Exponential Reservoir Sampling for Streaming Language Models

Miles Osborne (Edinburgh)
Ashwin Lall (Denison)
Benjamin Van Durme (JHU)

Building a Language Model

- Collect many examples of language use
  - Transcribed spoken utterances
  - Written sentences

- Build a table, mapping phrases to how many times they appear in the collection

Count(*the dog ran*) = 52,
Count(*the dog*) = 109,
Count(*the*) = 1,370
…

Problem!

There are a lot of unique phrase combinations

- Requires a very big table to store counts
- LMs require *pruning* in order to fit into memory

Language changes over time

- New words and phrases introduced, exacerbating the problem

Solution?

There are a lot of unique phrase combinations

- Requires a very big table to store counts
- LMs require *pruning* in order to fit into memory
- **Sub-sample data**

Language changes over time

- New words and phrases introduced, exacerbating the problem
- **Update sample over time**

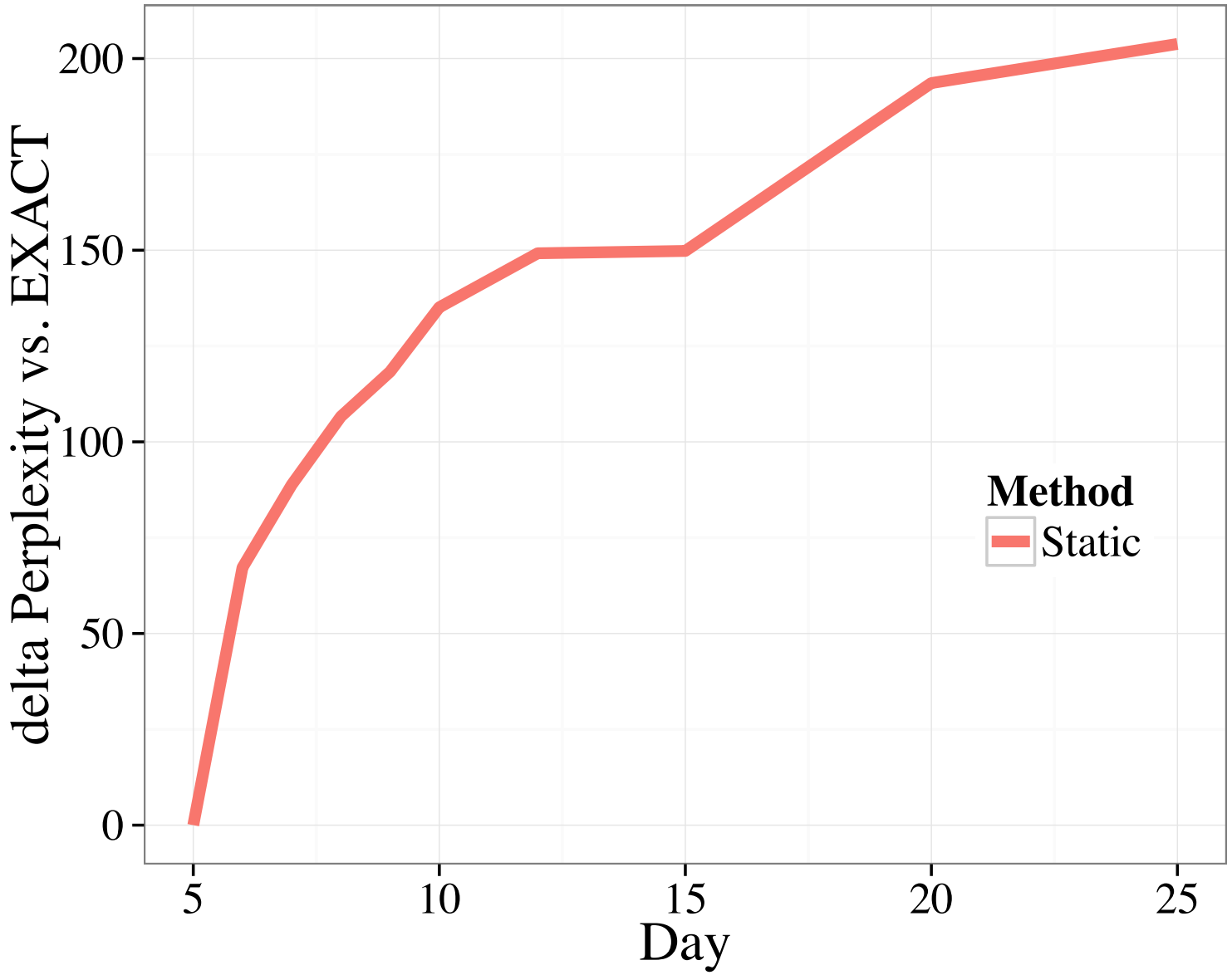
Looking ahead to the experiment

- Move through collection of ***tweets*** (or newspaper articles), sorted by time
- Sub-sample 5 days of content, with various strategies
- Build LM from sample, compute **perplexity** on the next day's material

