

KARAN THUKRAL - 20460691

ENGINEERING LOGBOOK

SYDE 461

Contents

Week 1: Sept. 11 - 17 3

Week 2: Sept. 18 - 24 4

Week 3: Sept. 15 - Oct. 1 9

Bibliography 15

Week 1: Sept. 11 - 17

- Discussion with team members to clarify the idea and discuss the potential interest
- Discussions with Lyrical Security¹ regarding the data access policy along with NDA and IP agreements. ¹<http://lyricalsecurity.com/>
- Discussion around potential supervisor

Week 2: Sept. 18 - 24

- Team met with and signed Alex Wong as our supervisor
- Back and forth with Lyrical Security about details of NDA and IP agreement
- Finished team and advisor contracts
- Talked with Alex Wong regarding the details of the NDA

Sept 23: Speed Dating Round 1

Team 7: Snow Removal

- No good fully automated way
- Risk of injury while removing snow
- Need more affordable options
- Has a good mix of skills in the team members to pull it off

Team 10: International Shipping

- Using US + Mexico border as example
- lost of illicit goods being shipped
- Not sure how to tackle it yet
- Looking into modelling the physical steps in shipping and then find ways to improve it.
- One idea is a tamper proof seal

Team 5: Heat Exhaustion

- Early detection of heat exhaustions
- #2 cause of death for athletes in US

- Health damage or death can happen from heat exhaustion
- Need to measure a good approximation of internal body temp to be able to detect it
- Prof Stashuk as advisor. Good choice

Team 1: Real Time Wait Time

- How busy is the restaurant I want to go to?
- How long will be the wait time?
- One challenge is to take into consideration crowds inside and outside the location
- Potential for ML

Team 2: Women's Health

- Uncomfortable topic to talk about
- Lot of stigma
- Chat bot to make this conversation easier
- Source info and knowledge from doctors
- Need to scrape existing forums etc to train the NLP model. This will be challenging
- Should not use conversations to train the model since can lead to mis-training and ruin the purpose - example Microsoft bot

Team 4: Pressure Ulcers

- Why does it happen?
- How do you prevent it?
- How do you take proactive action towards it?

Team 8: Understand Products

- Understand existing info about products by using forums, social media reviews etc
- structure this unstructured data somehow - NLP problem

Feedback for Us

- Try and narrow the scope
- Cannot process each packet in realtime without adding overhead
- Have very clear ways of testing and validating it
- Look at Cloudflare. Potentially have a contact there through someone in class

Tensorflow Example

- Started work on a tensorflow example to learn details about neural nets

TF Softmax Regression

Saturday, 24, 2016

- Must

- Each image is 28×28

~~can be~~

- 55K images = $[55000, 784]$ array

- labels are one-hot

→ eg 3 = $[0, 0, 0, 1, 0, 0, 0, 0, 0]$

→ labels = $[55000, 10]$ array

- Softmax is a simple model for when the alg might want to assign probabilities of classification.

Gives a # b/w 0, 1 - to each class
& adds up to 1

- step 1 → Add evidence of our input being in certain classes,

step 2 → convert into probabilities

- Must

evidence → weighted sum of pixel intensities. is -ve if in favour of not being in class, +ve otherwise
↙ input img

$$\text{evidence } i = \sum_j W_{i,j} x_j + b_i$$

↙ weights ↘ bias
 ↘ summing index

$$y = \text{prob} = \text{softmax}(\text{evidence})$$

↘ find out → softmax

turns evidence into prob dist at the output layer

Athulal
24/09/16

Week 3: Sept. 15 - Oct. 1

Tensorflow Example Continued

Sunday, 25, 2016

- Since softmax turns evidence into prob
dist its used as the last layer
even in complicated models.

- ~~Soft~~ i = class
 j = summing index over all classes
 x = input image

- for this softmax is serving as a link func ~~out~~ shaping the output of our linear func into the form we want!!
- its tallying the evidence into prob for our inputing

$$\text{softmax}(x) = \text{normalize}(\exp(x))$$

$$\text{softmax}(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

↳ exponentiating its inputs & then normalizing them

- No hypothesis has 0 or -ve weights
- The exp inc the weight for one unit of +ve evidence & reduce multiplicatively similarly for -ve evidence

$$y = \text{softmax}(wx + b)$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \text{softmax} \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Regression

- Numpy ~~comp~~ does computations in C & thus cares about data types. There is overhead in switching b/w the two though.
- Tensorflow does heavy lifting outside Python. You describe the graph in py & execute outside of py.

Training

- Define what it means for the model to be bad?
 - ↳ cost func, loss func etc
- minimize the error
- cross-entropy

$$H_{y'}(y) = - \sum_i y'_i \log(y_i)$$

y = predicted prob ~~dist~~ dist

y' = true dist

measure ~~at~~ how inefficient our prediction is

→

Back Propagation

- Reverse mode differentiation
- get → ~~how~~ gives the derivative of the output with respect to every single node

Alhulwail
25/09/16

Back Propagation

- Reverse mode differentiation
- Regular chain rule gives you the derivative of the output with respect to one input/node
- Doing that for all nodes is intractable
- To extend this to find $\frac{\partial \text{output}}{\partial}$ with respect to all nodes and inputs in the neural net/graph you start using the chain rule from the other end (output) and go till the input layer. This is essential for neural networks. ²

² Christopher Olah. Calculus on computational graphs: Backpropagation. <https://colah.github.io/posts/2015-08-Backprop/>, August 15, 2015

Bibliography

Christopher Olah. Calculus on computational graphs: Backpropagation. <https://colah.github.io/posts/2015-08-Backprop/>, August 15, 2015.