Aalto University
School of Science

Computational problems in mining urban data

Aristides Gionis

Department of Computer Science, Aalto University

users.ics.aalto.fi/gionis

SPIRE 2015, King’s College London, UK

September 1, 2015

urban data

- popular social-media applications are equiped with geolocation functionalities

  facebook, twitter, foursquare, instagram, flickr, ...

- additional sensor data and open data

  traffic sensors, mobile devices, emergency requests, crime, public transportation, food inspections, ...

- a lot of information about us and our relation with our environment

  places we go, how we move, when and with whom, what we do, what we discuss, and (potentially) how we feel in each place

mining urban data – motivation

- how to take advantage of the available information?

  improve existing services and resource allocation

  improve city planning

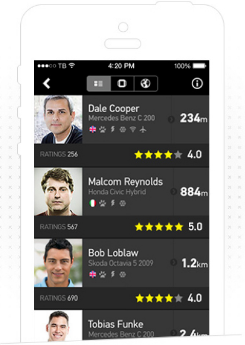
  increase safety

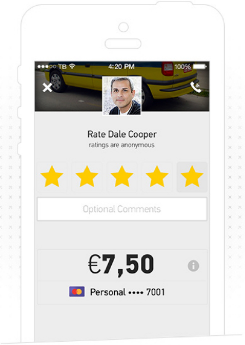
  increase public engagement

  improve city design and citizens well-beingness

  discover and enjoy the city

pick your taxi – taxibeat





Choose your driver

There is always a Taxibeat driver near you. Before you hail, you can text your driver and also select your payment method, all in a few taps on your screen. Select your driver from the

Enjoy and rate your experience

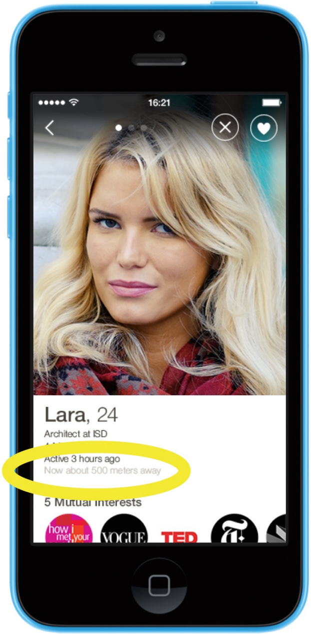
When your driver arrives, you will get notified to get on board. Sit back, relax, and enjoy your ride. Once the ride is completed, rate your

or find love – happn

You want to get in touch with someone you've crossed paths with?

You can check out their profile at any time and see the time and place of your last encounter.
You happen to find someone you like?
Like them secretly with the Heart button: they won't find out... unless the interest is mutual!
And if you wish to be noticed, charm them to send them a notification.
(For men, sending a charm costs 1 credit.)

or find love – happn



You want to get in touch with someone you've crossed paths with?

You can check out their profile at any time and see the time and place of your last encounter.
You happen to find someone you like?
Like them secretly with the Heart button: they won't find out... unless the interest is mutual!
And if you wish to be noticed, charm them to send them a notification.
(For men, sending a charm costs 1 credit.)

agenda

- overview of a few problems on mining urban data
- discussion of the underlying algorithmic problem

  application of existing methods or tailor-made techniques use urban data to reason about travel itineraries

travel itineraries

- tourists in a city leave a digital footprint
- where they go, points of interest, entertainment,

  how much time they spend where, where they go after
- can we combine the available info in order to

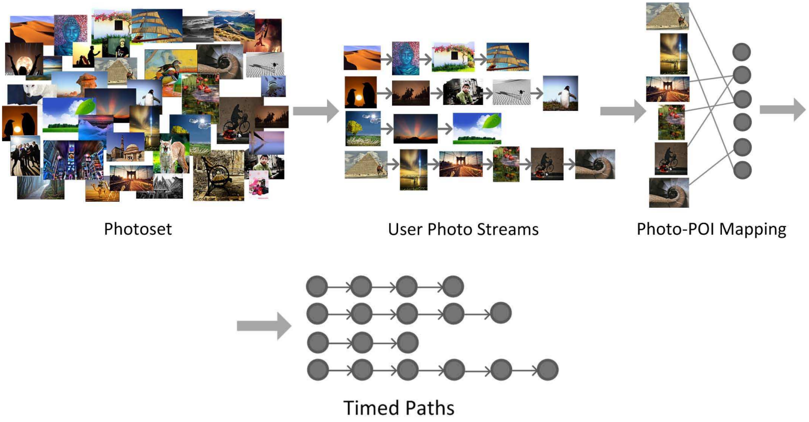
  understand what is popular and worth visiting

  suggest meaningful new itineraries, given constraints

constructing travel itineraries

[De Choudhury et al., 2010] :

