A Probabilistic Query Suggestion Approach Without Using Query Logs

Meher T. Shaikh Maria Soledad Pera Yiu-Kai Ng
Computer Science Department
Brigham Young University
Provo, Utah, USA

Web Searches

- Problems encountered by using keyword searches:
  - Queries tend to be short and ambiguous
  - Users may lack knowledge on formulating queries for difficult topics

Widely-adopted solution

Query Suggestion (QS) Module:

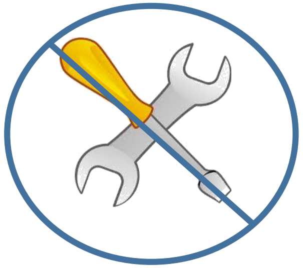
Recommends *useful queries* for a user's input keywords/query

- Benefits
  - Assist users in formulating queries that capture their information needs

Query Suggestions

- Current design issues affecting existing QS modules:

Depend on very large query logs



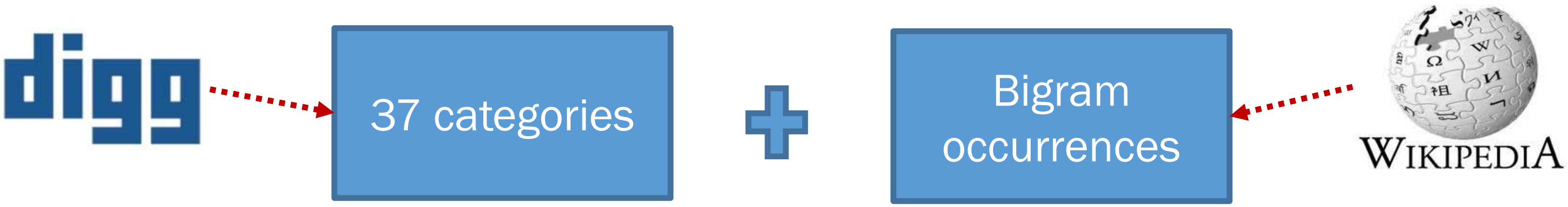
Cannot be adapted for small-scale search applications

Proposed solution

A *Probabilistic Query Suggestion (PQS)* Module

Probabilistic Query Suggestion (PQS)

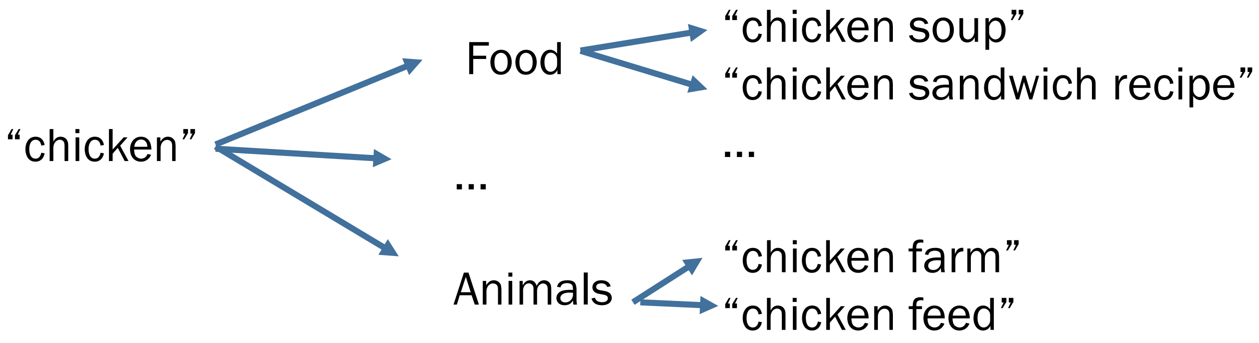
STEP 1. *Candidate suggestions, CSs,* given a user's query *Q*



37 categories

+

Bigram occurrences



"chicken"

Food

"chicken soup"

"chicken sandwich recipe"

…

…

Animals

"chicken farm"

"chicken feed"

