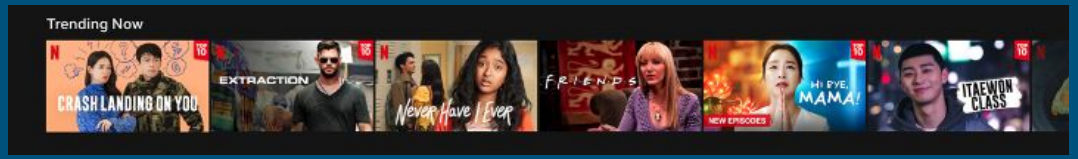
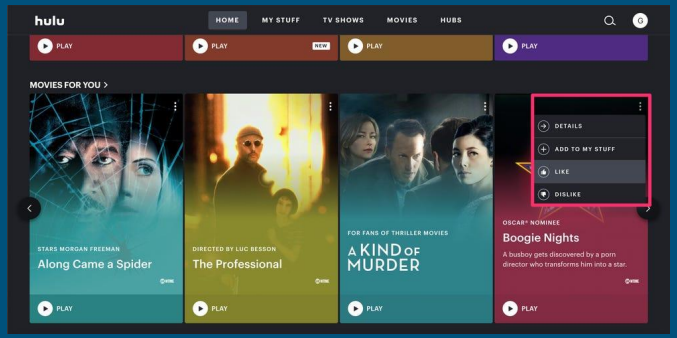
w207: Predicting Movie Performance

Ryan Brown, Trevor Dalton, Dimitrios Psaltos, Noah Prozan

Motivation

- Machine Learning is being increasingly used in the movie industry
  - **Streaming platforms** (ie. Netflix, HBO), attempt to maximize customer retention using targeted recommendations (SVD & RBM models)
  - **20th Century Fox,** predicted movie audience using movie trailers (neural network)
- Companies across industry are interested in data driven insights





Our Task

Movie merchandising: *Products/commodities based on a movie which are used to promote a film and act as a revenue stream.*

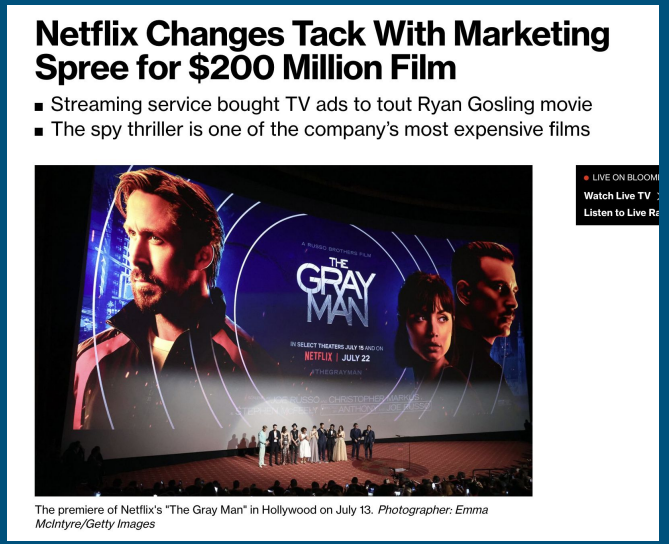
- Star Wars - $40B in merchandising revenue
- Harry Potter - $25B in merchandising revenue

Question: ***Using publicly available data, can we predict whether a movie will be well liked, prior to its release?***

Value Proposition: If a movie is expected to be well liked, studios can start marketing campaigns earlier and expand revenue opportunity

Our Reasoning:

1) Critics are 1st to see and rate a movie - we care about their opinions
2) Analyze movie data to identify features indicative of ratings
3) Build a ML model to predict a movie's rating based on these features



Netflix Changes Tack With Marketing Spree for $200 Million Film

- Streaming service bought TV ads to tout Ryan Gosling movie
- The spy thriller is one of the company's most expensive films

The premiere of Netflix's "The Gray Man" in Hollywood on July 13. *Photographer: Emma McIntyre/Getty Images*

Data - Rotten Tomatoes Movies Data Set

Rotten Tomatoes: One of the largest movie rating sites

History: Dataset scraped from Rotten Tomatoes website (up to 2020-10-31) and available on Kaggle

2 Datasets (movies and critics) - using movies

Movies Dataset: Movie level info (movie title, release date, description, genre, directors, actors, etc…)

- Size = 17712 x 23

Data Includes:

- **Movie Fixed Effects:** Runtime, content rating, release month (movie and streaming), production company, title, directors, actors, genres (ie. action & adventure, comedy, drama, romance (21 total))
- **Reviews**: Critics consensus - summary of multiple reviews
- **Target**: Tomatometer status - unique Rotten Tomato movie rating (Fresh or Rotten)

TOP GUN: MAVERICK

PG-13 2022, Action/Adventure, 2h 11m

CERTIFIED FRESH Rotten Tomatoes **97%**

TOMATOMETER
433 Reviews

99%

AUDIENCE SCORE
50,000+ Verified Ratings

JACK AND JILL

PG 2011, Comedy, 1h 30m

3%

TOMATOMETER
116 Reviews

36%

AUDIENCE SCORE
50,000+ Ratings

Link: https://www.kaggle.com/datasets/stefanoleone992/rotten-tomatoes-movies-and-critic-reviews-dataset

EDA and Feature Generation

Data Cleaning:

- Tomatometer Status (Fresh - 1/Rotten - 0)
- Only use movies after Rotten Tomatoes creation (1998)
- Rows with missing values were removed

Final Featureset:

- 33 features
- N = 6510 - (1 =54% | 0 =46%)

Derived Movie Features:

- Production Company sum (frequency)
- Genres sum (frequency)
- Delta Runtime (difference from average)
- Title length (n characters)
- Release date (month #) - movie and streaming release
- Content Rating (label encoder)
- Critic Consensus length (n characters)
- Director count and sum (frequency*)
- Actor count and sum (frequency*)
- Genres (one hot encoding)
- Critic Consensus (Text Embedding**)

* Actors and Directors in <2 movies were treated as 0s for sum feature, ** For BERT and LSTM models

	runtime difference from average	content rating	original release month	streaming release month	genres sums	production Comp sums	title length	critic consensus length	directors counts	actors counts	actors value	directors value
count	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0	6510.0
mean	13.0	3.9	6.7	6.8	3929.2	99.9	16.0	140.0	1.1	28.3	190.8	4.4
std	12.4	1.2	3.4	3.5	1880.8	109.5	9.7	39.6	0.7	22.1	151.6	4.5
min	0.0	0.0	1.0	1.0	49.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0
0.3	5.0	3.0	4.0	4.0	2240.0	10.0	10.0	115.0	1.0	11.0	65.0	2.0
0.5	10.0	4.0	7.0	7.0	3751.0	51.0	14.0	145.0	1.0	22.0	168.0	4.0
0.8	17.0	5.0	10.0	10.0	5288.0	188.0	19.0	166.0	1.0	40.0	282.0	6.0
max	166.0	5.0	12.0	12.0	10750.0	332.0	99.0	528.0	31.0	194.0	1177.0	109.0

EDA and Feature Generation

Data Cleaning:

- Tomatometer Status (Fresh - 1/Rotten - 0)
- Only use movies after Rotten Tomatoes creation (1998)
- Rows with missing values were removed

Final Featureset:

- 6510 movies x 33 features
- (1 =54% | 0 =46%)

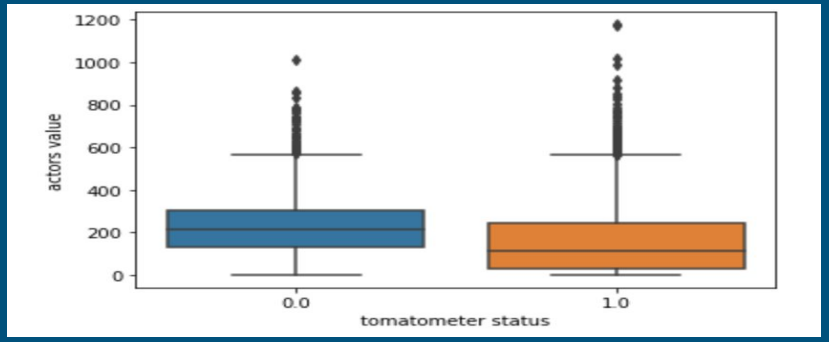
Feature	mean - 0	std - 0	mean - 1	std - 1
runtime_difference_from_average	11.23	9.97	14.45	13.98
content_rating	4.17	1.02	3.76	1.36
original_release_month	6.42	3.41	6.85	3.33
streaming_release_month	6.82	3.50	6.73	3.53
genres_sums	3961.57	1827.55	3902.19	1924.04
prodComp_sums	110.15	114.97	91.25	103.94
title_length	15.25	8.31	16.66	10.68
critic_consensus_length	134.14	41.55	144.82	37.26
directors_counts	1.12	0.72	1.15	0.76
actors_counts	32.19	22.12	25.08	21.53
actors_value	229.12	134.31	158.78	157.69
directors_value	4.38	4.37	4.40	4.69

