Is It Funny?

Joke classification with BiLSTM and NLTK

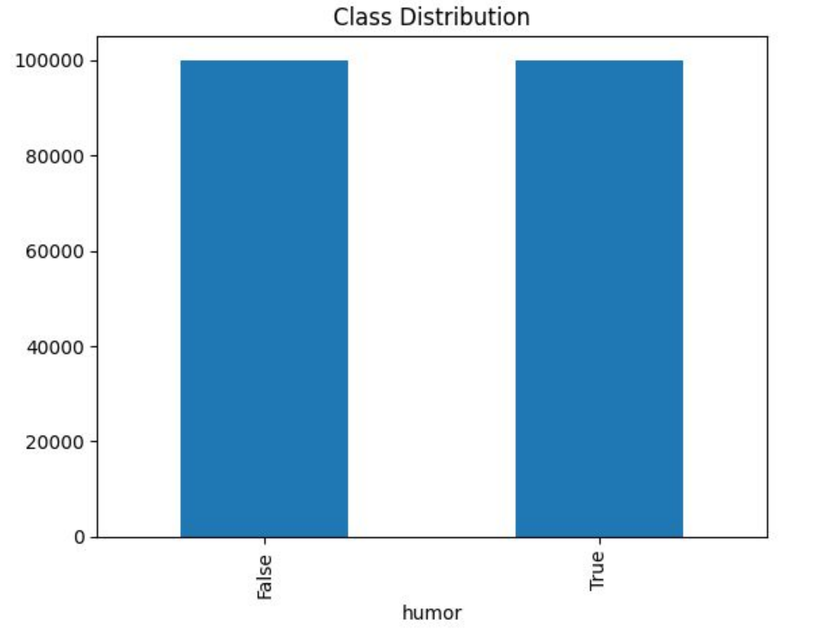
Purpose

- Which is the joke?
  - Why did the chicken cross the road?
  - Why did the man cross the road?
- Can a deep learning model recognize a joke?
- Tools:
  - NLTK to process the texts
  - BiLSTM to identify patterns and classify texts

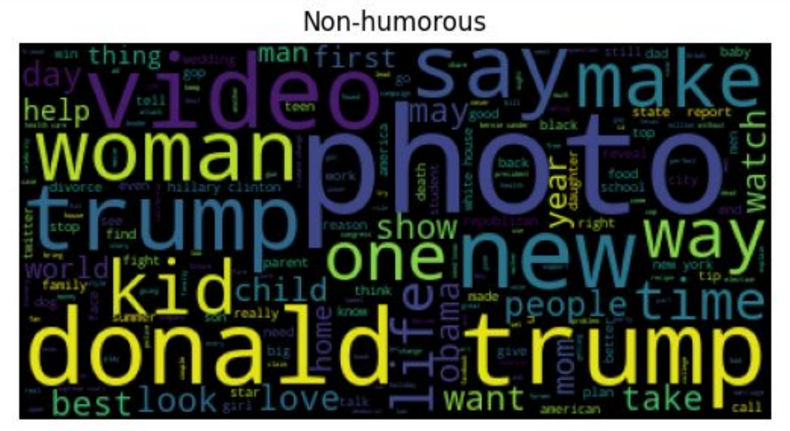
Dataset

- Generated for a paper on BERT Sentence Embedding
- 200,000 short texts
- 2 classes: humor vs non-humor
- Text and class are the only features
- The mean character count for jokes is slightly larger (70) than for non-jokes (65)

EDA


Class Distribution
100000
80000
60000
40000
20000
0
False
True
humor


Humorous
say
call
one
know
woman
guy
go
people
always
got
make
joke
think
want
man
never
wife
called
two
difference
girl
knock
really
love
someone
time
day
life
dog
tell
today
common
walk
bar
many
said
work
kid


Non-humorous
photo
donald trump
video
say
make
woman
trump
new
way
one
kid
time
people
life
obama
show
child
best
look
love
want
take
year
watch
help
day
thing
man
first
world
may

EDA

- Most frequent n-grams in joke texts
- Can recognize some common joke cliches

1	2	3
(fuck, 1761)	((knock, knock), 2478)	((like, woman, like), 1053)
(fucking, 1608)	((get, cross), 1101)	((woman, like, like), 774)
(whats, 1524)	((like, coffee), 960)	((take, change, lightbulb), 480)
(favourite, 1116)	((take, change), 864)	((man, walk, bar), 465)
(lightbulb, 1032)	((yo, mama), 801)	((chicken, cross, road), 447)
(midget, 951)	((call, mexican), 573)	((like, coffee, like), 408)
(dyslexic, 606)	((black, guy), 570)	((wan, na, hear), 387)
(im, 561)	((take, screw), 567)	((like, like, coffee), 372)
(cunt, 453)	((chicken, cross), 543)	((change, light, bulb), 351)
(cant, 420)	((change, lightbulb), 513)	((yo, mama, fat), 345)
(til, 381)	((whats, difference), 480)	((take, change, light), 330)
(constipated, 360)	((hear, joke), 477)	((walk, bar, bartender), 324)

