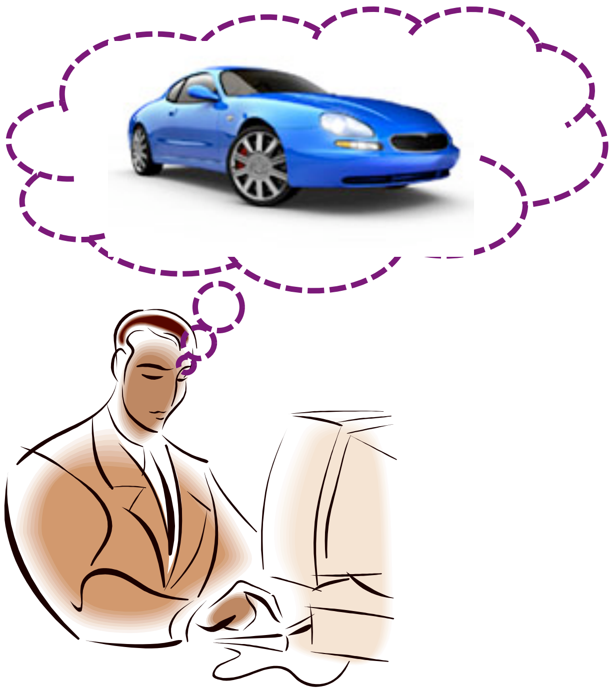
Discover Users' Specific Geo Intent in Web Search

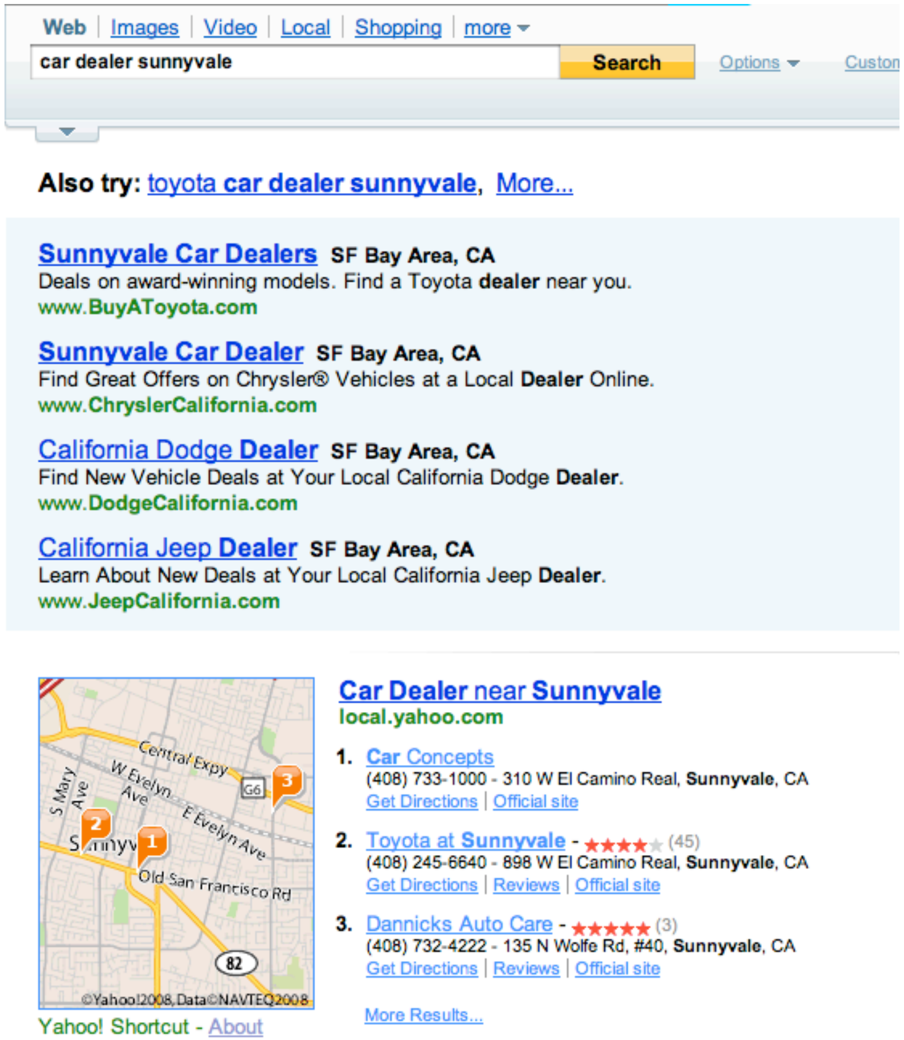
Xing Yi[1], **Hema Raghavan**[2] and **Chris Leggetter**[2]