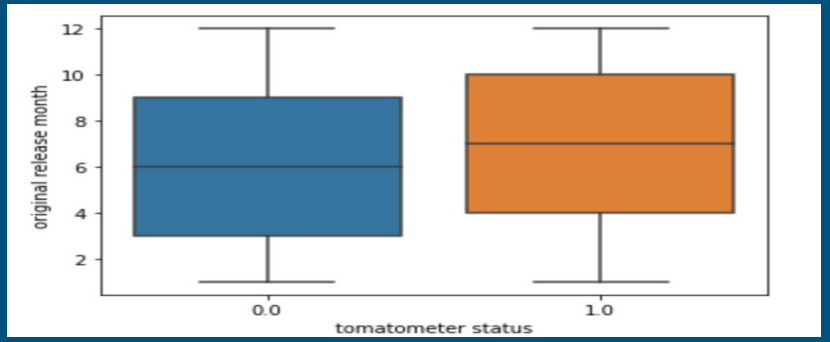
Derived Movie Features:

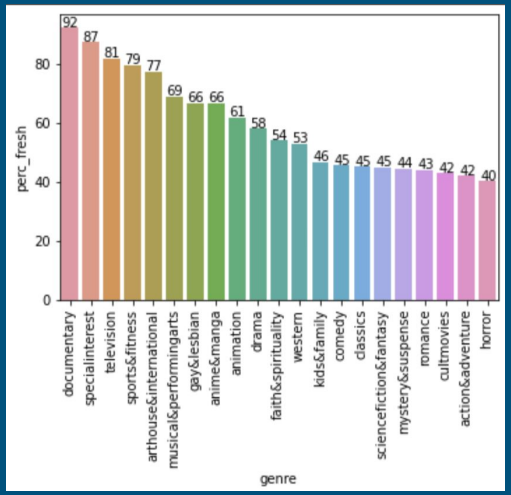
- Production Company sum (frequency)
- Delta Runtime (difference from average)
- Title length (n characters)
- Release date (month #) - movie and streaming release
- Content Rating (label encoder)
- Critic Consensus length (n characters)
- Director count and sum (frequency*)
- Actor count and sum (frequency*)
- Genres (one hot encoding)
- Critic Consensus (Text Embedding**)

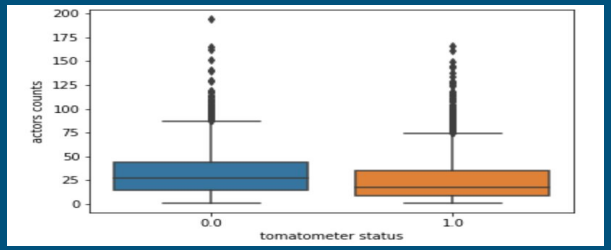
* Actors and Directors in <2 movies were treated as 0s for sum feature, ** For BERT and LSTM models

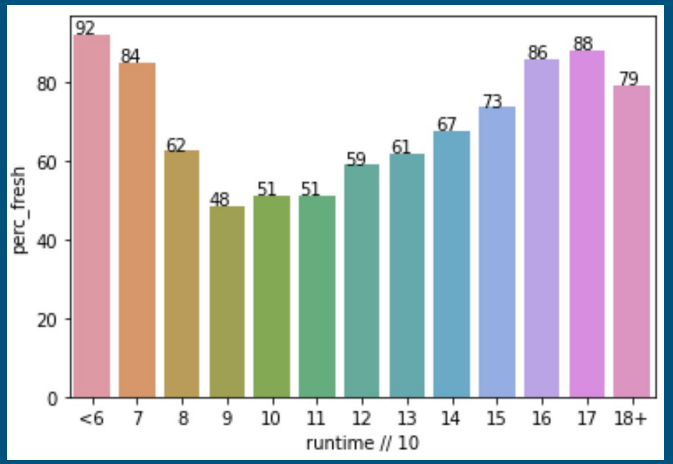
EDA/Relevant Statistics



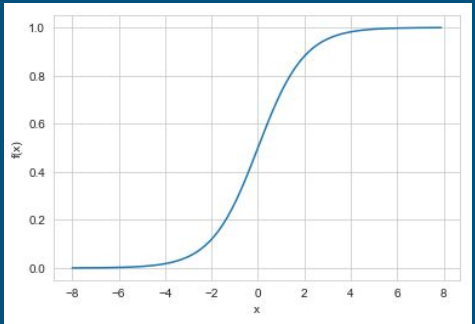








Approach



Model	Features	Accuracy (Test Set)
Baseline (Log Reg)	movie data	0.677
LSTM	movie data + critic review	0.861
BERT	critic review	0.904

Data preprocessing

- Train, val, test split: [60, 20, 20]
- SKLearn - Standard Scaler
- Test embedding for critic consensus

Rational behind approach:

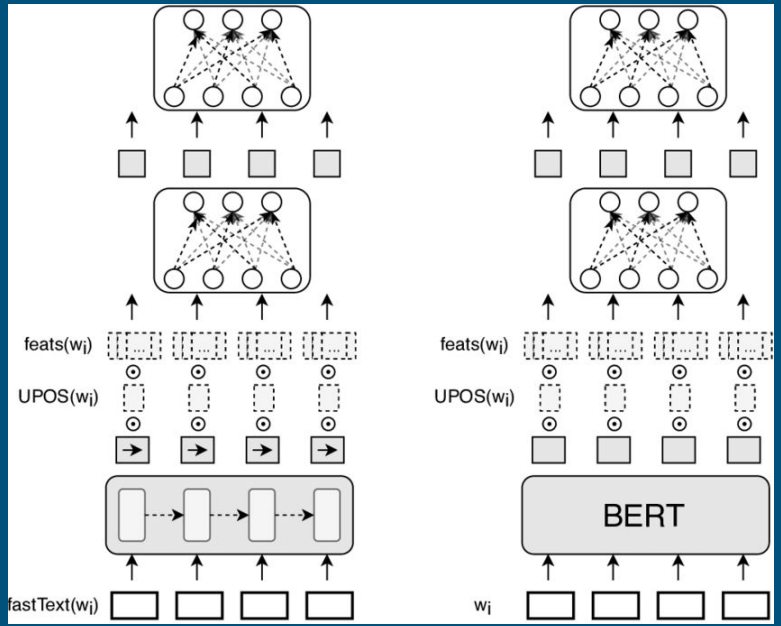
- Can just movie fixed effects be used?
- Can just a single review be used?
- Do we need a combination?

Models used:

- Baseline: Logistic Regression (Widely used for binary classification)
- LSTM (Allows for multiple feature types - review text + movie features)
- BERT (Currently best for text analysis)

Metrics:

- Accuracy and AUC



Baseline Model - Logistic Regression

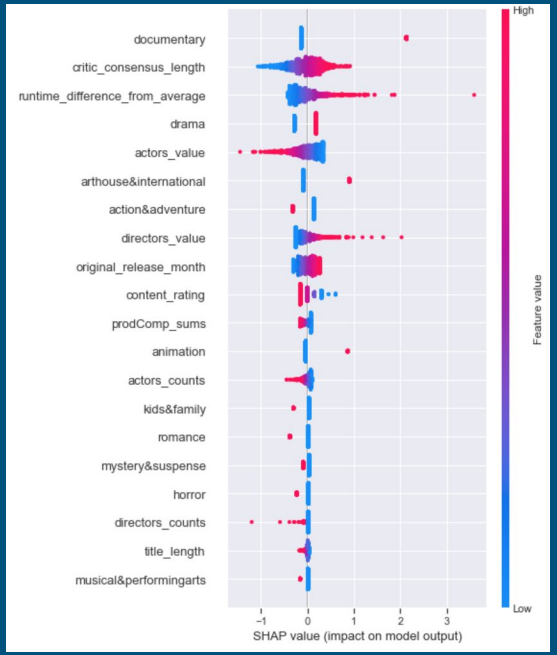
Features: Movie Fixed Effects - (33 features)

Implementation: SKlearn Linear Model - Logistic Regression

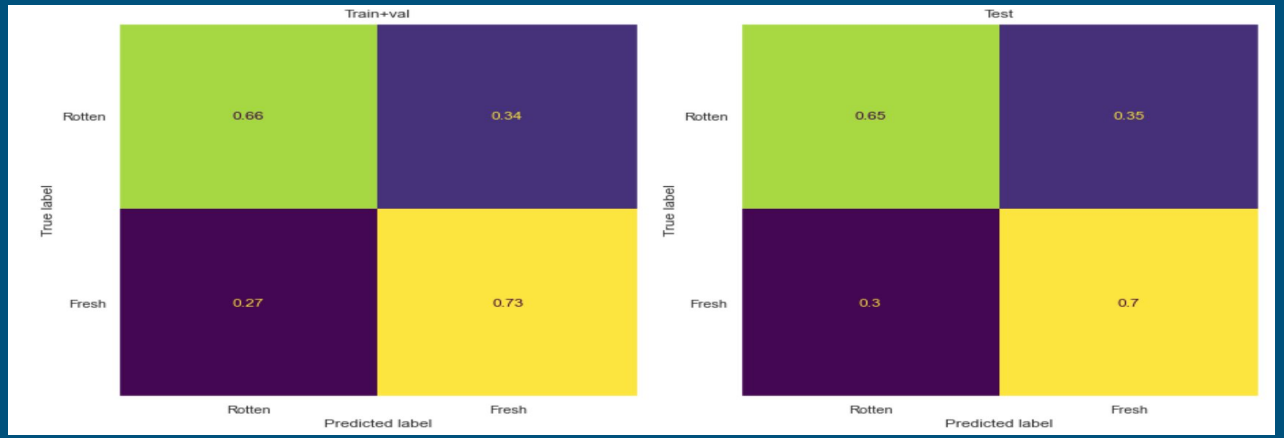
Hyperparameters: (C - regularization strength, Penalty)

