Building Ranking Infrastructure: Data-Driven, Lean, Flexible

Sergii Khomenko, Data Scientist, STYLIGHT
sergii.khomenko@stylight.com, @lc0d3r

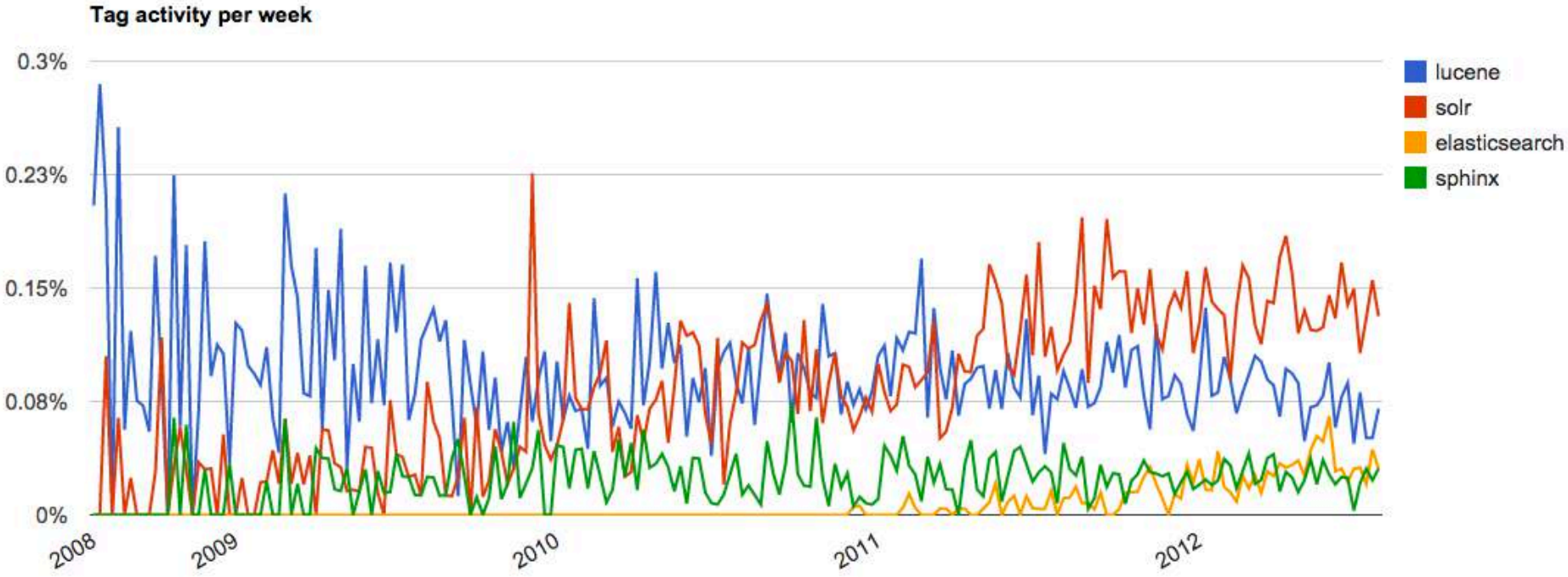
APACHECON EUROPE

CORINTHIA HOTEL
BUDAPEST, HUNGARY
NOVEMBER 17-21, 2014

Agenda

1. Problem definition
2. Boosting
