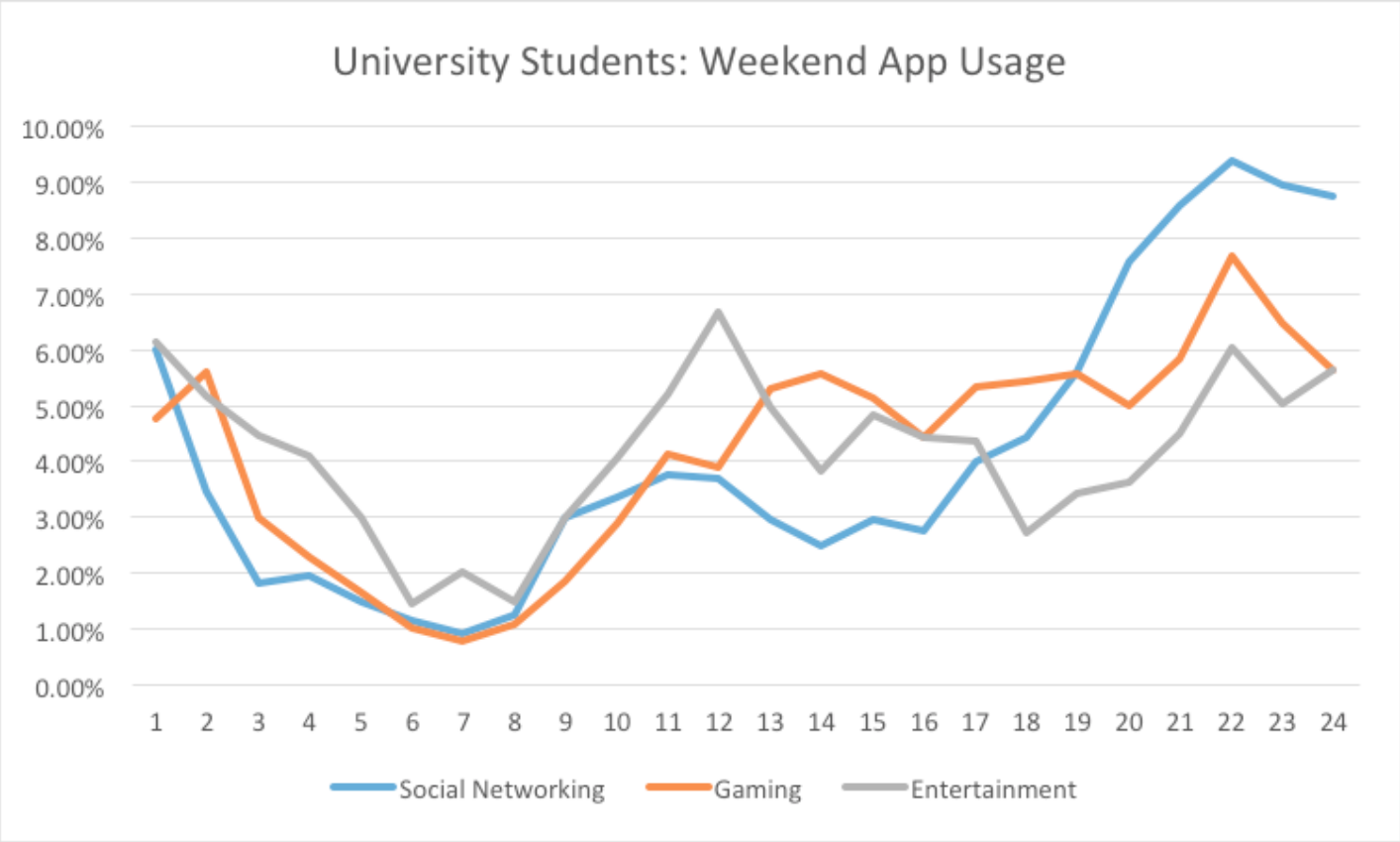
Frequent Business Travellers to LAX



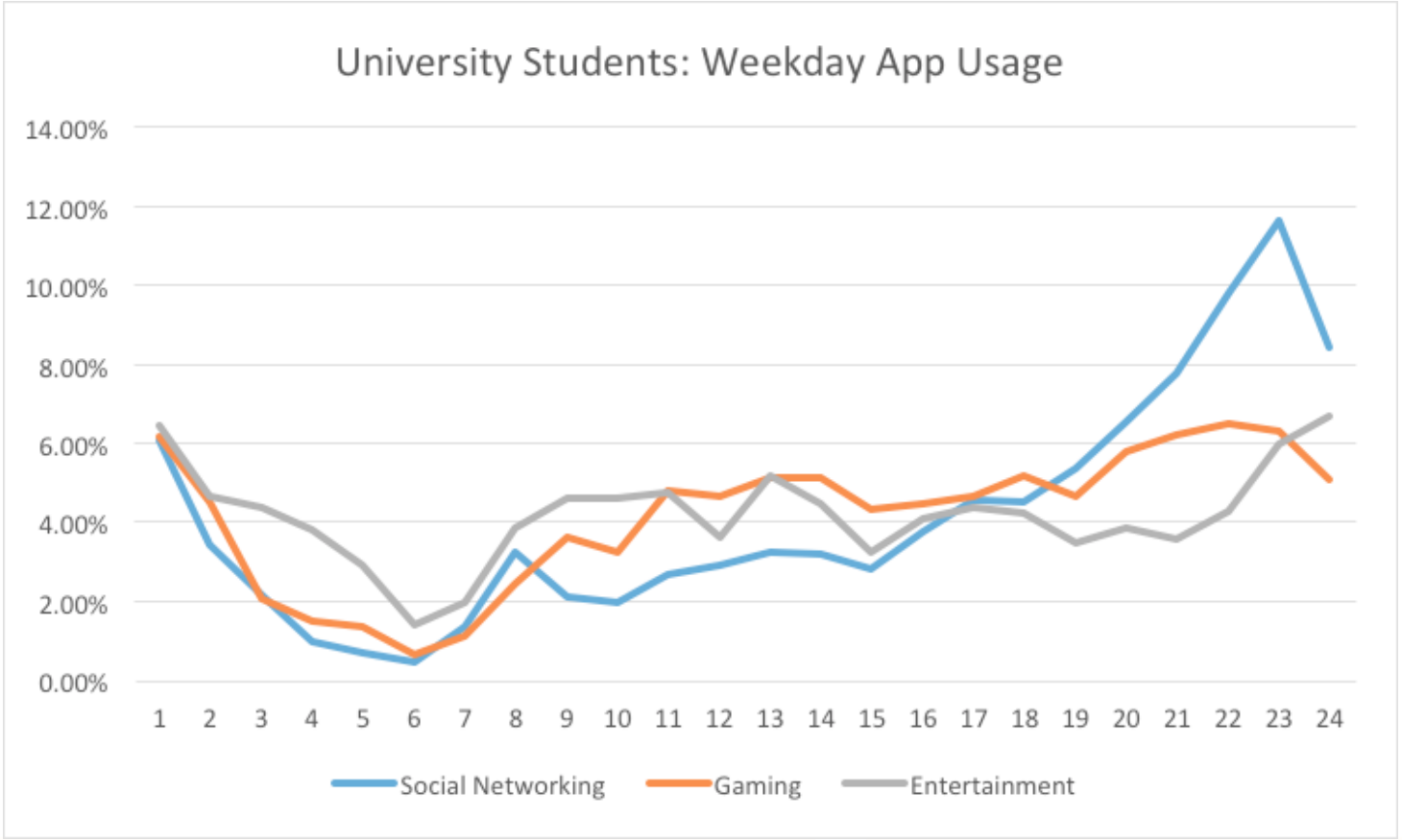
Using classifications: App Usage

University Students	General Population	Category
0.01%	6.65%	Productivity
0.14%	5.77%	Health & Fitness
0.01%	4.44%	Lifestyle
3.14%	4.26%	Entertainment
0.00%	3.33%	News
0.03%	2.91%	Reference
9.55%	2.42%	Tools
4.39%	0.28%	Social
5.26%	0.21%	Media & Video

Using classifications: App Usage



Using classifications: App Usage



Wider Spark Usage at InMobi

- Migration towards spark as the runtime of choice for data processing
- Legacy Pig jobs being switched to the Spark backend using Pig on Spark
- Pure MR applications being rewritten to use the Spark Java API

INMOBI™

Special Thanks to Paul Duff, Senior Research Scientist

Thank you
&
Questions