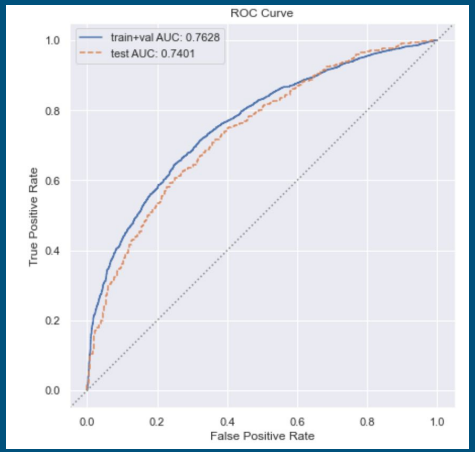
- GridSearchCV approach
- Tuned to validation dataset (predefined split)
- Best Hyperparameters: (C=0.234, penalty='l1', solver='liblinear')

	model	accuracy	train_auc	test_auc	train_AP	test_AP
0	Untuned (train | val)	0.688	0.7661	0.7444	0.8028	0.7816
1	Tuned (train | val)	0.689	0.7659	0.7439	0.8028	0.7809
2	Tuned (train + val | test)	0.677	0.7628	0.7401	0.7979	0.7674



Logistic Regression Results





LSTM w/ spaCy word embeddings

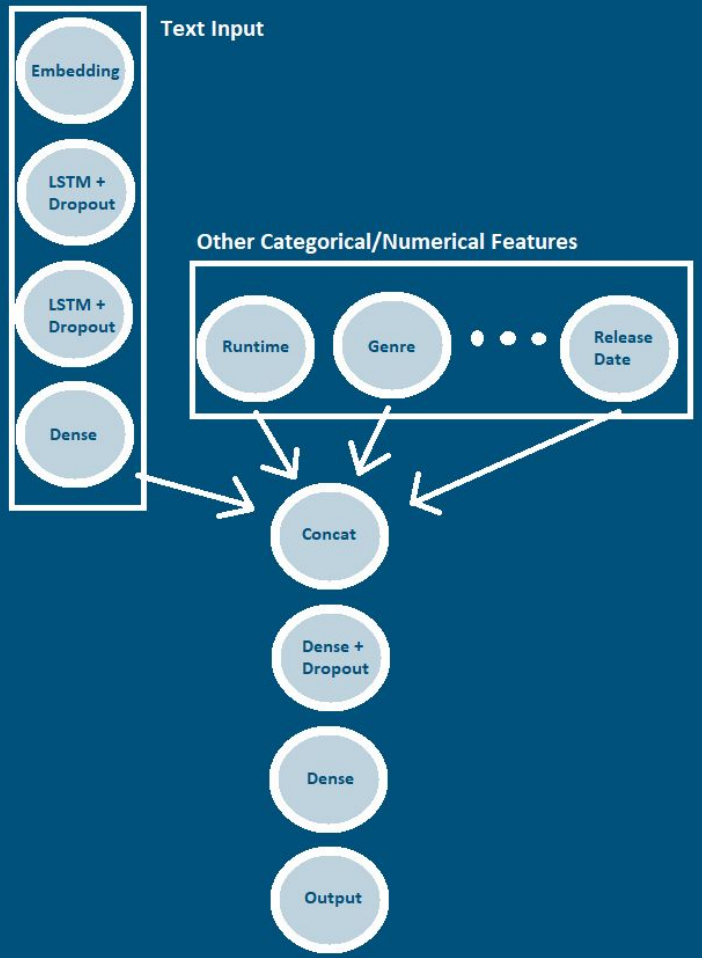
Data: Movie features + Critics Consensus - (34 features)

- Critics Consensus found to be far more valuable than the fixed features

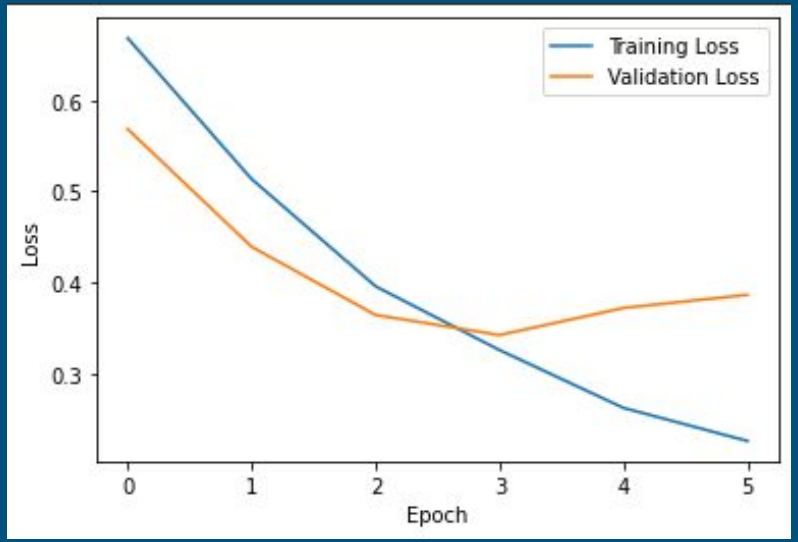
Implementation: LSTM

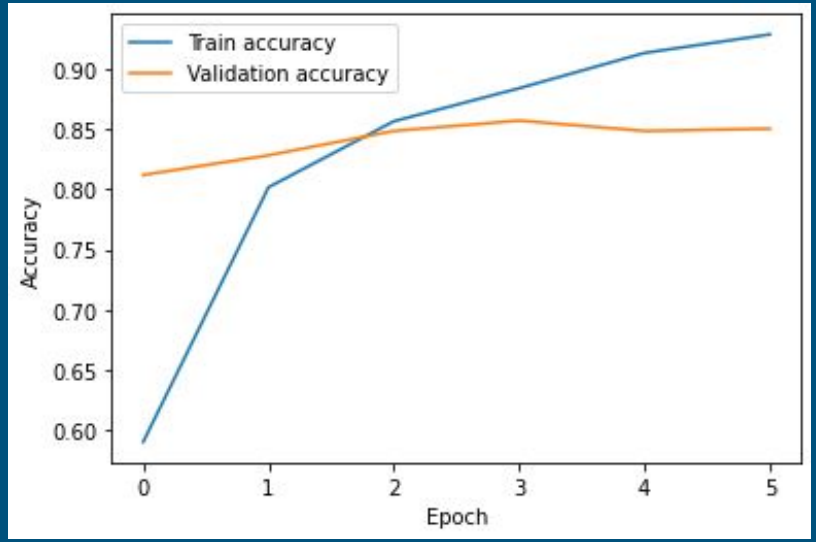
Hyperparameters: (LSTM units, dense units, dropout rate, etc.)