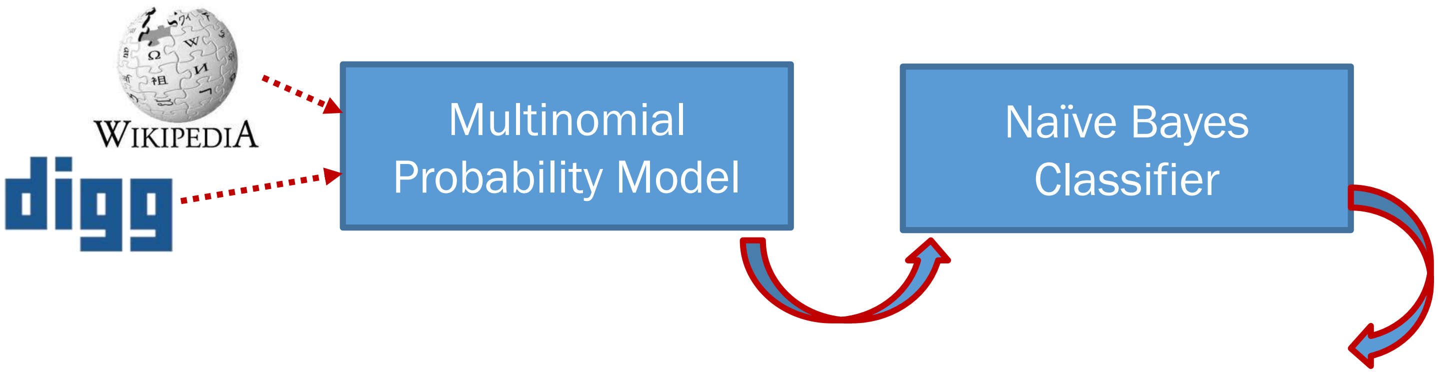
Query

Categories

Candidate suggestions, which are generated by analyzing bigram occurrences in Wikipedia documents

PQS

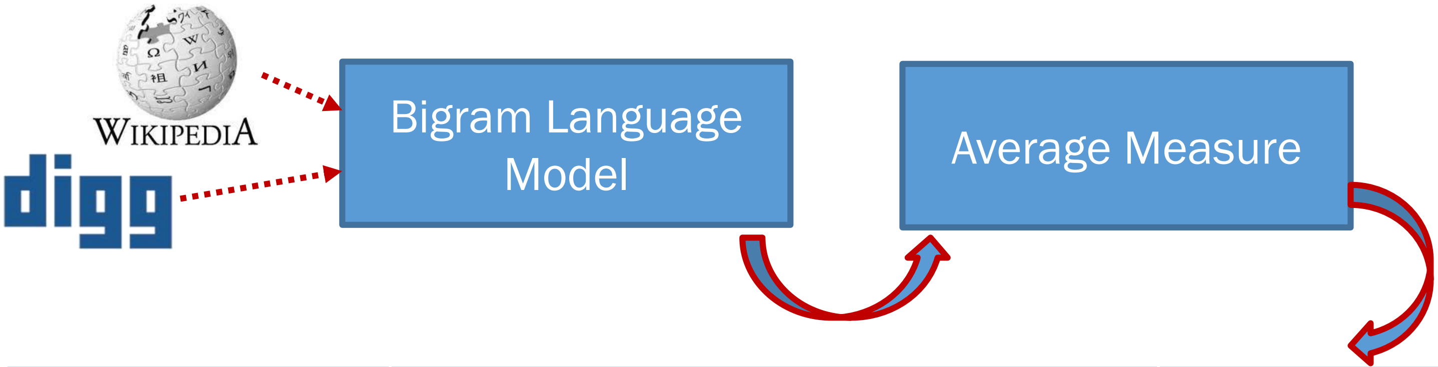
STEP 2(a). *Probability* of candidate suggestions from diverse categories



Q	Categories	Category Likelihood Score (CL)
Chicken	Animals	0.25
	…	…
	Food	0.32
	…	…