- accomplish task using flickr data
- utilize location and timestamp of photos



constructing travel itineraries

[De Choudhury et al., 2010]

nice mix of computational problems

- data cleaning issues
- data mining problems on finding frequent transitions
  and assigning photos to points of interest
- but also theoretical abstractions

– set up itinerary construction as variant of orienteering

find a path to maximize reward, while satisfying constraints

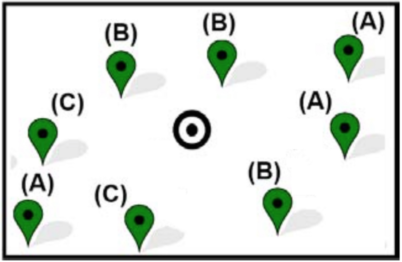
(quasipolynomial) logarithmic approximation

call for customization

- locations have types (art, restaurant, shopping, . . .)

  a user is interested in certain activities
- find a group of locations that satisfies user requirements,

  and are in geographic proximity



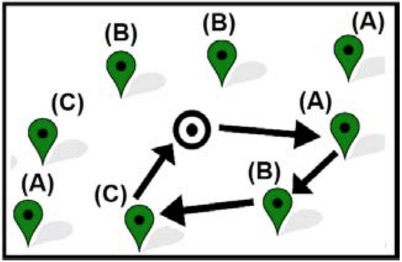
[AG et al., 2014]

call for customization

- venues have types (art, restaurant, shopping, . . .)

  a user is interested in certain activities
- find a group of venues that satisfies user requirements,

  and are in geographic proximity



[AG et al., 2014]

problem setup

- starting and ending point (e.g., my hotel)
- type preferences: visit only specific types
  e.g., museums, coffee shops, shopping malls, ...
- type ordering: visit types in specific order
  e.g., start $\rightarrow$ breakfast $\rightarrow$ museum $\rightarrow$ lunch $\rightarrow$
  $\rightarrow$ shopping $\rightarrow$ dinner $\rightarrow$ drinks $\rightarrow$ end
- distance constraints: travel at most distance D
- objective: maximize satisfaction

measuring satisfaction

additive:

- assign a score to each venue
  (allows personalization for a user profile)
- find the tour that maximizes total score

coverage:

- each venue covers a set of desirable features
  (e.g., local attractions, famous photo spots)
- overlap among covered sets is possible (and probable)
- find the tour that covers the most distinct features

algorithmic solution

- simple dynamic-programming solution
- for additive: optimal (pseudopolynomial) and FPTAS
- problem structure arises from the ordering constraint

  restrictive for the user, in practice

  can be relaxed to

  partial orders, super types, type skips

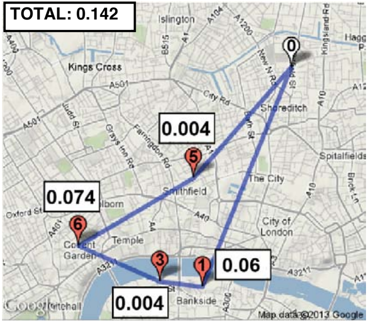
evaluation

- collect data from foursquare
- nine venue types, three cities
- satisfaction proportional to popularity within each type
- outperforming simple greedy baselines

details in [AG et al., 2014]

in London

(a) Cover-DP



(b) Cov-Greedy

distance = 6 miles

(1) = arts & entertainment, (3) = food, (6) = shop & services, (5) = nightlife

use urban data to reason about events

monitoring activity in the city

- understanding what is going on in the city
- events: collective activity, in time and place,
  which takes place out of the normal life cycle
  - social events, festivities, traffic accidents, weather disasters
- how to monitor activity data and detect events?