- Best hyperparameters found through the Keras Tuner HyperBand algorithm

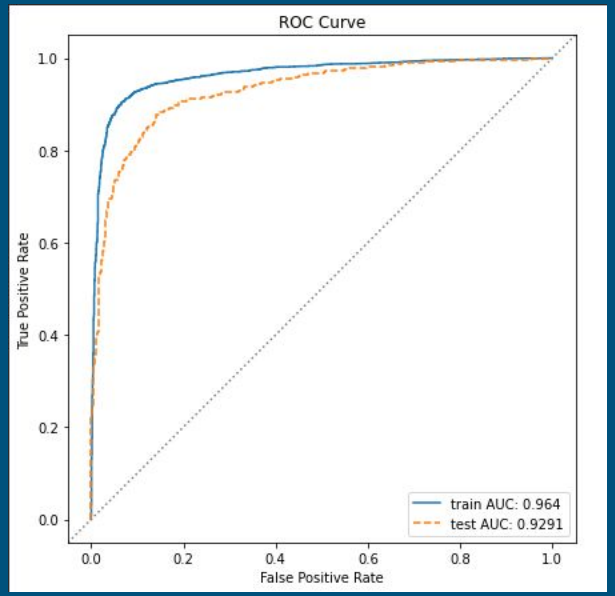


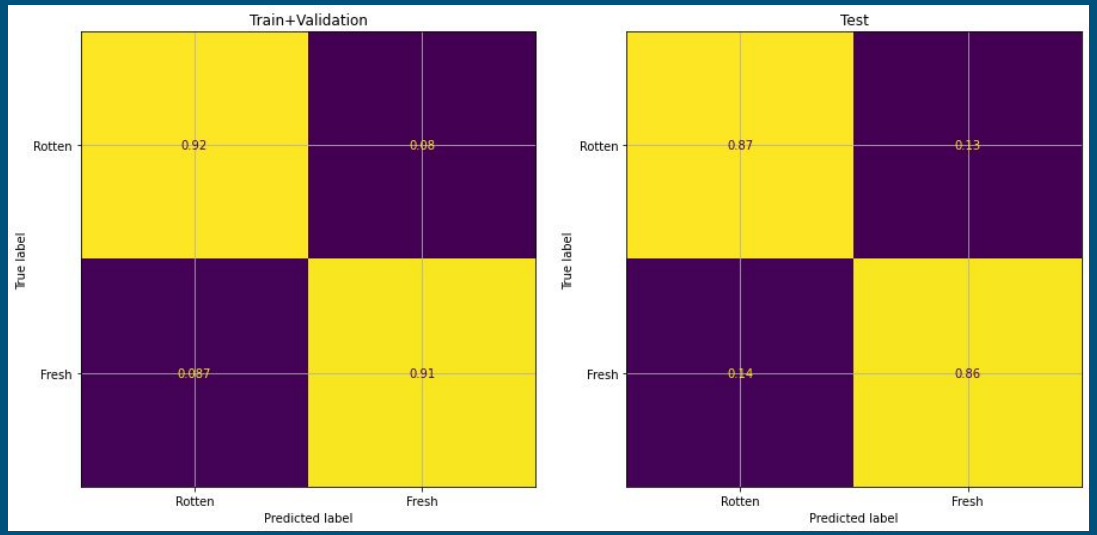
LSTM w/ spaCy embeddings Graphs



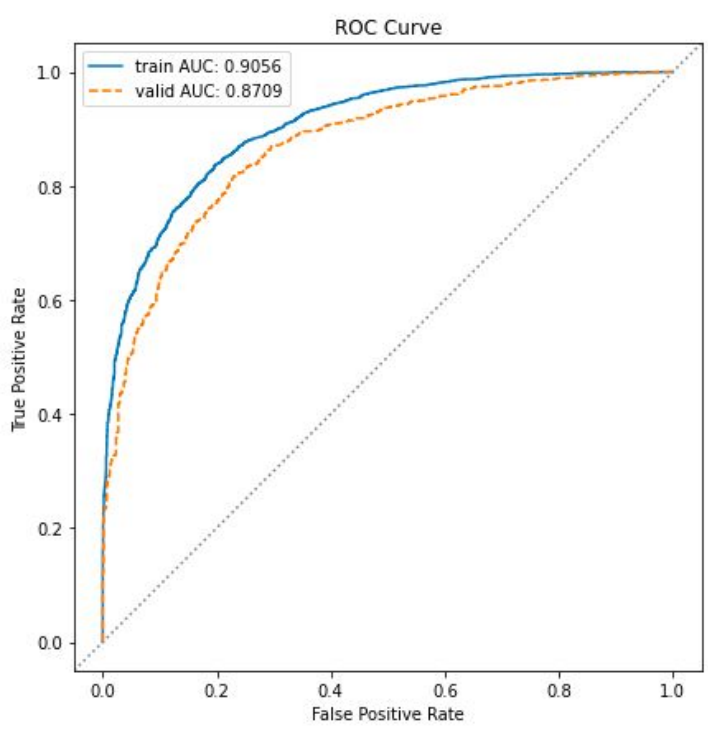


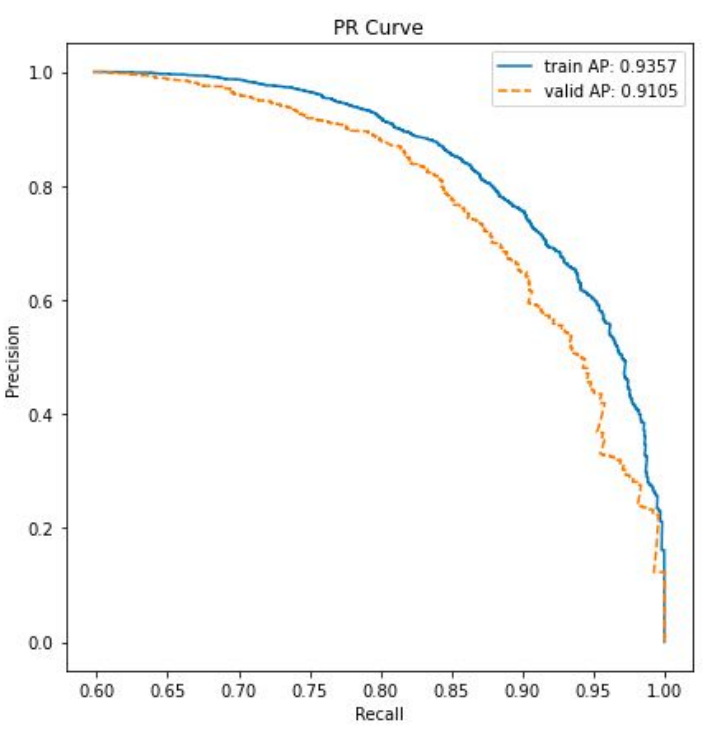
LSTM w/ spaCy embeddings Graphs



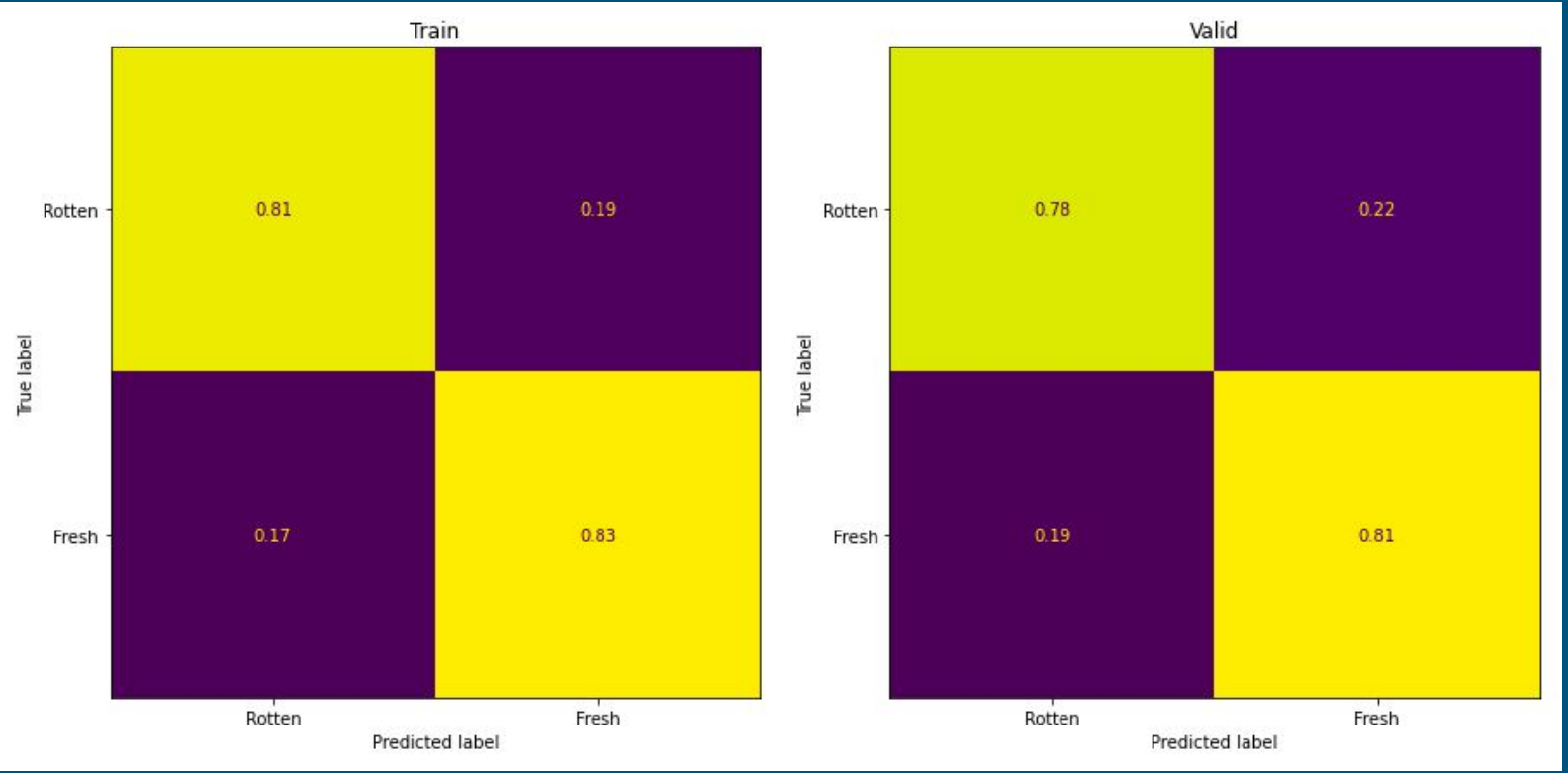


Revisiting critics consensus: Logistic + Ridge Regularization baseline on token counts

