Objective Function for DSSM

1. Current objective function

The current objective function for training a DSSM is derived as the follows:

Suppose a set of positive/relevant query-document pairs are provided as \((q_1, d_1), (q_2, d_2), \ldots, (q_n, d_n)\). To model the joint likelihood of these observations, we first model the probability \(P(d_i|q_i)\) as

\[
P(d_i|q_i; \bar{w}) = \frac{\exp(y R(q_i, d_i; \bar{w}))}{\exp(y R(q_i, d_i; \bar{w}))) + \sum_{d \in D^{-}_i} \exp(y R(q_i, d; \bar{w}))},
\]

where \(R(q, d; \bar{w})\) is the similarity between query \(q\) and document \(d\), parameterized by \(\bar{w}\), and \(D^{-}_i\) is a set of \(j\) randomly selected (irrelevant) documents for \(q_i\). \(\gamma > 0\) is a pre-determined smooth parameter. Then, assuming that those query-document pairs are independent, the joint likelihood of seeing those query-document pairs is \(\prod_{i=1}^{n} P(d_i|q_i; \bar{w})\) so that the cost function to minimize becomes

\[
L^1(\bar{w}) = -\sum_{i=1}^{n} \log(P(d_i|q_i; \bar{w})).
\]  

2. Generalized objective function

In practice, \(\{(q_i, d_i): i = 1, 2, \ldots, n\}\) are \(n\) query-document pairs such that \(d_i\) is clicked under \(q_i\) for \(i = 1, 2, \ldots, n\). One issue raised here is that two different clicked query-document pairs might have different click-sIGNALS. Even for a same query, two clicked documents might be clicked different times by different numbers of people. Based on this point, a general setting is to start with a set of positively scored query-document pairs \((q_1, d_1, s_1), (q_2, d_2, s_2), \ldots, (q_n, d_n, s_n)\) where \(s_i > 0\) is the score for the pair \((q_i, d_i)\). An example of the score is number of clicks, or number of people who clicked. The generalized objective function for training a DSSM is derived as the follows:

Let \(y_i = \frac{1 - \exp(-\sigma s_i)}{1 + \exp(-\sigma s_i)}\) with \(\sigma > 0\) as a pre-determined smooth parameter. Since \(s_i > 0\), we have \(0 < y_i < 1\). We see that \(y_i\) can be treated as the probabilistic label for \((q_i, d_i)\). We still use (1) to model \(P(d_i|q_i)\). Furthermore we model the probability \(P(s_i|d_i, q_i)\) as

\[
P(s_i|q_i, d_i; \bar{w}) = P(d_i|q_i; \bar{w})^{y_i}(1 - P(d_i|q_i; \bar{w}))^{1-y_i}.
\]  

Then we consider the joint likelihood of observing \(s_1, s_2, \ldots, s_n\) given \((q_1, d_1), (q_2, d_2), \ldots, (q_n, d_n)\) and make independence assumption. That is

\[
P(s_1, s_2, \ldots, s_n|q_1, d_1), (q_2, d_2), \ldots, (q_n, d_n) = \prod_{i=1}^{n} P(s_i|q_i, d_i; \bar{w}),
\]

which leads to the cost function

\[
L^2(\bar{w}) = -\sum_{i=1}^{n} \log(P(s_i|q_i, d_i; \bar{w}))
\]

\[
= -\sum_{i=1}^{n} [y_i \cdot \log(P(d_i|q_i; \bar{w})) + (1 - y_i) \cdot \log(1 - P(d_i|q_i; \bar{w}))].
\]

If \(y_i \to 1\) for all \(i\), then this cost function reduces to the original one (2). Suppose \(\sigma = 1.237\), then we have the following table for the mapping from \(s_i\) to \(y_i\):

<table>
<thead>
<tr>
<th>(s_i)</th>
<th>(y_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.55</td>
</tr>
<tr>
<td>2</td>
<td>0.84453</td>
</tr>
</tbody>
</table>
Basically, if there is only one click for a query-document pair \((q_i, d_i)\), then we keep some uncertainty so that the corresponding terms in (5) has 0.55 weight on \(\log(P(d_i|q_i; \overline{w}))\) and 0.45 weight on \(\log(1 - P(d_i|q_i; \overline{w}))\). If there are more clicks such that \(s_i \geq 3\), then the corresponding terms in (5) is pretty much weighted on \(\log(P(d_i|q_i; \overline{w}))\) by 0.952.

Here is a sample data set “qd_pairs_DSSM.tsv” in //Msrr-deep-01/IPEsearch/data.

This sample data set is extracted from the Xbox One’s query/click-logs from Dec. 2013 to May 2014. The schema is

(query, bingId, title, mediaType, numberUsersWhoClicked)

It’s first sorted by bingId, then sorted by numberUsersWhoClicked so that it’s clear to see under what queries a document is clicked and by how many users.