3. Lean approach to ranking infrastructure
4. Real-word examples

LUCENE, SOLR, ELASTICSEARCH



SOLR USERS


Instagram
Fast beautiful photo sharing


THE WHITE HOUSE
WASHINGTON


Zappos
.com


digg


STYLIGHT

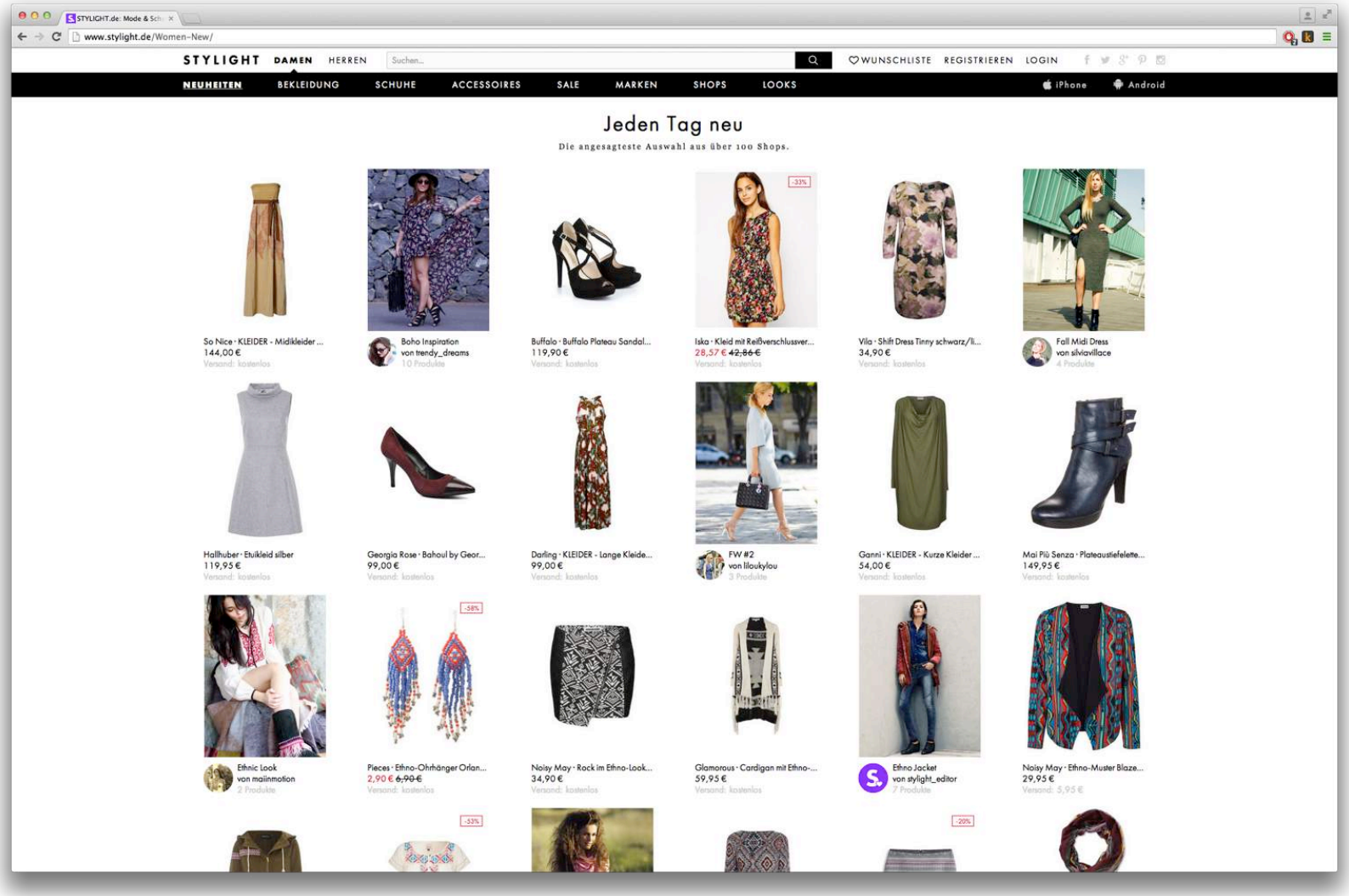

theguardian


NASA

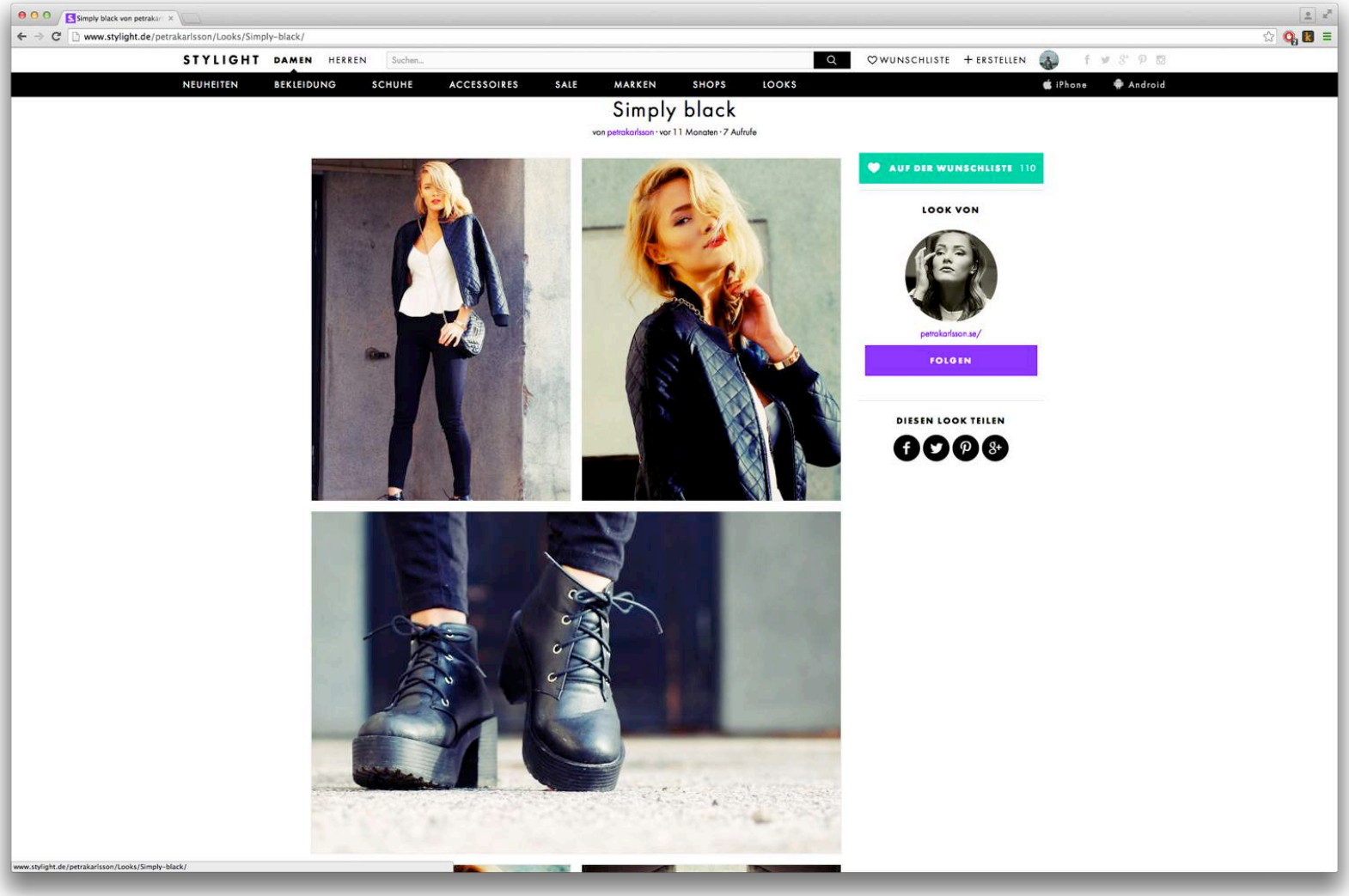

NETFLIX


