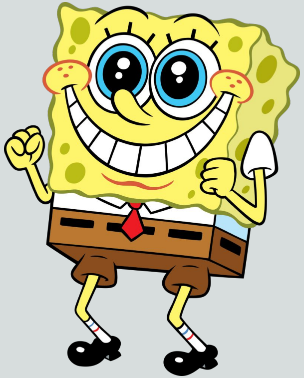
Using NLP Sentiment Analysis to Develop Trading Strategies

Daniel Setiawan & Laurel He
Capstone Project: 3/10/23

FEELING A LIL NEURAL-TASTIC TODAY

- While AI may *never* have feelings, people do.
- Sentiment analysis is significant for analyzing trends, business insights, reputation management
- Throughout this presentation, we’ll use NLP to characterize language into sentiment for a more in-depth analysis.



Positive

Negative

Patrick

GOO-AL

1. **Train** NLP sentiment classifier with StockTwits.

2. **Predict** sentiment on scraped Tweets.

3. **Utilize** predicted sentiment to devise trading strategies.

TRAIN — StockTwits

User-labeled Data

900K observations

5 companies

KAGGLE LINK

TEST — Twitter

Scraped & Unlabeled

4.7M + 1.7M tweets

19 companies + 9 topics

COMP — Yahoo Finances

Scraped Data

Span: 2015 - 2023

19 companies

BONUS — US Statistics

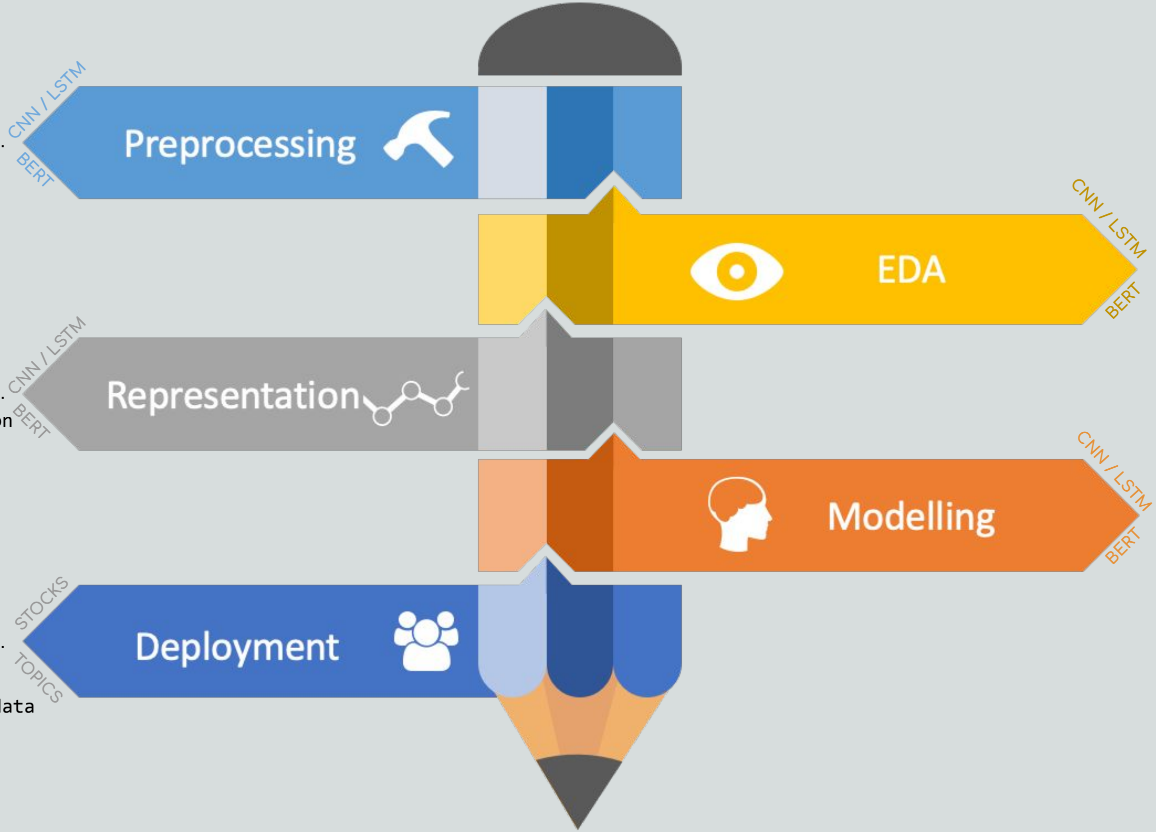
13 datasets

Span: 2022 - 2023

One datapoint / month

DATA LINK

FROM ZERO TO HUNDRED



Preprocessing

CNN / LSTM

1. Remove !^!@#$%
2. Remove stopwords
3. Lemmatization

BERT

1. Remove # @ //
