Predicting Success of Cable Sitcoms using NLP on Script Data

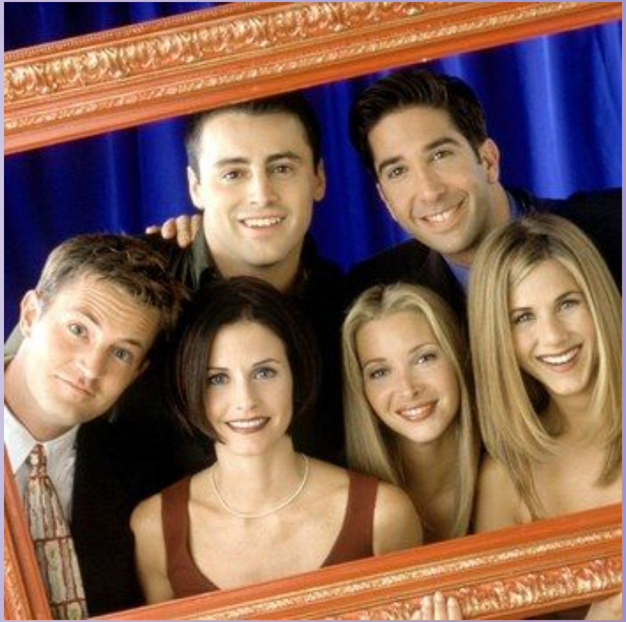
Emily Pfeifer
Capstone Project

Obtaining the data

- 50 sitcom pilots - 25 “failed” and 25 “successful”
- 14 columns of descriptive data - year aired, network, etc.
- Collect script for each pilot using web-scraping API “Selenium”
- Generate dataframe

Criteria for Chosen Programs

- All “Situational Comedies”
- Aired on American Cable Network
- Made in last 30 years
- Successful Sitcoms:
  - Rotten Tomatoes Score > 60%
  - IMDB Score > 7
  - Minimum of 3 Seasons
- Failed Sitcoms:
  - Rotten Tomatoes Score < 45%
  - IMDB Score < 6.5
  - Canceled after 1 season



A Closer look

- Average RT Score – Success: **88.93**
- Average RT Score – Failed: **15.33**

- Average IMDB Score – Success: **8.18**
- Average IMDB Score – Failed: **5.17**

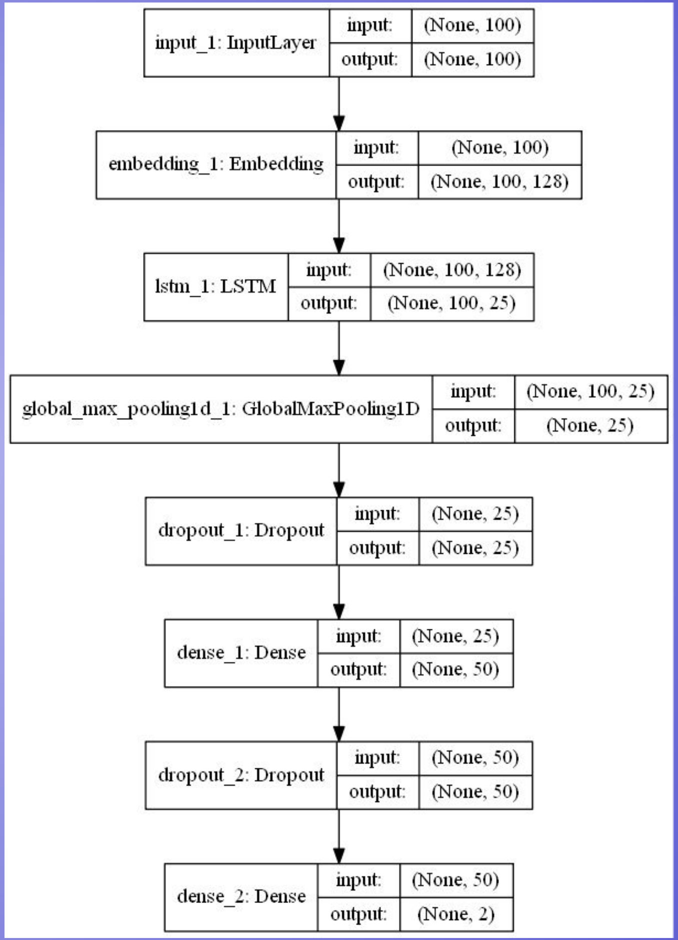
- Average # of Episodes – Success: **168.3**
- Average # of Episodes – Failed: **9.0**

Classification

- TF-IDF
    - Highest Test Score - 62%
    - Highest Train Score - 94%
- TF-IDF with LSA
- TF-IDF with Text Descriptive Features
- Text Descriptive Features Alone

Deep Learning

- LSTM Recurrent Neural Network
- Word embeddings
- K-folds cross evaluation
- Average accuracy of 80%



input_1: InputLayer — input: (None, 100) — output: (None, 100)

embedding_1: Embedding — input: (None, 100) — output: (None, 100, 128)

lstm_1: LSTM — input: (None, 100, 128) — output: (None, 100, 25)

global_max_pooling1d_1: GlobalMaxPooling1D — input: (None, 100, 25) — output: (None, 25)

dropout_1: Dropout — input: (None, 25) — output: (None, 25)

dense_1: Dense — input: (None, 25) — output: (None, 50)

dropout_2: Dropout — input: (None, 50) — output: (None, 50)

dense_2: Dense — input: (None, 50) — output: (None, 2)

LIMITATIONS

- ❑ Small sample size
- ❑ Uneven Data
- ❑ Availability of scripts
- ❑ Time

Future Work

- Data
  - Increase sample size
  - Expand criteria (include HBO/Netflix, older shows)
  - Add more descriptive features
- Technical
  - Data exploration with unused variables (actors, viewer data, etc.)
  - Additional feature engineering
  - Hyperparameter tuning classifiers
  - Hyperparameter tuning RNN