Algorithms for Graph-Based Supervised Learning

Brian Dolan

September 1, 2017

Hello

1 Modified Adsorption

This routine is described in Talukdar below.

I'm looking for a good parallelization strategy for this function. I'm going to expand it as far as I can to see if something presents itself. We are aiming to find c_v for each vertex in the graph.

$$
\begin{aligned}
p(a|b) &= \frac{W_{a,b}}{\sum_u W_{u,b}} && (1)\\
H(x) &= -\sum_y p(y|x)\log p(y|x) && (2)\\
&= -\sum_y \left(\frac{W_{y,x}}{\sum_u W_{u,x}}\right)\log\left(\frac{W_{y,x}}{\sum_u W_{u,x}}\right) && (3)\\
&= \frac{-1}{\sum_u W_{u,x}}\sum_y W_{y,x}\left[\log W_{y,x} - \log\sum_u W_{u,x}\right] && (4)
\end{aligned}
$$

So we can define

$$m(x) = \sum_u W_{u,x}$$

and reduce the above to

$$
\begin{aligned}
H(x) &= -m(x)^{-1}\sum_y W_{y,x}\left[\log W_{y,x} - \log m(x)\right] && (5)\\
&= -m(x)^{-1}\left[\sum_y W_{y,x}\log W_{y,x} - \log m(x)\sum_y W_{y,x}\right] && (6)
\end{aligned}
$$

Next we have the smoothing function for a given β

$$
\begin{aligned}
f(x) &= \frac{\log \beta}{\log(\beta + e^x)} && (7) \\
c_x &= f(H(x)) && (8) \\
&= \log \beta \left[\log(\,\beta + e^{H(x)}\,)\right]^{-1} && (9)
\end{aligned}
$$

2 References

BibTeX is a pain, so for right now I'm going to do

```
@article{doi:10.2200/S00590ED1V01Y201408AIM029,
author = {  Amarnag
  Subramanya  and   Partha Pratim
  Talukdar },
title = {Graph-Based Semi-Supervised Learning},
journal = {Synthesis Lectures on Artificial Intelligence and Machine Learning},
volume = {8},
number = {4},
pages = {1-125},
year = {2014},
doi = {10.2200/S00590ED1V01Y201408AIM029},

URL = {
        https://doi.org/10.2200/S00590ED1V01Y201408AIM029

},
eprint = {
        https://doi.org/10.2200/S00590ED1V01Y201408AIM029

}

}
```