Stream	Interval	Total (toks)	Test (toks)
Twitter	Dec 2013	3282M	105M

What if we just use the first 5 days?

- Build a **Static** language model on the first 5 days of tweets
- Measure perplexity on subsequent days
- Measure difference in perplexity against a model trained on all data (**Exact**)


delta Perplexity vs. EXACT
200
150
100
50
0
5
10
15
20
25
Day
Method
Static

Sampling

- Build an LM based on a corpus formed from a sample from the stream
- As the stream updates, update the sample, retrain the LM
- But how to sample efficiently?
  - Content keeps coming in, we don't want to store all of it, forever

Uniformly sample over stream

=

Reservoir Sampling

Reservoir Sampling (Vitter ‘85)

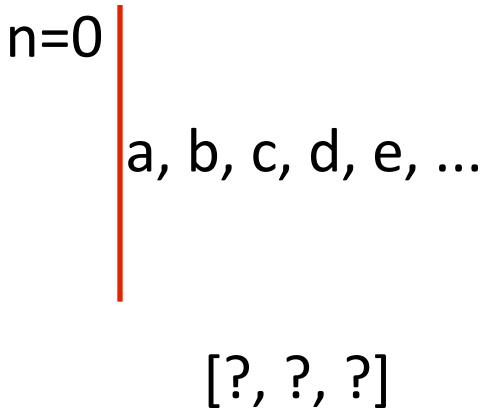
a, b, c, d, e, ...

Reservoir Sampling

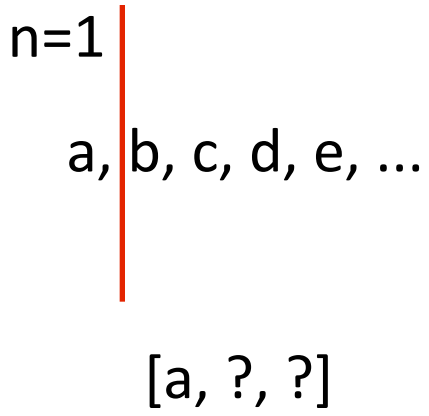
a, b, c, d, e, ...

reservoir of size k=3 → [?, ?, ?]

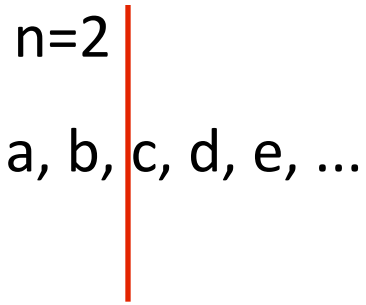
Reservoir Sampling



Reservoir Sampling

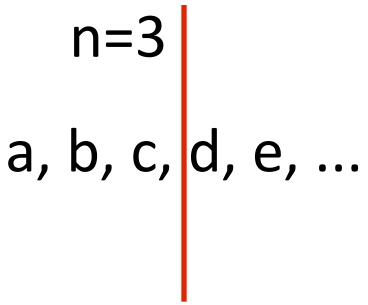


Reservoir Sampling



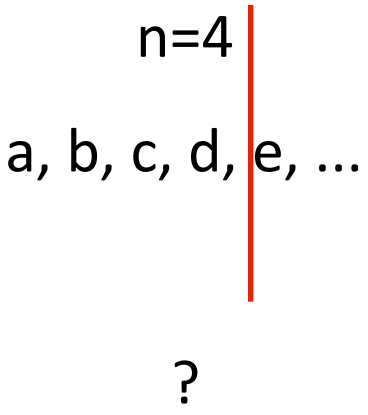
[a, b, ?]