2. Demojize → Unicode

EDA

CNN / LSTM

1. Word2Vec Embedding
2. Spelling Checker

BERT

1. Self-Attention Embedding

Representation

CNN / LSTM

1. Word Tokenization
2. Numericalized
3. Truncated @ 31

BERT

1. Subword Tokenization
2. Attention Masking
3. Truncated @ 64

Modelling

CNN / LSTM

1. Adam, Sigmoid Activation
2. Learning Rate: 5e-4
3. Loss: Binary Cross-Entropy

BERT

1. Optimizer: Adam
2. Learning Rate: 3e-5
3. ε: 3e-8

Deployment

STOCKS

1. MACD / RSI
2. Winning Rates
3. Granger Causality

TOPICS

1. Granger Causality
2. Predicted against data

SENTIMENT ANALYSIS

TRADITIONAL NEURAL NETS

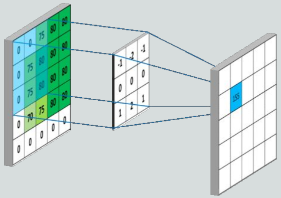
Neural Network Model	Validation Accuracy	Model Highlights
CNN	0.71	Uses a convolution window to slide over large inputs of text to extract useful features (eg. phrases, semantics)
LSTM (RNN)	0.77	Effective at capturing long-term dependencies without its gradient exploding or vanishing (unlike RNNs)
CNN + LSTM	0.75	Able to learn both local and global features in sequential data

CONVOLUTIONAL NEURAL NETWORK

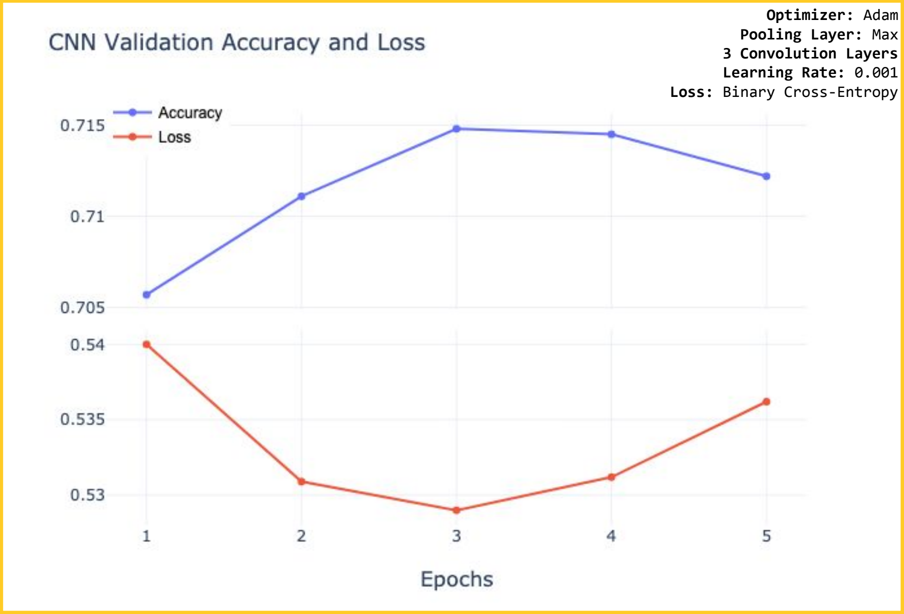
Known for: Ability to learn hierarchies / patterns in input vector.