events in the city

e.g., in Barcelona :

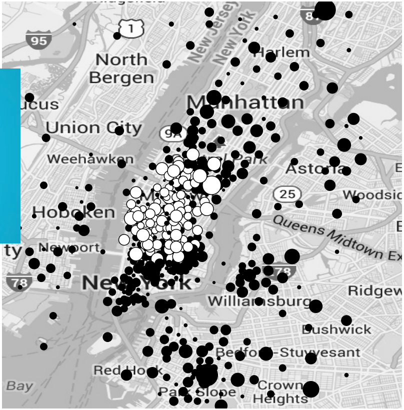
ordinary day, no events

an eventful day

data we can monitor

- example I :

  social media and location-based social networks

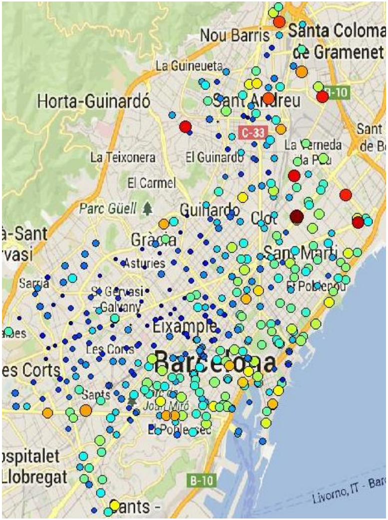


data we can monitor

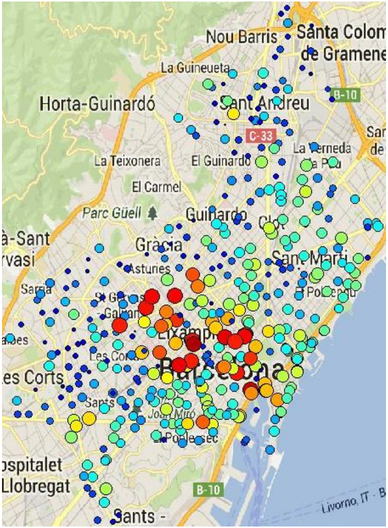
- example II :
  sensor networks and traffic measurements

using the bicing data

15.11.2012
ordinary day, no events



11.09.2012
Catalunya national day



setting up the problem

- given a graph $G = (V, E, d, w)$
  with a distance function $d : E \rightarrow \mathbb{R}$ on edges
  and weights on vertices $w : V \rightarrow \mathbb{R}$
- find a subset of vertices $S \subseteq V$
  so that

1. total weight in S is high
2. vertices in S are close to each other

[Rozenshtein et al., 2014]

setting up the problem

- what does total weight and close to each other mean?
- total weight

$$W(S) = \sum_{v \in S} w(v)$$

- close to each other

$$D(S) = \sum_{u \in S} \sum_{v \in S} d(u, v)$$

- want to maximize $W(S)$ and minimize $D(S)$
- maximize

$$Q(S) = \lambda\, W(S) - D(S)$$

[Rozenshtein et al., 2014]

remarks

1. not a temporal model

    working with snapshots

    temporal information is used to infer node weights

    large weight $\rightarrow$ abnormal activity

2. not a geometric model

    building a proximity graph and working with it

    considering geometry would allow more efficient methods

    but we can discover events of arbitrary shape

application to general networks

- find events in any social network with activity recordings

  twitter, blogs, entity graphs, news feeds

the event detection problem

- maximize $Q(S) = \lambda\, W(S) - D(S)$
- objective can by negative
- add a constant term to ensure non-negativity
- maximize $Q(S) = \lambda\, W(S) - D(S) + D(V)$

algorithmic solution

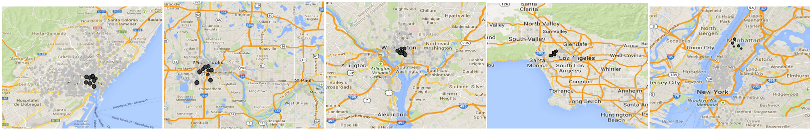
- maximize $Q(S) = \lambda\, W(S) - D(S) + D(V)$
- objective is submodular (but not monotone)
- can obtain $\frac{1}{2}$-approximation guarantee

  [Buchbinder et al., 2012]

- problem can be mapped to the max-cut problem
  which gives 0.868-approximation guarantee

  [Rozenshtein et al., 2014]

events discovered with foursquare and bicing data



(a) Barcelona: 11.09.12 National Day of Catalonia

(b) Minneapolis: 4.07.12 Independence Day

(c) Washington, DC: 27.05.13 Memorial Day

(d) Los Angeles: 31.05.10 Memorial Day

(e) New York: 6.09.10 Labor Day

Figure 4: Public holiday city-events discovered using the SDP algorithm.