Reservoir Sampling

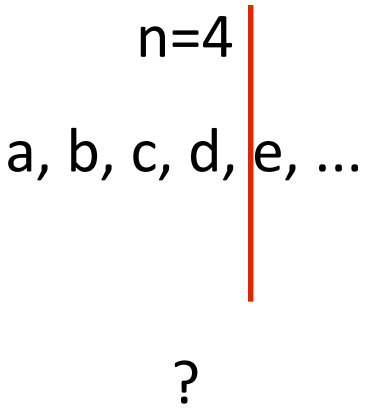


[a, b, c]

Reservoir Sampling

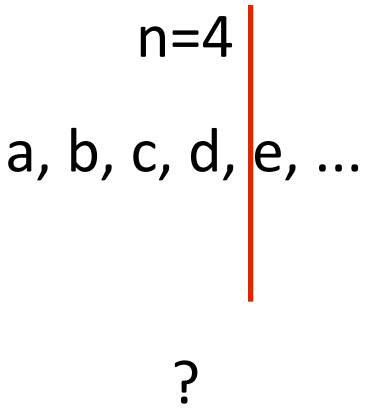


Reservoir Sampling



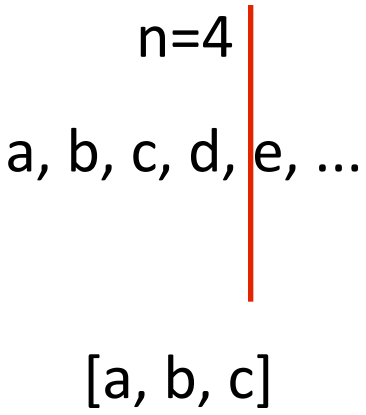
1. accept d with probability: k/n = 3/4

Reservoir Sampling



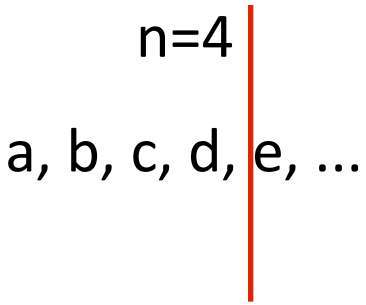
1. accept d with probability: k/n = 3/4
2. if accept, then evict a random element from reservoir

Reservoir Sampling



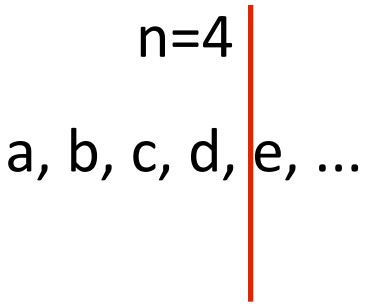
1. accept d with probability: k/n = 3/4
2. if accept, then evict a random element from reservoir

Reservoir Sampling



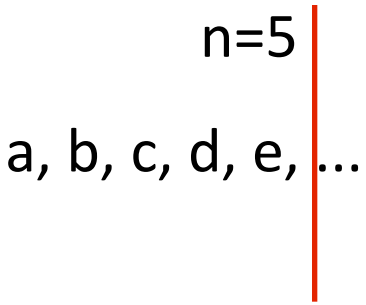
1. accept d with probability: k/n = 3/4
2. if accept, then evict a random element from reservoir

Reservoir Sampling



1. accept d with probability: k/n = 3/4
2. if accept, then evict a random element from reservoir

Reservoir Sampling



[a, d, c]