Convolution is achieved through stacked layers:

1. **Convolution**
2. **Pooling**
3. **Fully Connected**



Feature maps are created, then dimensionality is reduced. Layers are then transformed to produce output.

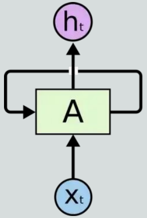


LONG SHORT-TERM MEMORY

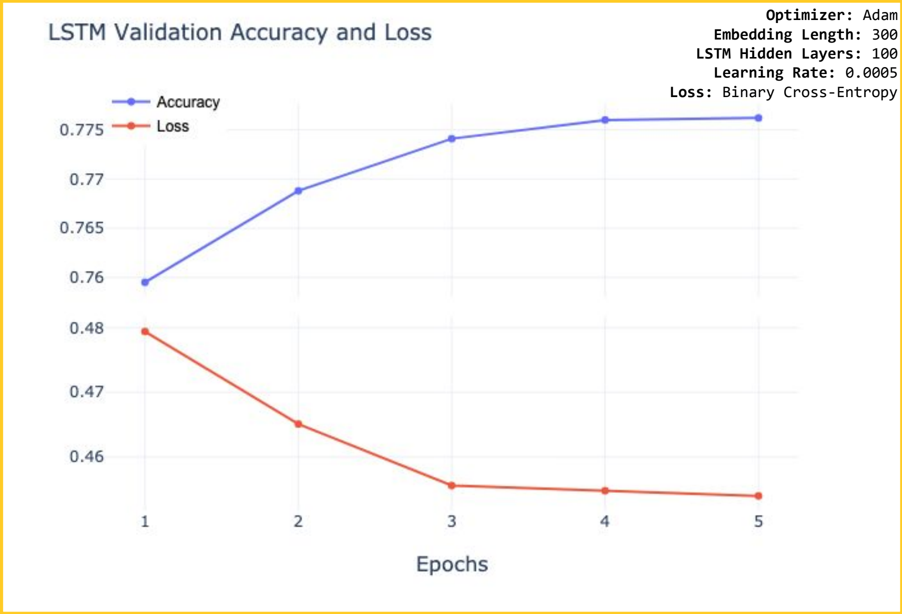
Known for: Ability to capture sequential long-term dependencies.

Uses a gating mechanism to recursively update hidden states:

1. **Input Gate**
2. **Forget Gate**
3. **Output Gate**



Input vectors are processed sequentially; gates control information to retain / forget.



TRANSFER LEARNING : TRANSFORMERS

Known for: Ability to capture sequential long-term dependencies.

Model Differences

RNN: ***Sequential Model***, sees words one after the other

GPT: ***Autoregressive Model***, masks future tokens, only consider the left context when making predictions

BERT: ***Bidirectional Model***, entire sentence run through model to predict masked words, consider left and right context for predictions

Other flavors of BERT used

RoBERTa - robust optimized BERT

FinBERT - BERT trained on financial texts

BERTweet - BERT trained on Tweets

Bidirectional Encoder Representation from Transformers ***(BERT)***

1. General self-supervised (not labeled) language model (not useful for specific tasks)
2. Transfer learning to fine-tune on a specific task with labeled data (supervised learning)
3. Pretrained with the Masked Language Modeling (MLM) objective
4. Learns an inner representation of the English language
5. Used to extract features useful for downstream tasks, eg. train sentiment analysis/ classification using features produced by BERT as inputs