PQS

STEP 2(b). *Probability* of generating the *word sequence* of *CSs* for *Q*



Bigram Language Model

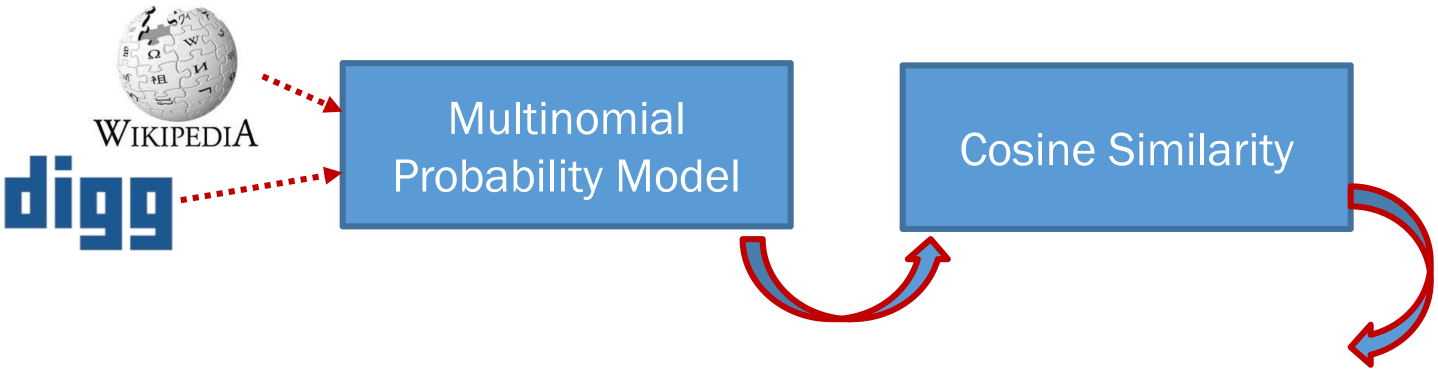
Average Measure

Candidate suggestion	Bigram probabilities	*N-gram Probability Score* (NGS)
Chicken soup	P(soup|chicken) , P(EOS|soup)	0.65
Chicken sandwich recipe	P(sandwich|chicken), P(recipe|sandwich), P(EOS|recipe)	0.39

Special case

PQS

STEP 2(c). *Cohesiveness* of (words in) *CSs* with respect to their corresponding *categories*



Multinomial Probability Model

Cosine Similarity

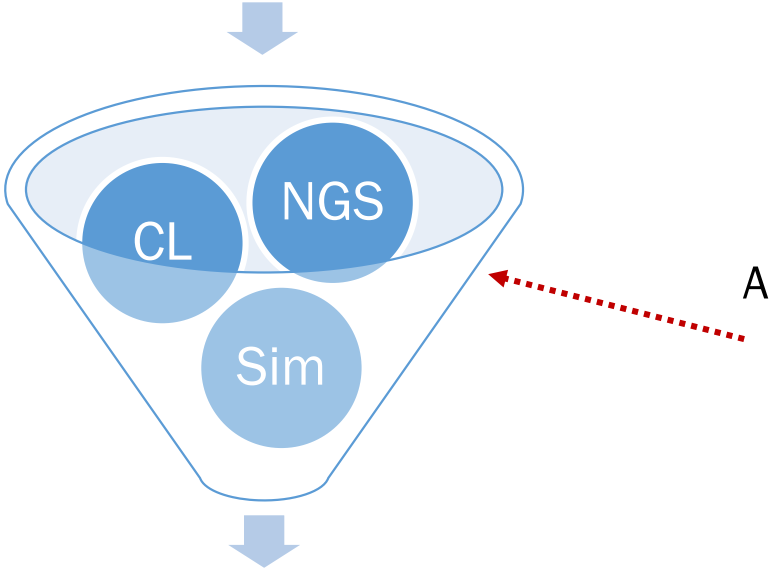
Candidate suggestion	Category	Probability of a word given a category	Degree of Cohesiveness Score (Sim)
Grand prix	Cars	P(grand|cars), P(prix|cars)	0.99
Grand slam	Cars	P(grand|cars), P(slam|cars)	0.71

PQS

STEP 3. Rank CSs for *Q*

- *Top-k (k ≥ 1)* candidate suggestions with the *highest* ranking score are recommended to its user who creates *Q*

A candidate suggestion (*CS*)



