

# EEL6935 Course Project

## Final Report

---

Caleb Bryant<sup>1</sup> Jixin Feng<sup>2</sup>

EEL6935 T21

Department of

<sup>1</sup>Computer & Information Science & Engineering

<sup>2</sup>Electrical & Computer Engineering

University of Florida, Gainesville, FL

# Sentiment Analysis

Goal: assign sentiment labels to a sentence.

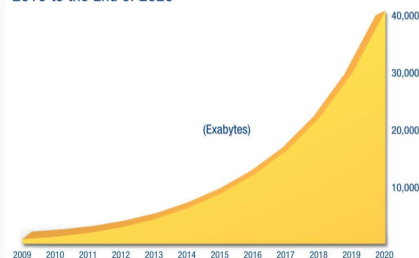
$$f : \mathcal{D} \rightarrow \mathcal{L}$$

- $\mathcal{D} = \{d_0, d_1, \dots, d_{n-1}\}$  is the set of sentence
- $\mathcal{L} = \{l_0, l_1, \dots, l_{k-1}\}$  is the set of labels.

Movie review is ideal for sentiment analysis

- Most dataset of movie review are already associated with scores
- The scores in dataset are reliable

The Digital Universe: 50-fold Growth from the Beginning of 2010 to the End of 2020



---

<sup>1</sup>Gantz, J., & Reinsel, D. (2012). The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. IDC iView: IDC Analyze the future, 2007(2012), 1-16.

# Text Encoding

- Define the collection of text documents to be

$$\mathcal{D} = \{d_1, d_2, \dots, d_D\}$$

- Define vocabulary to be

$$\mathcal{V} = \{v_0, v_1, \dots, v_{m-1}\}$$

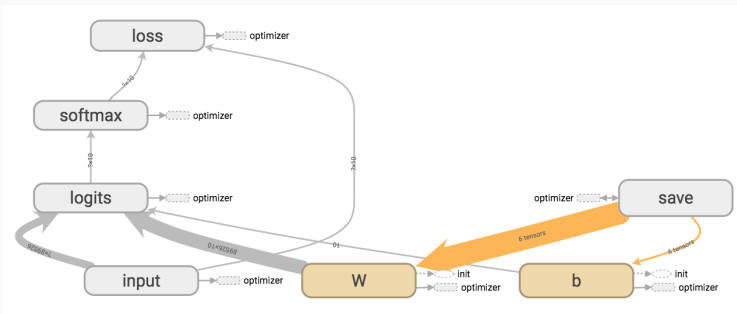
- The notation of frequency of a word  $v \in \mathcal{V}$  occurred in document  $d \in \mathcal{D}$  is  $f_d(v)$ , hence a document can be represented as a vector

$$\vec{d} = \{f_d(v_1), f_d(v_2), \dots\}$$

- Define total number of documents  $d \in \mathcal{D}$  containing the word  $w$  is represented as

$$f_{\mathcal{D}}(v)$$

## Backend: Logistic Regression



- Multiply input vectors by  $W$ , add  $b$
- Convert output into probabilities with Softmax
- pick  $i = \operatorname{argmax}(\hat{y}_i)$
- Train the model using Adam Optimizer
- Common baseline model and simple to implement

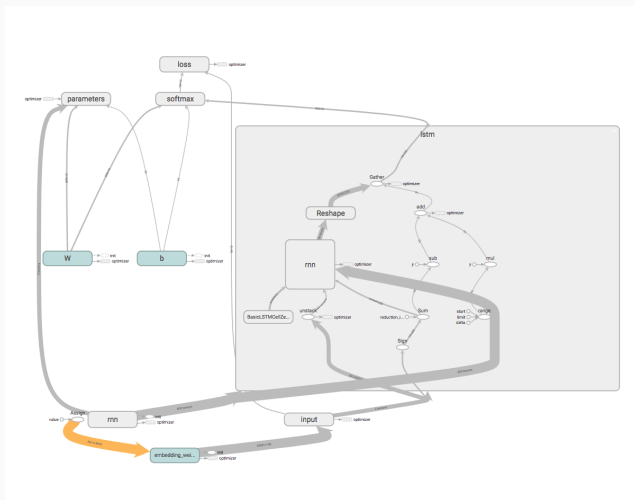
# Backend: LSTM Model

Given a word sequence

$S = \{v_0, v_1, \dots, v_{l-1}\}$  with length  $l$ , the states of LSTM are updated as:

$$\begin{bmatrix} i_t \\ f_t \\ o_t \\ c_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} S[h_{t-1}, x_t]$$