Walt Disney
PICTURES

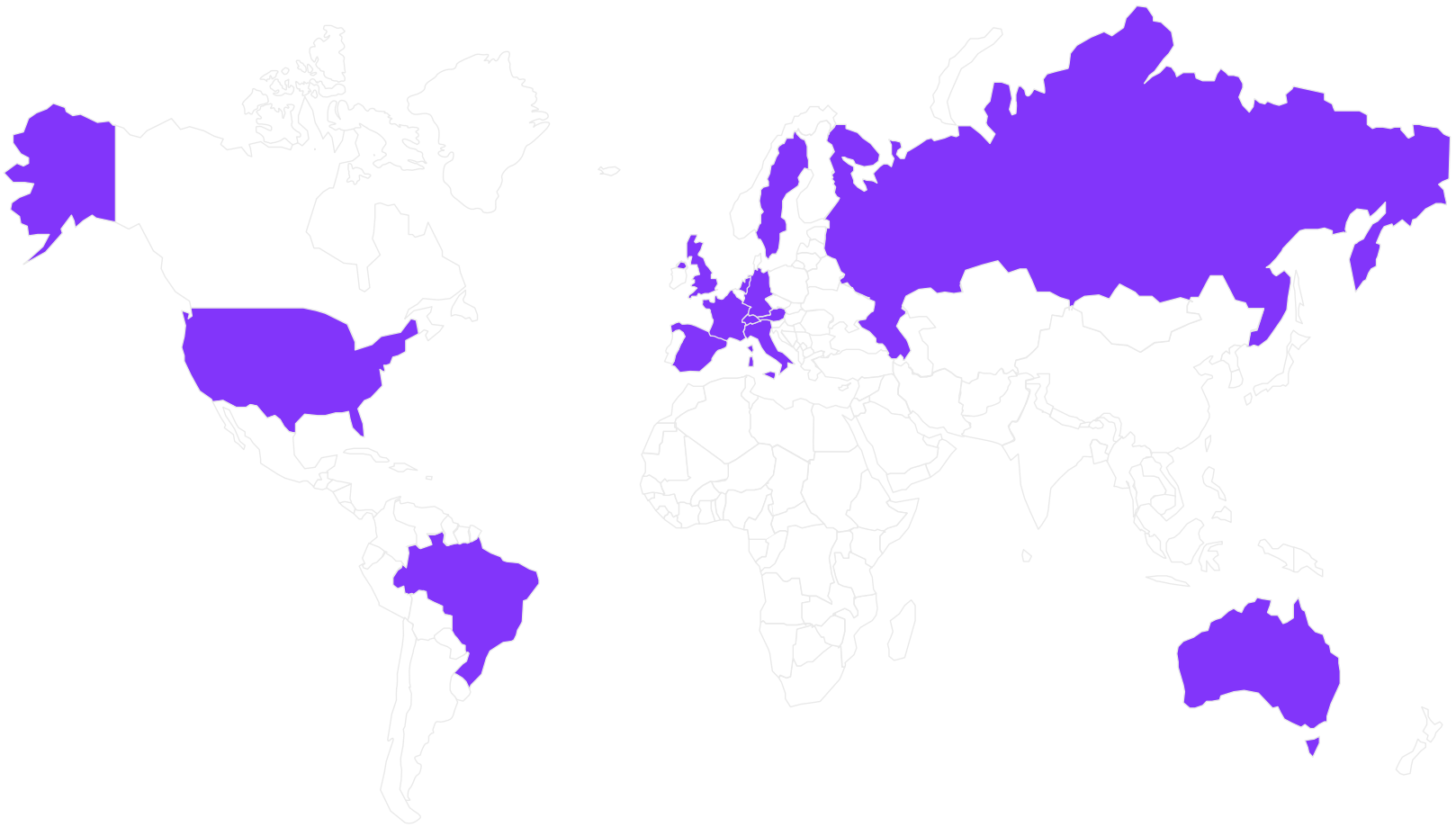
THE BEST PLACE TO DICOVER FASHION

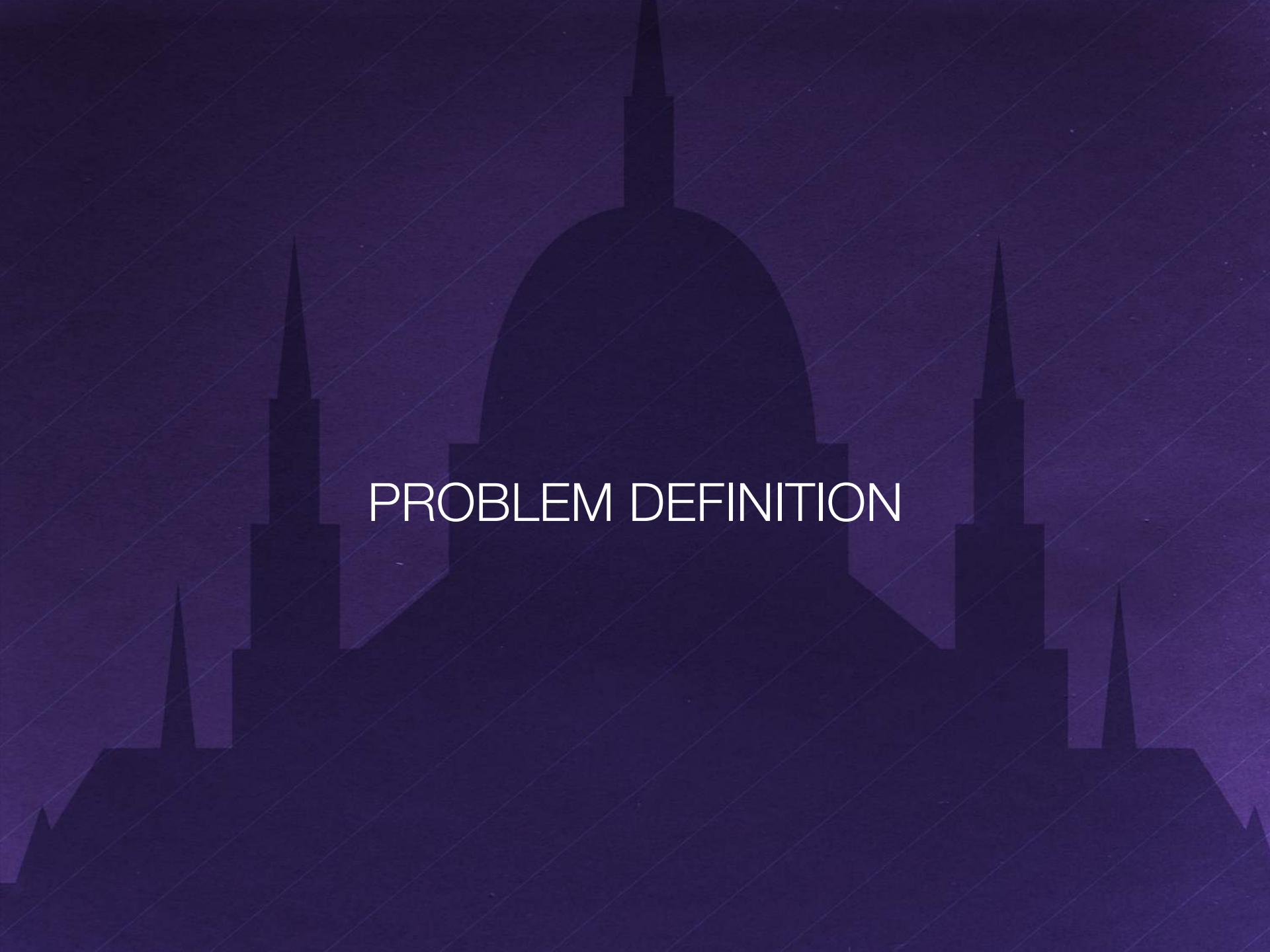


GET INSPIRED BY LOOKS CREATED BY COMMIUNITY

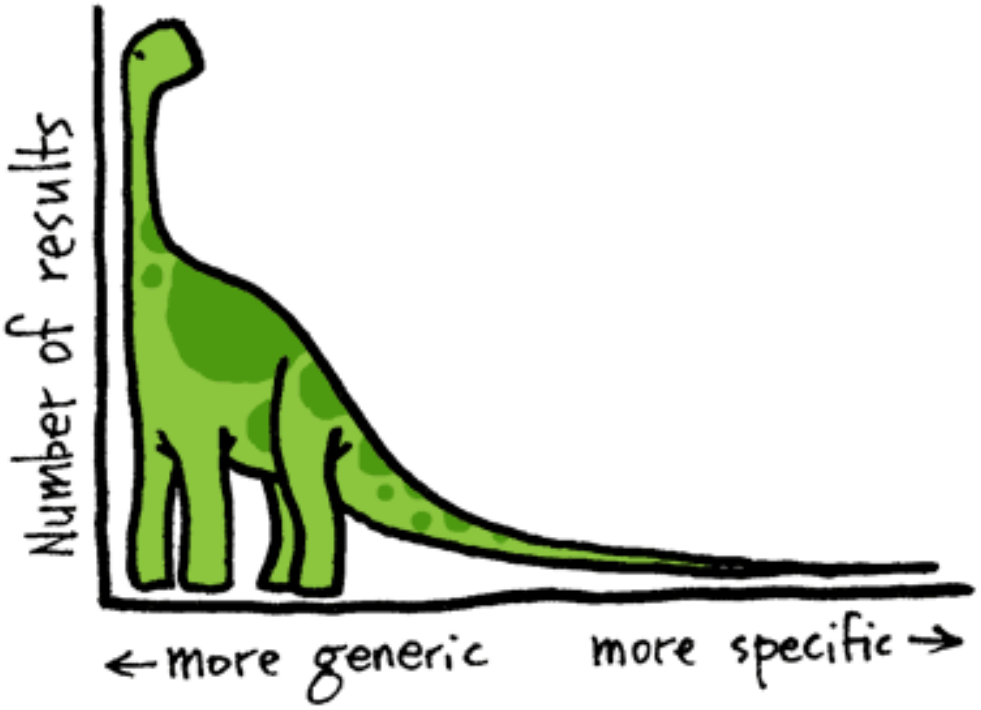


STYLIGHT – INTERNATIONAL COMMUNITY




PROBLEM DEFINITION

Problem definition



Problem definition

Ranking specifics:

- Seasonal influence
- Trends
- Cold start of new countries, shops
- Multiple dimensions of ranking model


IMPROVING RELEVANCE

TF-IDF - default scoring model in Lucene/Solr

$$coord(q,d) \cdot \sum_{t \in q} \left(tf(t,d) \cdot idf(t)^2 \cdot norm(field(t),d) \right)$$