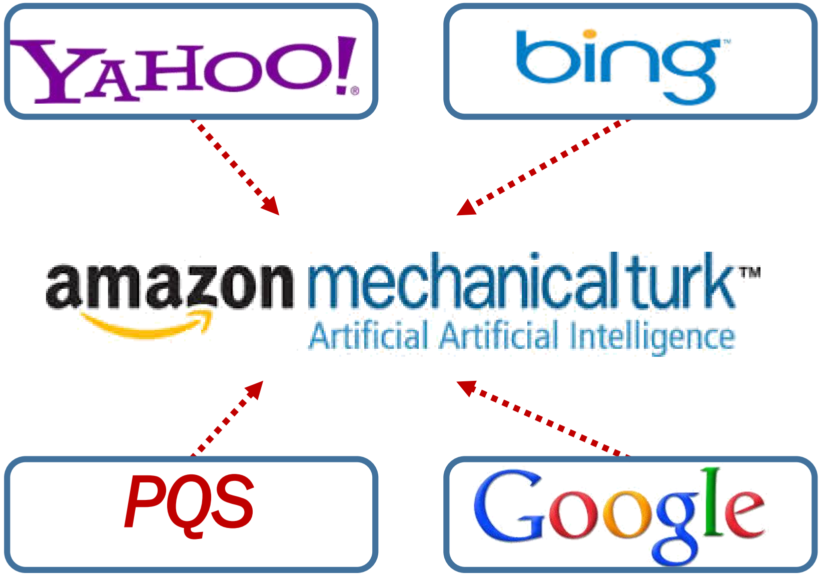
Applying an aggregation strategy

Ranking score of *CS*

Experimental Results

- Dataset

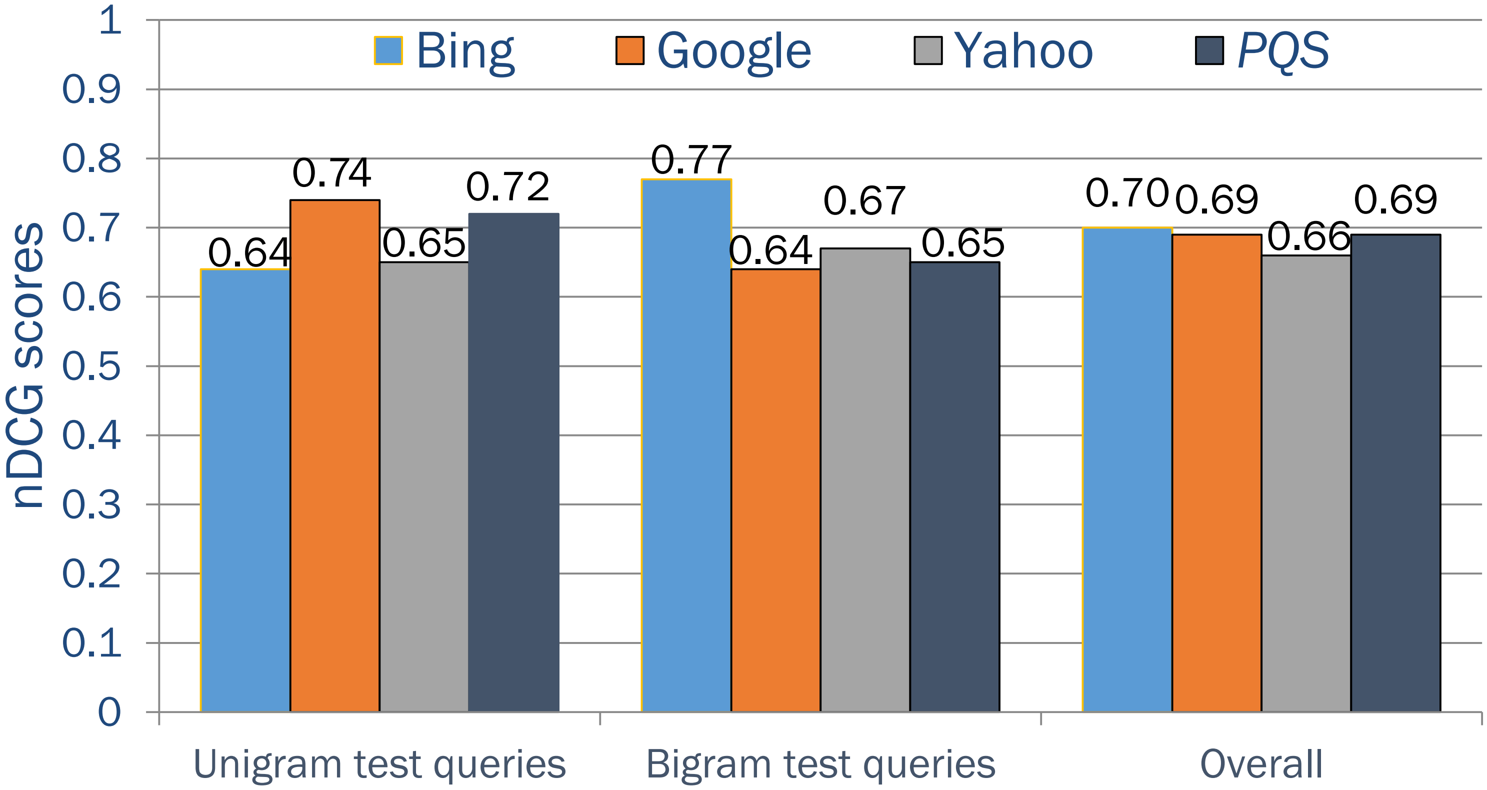
36 Test Queries + Gold Standard



- Metric
  - Normalized Discounted Cumulative Gain (*nDCG*)
    - Applies logarithmic penalization on *useful* suggestions ranked *lower* in the list of suggested queries

Experimental Results

- PQS vs. QS modules used by commercial web search engines



Conclusions

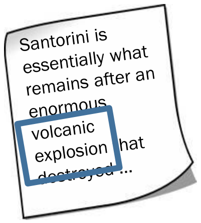
- Resources

publicly-accessible web documents

topical categories

- Strategy

Probability distribution



Santorini is essentially what remains after an enormous volcanic explosion that destroyed ...

Likelihood of co-occurrence

Cohesiveness

Analyzes word sequences in each candidate suggestion for a user query

- Performance

It is comparable to commercial search engines