EDA

- Most frequent n-grams in non-joke texts

1	2	3
(huffpost, 1239)	((morning, email), 777)	((huffpost, rise, need), 354)
(reportedly, 900)	((jimmy, fallon), 531)	((rise, need, know), 354)
(allegedly, 666)	((bill, maher), 468)	((new, york, fashion), 303)
(amid, 633)	((photo, poll), 444)	((york, fashion, week), 303)
(fallon, 564)	((week, photo), 432)	((photo, donald, trump), 240)
(defends, 555)	((pope, francis), 429)	((woman, business, q), 222)
(debut, 552)	((kate, middleton), 396)	((wednesday, morning, email), 174)
(middleton, 531)	((gps, guide), 393)	((tuesday, morning, email), 165)
(warns, 528)	((chrissy, teigen), 381)	((friday, morning, email), 162)
(huffpollster, 495)	((roy, moore), 378)	((everything, need, know), 159)
(maher, 489)	((seth, meyers), 372)	((cute, kid, note), 147)
(beyoncé, 483)	((huffpost, rise), 363)	((kid, note, day), 147)

Preprocessing

- Convert text to lowercase
- Tokenize
- Remove stop words
- Lemmatize

```
0    Joe biden rules out 2020 bid: 'guys, i'm not r...
1    Watch: darvish gave hitter whiplash with slow ...
2    What do you call a turtle without its shell? d...
3        5 reasons the 2016 election feels so personal
4    Pasco police shot mexican migrant from behind,...
```

```
0                          joe biden rule bid running
1          watch darvish gave hitter whiplash slow pitch
2                       call turtle without shell dead
3                         reason election feel personal
4     pasco police shot mexican migrant behind new a...
```

Tokenization and Word Embedding

- Tokenized words converted to sequences and padded to ensure equal lengths
- Pretrained Word2Vec word embedding
    - Boosts performance
- Creates a vector representation of the texts to feed into the models

BiLSTM

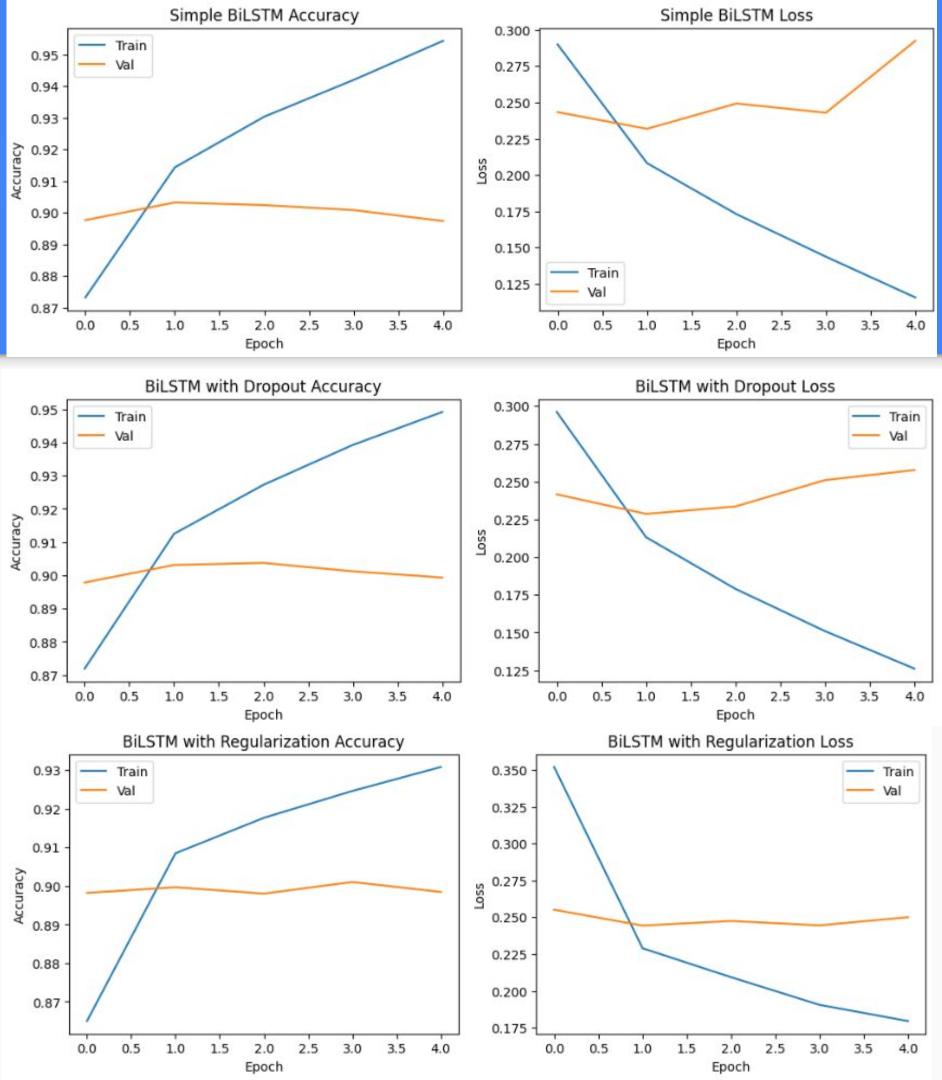
- LSTM maintains a long term memory store
- A BiLSTM uses 2 LSTM layers
  - Forward process
  - Backward process
  - Allows for better learning of sequences
- 3 candidate architectures
  - Simple BiLSTM
  - BiLSTM with a dropout layer
  - BiLSTM with L2 regularization