- $c_t = f_t \circ c_{t-1} + i_t \circ \hat{c}_t$
- $h_t = o_t \circ \tanh c_t$



## Backend: LSTM Model

To build our LSTM model:

- Define a dictionary

$$D = [w_1, w_2 \dots w_N]$$

to contain the set of all words representable by our LSTM network

- The  $i$ -th input to our LSTM is a vector

$$v_i = [\pi_{i1}, \pi_{i2} \dots \pi_{iL}]$$

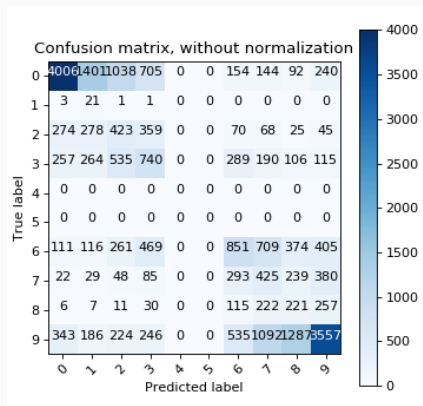
corresponding to a movie review  $r_i$  of length  $L$ , where  $\pi_{ij}$  is the dictionary index of the  $j$ -th word of  $r$

- $D = [\text{cat}, \text{dog}, \text{meow}]$

$$r = [\text{dog}, \text{dog}, \text{cat}, \text{meow}] \Rightarrow v = [1, 1, 0, 2]$$

# Simulation Platform Specs

- Training based on Stanford Large Movie Review Dataset
  - 50,000 highly polar movie reviews
  - 25,000 for training, 25,000 for testing
- Code in Python with Tensorflow/Scikit-Learn
- simulations will be conducted on computers with
  - 3.1GHz Intel Core i7 CPU with 8MB cache
  - nVidia GTX 1050 GPU with 4GB Memory
  - 16GB RAM
  - Ubuntu 14.04 LTS



**Table 1:** System Performance

Method	Epochs	Binomial Training	Binomial Testing	Multinomial Training	Multinomial Testing
scikit-learn LR	N/A	0.9981	<b>0.8697</b>	N/A	N/A
tensorflow LR	20	0.8670	0.8583	0.9982	0.3734
larger LSTM	8	N/A	N/A	0.6693	0.3657
smaller LSTM	2	N/A	0.8507	0.5622	<b>0.4098</b>



# Front End: Flask Web App

- Web App via Flask
- Allow model selection
- Accept text input and sentiment analysis output
- Hosted on low spec VM
  - Ubuntu 16.04 LTS
  - Single Core CPU
  - 2GB RAM
  - No standalone GPU

The screenshot shows the web application interface on a mobile device. The browser address bar displays 't21.eceigator.com'. The header section reads 'EEL6935 | Team 21'. Below this, the title 'Sentiment Analysis' is centered. The main form area contains a text input field labeled 'Movie Review:' with the text 'I love this movie!'. Below the input field, there are two dropdown menus: 'Model:' set to 'LSTM' and 'Granularity:' set to 'Binary'. A 'submit' button is located below these dropdowns. At the bottom of the screen, a table displays the results of the sentiment analysis. The table has two rows: 'Baseline' and 'LSTM'. The 'LSTM' row is highlighted, indicating the current model's output.

Model
Baseline
LSTM

The screenshot shows the web application interface on a mobile device, displaying the output of the sentiment analysis. The browser address bar displays 't21.eceigator.com'. The header section reads 'EEL6935 | Team 21'. Below this, the title 'Sentiment Analysis' is centered. The main content area shows the results of the sentiment analysis. The first row is labeled 'lstm' and the second row is labeled '10'. Below these, a 'Do It Again!' button is visible. At the bottom of the screen, a footer section contains the text 'Design and implemented by Caleb Bryant and Jixin Feng' and 'Built using [mini.css](#)'. The bottom navigation bar includes icons for back, forward, share, book, and tabs.

lstm

10

binary

Do It Again!

Design and implemented by Caleb Bryant and Jixin Feng  
Built using [mini.css](#)

## More About Us



Code Repo



Web App



Video Demo



# Thank You!

Questions?