1. accept e with probability: k/n = 3/5

Seen elsewhere

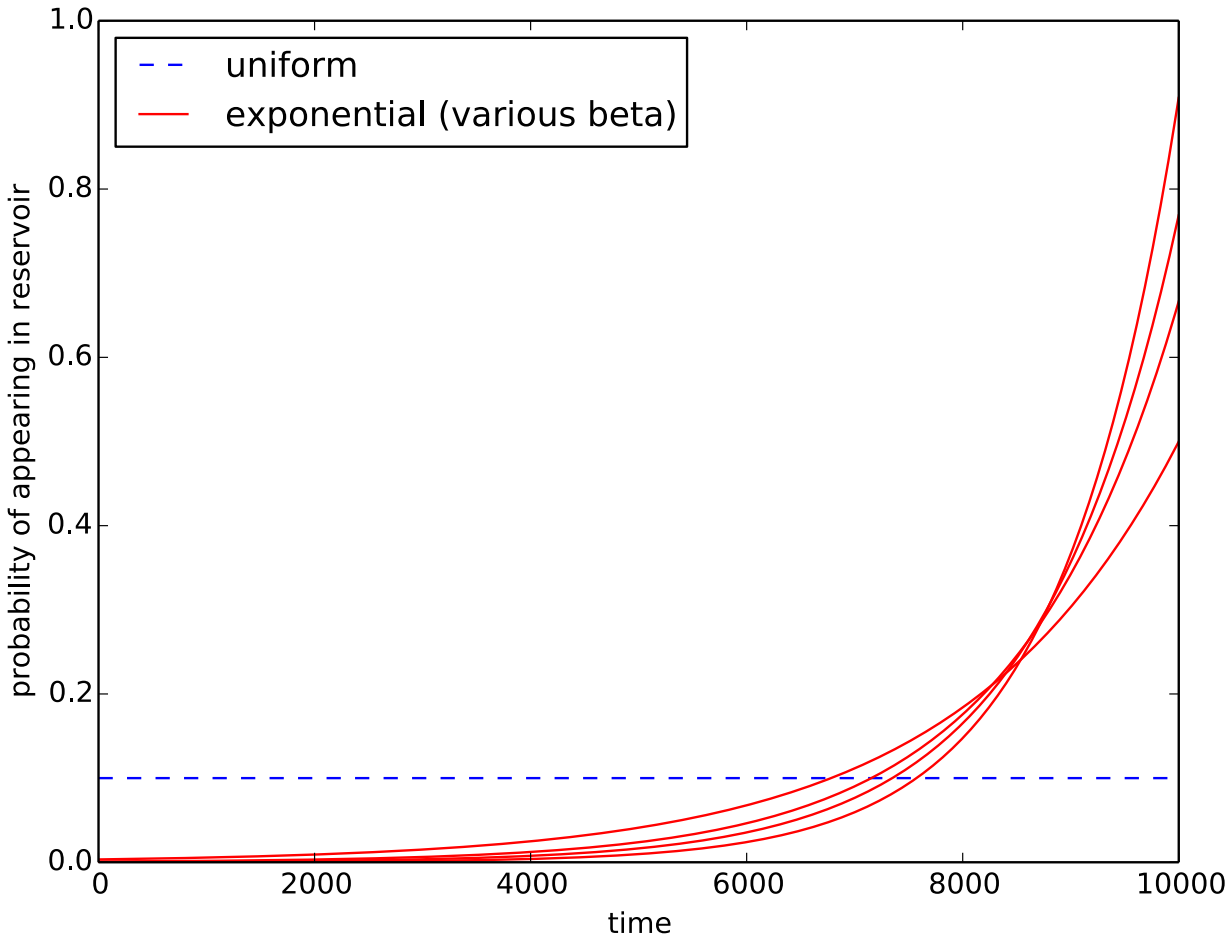
Particle Filter Rejuvenation and Latent Dirichlet Allocation
Chandler May, Alex Clemmer and Benjamin Van Durme
ACL. 2014.

Streaming Analysis of Discourse Participants
Benjamin Van Durme
EMNLP. 2012.

Efficient Online Locality Sensitive Hashing via Reservoir Counting
Benjamin Van Durme and Ashwin Lall
ACL. 2011.

Exponential Reservoir Sampling

Exponential Reservoir Sampling



Acceptance Probabilities

Uniform Reservoir:

$$f(n,k) = \frac{k}{n}$$

Approximation to Exponential Reservoir (Aggarwal '06) :

$$f(n,k) = \frac{k}{\beta}$$

Exact Exponential Reservoir (this work) :

$$f(n,k) = ck(1 - e^{-1/\beta})$$

Experiment

- Move through collection of ***tweets***, sorted by time (could be news articles, spoken utterances, ...)
- Sub-sample 5 days of content
- Build LM from sample, compute **perplexity** on the next day's material

Stream	Interval	Total (toks)	Test (toks)
Twitter	Dec 2013	3282M	105M

Sampling Strategies

Sample window of 5 days worth of content:

- Last 5 days seen (Window)
- Uniform sample (Uniform)
- Exponentially biased sample (Exp)

Sampling Strategies

Sample window of 5 days worth of content:

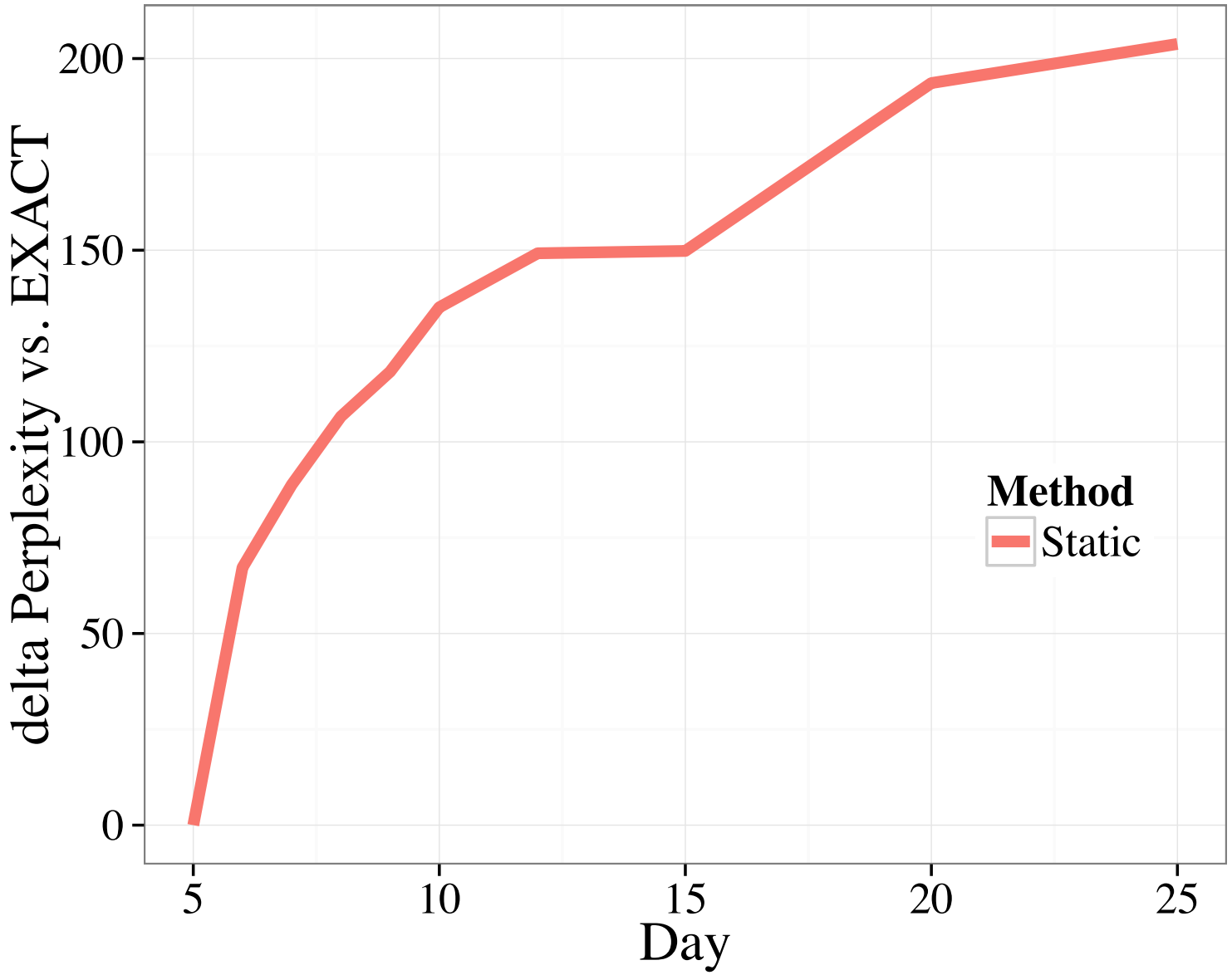
- Last 5 days seen (Window)
- Uniform sample (Uniform)
- Exponentially biased sample (Exp)

