Machine Learning
In Search Quality
At Yandex
Russian Search Market

Source: LiveInternet.ru, 2005-2009
A Yandex Overview

1997
Yandex.ru was launched

No 7
Search engine in the world * (# of queries)

150 mln
Search Queries a Day

Offices
> Moscow
> 4 Offices in Russia
> 3 Offices in Ukraine
> Palo Alto (CA, USA)

* Source: Comscore 2009
15 countries with cyrillic alphabet
77 regions in Russia

Variety of Markets

> Different culture, standard of living, average income
  for example, Moscow, Magadan, Saratov

> Large semi-autonomous ethnic groups
  Tatar, Chechen, Bashkir

> Neighboring bilingual markets
  Ukraine, Kazakhstan, Belarus
Geo-specific queries

Relevant result sets vary across all regions and countries

[wedding cake]
[gas prices]
[mobile phone repair]
[пицца]  Guess what it is?
pFound
A Probabilistic Measure of User Satisfaction
Probability of User Satisfaction

Optimization goal at Yandex since 2007

> $p_{\text{Found}}$ – Probability of an answer to be FOUND

> $p_{\text{Break}}$ – Probability of abandonment at each position (BREAK)

> $p_{\text{Rel}}$ – Probability of user satisfaction at a given position (RELEVANCE)

$$p_{\text{Found}} = \sum_{r=1}^{n} (1 - p_{\text{Break}})^{r-1} p_{\text{Rel}_r} \prod_{i=1}^{r-1} (1 - p_{\text{Rel}_i})$$

Similar to ERR, Chapelle, 2009, Expected Reciprocal Rank for Graded Relevance
Geo-Specific Ranking
An initial approach

query $\rightarrow$ query + user’s region

Ranking feature e.g.: “user’s region and document region coincide”
## An initial approach

### query → query + user’s region

<table>
<thead>
<tr>
<th>Problems</th>
<th>Hard to perfect single ranking</th>
<th>Cache hit degradation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&gt; Very poor local sites</td>
<td>&gt; Twice as much queries</td>
<td></td>
</tr>
<tr>
<td>in some regions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&gt; Some features</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(e.g. links) missing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&gt; Countries</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(high-level regions)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>are very specific</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Alternatives In Regionalization

<table>
<thead>
<tr>
<th>Separated local indices</th>
<th>VS</th>
<th>Unified index with geo-coded pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>One query</td>
<td>VS</td>
<td>Two queries: original and modified (e.g. +city name)</td>
</tr>
<tr>
<td>Query-based local intent detection</td>
<td>VS</td>
<td>Results-based local intent detection</td>
</tr>
<tr>
<td>Single ranking function</td>
<td>VS</td>
<td>Co-ranking and re-ranking of local results</td>
</tr>
<tr>
<td>Train one formula on a single pool</td>
<td>VS</td>
<td>Train many formulas on local pools</td>
</tr>
</tbody>
</table>

Yandex
Why use MLR?

Machine Learning as a Conveyer

> Each region requires its ranking
  Very labor-intensive to construct

> Lots of ranking features are deployed monthly
  MLR allows faster updates

> Some query classes require specific ranking
  Music, shopping, etc
MatrixNet
A Learning to Rank Method
MatrixNet

A Learning Method

> boosting based on decision trees
  We use oblivious trees (i.e. “matrices”)

> optimize for pFound

> solve regression tasks

> train classifiers

Based on Friedman’s Gradient Boosting Machine, Friedman 2000
MLR: complication of ranking formulas

MatrixNet

2006 2007 2008 2009 2010

Yandex

0.02 kb 1 kb 14 kb 220 kb 120 000 kb
MLR: complication of ranking formulas

A Sequence of More and More Complex Rankers

> pruning with the Static Rank (static features)
> use of simple dynamic features (such as BM25 etc)
> complex formula that uses all the features available
> potentially up to a million of matrices/trees for the very top documents

See also Cambazoglu, 2010, Early Exit Optimizations for Additive Machine Learned Ranking Systems
Geo-Dependent Queries: pfound

Other search engines

Yandex

Google

2009

2010
Geo-Dependent Queries

Number of Local Results (%)

Source: AnalyzeThis.ru, 2005-2009
Lessons

**MLR** is the only key to **regional search**: it provides us the possibility of tuning many geo-specific models at the same time.
Challenges

> **Complexity** of the models is increasing rapidly
  Don’t fit into memory!

> MLR in its current setting does not fit well to **time-specific queries**
  Features of the fresh content are very sparse and temporal

> **Opacity of results of the MLR**
  The back side of Machine Learning

> Number of features grows faster than the number of judgments
  Hard to train ranking

> Learning from clicks and user behavior is hard
  Tens of Gb of data per a day!
Yandex and IR

Participation and Support
Yandex MLR at IR Contests

MatrixNet at Yahoo Challenge: #1, 3, 10 (Track 2), also BagBoo, AG
Support of Russian IR

Schools and Conferences

> **RuSSIR**, since 2007, – Russian Summer School for Information Retrieval


> **Yandex School of Data Analysis**, since 2007 – 2 years master program

Grants and Online Contests

> **IMAT (Internet Mathematics) 2005, 2007** – Yandex Research Grants; 9 data sets

> **IMAT 2009** – Learning To Rank (in a modern setup: test set is 10000 queries and ~100000 judgments, no raw data)

> **IMAT 2010** – Road Traffic Prediction

http://company.yandex.ru/academic/grant/datasets_description.xml
http://imat2009.yandex.ru/datasets
http://www.romip.ru
We are hiring!