(a) 01.06.12 Primavera sound music festival

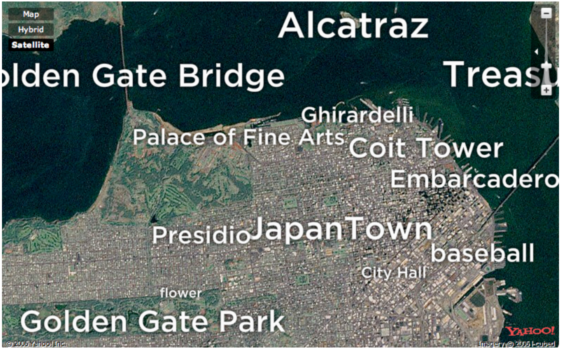
(b) 18.09.12 festival of the Poblenou neighborhood

(c) 31.10.12 Halloween

use urban data to reason about city neighborhoods

from social-media data to city maps

[Kennedy et al., 2007]



- spatial scan methods for finding high discrepancy areas

[Kulldorff, 1997]

questions to consider

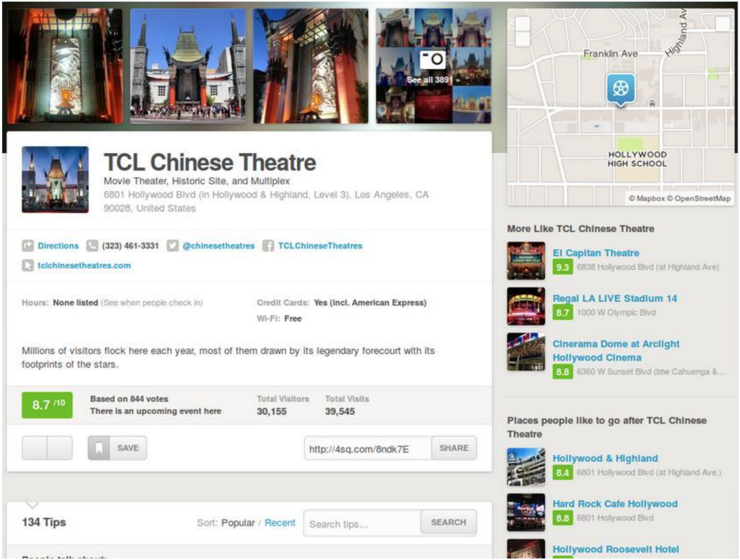
- how people experience and interact with their cities
- how are neighborhoods defined
- what is happening, what is unique in each neighborhood
- which neighborhood is similar to which?
  (in the same city or across cities)
- application: recommendations

the data

- venues (location, category) from foursquare
- check-ins (person, venue, time) from foursquare
- photos (person, location, time, tags) from flickr

[Le Falher et al., 2015]

foursquare venues



foursquare checkins

Swarm

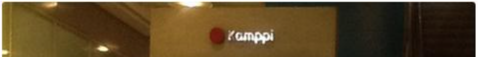
Pegre checked in at **DigiTintamareski1**

Kamppi | September 13, 2012 via foursquare for iPhone

This is always so much fun. @ClearChannelFI #Tintamareski At Kamppi level E

First check-in at DigiTintamareski1.

First of friends to check in at DigiTintamareski1.