ATTENTION-BASED NEURAL NETS

Transformer Model	Validation Accuracy *(80%-20% train-test split)*	Model Highlights
BERT-Base	0.80	Bidirectional Encoder, Masking, trained on 16GB English text dataset
FinBERT	0.84	Training BERT on financial text
RoBERTa - Base	0.85	Robustly Optimized BERT, improved downstream task performance, trained on 160GB text, full sentences, larger mini-batches and learning rate
BERTweet	0.86	Training BERT on Tweets
RoBERTa - Base fine-tuned on 3.2M StockTwits[1]	0.88	Fine tuned on the same data source as our training dataset but larger

[1] https://huggingface.co/zhayunduo/roberta-base-stocktwits-finetuned

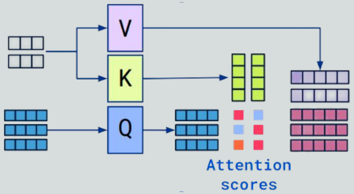
Final Model

TRANSFORMERS

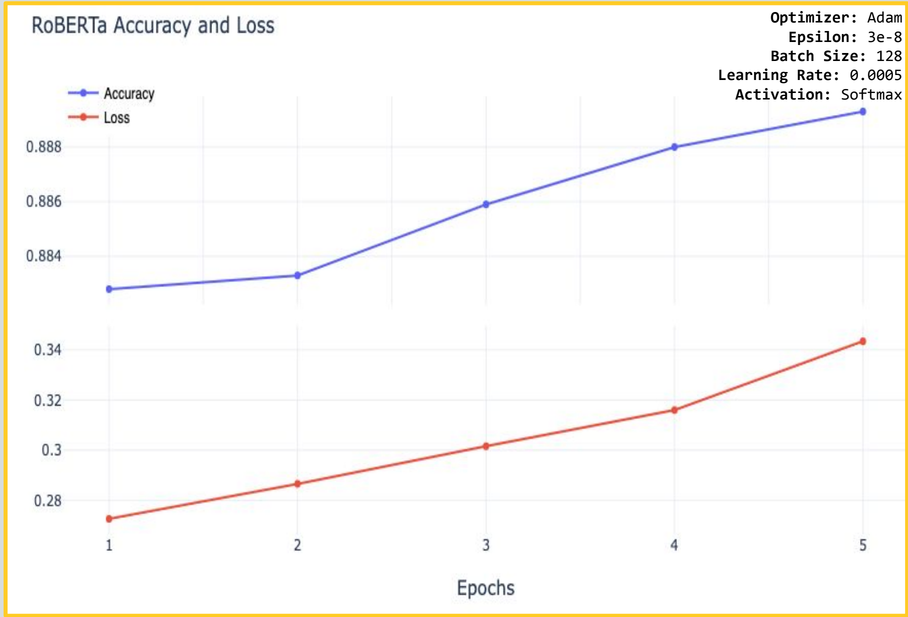
Known for: Ability to process long input sequences efficiently + effectively

Uses a self-attention mechanism to add weights to certain words:

1. **Query Vector**
2. **Key Vector**
3. **Value Vector**



Input vectors are processed non-sequentially. All inputs are *indirectly* connected to output vectors.



DEVISING A SENTIMENT-BASED TRADING STRATEGY

1. From scraped unlabeled Tweets, predict sentiment (score between 0 (negative) and 1 (positive).

2. Plot long-term (200 day) and short-term (50 day) sentiment moving average (MA).

3. Categorize market sentiment from rule of thumb: Whenever the short-term MA line crosses the long-term MA line from below, it is an indication of bullish market and vice versa.

4. Buy and sell signals: buy at the start of bullish market, and sell at the start of bearish market.

5. Compute profit/winning rate, compare with **conventional RSI trading strategy*** winning rate.

*** Conventional RSI Trading Strategy**

If 10-period RSI of the stock is below 30, buy on the next day's open

If 10-period RSI is above 40 or after 10 trading days, sell on the next day's open