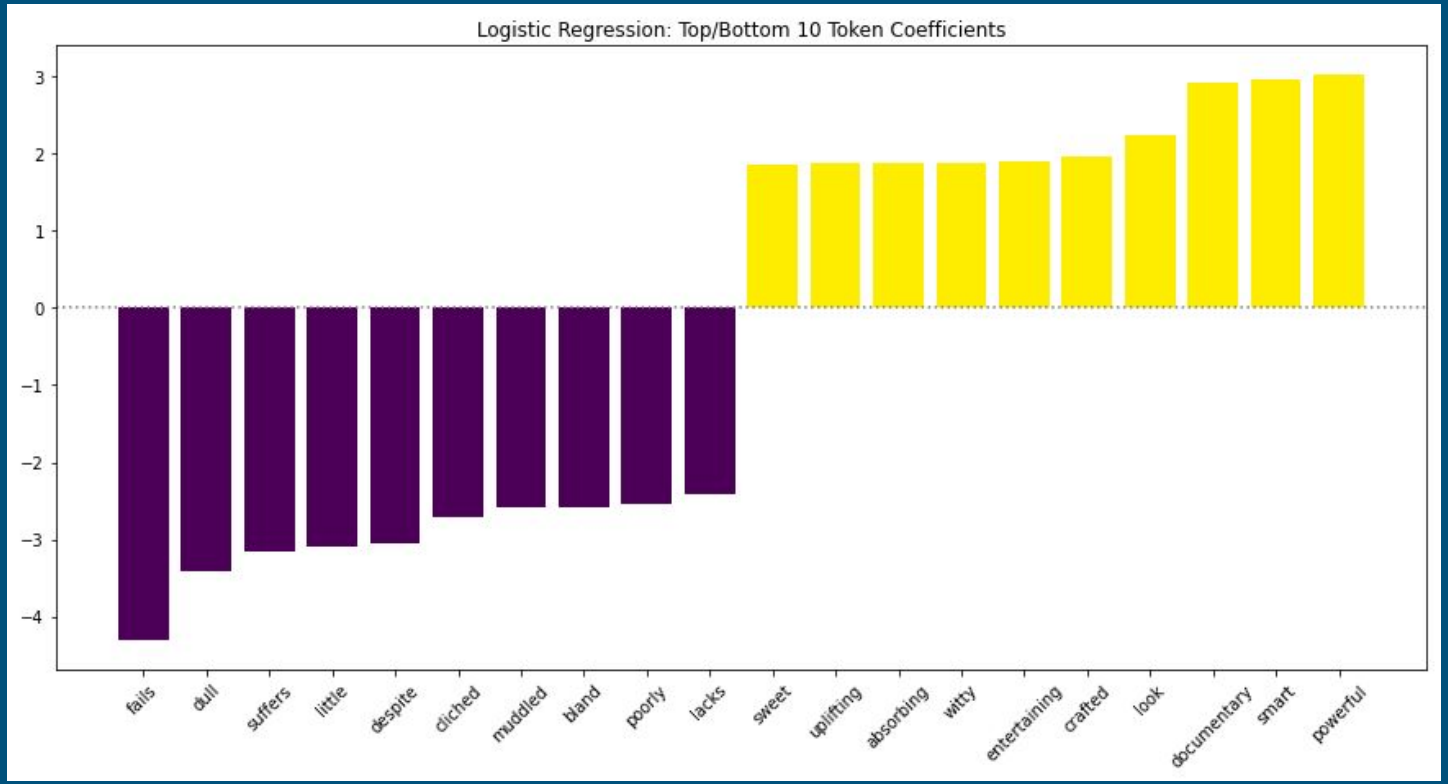




Logistic baseline continued



Model Explainability: Logistic Baseline

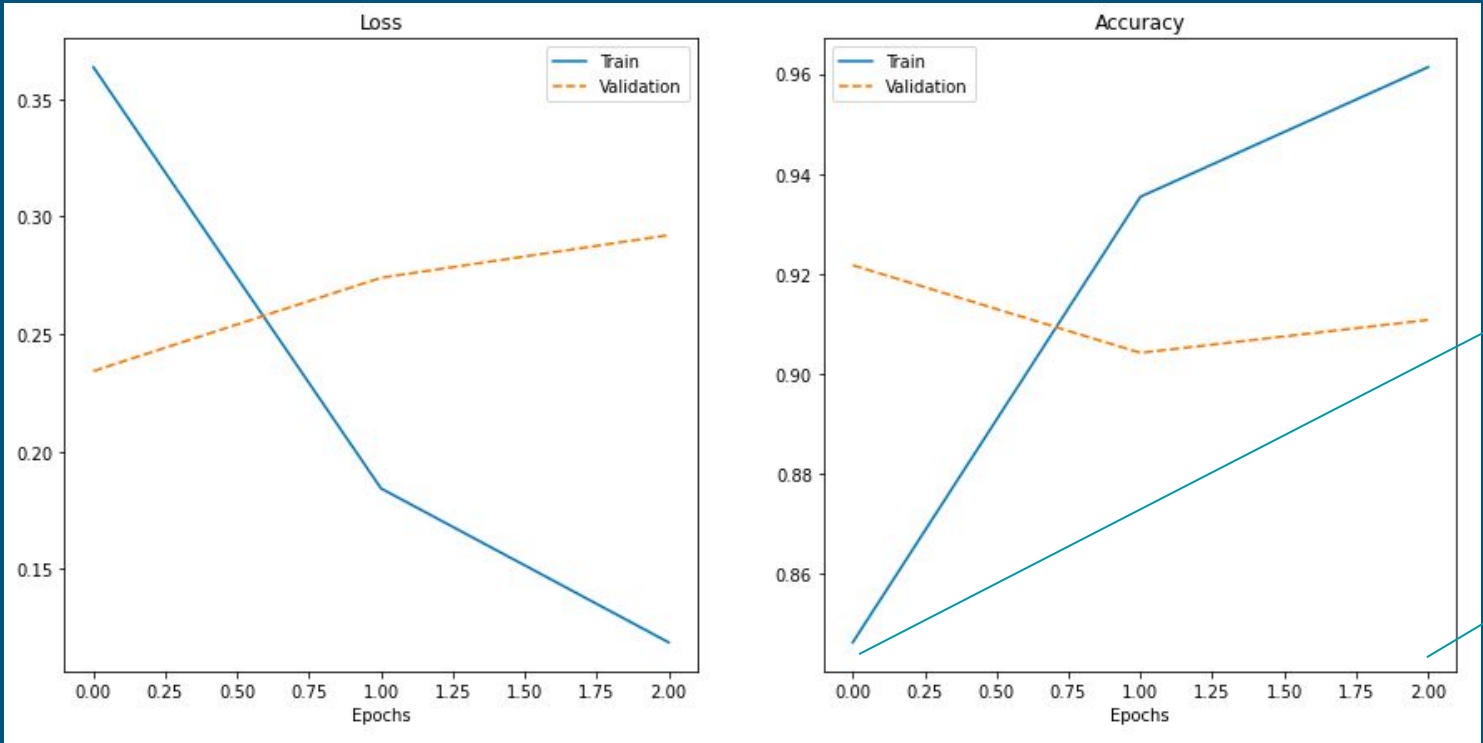


BERT on Critics Consensus: Hypertuning

trial id	max seq. len.	batch size	learning rate	init. epoch	epochs	train loss	valid loss	train accuracy	valid accuracy
0000	64	240	0.00001	0	2	0.413369	0.309384	0.866788	0.892720
0001	128	240	0.00003	0	2	0.235881	0.243045	0.919161	0.912425
0002	64	120	0.00002	0	2	0.445402	0.267980	0.810949	0.900930
0003	320	16	0.00003	0	2	0.298277	0.263688	0.879015	0.898741
0004	384	6	0.00001	0	2	0.167343	0.250750	0.943248	0.906951
0005	64	64	0.00002	0	2	0.196284	0.236402	0.931752	0.912972
0006	64	64	0.00002	2	3	0.410101	0.247831	0.806204	0.909688
0007	128	240	0.00003	2	3	0.542782	0.312955	0.754745	0.893268
0008	384	6	0.00001	2	3	0.311370	0.242066	0.874088	0.906951
0009	64	64	0.00002	3	5	0.195773	0.239055	0.932847	0.914614
0010	384	6	0.00001	3	5	0.167203	0.264696	0.943613	0.903667
0011	384	16	0.00003	0	3	0.157351	0.288256	0.947263	0.899836
0012	64	64	0.00005	0	3	0.332601	0.234468	0.859489	0.918993
0013	64	12	0.00001	0	3	0.320467	0.246024	0.873905	0.908593
0014	128	64	0.00003	0	3	0.189137	0.245320	0.933942	0.912972
0015	64	64	0.00005	3	5	0.329265	0.229368	0.853832	0.916256
0016	128	64	0.00003	3	5	0.373965	0.235005	0.847080	0.919540
0017	384	12	0.00002	0	5	0.164985	0.272897	0.945438	0.894910
0018	256	240	0.00005	0	5	0.209245	0.236770	0.928650	0.912425
0019	320	64	0.00005	0	5	0.332467	0.232741	0.860949	0.915161
0020	512	14	0.00005	0	5	0.092615	0.385854	0.967336	0.889436

Trials 14/16 selected:
{'learning_rate': 3e-05,
'max_length': 128,
'batch_size': 64}