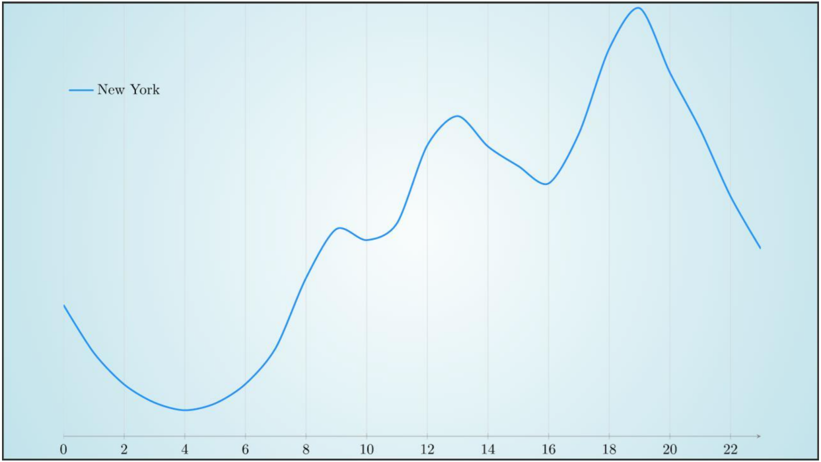
flickr photos

location: Helsinki, Finland
time: Dec 4, 2013, 11am
tags: foodporn, stockmann, helsinki

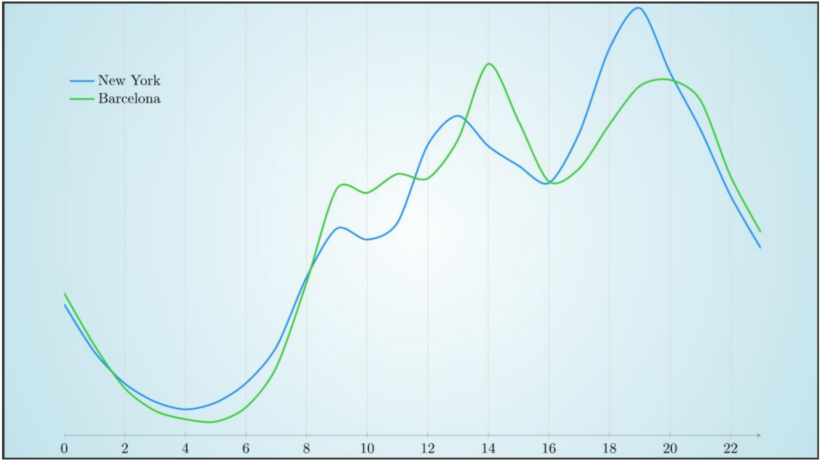
location: Helsinki, Finland
time: Feb 20, 2011, 6pm
tags: white cathedral, snow, helsinki