3. Gradient computation

By the cost function (2), we have

\[
\frac{\partial L^1(\overline{w})}{\partial \overline{w}} = -\sum_{i=1}^{n} \frac{1}{p(d_i|q_i; \overline{w})} \cdot \frac{\partial}{\partial \overline{w}} P(d_i|q_i; \overline{w}).
\]  

(6)

By the cost function (5), we have

\[
\frac{\partial L^2(\overline{w})}{\partial \overline{w}} = -\sum_{i=1}^{n} \left[ y_i \cdot \frac{\partial}{\partial \overline{w}} P(d_i|q_i; \overline{w}) + (1 - y_i) \cdot \frac{\partial}{\partial \overline{w}} P(d_i|q_i; \overline{w}) \right]
\]

\[
= -\sum_{i=1}^{n} \left[ \frac{y_i}{p(d_i|q_i; \overline{w})} - \frac{(1 - y_i)}{1 - p(d_i|q_i; \overline{w})} \right] \cdot \frac{\partial}{\partial \overline{w}} P(d_i|q_i; \overline{w}).
\]  

(7)

Therefore the only difference in computing the gradient for (5) is that the coefficient of \(\frac{\partial}{\partial \overline{w}} P(d_i|q_i; \overline{w})\) is changed from \(-\frac{1}{p(d_i|q_i; \overline{w})}\) to \(-\frac{y_i}{p(d_i|q_i; \overline{w})} + \frac{(1 - y_i)}{1 - p(d_i|q_i; \overline{w})}\).
4. Implementation Notes:

\[ P(d_i|q_i; \tilde{w}) = \frac{\exp(\gamma R(q_i, d_i; \tilde{w}))}{\exp(\gamma R(q_i, d_i; \tilde{w})) + \sum_{d \in D_i^-} \exp(\gamma R(q_i, d; \tilde{w}))} \]

\[ = \frac{1}{1 + \sum_{d \in D_i^-} \exp(\gamma R(q_i, d; \tilde{w}) - \gamma R(q_i, d_i; \tilde{w}))} \]

Define \( \Delta_i^d = R(q_i, d_i; \tilde{w}) - R(q_i, d; \tilde{w}) \), then

\[ P(d_i|q_i; \tilde{w}) = \frac{1}{1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)} \]

\[ 1 - P(d_i|q_i; \tilde{w}) = \frac{\sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)}{1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)} \]

\[ \log P(d_i|q_i; \tilde{w}) = -\log(1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)) \]

\[ \log(1 - P(d_i|q_i; \tilde{w})) = \log(\sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)) - \log(1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)) \]

\[ \nabla_{\tilde{w}} \log P(d_i|q_i; \tilde{w}) = -\frac{\gamma \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)\nabla_{\tilde{w}} \Delta_i^d}{1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)} = -\sum_{d \in D_i^-} \alpha_i^d \cdot \nabla_{\tilde{w}} \Delta_i^d, \]

where \( \alpha_i^d = \frac{-\gamma \exp(-\gamma \Delta_i^d)}{1 + \sum_{d' \in D_i^-} \exp(-\gamma \Delta_i^{d'})} \).

\[ \nabla_{\tilde{w}} \log(1 - P(d_i|q_i; \tilde{w})) = \nabla_{\tilde{w}} \log(\sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)) - \nabla_{\tilde{w}} \log(1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)) \]

\[ = -\frac{\gamma \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)\nabla_{\tilde{w}} \Delta_i^d}{\sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)} - \frac{-\gamma \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)\nabla_{\tilde{w}} \Delta_i^d}{1 + \sum_{d \in D_i^-} \exp(-\gamma \Delta_i^d)} \]

\[ = \sum_{d \in D_i^-} (\beta_i^d - \alpha_i^d) \cdot \nabla_{\tilde{w}} \Delta_i^d, \]

where \( \beta_i^d = \frac{-\gamma \exp(-\gamma \Delta_i^d)}{\sum_{d' \in D_i^-} \exp(-\gamma \Delta_i^{d'})} \).

Therefore for the original loss function \( L^1(\tilde{w}) = -\sum_{i=1}^n \log P(d_i|q_i; \tilde{w}) \), the gradient can be expressed as

\[ \nabla_{\tilde{w}} L^1(\tilde{w}) = -\sum_{i=1}^n \nabla_{\tilde{w}} \log P(d_i|q_i; \tilde{w}) \]

\[ = \sum_{i=1}^n \sum_{d \in D_i^-} \Delta_i^d \cdot \nabla_{\tilde{w}} \Delta_i^d, \]

For the new loss function
\[ L^2(\mathbf{w}) = -\sum_{i=1}^{n}[y_i \cdot \log(P(d_i|q_i;\mathbf{w})) + (1 - y_i) \cdot \log(1 - P(d_i|q_i;\mathbf{w}))], \]

the gradient can be expressed as

\[
\nabla_{\mathbf{w}} L^2(\mathbf{w}) = -\sum_{i=1}^{n}[y_i \cdot \nabla_{\mathbf{w}} \log(P(d_i|q_i;\mathbf{w})) + (1 - y_i) \cdot \nabla_{\mathbf{w}} \log(1 - P(d_i|q_i;\mathbf{w}))]
\]

\[
= \sum_{i=1}^{n}[y_i \cdot \sum_{d \in D_i} \alpha_d^i \cdot \nabla_{\mathbf{w}} \Delta_d^i + (1 - y_i) \cdot \sum_{d \in D_i} (\alpha_d^i - \beta_d^i) \cdot \nabla_{\mathbf{w}} \Delta_d^i]
\]

\[
= \sum_{i=1}^{n} \sum_{d \in D_i} \left[\alpha_d^i - (1 - y_i) \beta_d^i \right] \cdot \nabla_{\mathbf{w}} \Delta_d^i. \tag{9}
\]

Now, for the package (release_06-25-2014), we need to change the coefficient of \( \nabla_{\mathbf{w}} \Delta_d^i \) from \( \alpha_d^i \) to \( \left[\alpha_d^i - (1 - y_i) \beta_d^i \right] \). In the special case that all \( y_i \) are 1, then (9) reduces to (8). Note that the only difference between \( \alpha_d^i \) and \( \beta_d^i \) is “1” in the denominator.