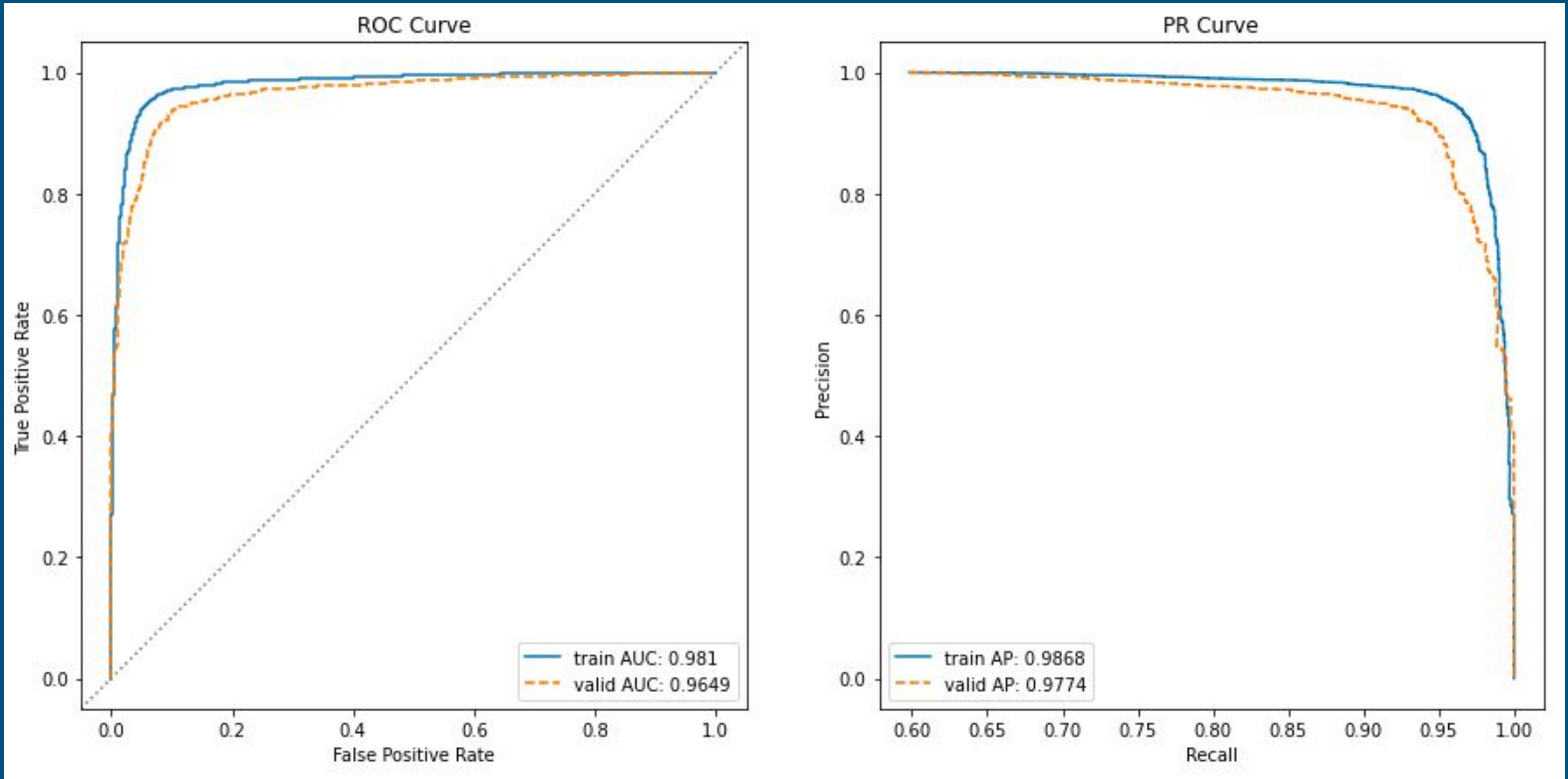
Retrain on hyperparameters over 5 epochs