20 cities, 5 million checkins, 8 million photos

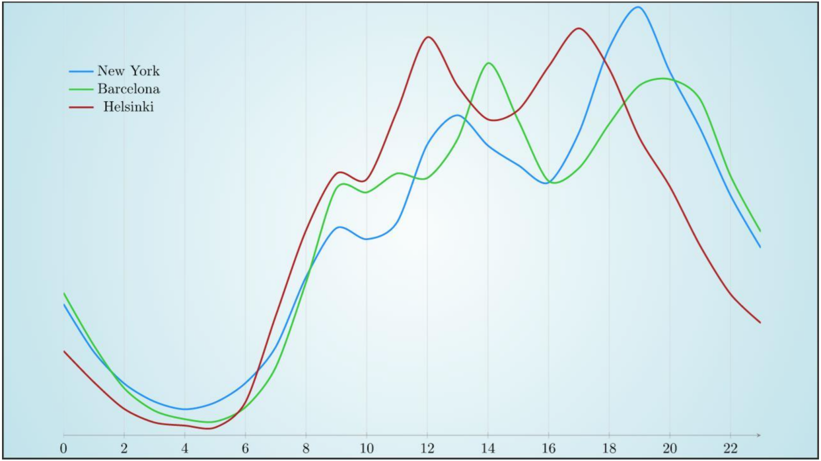
some data exploration



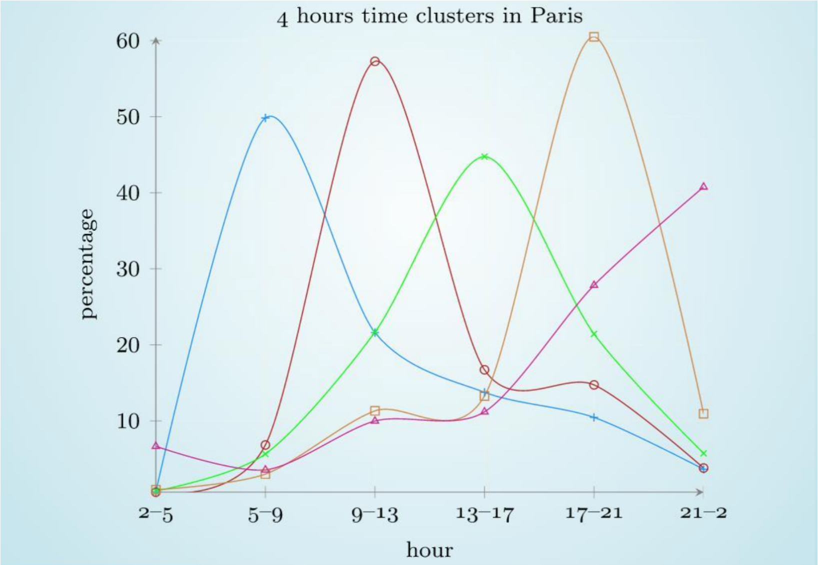
hourly check-in frequency



hourly check-in frequency



hourly check-in frequency



time activity of different venues

high-entropy venues

Paris

Eiffel tower

Gare SNCF de Paris Nord

Barcelona

sagrada familia

estacio sants

data model and problem setting

- each venue has geo coordinates (x, y)
- each venue described by a feature vector ($\dim = 30$)
- city / neighborhood : set of geo-located feature vectors
- the similarity search problem:

  find the most similar neighborhood to a given one

  (similarity? efficiency?)

comparing feature vectors distributions

- a number of different options

  earth mover's distance (EMD)

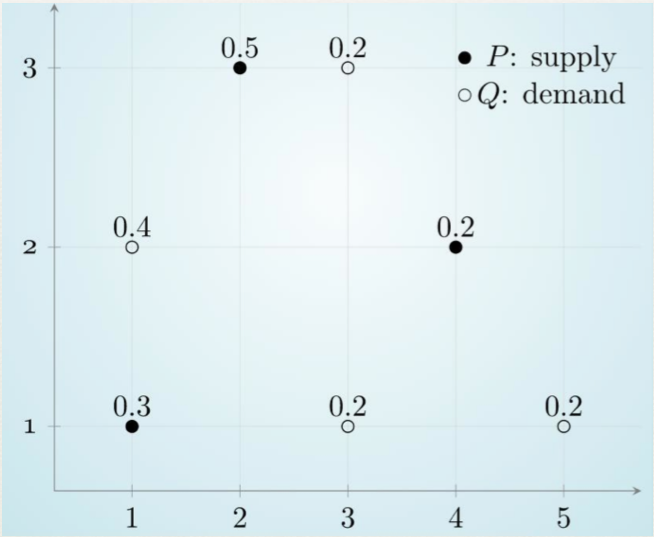
  Jensen–Shannon divergence (JSD)

  min-cost matching (MCM) on a set of centroids

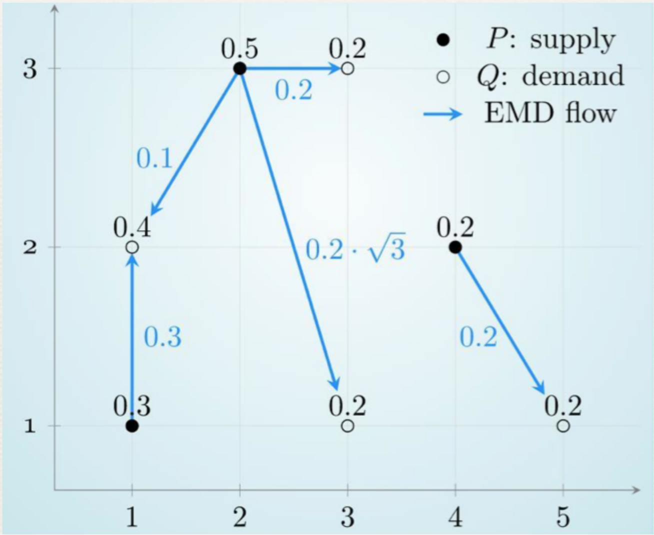
  …

- which one works the best for our setting?

