YJRS at the NTCIR-14
OpenLiveQ-2 Task

June 13, 2019
Tomohiro Manabe, Sumio Fujita, Akiomi Nishida
Yahoo Japan Corporation
Abstract

6 simple approaches:
• CA (training method)+basic (features)
• CA+BM25F
• CA+translated
• CA+all
• ListNet+basic
• LightGBM+basic

Evaluation findings:
• 4 approaches passed the first round.
• ListNet+basic was quite promising.
Our Approaches
Our Approaches

They are combinations of 3 ranking models (or training methods) and 4 feature sets.

<table>
<thead>
<tr>
<th>Run IDs</th>
<th>Ranking Model</th>
<th>Training Method</th>
<th>Feature Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>91, 95</td>
<td>Linear Combination</td>
<td>Coordinate Ascent</td>
<td>Basic</td>
</tr>
<tr>
<td>92</td>
<td>SQPBIQBJPO</td>
<td>Coordinate Ascent</td>
<td>Basic + BM25F-like</td>
</tr>
<tr>
<td>93</td>
<td>SQPBIQBJPO</td>
<td>Coordinate Ascent</td>
<td>Basic + translated</td>
</tr>
<tr>
<td>136</td>
<td>SQPBIQBJPO</td>
<td>Coordinate Ascent</td>
<td>All</td>
</tr>
<tr>
<td>100, 113</td>
<td>SQPBIQBJPO</td>
<td>ListNet</td>
<td>Basic</td>
</tr>
<tr>
<td>144, 148</td>
<td>SQPBIQBJPO</td>
<td>LightGBM</td>
<td>Basic</td>
</tr>
</tbody>
</table>
CA + Basic

It optimizes a linear combination of 77 features with Coordinate Ascent (CA).
- 17 LETOR features of 4 textual fields
  - TF, TFIDF, BM25, field length, …
- 9 numeric fields
  - Answer count, page view count, baseline rank, timestamp, …


README of the official tool
https://github.com/mpkato/openliveq/blob/master/README.md
CA + Basic

4 Textual Fields
- Question title
- Question snippets
- Question body
- Best answer body

4 fields → 4 x 17 = 68 features → Basic Feature Extractor

9 Numeric Fields
- Answer count
- Page view count
- Baseline rank
- Timestamp
- ...

9 features → Linear Combination Model

Score
CA + BM25F

It is one of the best performers in the previous round of this task. It uses:

• 3 BM25F-like features in addition to the 77 basic features
• nDCG@10 as the objective function of CA
• 5-fold cross validation (5cv)

Ref:
Manabe et al. YJRS at the NTCIR-13 OpenLiveQ Task, NTCIR 2017.
CA + BM25F

- **4 Textual Fields**
  - Question title
  - Question snippets
  - Question body
  - Best answer body

- **9 Numeric Fields**
  - Answer count
  - Page view count
  - Baseline rank
  - Timestamp
  - ...

- **Basic Feature Extractor**
  - 4 fields
  - 4 x 17 = 68 features

- **BM25F-like Feature Extractor**
  - 3 features

- **Linear Combination Model**
  - 9 features

- **Score**
It is inspired from another best performer in the previous round.

• Vocabulary of query must be more similar to questions than to answers.

• This method translates best answer body text into “question language.”

• The translation model is based on GIZA++ toolkit and a public corpus.

Refs: Chen et al. Erler at the NTCIR-13 OpenLiveQ task. NTCIR 2017
Och et al. Improved statistical alignment models. ACL 2000

Yahoo! Chiebukuro data (second edition)
https://www.nii.ac.jp/dsc/idr/yahoo/chiebkr2/Y_chiebukuro.html
CA + Translated

4 + 1 Textual Fields
- Question title
- Question snippets
- Question body
- Best answer body
- Best answer body in question language

9 Numeric Fields
- Answer count
- Page view count
- Baseline rank
- Timestamp
- ...

Basic Feature Extractor

5 fields → 5 x 17 = 85 features

Translation Model

Translation Model → 9 features

Linear Combination Model

Linear Combination Model → Score

5 x 17 = 85 features

Score

Copyright © 2019  Yahoo Japan Corporation. All Rights Reserved.
CA + All

It trains a model with all of:
• Basic features
• BM25F-like features
• Translated features
CA + All

4 + 1 Textual Fields
- Question title
- Question snippets
- Question body
- Best answer body
- Best answer body in question language

9 Numeric Fields
- Answer count
- Page view count
- Baseline rank
- Timestamp
- ...

5 fields
- Basic Feature Extractor
  - 5 x 17 = 85 features

3 features
- BM25F-like Feature Extractor
  - 3 features

9 features
- Linear Combination Model

Score
ListNet + Basic

We can arbitrary replace the linear combination model with a neural ranking model for example.

• We generated a simple neural ranking model with ListNet.

• The model is 3-layered fully-connected feed-forward neural network with 200 hidden nodes.

• It inputs 77 basic features.