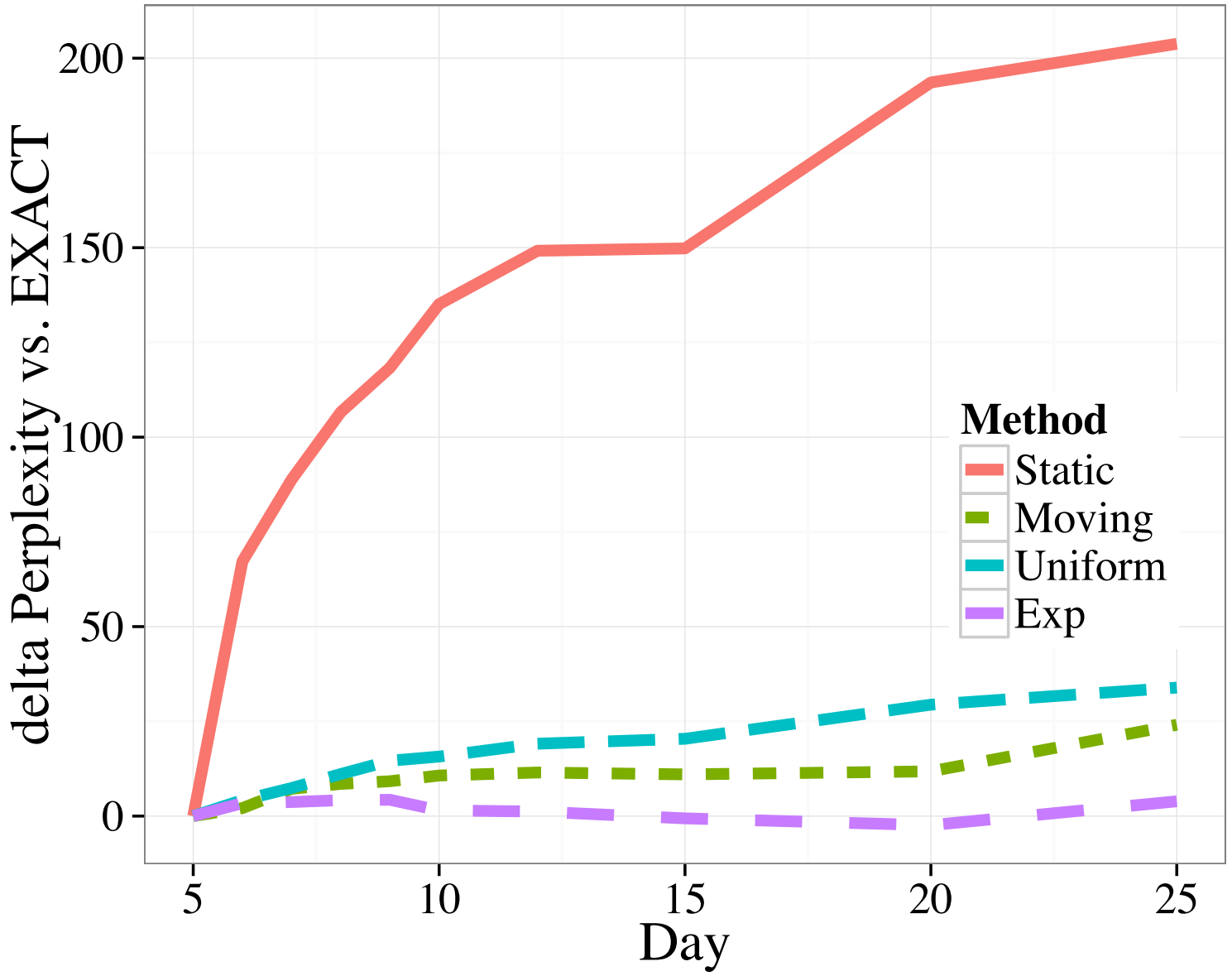
Contrast against:

- the first 5 days (Static)
- all data seen thus far (Exact)

Day	Static	Moving	Uniform	Exp	Exact
5	619.4	619.4	619.4	619.4	619.4
6	664.8	**599.7**	601.8	601.0	597.6
7	684.4	602.8	603.0	**599.3**	595.6
8	710.1	612.0	614.6	**607.7**	603.5
9	727.0	617.9	623.3	**613.0**	608.7
10	775.6	651.2	656.2	**642.0**	640.5
12	776.7	639.0	646.6	**628.7**	627.5
15	777.1	638.3	647.7	**626.7**	627.3
20	800.9	619.1	636.7	**604.9**	607.3
25	801.4	621.7	631.5	**601.5**	597.6

Table 4: Perplexities for differently selected samples over Twitter (sample size = five days, $\beta = 1.1$). Results in **bold** are the best sampling results. Lower is better.


delta Perplexity vs. EXACT
200
150
100
50
0
5
10
15
20
25
Day
Method
Static


delta Perplexity vs. EXACT
200
150
100
50
0
5
10
15
20
25
Day
Method
Static
Moving
Uniform
Exp

Conclusion

Exponential > Sliding window > Uniform

It is better to prefer recent material over a uniform sample,

but better is to smoothly decay: prefer recent, while preserving some longer term history.