- matching more query terms is better
- more occurrences of a query term is better
- more novel terms increase doc score more than common terms

Improving relevance

Stages to improve relevance in Solr

- Editorial voting (QueryEvaluationComponent)
- Indexing time

  (analyzing content, text analysis)
- Query-time

  (function queries, boosting)

Improving relevance - Solr queries

q = +brand:adidas shop:monshowroom^3

q = +adidas monshowroom
defType = dismax
qf = brand shop^3
sort = user_ratings desc, score desc

qq = adidas
q = {!boost b=$b defType=dismax v=$qq}
b = prod(popularity, clicks)

Definition of solr.ExternalFileField

```
<types>

   <fieldType name="float" class="solr.FloatField"
omitNorms="true"/>

   <fieldType name="file_delta2"
class="solr.ExternalFileField" keyField="id" defVal="1.0"
indexed="false" stored="false" valType="float" />


</types>

<fields>
 <field name="delta2" type="file_delta2"/>
</fields>
```

Example of external file with boosting

```
\cores\de_DE\products\external_delta2.txt


15062471=0.5
15062479=0.2
15062507=0.41
```


LEAN APPROACH TO RANKING

Lean manufacturing, lean enterprise, or ***lean production***, often simply, "***lean***", is a production practice that considers the expenditure of resources for any goal other than the creation of value for the end customer to be wasteful, and thus a target for elimination.

Essentially, lean is centered on ***preserving value with less work***.

Lean approach to Ranking

Requirements:

- Decreasing time to implement new ranking model
- Possibility to use more dynamic ranking models
- Keeping working infrastructure alive

Lean approach to Ranking

Requirements:

- A/B testing without changing entire infrastructure
- Performance level - “still fast” and “transparent”

Python benchmark, consistency checker

--gaid gaid, -g gaid Google analytics site id.

--gadate gadate a date to fetch the most popular pages from Google Analytics

–-solr solr, -s solr Solr server to benchmark performance.

--pages number, -p number a number of top pages from Google Analytics.

--repeats number, -r number a number of repeats for an every page.

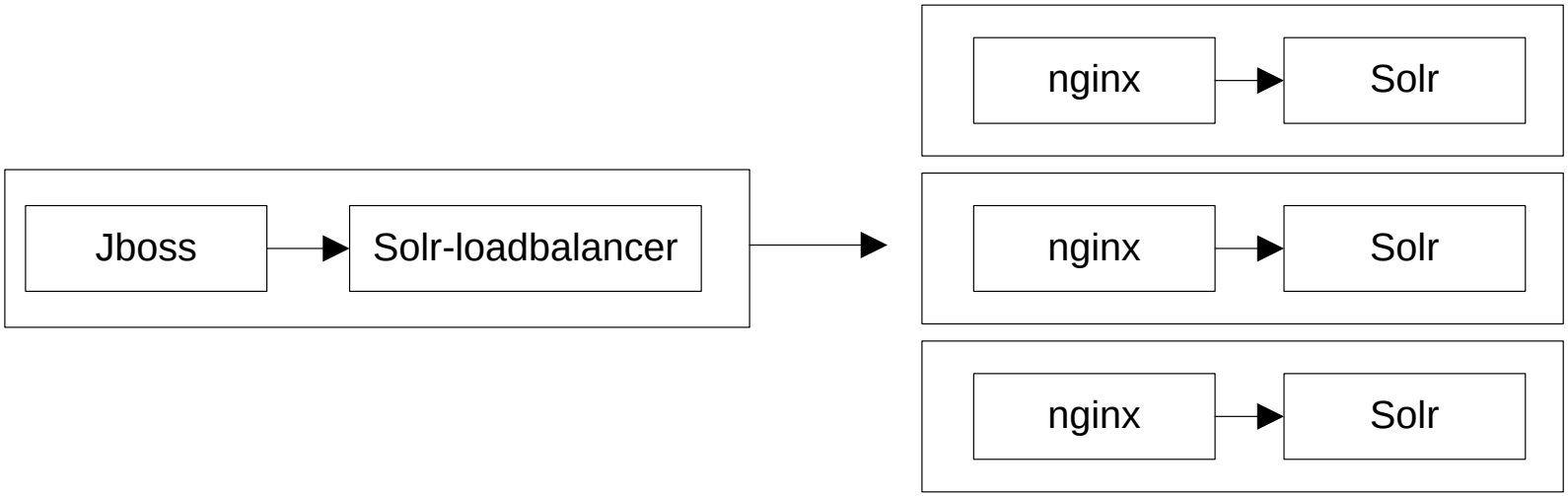
--compare compare, -c compare Different rankings algorithms to compare.

