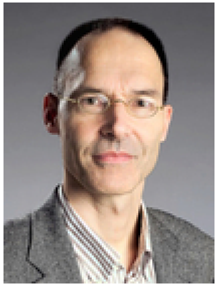
Identifying Queries in Instant Search Logs

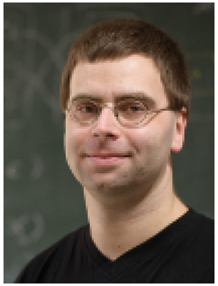
Markus Fischer

Kristof Komlossy



Benno Stein

Martin Potthast



Matthias Hagen

FRIEDRICH-SCHILLER-UNIVERSITÄT JENA

Bauhaus-Universität Weimar

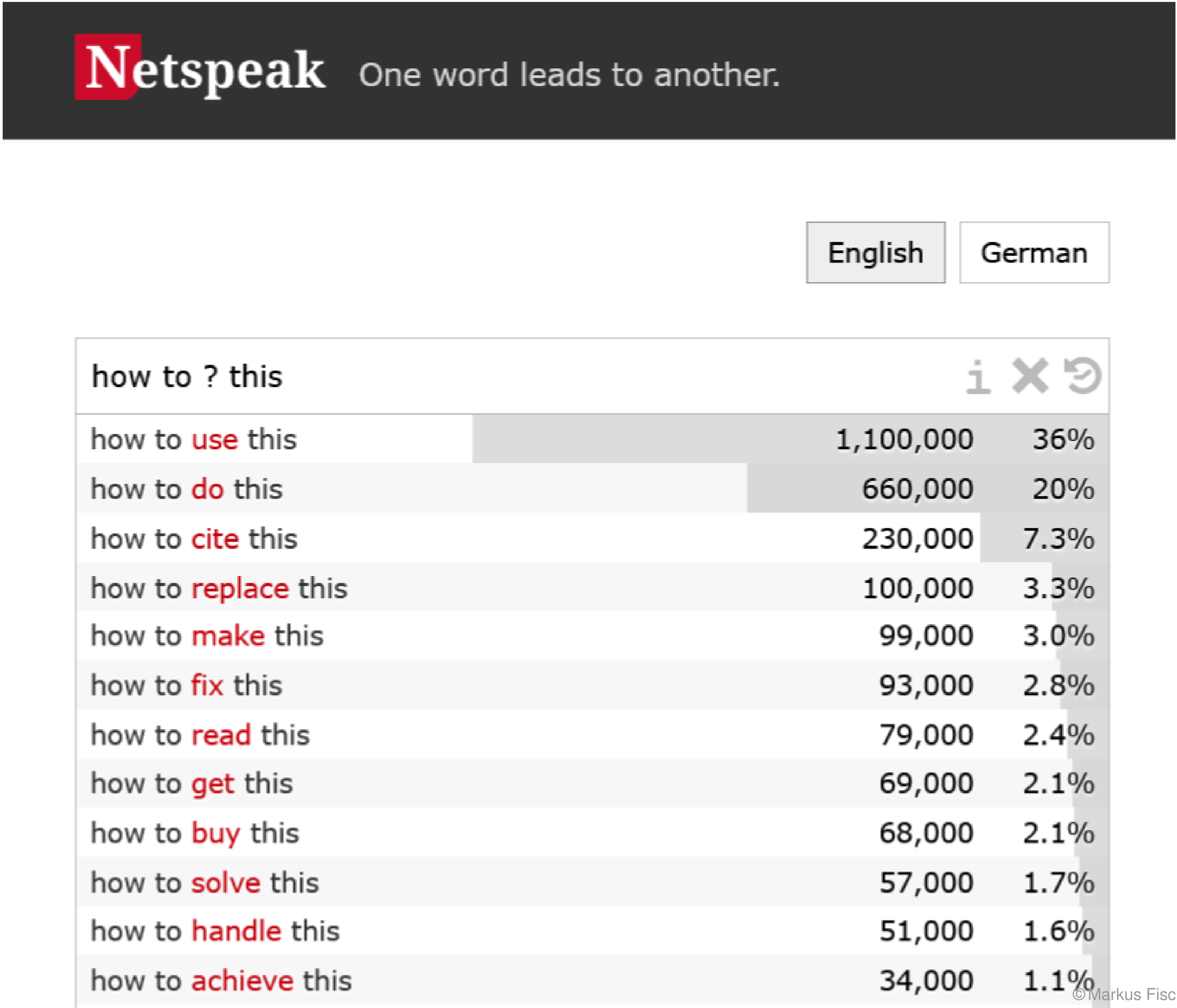
UNIVERSITÄT LEIPZIG

MARTIN-LUTHER-UNIVERSITÄT HALLE-WITTENBERG

[webis.de]

Introduction

Netspeak



how to ? this

how to use this	1,100,000	36%
how to do this	660,000	20%
how to cite this	230,000	7.3%
how to replace this	100,000	3.3%
how to make this	99,000	3.0%
how to fix this	93,000	2.8%
how to read this	79,000	2.4%
how to get this	69,000	2.1%
how to buy this	68,000	2.1%
how to solve this	57,000	1.7%
how to handle this	51,000	1.6%
how to achieve this	34,000	1.1%

