PRODUCTION CASE

REALTIME RECOMMENDATIONS ON SCALE USING K-NEAREST NEIGHBOURS

Stephan Sahm | Senior Consultant at Machine Learning Reply
| Data Scientist and Machine Learning Engineer
#1 fastest growing technology consulting firm

Annual turnover of 1 billion Euro

8,000 Employees

Source: Reply Group
From strategic approach to implementation and operation, Machine Learning Reply covers the entire lifecycle on generating data and turn valuable insights into efficient actions.
OUTLINE

1. GOAL
2. HOW WE TACKLED IT
3. DIMENSION REDUCTIONS
4. APPROXIMATE KNN
5. RESULTS
1. GOAL
Goal: intelligent coupon assignment for new partner

Customer: Benefit Program, running personalized and partner-specific promotions

Problem:
• for new partners, no usage data exists yet
• existing personalization models fail
Goal: intelligent coupon assignment for new partner

- A lot existing data

Cooperation with many Partners

- Happens frequently
- no data in the beginning

New Partners

- Make use of existing data
- SparkML

Unsupervised Model
2. HOW WE TACKLED IT
How we tackled it

<table>
<thead>
<tr>
<th>Customer</th>
<th>Neighbour1</th>
<th>Neighbour2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Julia</td>
<td>Hans</td>
<td>Franziksa</td>
</tr>
<tr>
<td>Georg</td>
<td>Ingrid</td>
<td>Peter</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

• if someone uses a coupon, also play it out to similar customers

Snowball Idea

Realtime Demands
• want to send out new coupons in realtime

Solution
• precompute similar customers
• KNN
Input Data Challenge

Input Data

- User * Partner Matrix
- how often they have used coupons
- 30+ Mio. Users
- 650+ Partners

Problems

- Curse of Dimensionality
- need to scale using Spark

Solution

compare different...
- distance functions
- dimension reductions
- KNN approximations

| User  | P1 | P2 | P3 | ...
|-------|----|----|----|---
| Sandra| 1  | 0  | 23 | ...
| Michael| 0 | 0  | 2  | ...
| Tom   | 4  | 11 | 3  | ...
| ...   | ...| ...| ...| ...
Worse yet, when $n>9$, we have [...] that $r_n > 2$, and thus the point $(r_n, 0, 0, \ldots, 0)$ on the central sphere lies outside the hypercube of side 4, even though it is "completely surrounded" by the unit-radius hyperspheres that "fill" the hypercube (in the sense of packing it). The central sphere "bulges" outside the hypercube in high-dimensional space."

3. DIMENSION REDUCTIONS
Collaborative Filtering = Matrix Factorization

\[ r_{ij} = \langle u_i, m_j \rangle, \forall i, j \]

Y. Zhou et al. 2008
Topic Modelling = Probabilistic Matrix Factorization

\[
p(w_d | \theta, \beta) = \langle \theta_d, \beta_w \rangle, \forall d, w
\]

\[
r_{ij} = \langle u_i, m_j \rangle, \forall i, j
\]

- Document-WordProbability matrix \( w_d \)
- Document-TopicProbability matrix \( \theta \)
- Topic-WordProbability matrix \( \beta \)

M.Hoffman et al. 2010
4. APPROXIMATE KNN
Approximate KNN

Problem:
- Brute Force KNN is **quadratic** in runtime
- Computation time: **a whole month** (on-premise)

Best Solution:
- Approximate with Local Sensitive Hashing (LSH)
- Computation Time: **3 hours**
- Constraint: Only subset of metrics supported
- Still challenge to scale this properly

[https://github.com/linkedin/scanns](https://github.com/linkedin/scanns)
5. RESULTS
Results

**Target Score:** How many neighbours actually use coupons at the same Partners?

**Distance:** Cosine Distance

**Dimension Reduction:** Collaborative Filtering worked best for Dimension Reduction

**Approximation:** LSH was that good in final performance, that we haven’t used dimension reduction at all in the final KNN

Running now in production
SUMMARY
Summary

- intelligent coupon selection without having training data
- Snowball idea + realtime requirement
- Curse of Dimensionality, n > 9
- Collaborative Filtering / Topic Modelling = Matrix Factorization / Probabilistic MF
- Best scalable KNN Approximation: LSH
- In production out there and assigning coupons today 😊
Thank you very much for your attention!

s.sahm@reply.de

We are hiring!

REPLY
MACHINE LEARNING