Ref: Cao et al. Learning to rank: from pairwise approach to listwise approach. ICML 2007
ListNet + Basic

4 Textual Fields
- Question title
- Question snippets
- Question body
- Best answer body

4 fields

Basic Feature Extractor

4 x 17 = 68 features

3-Layer Fully-Connected FF NN

9 Numeric Fields
- Answer count
- Page view count
- Baseline rank
- Timestamp
- ...

9 features

Score
Ensemble trees is another common model for learning to rank.

- We generated another simple model with LightGBM.
- It is linear combination of 100 regression trees with 15 leaves each.
- It inputs 77 basic features.

Ref: https://github.com/microsoft/LightGBM
LightGBM + Basic

4 Textual Fields
- Question title
- Question snippets
- Question body
- Best answer body

4 fields → Basic Feature Extractor → 4 x 17 = 68 features

9 Numeric Fields
- Answer count
- Page view count
- Baseline rank
- Timestamp
- ...

9 features → Ensemble of 100 Regression Trees → Score
Minor Modifications

We also submitted slightly modified versions of our runs:

- A retry of CA+basic (ID: 95)
- ListNet+basic with 5cv (ID: 113)
- LightGBM+basic with a rough grid search on hyper-parameter space (ID: 148)

Note: Our methods probabilistically generate ranking models.
Other Possible Modifications

• We could input our extended feature sets into ListNet or LightGBM.
  • E.g., ListNet+BM25F, LightGBM+translated, …
• We did not do that due to the time limitation.
## Summary of Our Runs
in descending order of Q-measure

<table>
<thead>
<tr>
<th>Run ID</th>
<th>Description</th>
<th>Q-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>93</td>
<td>CA + translated</td>
<td>0.46387</td>
</tr>
<tr>
<td>92</td>
<td>CA + BM25F</td>
<td>0.45609</td>
</tr>
<tr>
<td>95</td>
<td>CA + basic (retry)</td>
<td>0.39559</td>
</tr>
<tr>
<td>91</td>
<td>CA + basic</td>
<td>0.39124</td>
</tr>
<tr>
<td>136</td>
<td>CA + all</td>
<td>0.38514</td>
</tr>
<tr>
<td>148</td>
<td>LightGBM + basic (tuned)</td>
<td>0.37429</td>
</tr>
<tr>
<td>100</td>
<td>ListNet + basic</td>
<td>0.37340</td>
</tr>
<tr>
<td>113</td>
<td>ListNet + basic (5cv)</td>
<td>0.37240</td>
</tr>
<tr>
<td>144</td>
<td>LightGBM + basic</td>
<td>0.37228</td>
</tr>
</tbody>
</table>
Evaluations
Evaluations

The evaluation process is the key point of this task!
It is based on Pairwise Preference Multileaving on Yahoo! *Chiebukuro*.

- First round: 61 runs (9 are ours)
- Second round: 30 runs (5 are ours)
First Round
credits of our runs* and other team’s best runs

-2633.2 2489.0 2130.6 2006.3 1944.6 1923.0 1627.7 1592.1 316.2 162.8 -44.2 -315.4 -1420.9

92* 93* 111 118 113* 100* 95* 104 91* 148* 136* 144* 89
Among our runs, CA+BM25F and CA+ translated runs were on the top-tier. (consistent with offline evaluation)
First Round

credits of our runs* and other team’s best runs

The performance of our linear combination runs were not stable.
First Round
credits of our runs* and other team’s best runs

As an example: Performance of runs 95 and 91 were quite different although they are both CA+basic.
Performance of CA+all was quite worse than CA+BM25F and CA+translated, which are the same configurations except using only subsets of all the features.
First Round
credits of our runs* and other team’s best runs

Other than linear combination runs, ListNet+Basic runs were on the second-tier.
First Round
credits of our runs* and other team’s best runs

Our LightGBM+Basic did not work well.
Based on the result of this round, our LightGBM+basic and CA+all runs were excluded from the second round.
Second Round

credits of our runs* and other team’s best runs

<table>
<thead>
<tr>
<th>Run</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>113*</td>
<td>1867.4</td>
</tr>
<tr>
<td>92*</td>
<td>1846.7</td>
</tr>
<tr>
<td>100*</td>
<td>1815.9</td>
</tr>
<tr>
<td>93*</td>
<td>1416.7</td>
</tr>
<tr>
<td>111</td>
<td>1406.4</td>
</tr>
<tr>
<td>118</td>
<td>1129.6</td>
</tr>
<tr>
<td>95*</td>
<td>888.0</td>
</tr>
<tr>
<td>104</td>
<td>648.3</td>
</tr>
</tbody>
</table>

*Note: These are inferred values and may not align perfectly with the graphical representation.
Second Round
credits of our runs* and other team’s best runs

Our ListNet+basic runs occupied better positions than in the first round.
Second Round
credits of our runs* and other team’s best runs

The other tendencies were the same as in the first round.
Our ListNet+basic and CA+BM25F runs performed the best.
Conclusions

• Some combinations of a simple learning-to-rank method and reasonable features performed well.

• Namely, our ListNet+basic and CA+BM25F methods performed well.

• Our linear combinations resulted in unstable performance whereas ListNet+basic was quite promising.