recall: earth mover's distance



recall: earth mover's distance



a small-scale user study

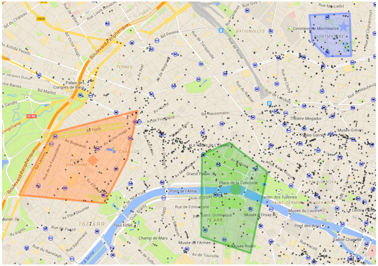
- ask locals to characterize neighborhoods in their cities
- 6 cities (Barcelona, NY, Paris, Rome, SF, Wash. DC)

target neighborhoods and answers for Paris

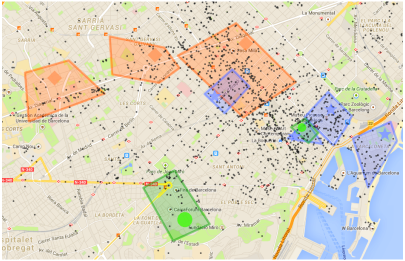
1 Fashion shops, luxurious places	Golden triangle
2 College & student neighborhood	Quartier Latin
3 Red light district	Pigalle
4 Touristic and artsy district	Montmartre
5 Government buildings	Official
6 LGBT neighborhood	Le Marais
7 Expensive residences	16th *arrondissement*
8 Parks & leisure	The banks of Seine

user-study interface

Paris



Barcelona



expensive residences, touristic and artsy, government buildings

user-study results

- which method agrees the most with user assessements

Query Source	Min cost matching	EMD-EUCL	EMD-LMNN	EMD-ITML	EMD-t-SNE	JSD	EMD-PARTIAL
Barcelona	.083	.078	**.084**	.033	.028	.042	.078
New York	**.059**	**.059**	**.059**	.049	.026	.057	.053
Paris	.061	**.091**	.078	.021	.044	.045	.061
Rome	.024	.042	.039	**.055**	.038	.021	.029
San Franc.	.045	.045	.040	**.060**	.042	.033	.044
Wash. DC	**.043**	.034	.038	.035	.026	.033	.038
Average	*.052*	**.058**	*.056*	*.042*	*.033*	*.038*	*.051*

- answer: EMD

similarity search

- challenges
  1. searching over distributions of feature vectors
  2. complex distance measure
  3. exponential search space
- option 1: exhaustive search

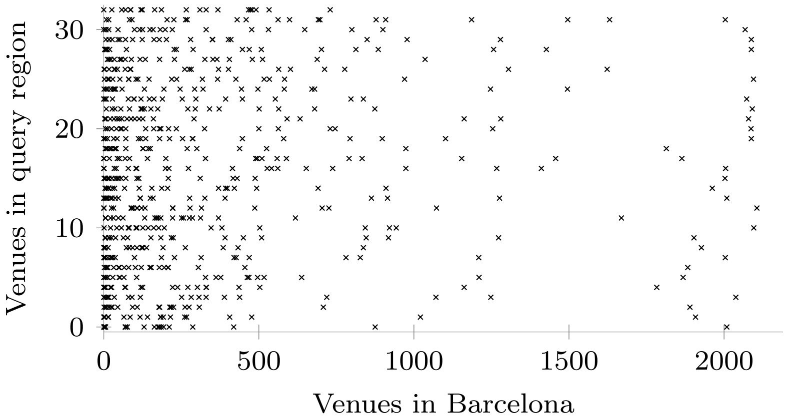
  search for a predefined shape

  still too slow
- option 2: prune the search space

  how exactly?

designing pruning strategy

- consider two areas with small EMD distance
- venues of one are in the top-k NN set of the other

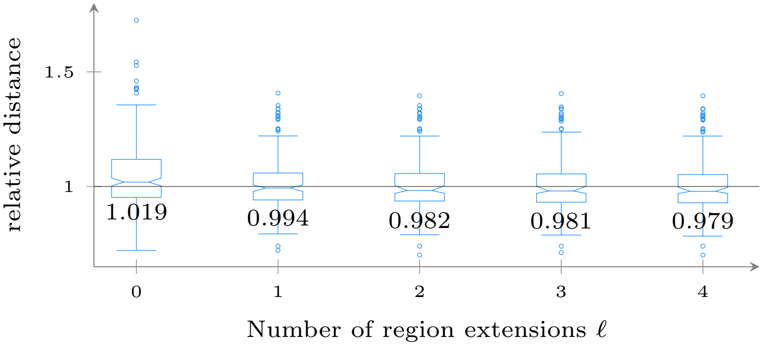


the pruning strategy

1. find matching locations
2. group locations by density-based clustering
3. expand and refine matching neighborhood