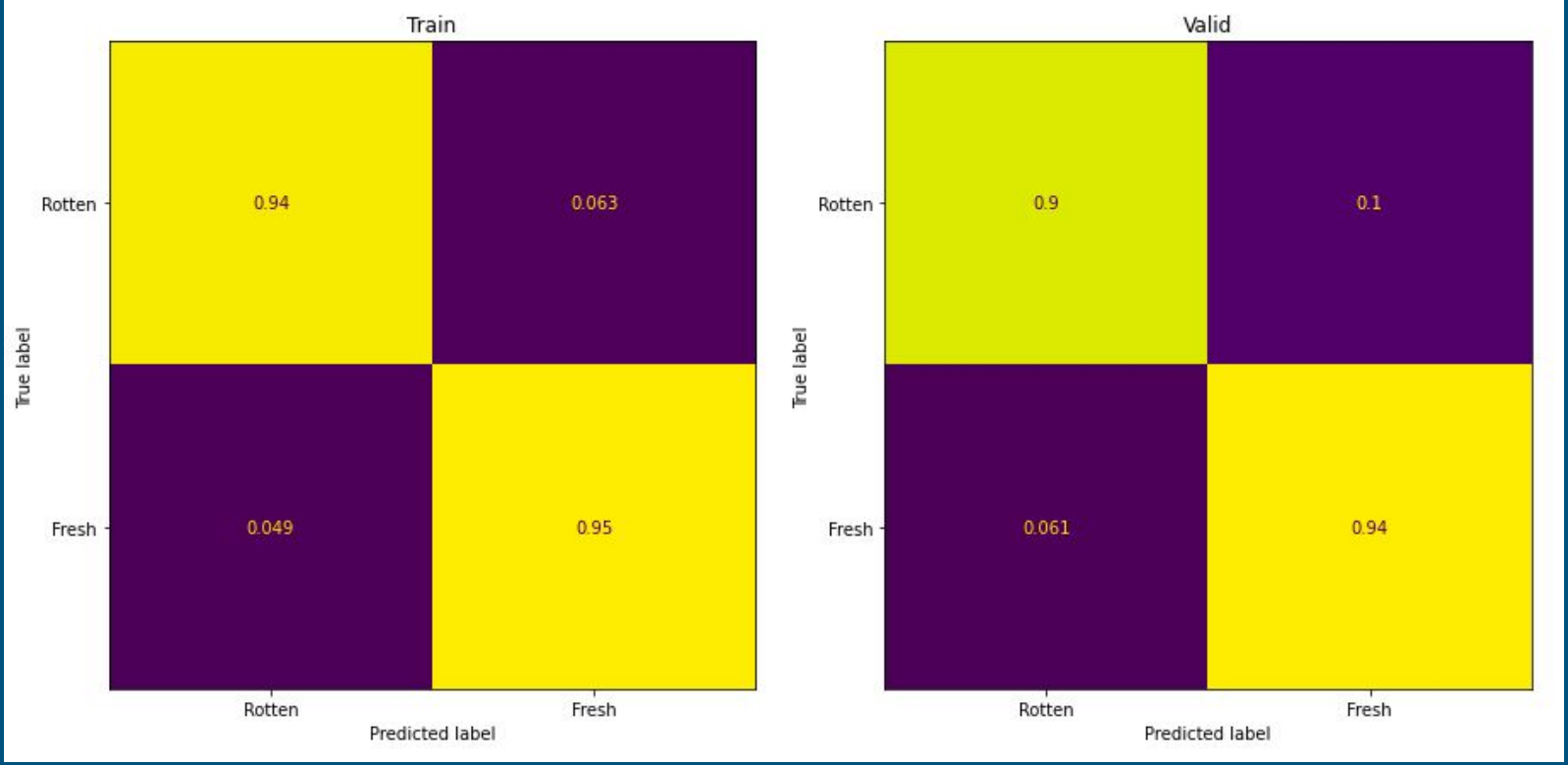
Best perf. After single epoch

Early stop at epoch 3

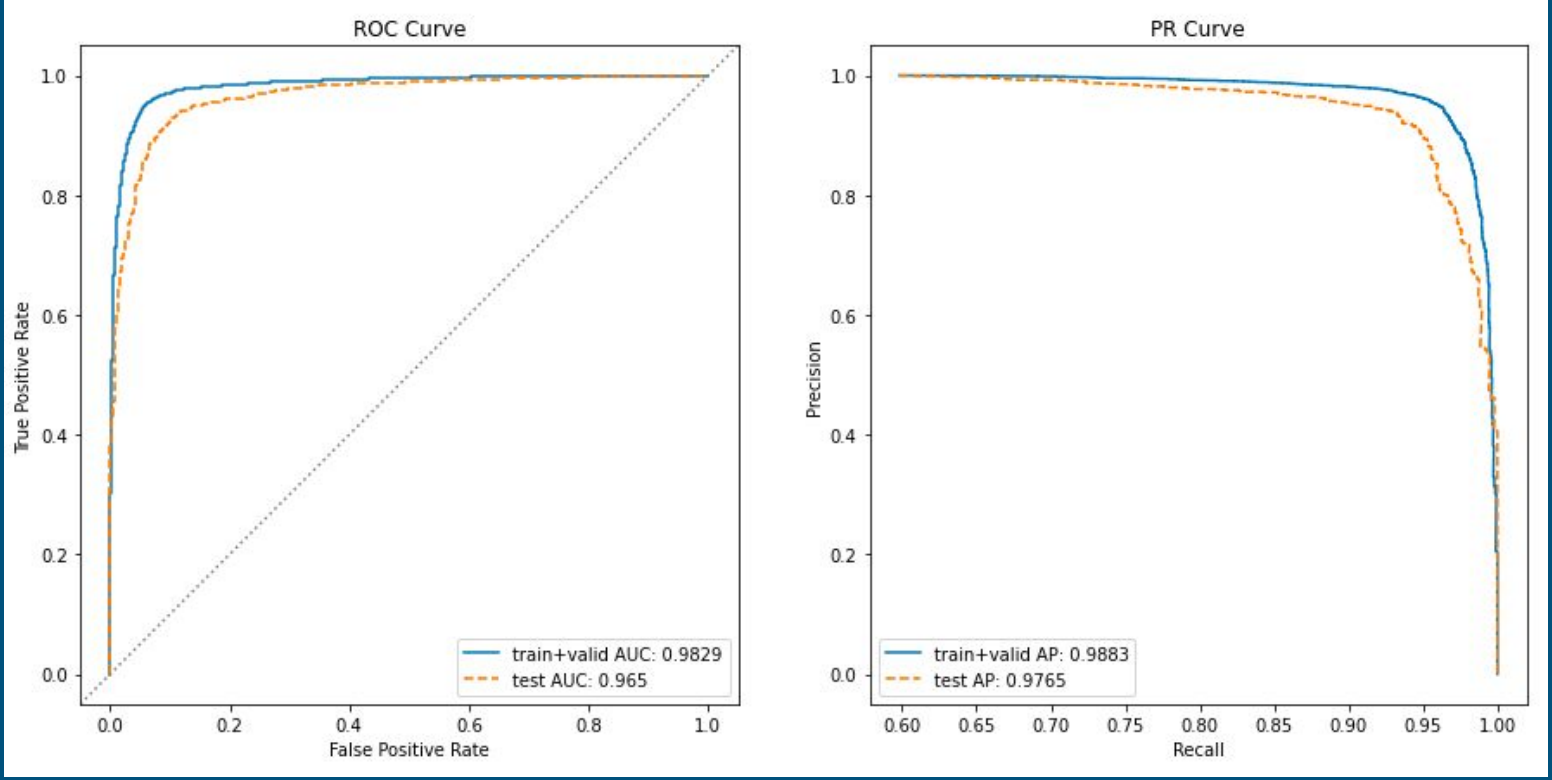
Hypertuned BERT AUC



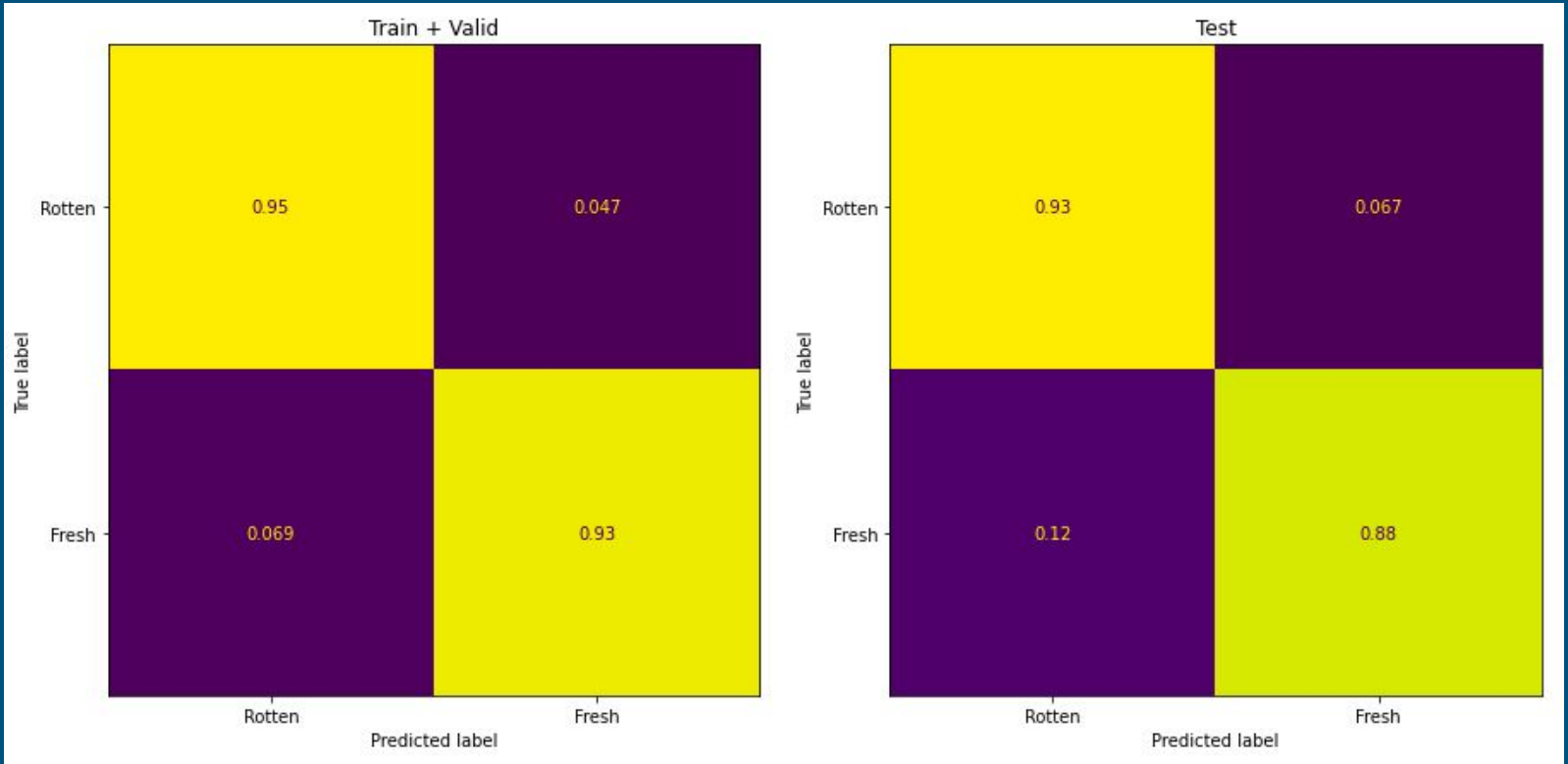
Hypertuned BERT Continued



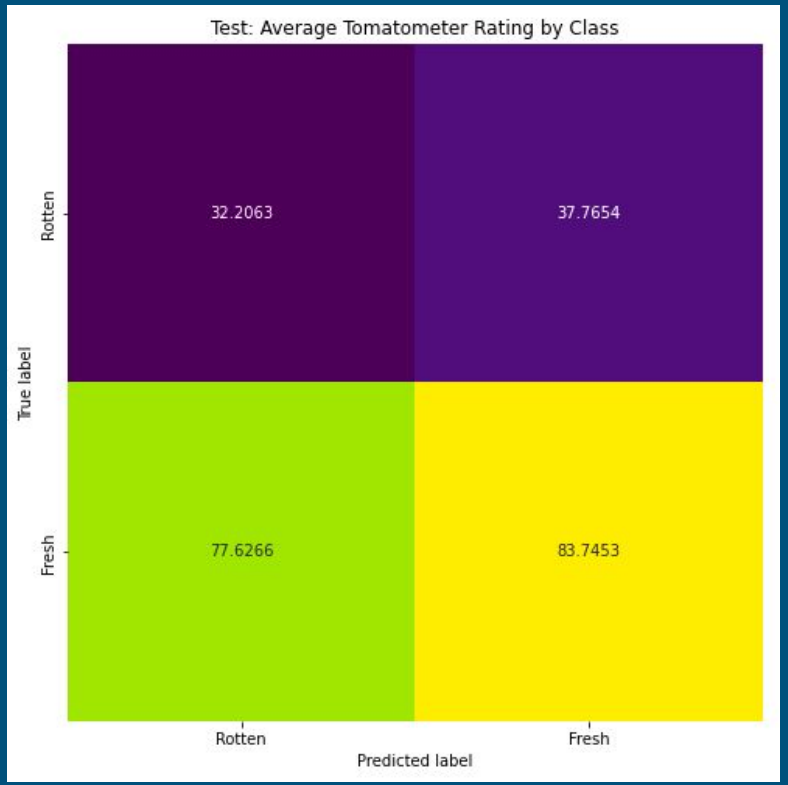
Retrain over One Epoch on Train + Valid



Final BERT Performance Continued



Exploring Association with Continuous Response



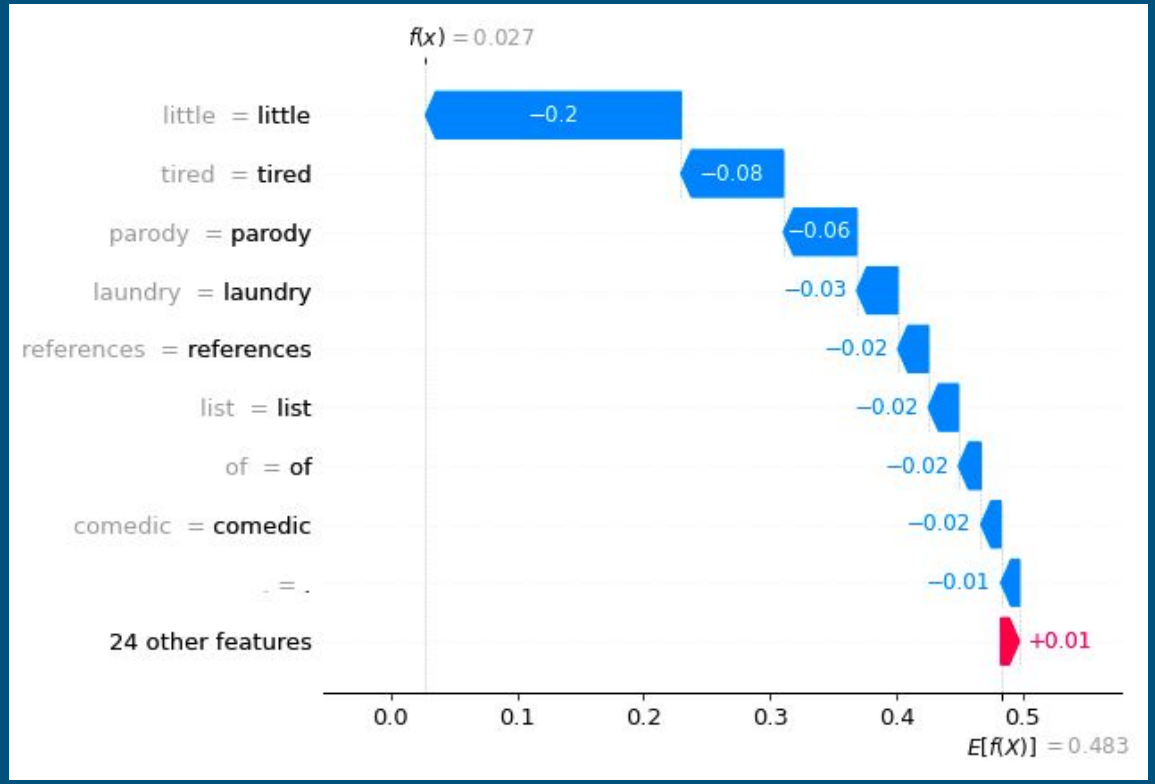
Clear association with higher/lower ratings for incorrect classifications

Comparing Performance

Run ID	Algorithm	Description	Train Loss	Valid Loss	Train Accuracy	Valid Accuracy
00	Logistic regression	L2 regularization	0.4195	0.4593	0.8197	0.7947
01	DistilBERT	baseline w/o finetuning	0.6880	0.6878	0.6011	0.6015
02	DistilBERT	Hyperparameter Tuned, early stopping	0.1741	0.2342	0.9453	0.9217

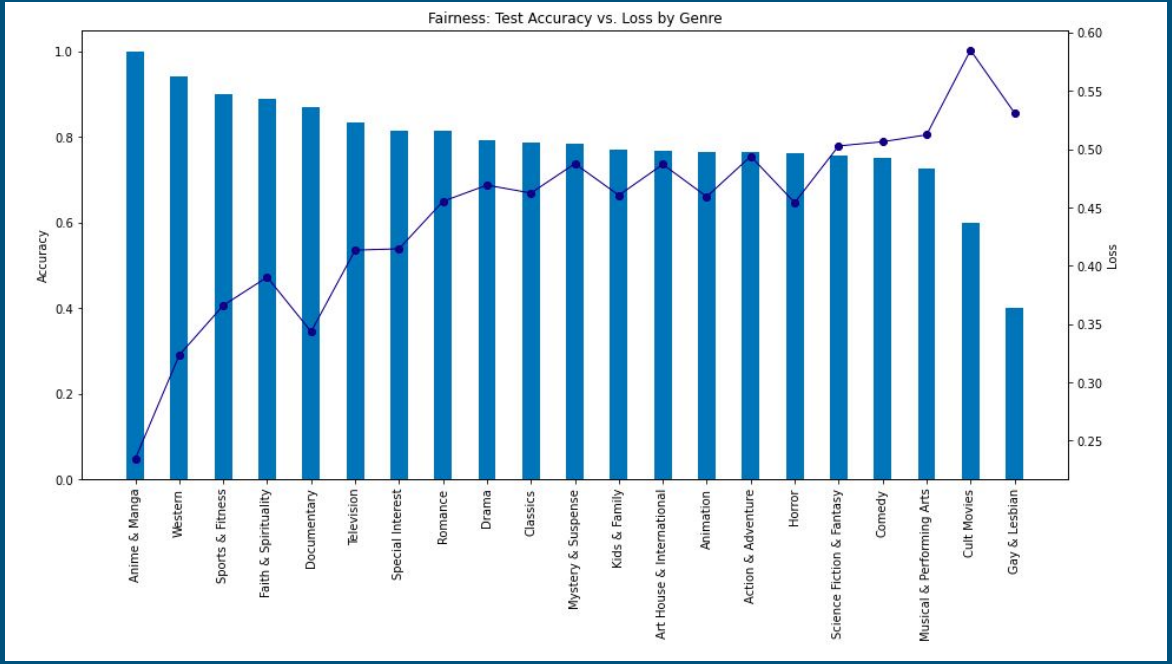
Run ID	Algorithm	Description	Train+Valid Loss	Test Loss	Train+Valid Accuracy	Test Accuracy
03	DistilBERT	hypertuned, retrained on train+valid	0.1832	0.262	0.9399	0.9037

Model Explainability: Final BERT SHAP



“Loaded Weapon 1 hits all the routine targets with soft squibs, yielding a tired parody that cycles through its laundry list of references with little comedic verve.”

Fairness



genre	n	loss	accuracy
Action & Adventure	408	0.4938	0.7647
Animation	85	0.4592	0.7647
Anime & Manga	3	0.2344	1.0000
Art House & International	227	0.4869	0.7665
Classics	94	0.4623	0.7872
Comedy	607	0.5063	0.7496
Cult Movies	10	0.5848	0.6000
Documentary	168	0.3434	0.8690
Drama	999	0.4691	0.7908
Faith & Spirituality	9	0.3902	0.8889
Gay & Lesbian	5	0.5311	0.4000
Horror	185	0.4541	0.7622
Kids & Family	153	0.4604	0.7712
Musical & Performing Arts	95	0.5122	0.7263
Mystery & Suspense	387	0.4875	0.7829
Romance	214	0.4552	0.8131
Science Fiction & Fantasy	213	0.5027	0.7559
Special Interest	108	0.4145	0.8148
Sports & Fitness	20	0.3659	0.9000
Television	12	0.4133	0.8333
Western	17	0.3233	0.9412

Conclusions

Key Results:

- ***Using publicly available data, we can predict whether a movie will be well liked, prior to its release with roughly 90% accuracy***