Python benchmark, consistency checker

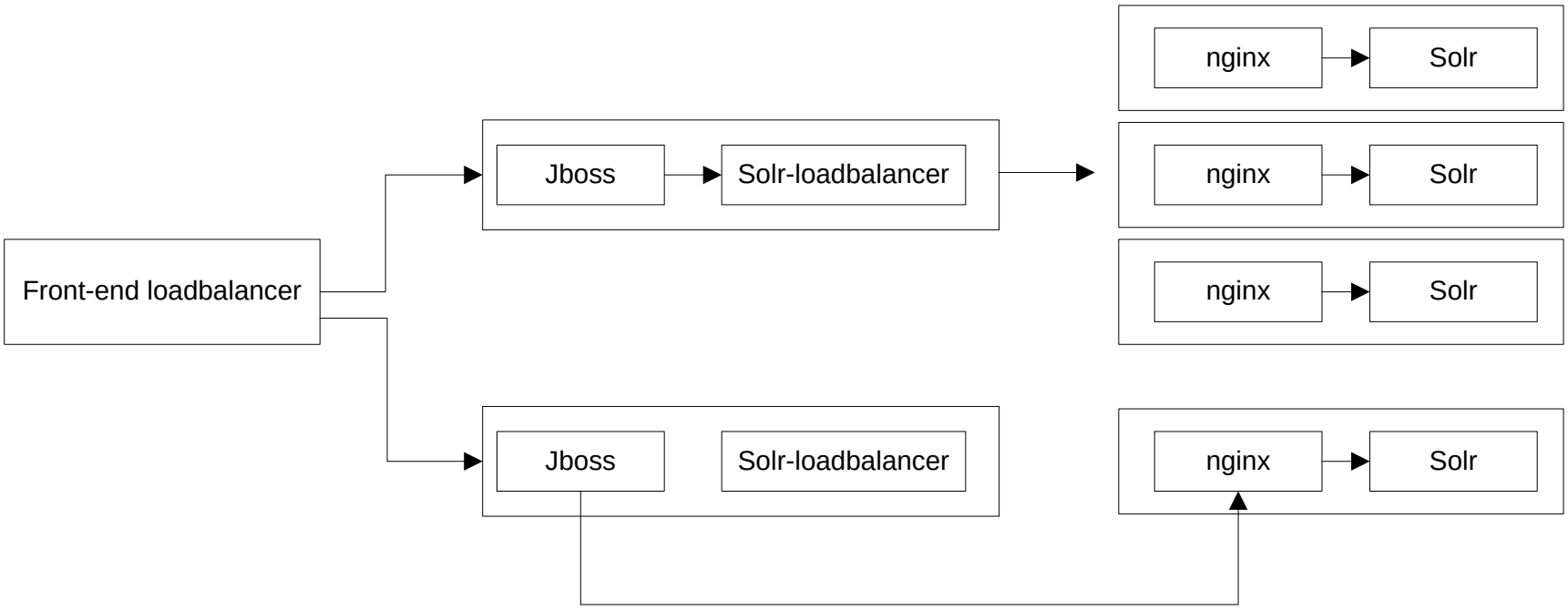
```
python solr-benchmark\benchmark.py -c
RankingClassical,RankingDelta2 --cmpmode
1
```

```
python solr-benchmark\benchmark.py -c
RankingClassical,RankingDelta2
```

Common search infrastructure



Updated infrastructure



manifests/nodes/solr0x-node.pp

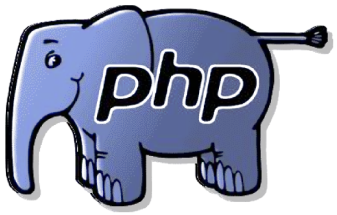
```
include nginx
nginx::config { "solr_dev": }
nginx::solr-ranking { "delta2":
  urls => [
      "/search.action?
gender=women&brand=2271&tag=1161&tag=877&tag
=468",
      "/search.action?
gender=men&brand=11235&tag=10203&tag=10299&ta
g=10326"
  ],
```

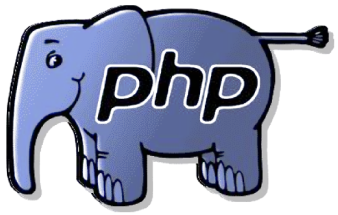
nginx / templates / conf / solr-rewrites.conf.erb

```
<% urls.each do |url| -%>
if ($args ~* <% if url['gender'] > 0 -%>gender_id%3A<%=
url['gender'] %>.*<% end -%><% url['tags'].each do |tag| -
%>tag_id%3A<%= tag %>.*<% end -%><% if url['brand'] > 0 -
%>brand_id%3A%28<%= url['brand'] %>%29<% end -%>) {
    set $orig $args;
    set $args "q={!boost+b=%24b+defType=dismax+v=%24qq}
&qq=id:*";
    rewrite ^(.*)$ "$1?$orig" break;
}
<% end -%>
```