Introduction

Instant Search Log

Time	Search box content
09:00:00	`search`
09:00:01	`searching f`
09:00:02	`searching for *`
09:05:10	`looking for results`
09:05:11	`looking`
09:05:41	`seraching`
09:05:45	`seraching for results`
09:05:47	`seching for results`
09:05:48	`seaching for results`
09:05:49	`searching for results`
09:06:20	`look`
09:06:21	`looking fo`
09:06:22	`looking for results`
09:06:30	`for results`
09:06:32	`sea for results`
09:06:35	`searching for results`
09:07:00	`* for results`

Introduction

Instant Search Log with Queries

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for *
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	* for results

Our Approach

Step 1: Time gap (split/defer)

Time	Search box content
09:00:00	`search`
09:00:01	`searching f`
09:00:02	`searching for ⋆`
09:05:10	`looking for results`
09:05:11	`looking`
09:05:41	`seraching`
09:05:45	`seraching for results`
09:05:47	`seching for results`
09:05:48	`seaching for results`
09:05:49	`searching for results`
09:06:20	`look`
09:06:21	`looking fo`
09:06:22	`looking for results`
09:06:30	`for results`
09:06:32	`sea for results`
09:06:35	`searching for results`
09:07:00	`⋆ for results`

Our Approach

Step 2: Containment (merge/defer)

Time	Search box content
09:00:00	`search`
09:00:01	`searching f`
09:00:02	`searching for *`
09:05:10	`looking for results`
09:05:11	`looking`
09:05:41	`seraching`
09:05:45	`seraching for results`
09:05:47	`seching for results`
09:05:48	`seaching for results`
09:05:49	`searching for results`
09:06:20	`look`
09:06:21	`looking fo`
09:06:22	`looking for results`
09:06:30	`for results`
09:06:32	`sea for results`
09:06:35	`searching for results`
09:07:00	`* for results`

Our Approach

Step 2: Containment (merge/defer)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for ⋆
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	⋆ for results

Our Approach

Step 2: Containment (merge/defer)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for *
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	* for results

Our Approach

Step 3: Lexical similarity (merge/defer)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for ⋆
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	⋆ for results

Our Approach

Step 3: Lexical similarity (merge/defer)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for ⋆
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	⋆ for results

Our Approach

Step 4: Lexical dissimilarity (split/defer)

Time	Search box content
09:00:00	`search`
09:00:01	`searching f`
09:00:02	`searching for *`
09:05:10	`looking for results`
09:05:11	`looking`
09:05:41	`seraching`
09:05:45	`seraching for results`
09:05:47	`seching for results`
09:05:48	`seaching for results`
09:05:49	`searching for results`
09:06:20	`look`
09:06:21	`looking fo`
09:06:22	`looking for results`
09:06:30	`for results`
09:06:32	`sea for results`
09:06:35	`searching for results`
09:07:00	`* for results`

Our Approach

Step 4: Lexical dissimilarity (split/defer)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for ⋆
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	⋆ for results

Our Approach

Step 5: Logistic regression (split/merge)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for *
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	* for results

Our Approach

Step 5: Logistic regression (split/merge)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for *
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	* for results

Our Approach

Step 5: Logistic regression (split/merge)

Time	Search box content
09:00:00	search
09:00:01	searching f
09:00:02	searching for *
09:05:10	looking for results
09:05:11	looking
09:05:41	seraching
09:05:45	seraching for results
09:05:47	seching for results
09:05:48	seaching for results
09:05:49	searching for results
09:06:20	look
09:06:21	looking fo
09:06:22	looking for results
09:06:30	for results
09:06:32	sea for results
09:06:35	searching for results
09:07:00	* for results

Evaluation

Our approach

	Step	Decided pairs	F_2	Run time
1	Time gap	9.1%	0.68	0.002 ms
2	Containment	25.0%	0.51	0.002 ms
3	Lexical similarity	63.7%	0.70	0.011 ms
4	Lexical dissimilarity	64.7%	0.75	(with Step 3)
5	Logistic regression	100.0%	0.93	0.811 ms

- Around 65% of all entries decided in very short time.
- Remaining 35% decided by slow Logistic regression.
- Throughput: 3500 entries per second.

  (2300 with rules and 1200 with Logistic regression)
- Nearly no errors after steps 1-4.

Evaluation

Comparison to previous methods

Approach	F_2	**Run time**
Our approach	**0.93**	0.82 ms
Kim and Li (2015)	0.88	0.06 ms
Hagen et al. (2013)	0.83	**0.01 ms**
Cetindil et al. (2012)	0.77	0.06 ms

❑ Kim and Li: Used time difference and normalized edit distance.

❑ Hagen et al.: Cascading approach for query log segmentation.
(Semantic steps were omitted for our task.)

❑ Cetindil et al.: Used normalized edit distance only.

Take-Home Message

Results

- ❑ Combined near-perfect-precision steps with downstream logistic regression.
- ❑ Achieving high accuracy with reasonable run time.
- ❑ Analysis on query level revealed: users revisit previous queries in short time frame.

Future Work

- ❑ Show previous queries as part of the user interface.

  Since about 25% of active users show the see-saw pattern.
- ❑ Investigate which log entry in a query actually gained attention.

Take-Home Message

Results

- Combined near-perfect-precision steps with downstream logistic regression.
- Achieving high accuracy with reasonable run time.
- Analysis on query level revealed: users revisit previous queries in short time frame.

Future Work

- Show previous queries as part of the user interface.

  Since about 25% of active users show the see-saw pattern.
- Investigate which log entry in a query actually gained attention.

Thank you for your attention!