- DistilBERT is a powerful model and just using one review can predict how accurate a movie is
- Review Level data is more accurate than fixed effect data
- Feature engineering improved models slightly
- Our models suffered from a fairness problem, particularly in genres

Moving forward:

- How can we collect a corpus of reviews to make movie level predictions more accurate?
- How do we communicate to key stakeholders about our movie prediction?

Model	Features	Accuracy (Test Set)
Baseline (Log Reg)	movie data	0.677
LSTM	movie data + critic review	0.861
BERT	critic review	0.904

Contributions/Primary Areas of Focus

	Ryan	Dimitri	Trevor	Noah
Theoretical/Market Research	X	X	X	X
EDA	X	X	X	X
Data Cleaning	X	X	X	X
Feature Engineering	X	X	X	X
Model Making	DistilBert	LogReg	LTSM	Model Review
Presentation Slides	X	X	X	X

Link to Code Repo

- https://github.com/trevorjd3141/w207-movie-project

References

- https://www.rottentomatoes.com/
- https://www.kaggle.com/datasets/stefanoleone992/rotten-tomatoes-movies-and-critic-reviews-dataset
- https://en.wikipedia.org/wiki/Scikit-learn
- https://towardsdatascience.com/logistic-regression-ca2d070a3eee
- https://www.bloomberg.com/news/articles/2022-07-15/netflix-changes-tack-with-marketing-spree-for-200-million-film
- https://cloud.google.com/blog/products/ai-machine-learning/how-20th-century-fox-uses-ml-to-predict-a-movie-audience
- https://medium.com/ds3ucsd/data-science-in-the-film-industry-part-2-movie-trailers-and-artificial-intelligence-64b3cbd267c1
- https://www.researchgate.net/figure/The-baseline-LSTM-based-left-and-BERT-based-right-models-for-the-NER-task-along-with_fig1_346373540
- https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc
- https://filmlifestyle.com/best-sites-for-rating-movies/
- https://www.ciiblog.in/movie-merchandising-is-a-potential-hit/
- https://www.openpr.com/news/1631327/movie-merchandise-market-analysis-strategic-assessment-and-trend-outlook-by-sony-pictures-paramount-pictures-warner-bros-huayi-brothers-enlight-media-lionsgate-films-nbc-universal-nickelodeon-toei-company-alpha-group-the-walt-disney-company-a.html