[1] **C**enter for **I**ntelligent **I**nformation **R**etrieval, University of Massachusetts Amherst

[2] **Yahoo!** Labs, 4401 Great America Pky, Santa Clara, CA

Queries with Geo Intent





Geo Intent

Geo intent – a user's information need has some kind of entity which has a geographic (geo) location associated with it:

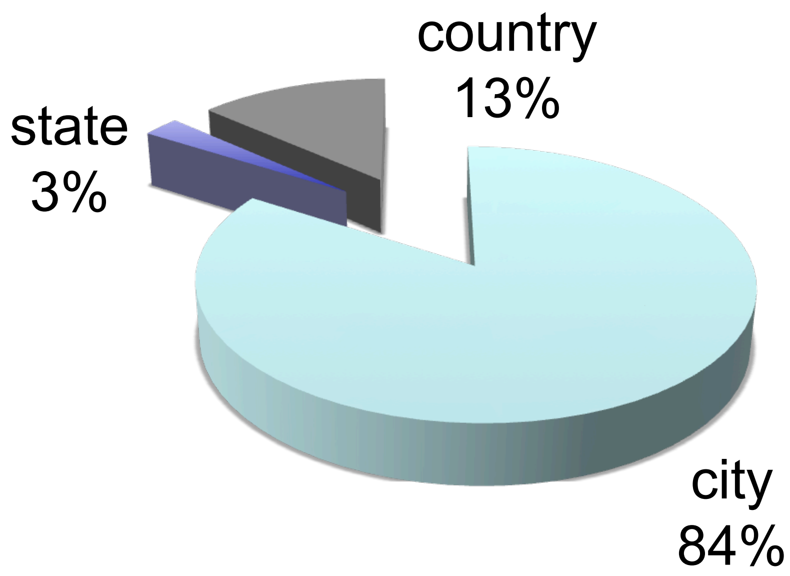
- **explicit:** “one bedroom apartment new york city”, “madrid guided tour”
  - An **explicit** geo query has two portions: e.g. “**car dealer** in **sunnyvale**”,

non location part *location part*

- **implicit:** “pizza delivery”, “dental care”, “day care”, “rockefeller center”

Observations about Web Geo queries

- many web queries contain geo info
  - About 13-14% queries have a place name (Jones *et al.* , Intl. J. of G.I. Science 2008, Sanderson & Kohler, SIGIR GIR workshop 2004)
  - About 30% queries may have geo intent; **only about half** of them have explicit geo info. (Welch & Cho, SIGIR 2008)

