Keeping Things That Matter: An Exploration on Online Learning with Delayed Feedback

Marco Tulio Ribeiro and Nacho Cano – University of Washington

March 14, 2014

The Problem

Online Learning:

Delayed Feedback:

Possible Applications

- Anomaly detection in network traffic.
- Pay per purchase systems.
- Wikipedia vandalism.

Approach

We implemented Logistic Regression with SGD and delayed regularization. We used three different step size policies: constant, 1/t decay and AdaGrad.

We proposed and evaluated different strategies for either compressing the data in memory and/or dropping instances or features, in order to maintain a bound on the memory used.

Dropping Instances

- **Random drop**: drop an example at random when memory is full.
- **LRU**: drop the least recently used example (like a FIFO queue).
- **Uncertainty drop**: drop the instance that we are more certain about (from Active Learning literature).
- **Ada drop**: drop the instance that has the lowest utility calculated as:

  \[ u(^i) = \sum_{j \in I} \ln \eta_j \]  

  where \( \eta_j \) = adagrad step size of the jth feature

Dropping Features

- **Ada feature drop**: drop the feature with the smaller step size (based on AdaGrad) from all examples.
- **Random Ada feature drop**: select an instance at random and drop the feature of that example that has the smaller step size (based on AdaGrad).
- **Random feature drop**:
  - **Example specific (fr1)**: select an example at random, select a feature from that example at random and remove it.
  - **All Examples (fr2)**: select a feature at random and remove it from all the examples.
- **Max Frequency feature drop**: remove the feature that appears the most in memory from a random example that contains it. This preserves diversity of features in the examples in memory.

Hashing Kernels

We used it to reduce the dimensionality of the feature space. Features are kept in a more compressed way. Any of the dropping strategies described above can be applied here as well.

Results

We ran our tests over a “delayed” version of the click-through rate dataset from 2012 KDD Cup Track 2. We induced a N(0, sigma) delay. We explored the 2010 PAN Wikipedia Vandalism dataset as well, preprocessed it and ran some experiments, but didn’t get any useful insights yet.

<table>
<thead>
<tr>
<th>Step Size Policy</th>
<th>NO DELAY</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Hashing</td>
<td>0.668481254</td>
<td>0.60639443</td>
</tr>
<tr>
<td>Hashing</td>
<td>0.676014587</td>
<td>0.60639445</td>
</tr>
</tbody>
</table>

Future Work

- Theoretical analysis with mathematical guarantees.
- Focus more on the dropping features policies.
- Add experiments with different datasets

Acknowledgments

Thanks to Tianqi Chen, Sameer Singh and Carlos Guestrin for their key contributions to the project.