REAL-WORLD EXAMPLES

Elephant-Driven architecture





Simplified Version

Simplified Version

kafka

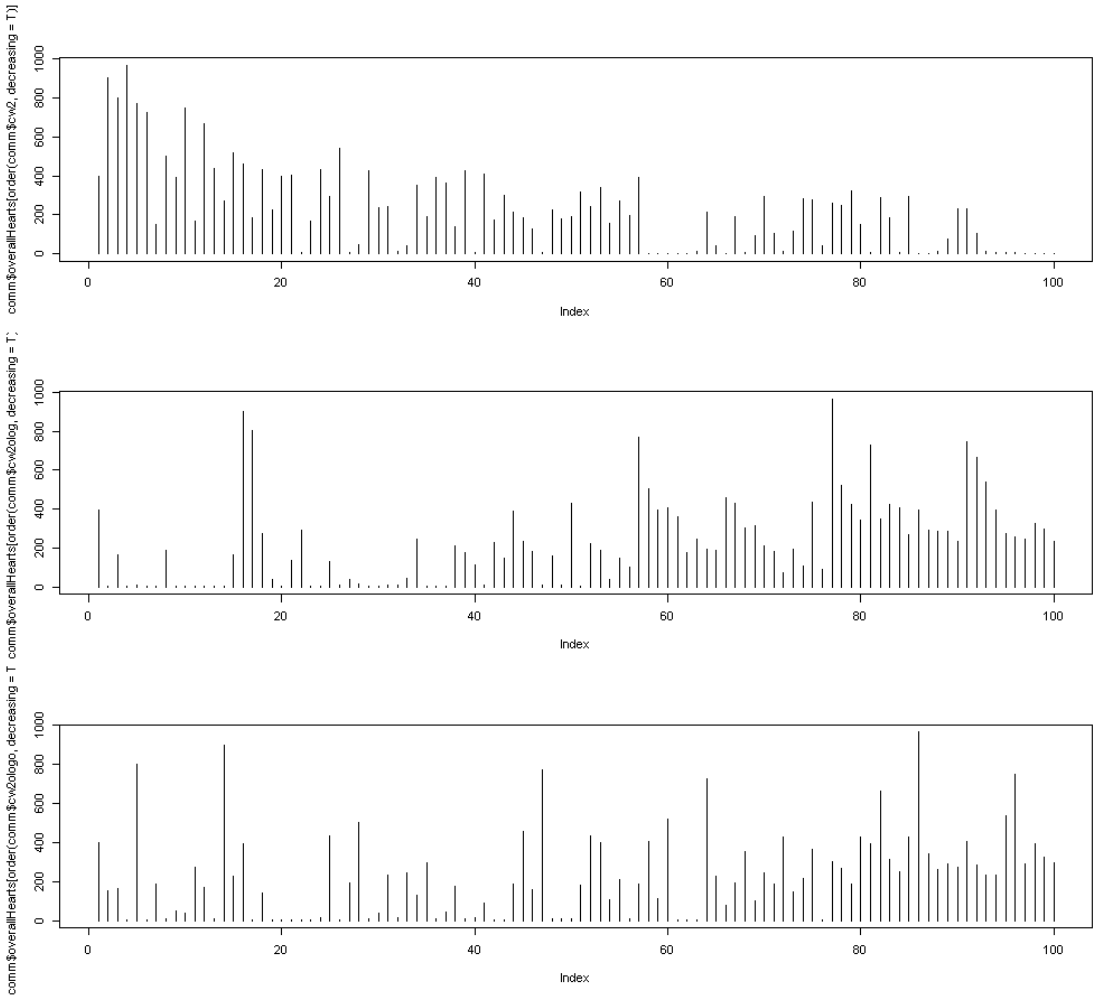
APACHE DRILL

samza

Apache Solr

Kinesis

Real-world examples

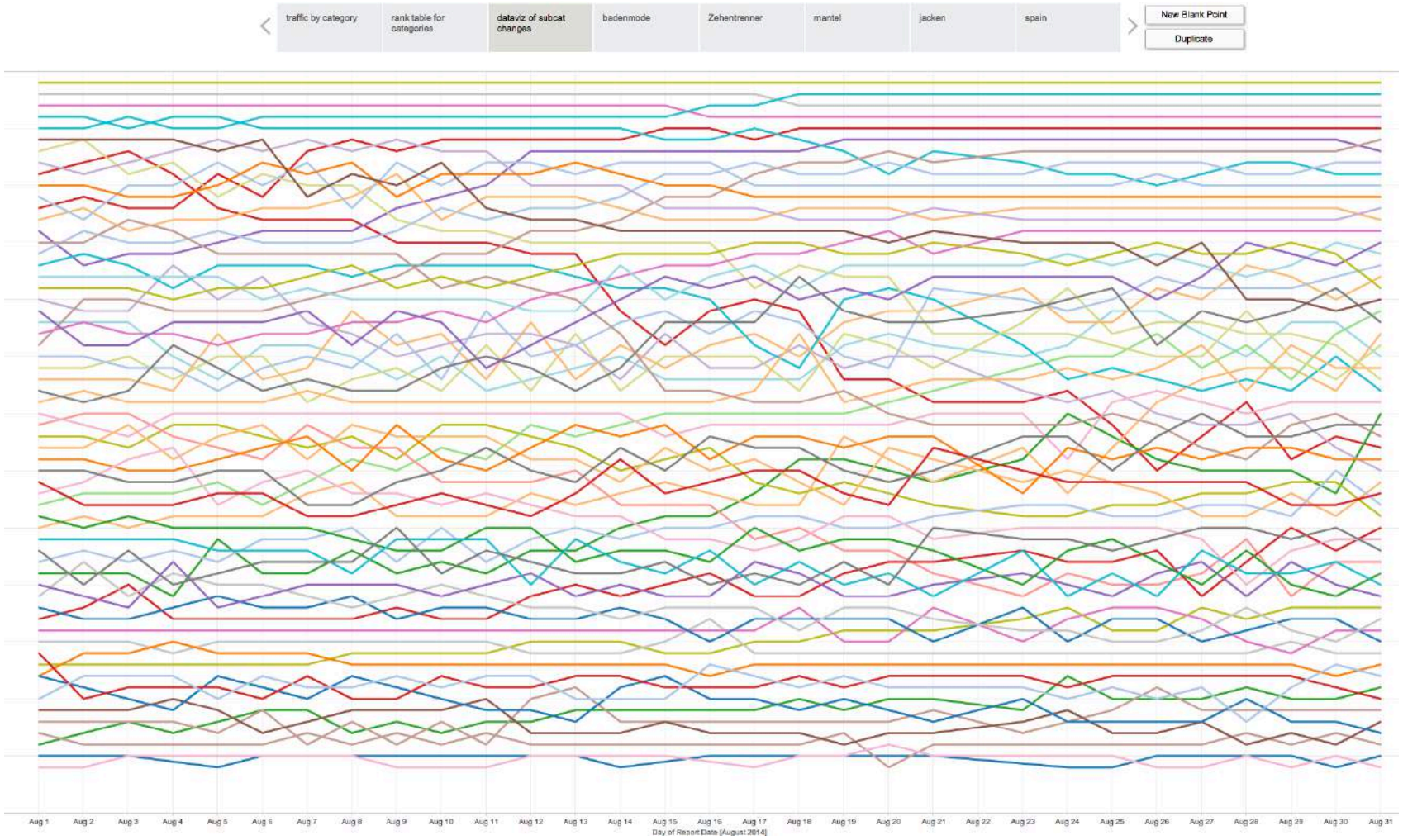


Multiple points to evaluate

Stages to evaluate the model:

- R ranking model
- Independent Solr-node
  - For internal use-cases
  - Testing for some of pages
  - A/B roll out for % of users
- Production roll out

Real-world examples

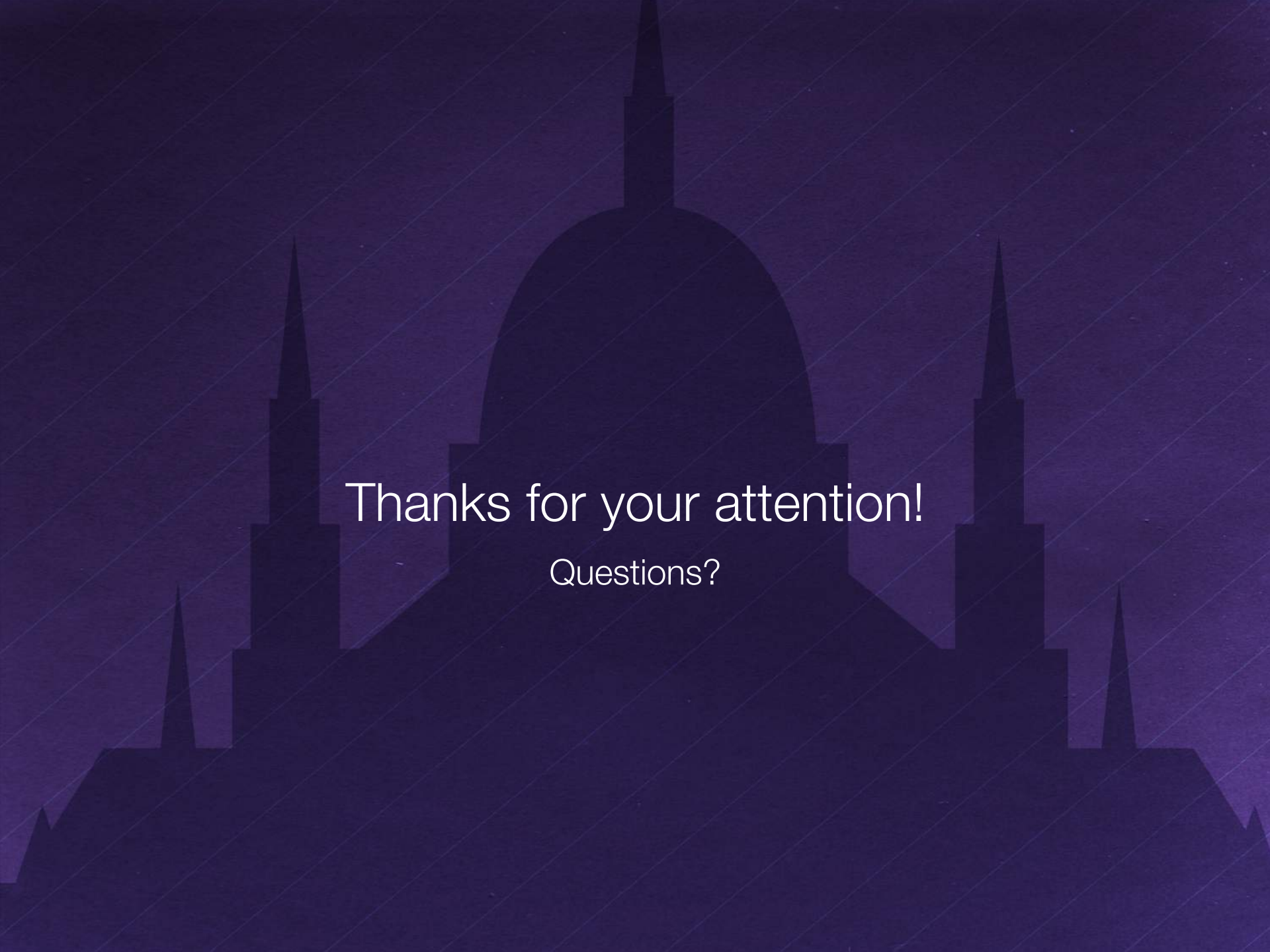


Real-world examples

Multiple points to evaluate

Stages to evaluate the model:

- R ranking model
- Independent Solr-node
  - For internal use-cases
  - Testing for some of pages
  - A/B roll out for % of users
- Production roll out


Thanks for your attention!
Questions?

Sergii Khomenko
Data Scientist
STYLIGHT GmbH
sergii.khomenko@stylight.com
@lc0d3r

Nymphenburger Straße 86
80636 Munich, Germany

STYLIGHT.COM

REFERENCE LIST

- Stack Overflow Tag Trends http://hewgill.com/~greg/stackoverflow/stack_overflow/tags/#!lucene+solr+elasticsearch+sphinx
- Public websites using Solr http://wiki.apache.org/solr/PublicServers
- CommonQueryParameters http://wiki.apache.org/solr/CommonQueryParameters
- Thoughts in plain text http://lc0.github.io/
- STYLIGHT Engineering http://www.stylight.com/Engineering/