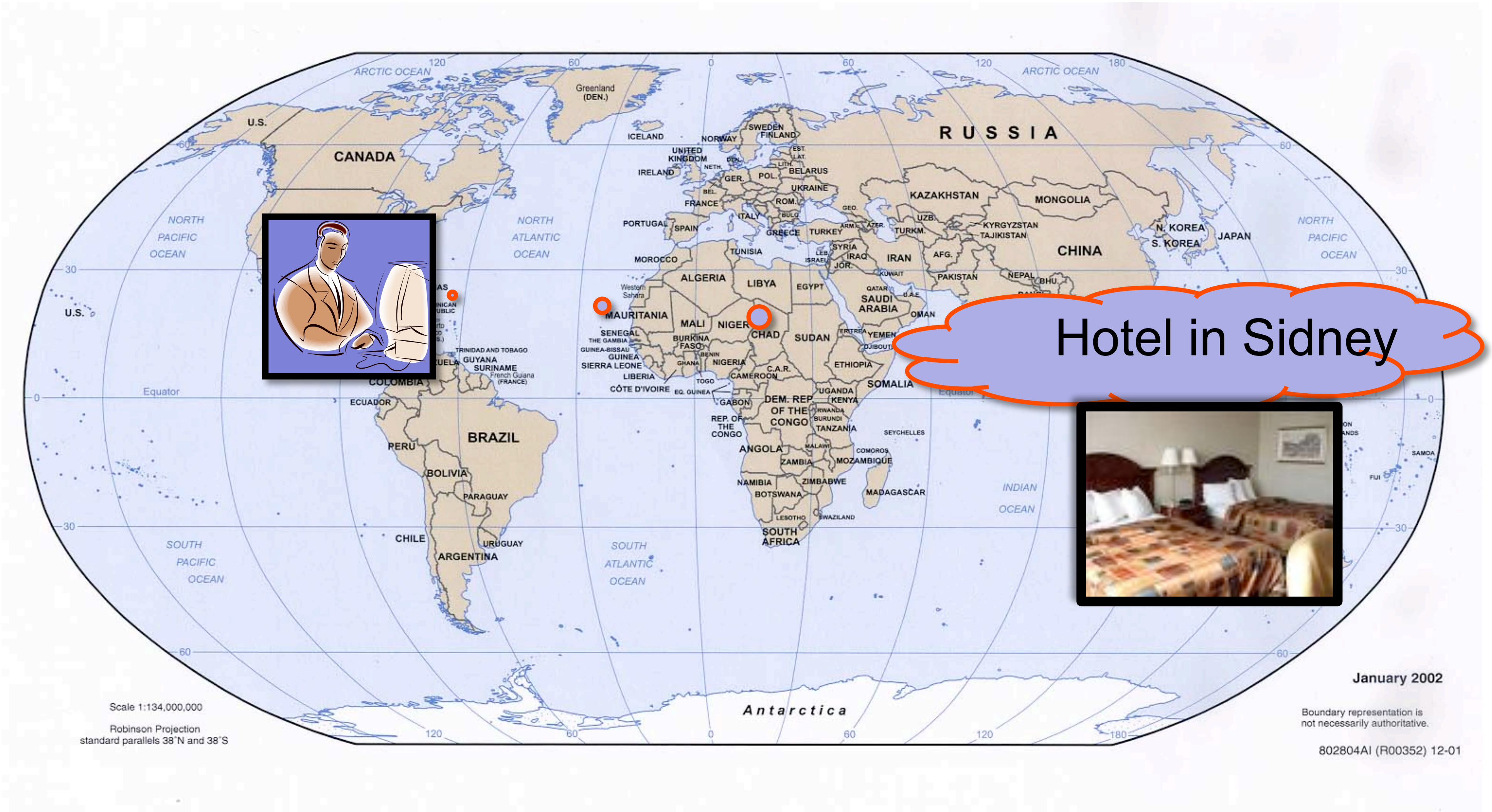


Research questions/tasks

① Given a set of queries with no mentions of any location (town/city/state), can we predict which of these have **implicit geo intent?**

Localization capability of a geo-intent query

Queries with Geo intent but not localized



Research questions/tasks

① Given a set of queries with no mentions of any location (town/city/state), can we predict which of these have **implicit geo intent?**

② What is the **localization capability** of a geo-intent query?

③ What is the **city** corresponding to the geo-intent?

Applications

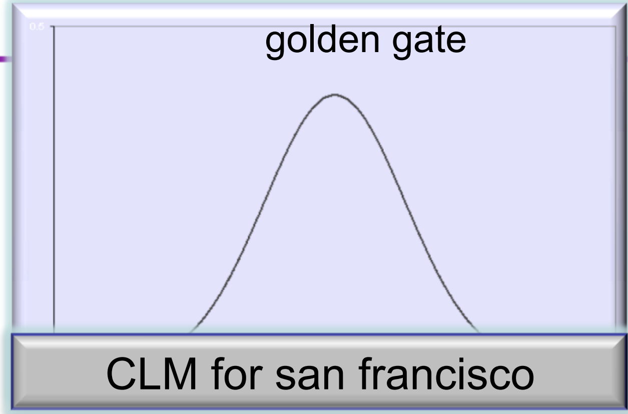
Benefits for finding users' geo intent:

- Personalizing web search results
- Better sponsored online advertisement matching

More benefits for finding users' specific geo intent at a fine-grained **city**/location level:

- Delivering more local goods and services
- Finding local news and events

Outline of remainder of the talk



- Feature extraction
  - City Language Models
    - Entity Language Models (Raghavan *et al.* ACM LinkKDD 2004)
- Experiments for each of the 3 tasks:
  - Label generation : **millions of training samples from click data.**
  - Evaluation
- Conclusions and Future Work

City Language Models

Feature Extraction

Use Internal tool

	query Q_{nc}	freq
san fancisco	pizza	200
	cheap hotel	150
	49ers	125
	zoo	100
	golden gate	75

- Bigram language model
- Smoothed
- Details in the paper

$$P(q \mid C_k) = \prod_{i=1}^{n} P(w_i \mid w_1^{i-1}, C_k) \approx \prod_{i=1}^{n} P(w_i \mid w_{i-1}, C_k)$$

City Language Models

Feature Extraction

- Calculate the posteriors:

$$P(C_i \mid q) \propto P(C_i)P(q \mid C_i)$$

- These posteriors are used for predicting the locations for *location-specific* queries

- Top-10 posteriors are used as features for classifications

Some examples

Feature Extraction

"Disney world ticket"		"Harvard University"	
City Name	$P(C_i \mid q)$	**City Name**	$P(C_i \mid q)$
Orlando	0.98011	Cambridge	0.63545
Kissimmee	0.01386	Princeton	0.05360
Anaheim	0.00240	Longwood	0.05334
New Castle	0.00135	Boston	0.01979
San Antonio	0.00044	Tuskegee	0.01719
…	…	…	…

Geo Information Units (GIU)

Feature Extraction

	Q_{nc}	freq	
san fancisco	pizza	200	Global information units
	cheap hotel	150	
	49ers	125	
	zoo	100	
	golden gate		

GIUs like ***pizza*** co-occur with many different city names

Features based on GIUs

Feature Extraction

Examples:

- Probability $P_g(w_i^{i+n-1})$ of a GIU appearing in geo queries
- Probability $P(w_i^{i+n-1})$ of this GIU appearing in all the queries
- The pair-wise mutual information (PMI) between the w_i^{i+n-1} and each location

Aggregate features and individual GIUs as features

Experiments

Overall Data Description

Three learning tasks:

- Classifier I: Detecting implicit geo queries
- Classifier II: Discriminating different localization capabilities of geo queries: local geo intent, neighbor region geo intent, *etc.*
- City language models: Predicting geo entities related to a query

Evaluations and Results

Data Description

Task 1

Slice of traffic:

- Training data
  - 1.44b queries in May 08
  - 96.2m are explicit geo queries (**training geo subset**)
- Testing data
  - 1.42b queries in June 08
  - 96.7m are explicit geo queries (**testing geo subset**)

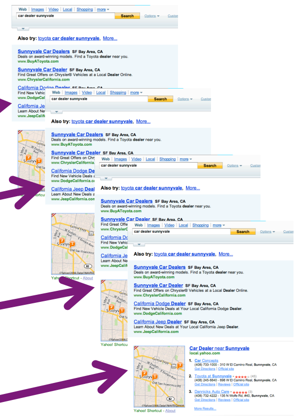
Weakly supervised automatic label generation

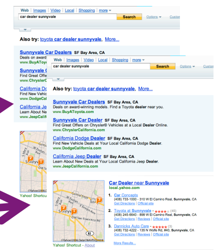
Generating labeled data for Classifier I

Task 1

training geo subset

san francisco	pizza
san francisco	cheap hotel
san francisco	49ers
san francisco	zoo
…………	
new york	pizza
new york	zoo





Logs of what was shown to the users

Step 1: get the clicked url for each query (domain name)

Generating labeled data for Classifier I

Task 1

Step 2: *DN1* is set of top 100 clicked domains from Step 1.

Step 3: *DN2* is set of top 100 clicked domains from queries in *training set* and not in *training geo subset*.