Training

- Each model was trained with a training and validation set to monitor performance
- Loss function: binary cross entropy
- Optimizer: ADAM

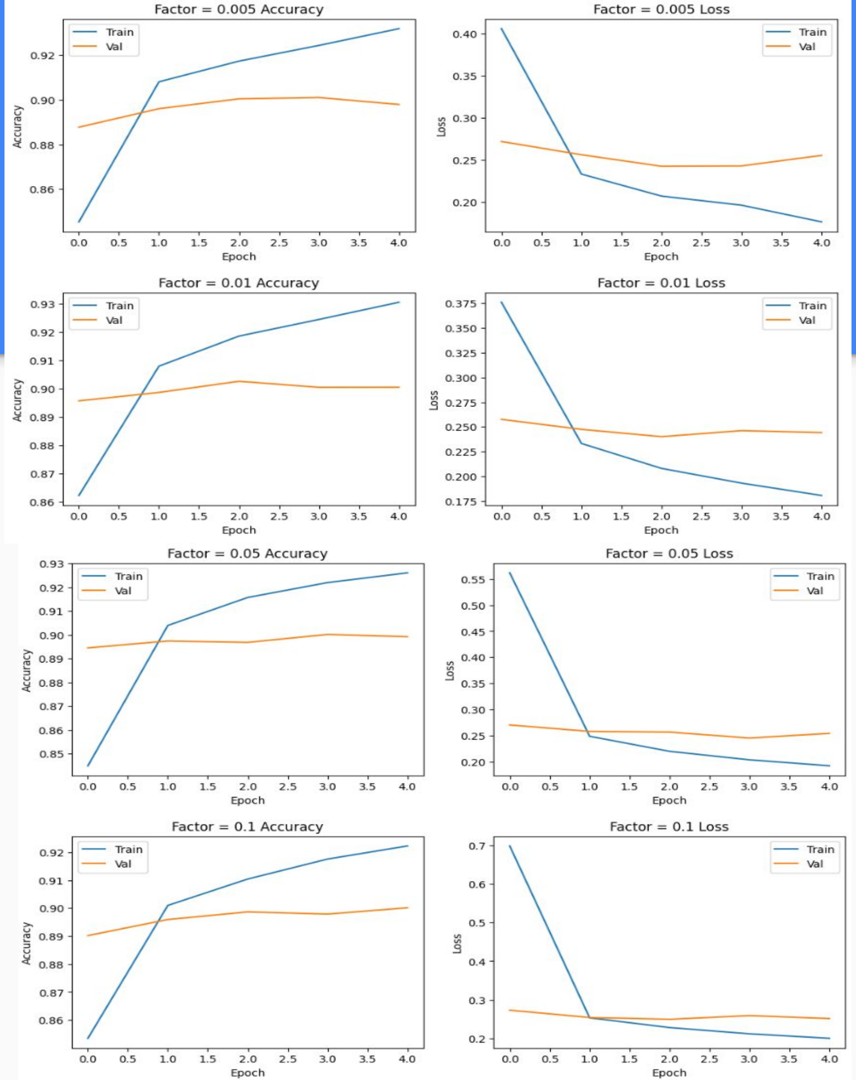
Evaluation

- Validation accuracy is flat across models
- Overfitting observed in losses
- Model with regularization shows best performance



Hypertuning

- Evaluated 4 L2 regularization factors
- As the regularization factor increased, the training and validation losses became more similar



Results

- Final model: BiLSTM with L2 regularization (0.1)
- Test accuracy = 0.90
- Test loss = 0.25

Conclusion and Future Directions

- BiLSTM successfully identified whether texts were jokes
- Regularization helped reduce overfitting
- Future expansion to longer texts would be interesting. Could the model identify jokes that require a longer set up?

References

- Data source: https://github.com/Moradnejad/ColBERT-Using-BERT-Sentence-Embedding-for-Humor-Detection/tree/master?tab=readme-ov-file
- https://towardsdatascience.com/simple-word-embedding-for-natural-language-processing-5484eeb05c06
- https://www.geeksforgeeks.org/bidirectional-lstm-in-nlp/
- https://www.ibm.com/topics/word-embeddings