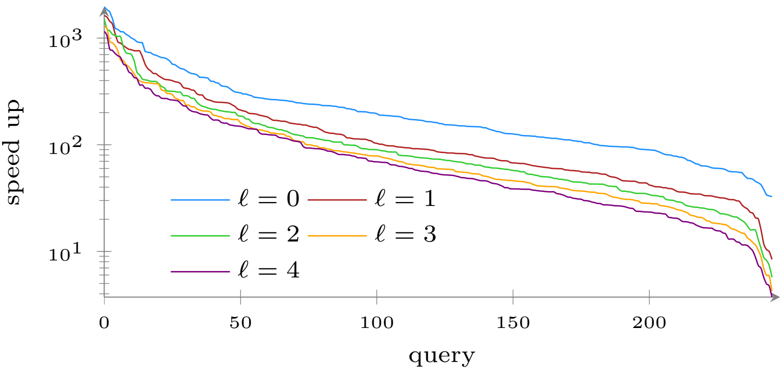
how well does it work?

accuracy



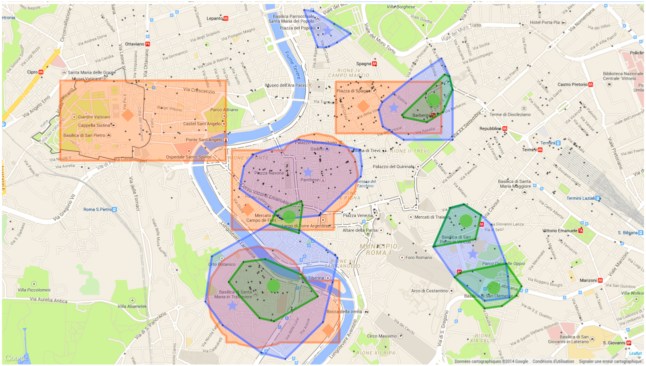
how well does it work?

efficiency



what does it actually find?

- quality depends on the available data



touristic and artsy neighborhoods in Rome

ground truth, and results with queries from Barcelona, Paris

conclusions

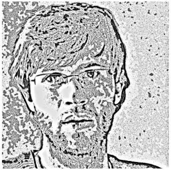
- wealth of data, wealth of problems

  mining, learning, recommendations, discovery, search
- challenges due to size, noise, heterogeneity,
  high dimensionality
- improve existing methods or work on new problems
- fun data to play and visualize

credits

Aris Anagnostopoulos

Ted Lappas



Géraud Le Falher

Michael Mathioudakis

Kostas Pelechrinis

Polina Rozenshtein

Nikolaj Tatti

Evimaria Terzi

references

Buchbinder, N., Feldman, M., Naor, J. S., and Schwartz, R. (2012).
A tight linear time (1/2)-approximation for unconstrained submodular maximization.
FOCS.

De Choudhury, M., Feldman, M., Amer-Yahia, S., Golbandi, N., Lempel, R., and Yu, C. (2010).
Automatic construction of travel itineraries using social breadcrumbs.
In *HT*.

Kennedy, L., Naaman, M., Ahern, S., Nair, R., and Rattenbury, T. (2007).
How flickr helps us make sense of the world: Context and content in community-contributed media collections.
In *International Conference on Multimedia*.

Kulldorff, M. (1997).
A spatial scan statistic.
Communications in Statistics-Theory and Methods, 26(6):1481–1496.

references (cont.)

Le Falher, G., Mathioudakis, M., and Gionis, A. (2015).
Where is the Soho of Rome? Measures and algorithms for finding similar neighborhoods in cities.
In *ICWSM*.

Rozenshtein, P., Anagnostopoulos, A., Gionis, A., and Tatti, N. (2014).
Event detection in activity networks.
In *KDD*.