Step 4:

- $DN+ = DN1 \notin DN2$ $DN- = DN2 \notin DN1$
- If a query in training geo subset has clicked domain in DN+ → positive sample
- *non-location* parts of positive samples as the final implicit geo intent queries.
- randomly sample 20,000 implicit geo queries and 20,000 non-geo queries to train classifiers

Some examples in DN+ and DN-

Task 1

DN+ – DN as Positive label	DN- – DN as Negative label
www.citysearch.com	en.wikipedia.org
www.yellowpages.com	answers.yahoo.com
local.yahoo.com	search-desc.ebay.com
www.local.com	www.youtube.com
travel.yahoo.com	www.amazon.com
www.tripadvisor.com	www.myspace.com
www.yellowbook.com	www.nextag.com
www.city-data.com	

Generating labels: Test data

Task 1

Two testing subsets from testing data

Testing data I:

- Same as training data process, but on testing data.
- 40,000 implicit geo queries + 40,000 negative queries

Testing data II:

- Extract all queries that have DNs in DN+ or DN-.
- Remove all possible location information using WOE
- Sample 40,000 implicit geo queries + 40,000 negative queries
- May have queries that had implicit geo intent to begin with.

Task 1

Evaluations

Three classifiers

- Support Vector Machines (linear kernel and RBF gaussian kernel)
- Gradient boosting decision trees (Treenet)
- Multinomial Logistic Regression

5-fold cross-validation

Results using CLM features + aggregated GIU features

Task 1

	P	R	A
Testing Set I			
SVM-Linear	**91.7%**	82.6%	87.6%
SVM-RBF	91.4%	86.0%	**89.0%**
Treenet	89.4%	**87.4%**	88.5%
Logistic-R	91.3%	83.5%	87.8%
Testing Set II			
SVM-Linear	**80.9%**	35.7%	63.7%
SVM-RBF	80.4%	36.2%	63.7%
Treenet	78.1%	**40.9%**	**64.7%**
Logistic-R	80.2%	36.4%	63.7%

Using aggregate GIU stats as well as GIUs as individual features.

Task 1

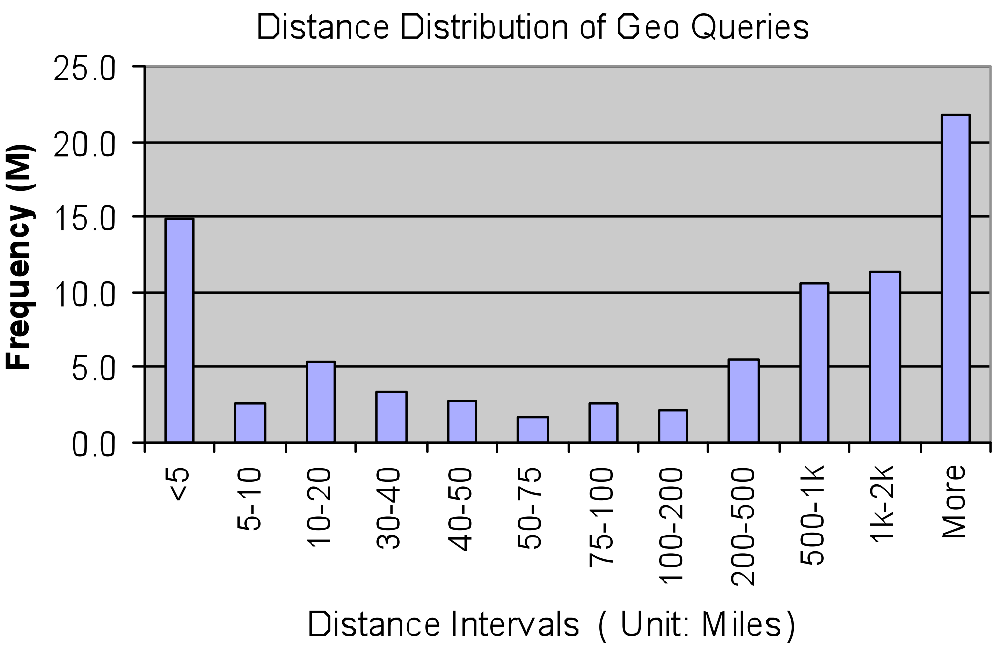
	P	R	A
Testing Set I			
SVM-Linear	**99.9%**	**66.0%**	**83.0%**
SVM-RBF	98.5%	62.8%	80.9%
Testing Set II			
SVM-Linear	**99.9%**	**48.8%**	**74.4%**
SVM-RBF	97.8%	48.0%	73.5%

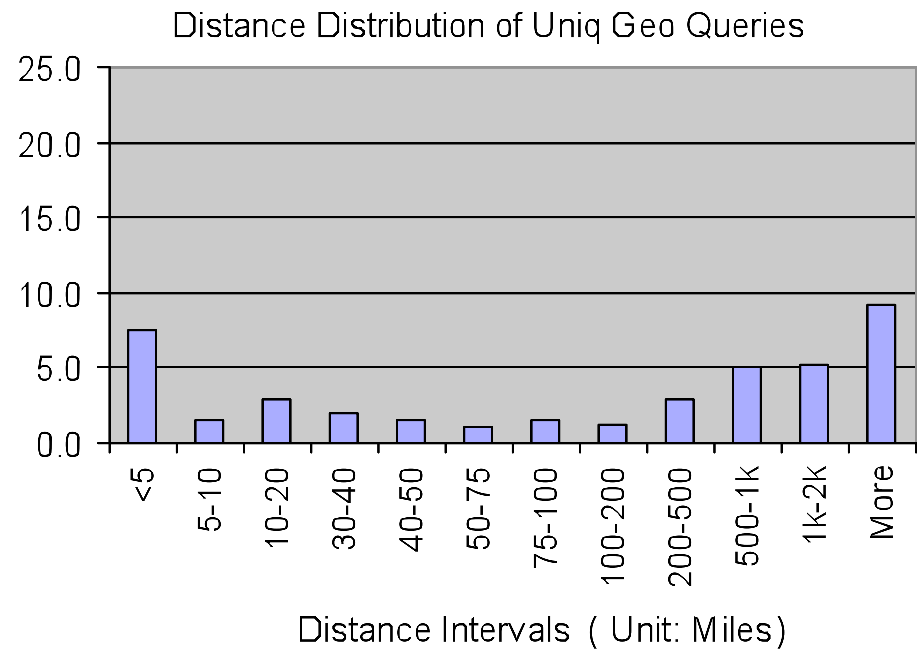
Classifier II: Localization capability of a query

Task 2

$L(q,C)$ = distance between the city C (Q_c) in a query q (Q_{nc}) and the user IP

$L^m(q)$ = median of all $L(q,C)>0$ for all cities C associated with q.





Data generation for Classifier II

Task 2

- 3 classes:
  - $L^m(q) <= 50$ miles → q is a **local geo query (LG)**
  - $50 < L^m(q) < 100$ miles → **neighbor region query (NG)**
  - **other geo queries (OG)**

Results

Task 2

	A (LG/OG)	B (LG/NG)	C (NG/OG)	D (ALL)
Case I	aggregate GIU features			
SVM-Linear	61.3%	53.5%	61.0%	42.6%
SVM-RBF	62.0%	53.9%	**61.8%**	43.2%
Treenet	**62.8%**	**54.2%**	60.8%	**44.1%**
Logistic-R	61.2%	53.4%	61.0%	42.6%
Case II	high dimensional features			
SVM-Linear	99.6%	97.2%	96.9%	87.0%
SVM-RBF	**99.6%**	**98.0%**	**98.0%**	**96.6%**

Predict Locations for Location-Specific Queries

Task 3

- Queries with mentions of an entity that is directly associated with a location: eg., hotels, local tv and radio channels, local newspapers, universities etc.
  - "airport check metro airport" → detroit
  - "woodfield mall jobs" → schaumburg

- Data is generated using several rules (refer paper).
- Top cities using the City Language Models ($P(C_i \mid q)$)were taken as predictions.

Task 3

Human Evaluation: key points

669 randomly sampled *location-specific* queries and their predicted related locations

Request annotators to answer two questions with `yes/no/?':

- whether the selected query was a *location-specific* query (84.5 % inter annotator agreement)
- Whether the predicted location was correct (73% agreement)

Of queries marked location specific, **accuracy** of predicting a location was **84.5%**.

Concluding Remarks

- Methods for
  - identifying users' implicit city-level geo intent.
  - discriminating different localization capabilities of geo queries.
  - predicting the city corresponding to the geo-intent in a *location-specific* query.
- The models are learned from large amounts of click-through data and involve little supervision.
- Future Work:
  - Incorporate our CLM into retrieval models.
  - Use geo intent analysis results for helping search engines provide better query suggestions.
  - Exploit other data sources

Questions?

THANK YOU !