GA Data Science Capstone Project: Predicting Attraction Popularity at Walt Disney World

Christopher Doughty

October 14, 2021

PROBLEM

- The Disney Parks branch of the Walt Disney Company wants to improve the guest experience by “imagineering” the best ride attractions possible – those that will have the most appeal to the broadest audience
- Accordingly, it is seeking to design a recommendation tool that will facilitate smoother park experiences from among existing attractions, as well as determine what types of attractions to develop for future projects

PROBLEM

Questions:

- Are there certain attributes that make a ride at Walt Disney World more or less appealing?
- Could we make an algorithm that would predict the popularity of a ride?

Objectives:

- Determine if there are any ride attributes that are likely to make an attraction more or less appealing.
- Create a model that can accurately predict the popularity (rating) of a ride.

DATA SET

- Obtained Walt Disney World Ride Data from data.world: https://data.world/lynne588/walt-disney-world-ride-data

	Ride	Park_location	Park_area	Ride_type_all	Ride_type_thrill	Ride_type_spinning	Ride_type_slow	Ride_type_small_drops	Ride_type_big_drops	Ride_type_dark	...	Age_interest_teens	Age_interest_adults	Height_req_inches	Ride_duration_min	Open_date	Age_of_ride_days	Age_of_ride_years	Age_of_ride_total	TL_rank	TA_Stars
0	Alien Swirling Saucers	HS	Toy Story Land	spinning	No	Yes	No	No	No	No	...	Yes	Yes	32	1.5	2018-06-30	1197	3.277207	3 years 3 months 11 days	31.0	NaN
1	Astro Orbiter	MK	Tomorrowland	spinning, slow	No	Yes	Yes	No	No	No	...	Yes	Yes	0	1.5	1995-02-25	9723	26.620123	26 years 7 months 14 days	43.0	3.5
2	Avatar Flight of Passage	AK	Pandora	thrill	Yes	No	No	No	No	No	...	Yes	Yes	44	5.0	2017-05-27	1596	4.369610	4 years 4 months 14 days	9.0	5.0

- Contains features such as park location, ride type, duration, ride age, height req., and age interest group for each ride
- Pre-cleaning: 46 rows x 28 columns | Post: 45 rows x 24 columns
- Deleted 1 row (no review page), updated review data (data was originally compiled on October 23, 2019), renamed/lowercased columns, changed data types (string to Boolean), deleted 4 columns (irrelevant info, duplicated info, high correlations)

DATA SET

- Scraped individual review data from TripAdvisor for each of the attractions in the WDW Ride Data (through Oct. 9, 2021): https://www.tripadvisor.com/Attractions-g34515-Activities-a_allAttractions.true-Orlando_Florida.html

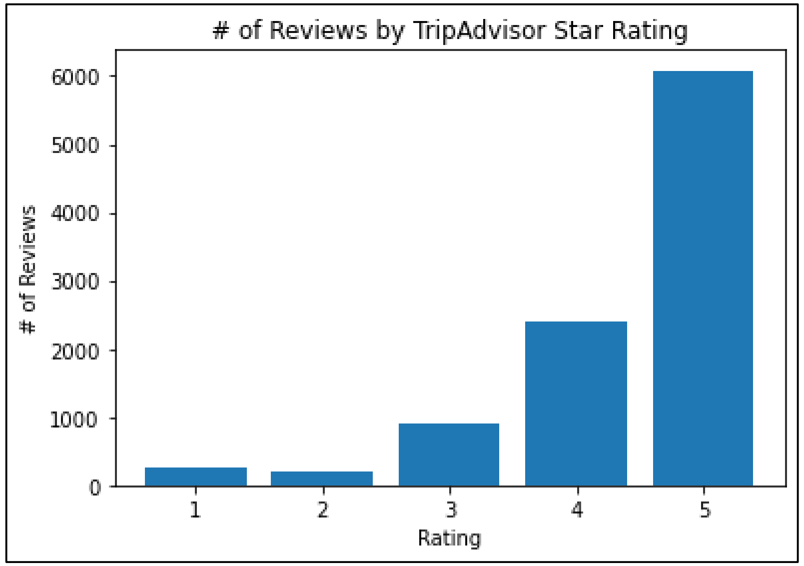
	ride	reviewer	reviewer_visit_group_type	review_date	review_title	review_text	rating
0	Astro Orbiter	Courtenay O	NaN	2020-12-20	Short & sweet ride to space!	This ride brings me closer to the sky than any...	5.0
1	Astro Orbiter	Love2Travel100	Family	2020-02-20	Fun both day and night	Recommend riding both in the day and at night ...	4.0

- Contains features like review date, review title, review text, and rating for each ride; initial cleaning in Excel, secondary in Jupyter
- Pre-cleaning: 9,843 rows x 7 columns | Post: 9,843 rows x 5 columns
- Dropped features not prepared to use for this analysis (nulls requiring imputation, time series data)

ANALYSIS

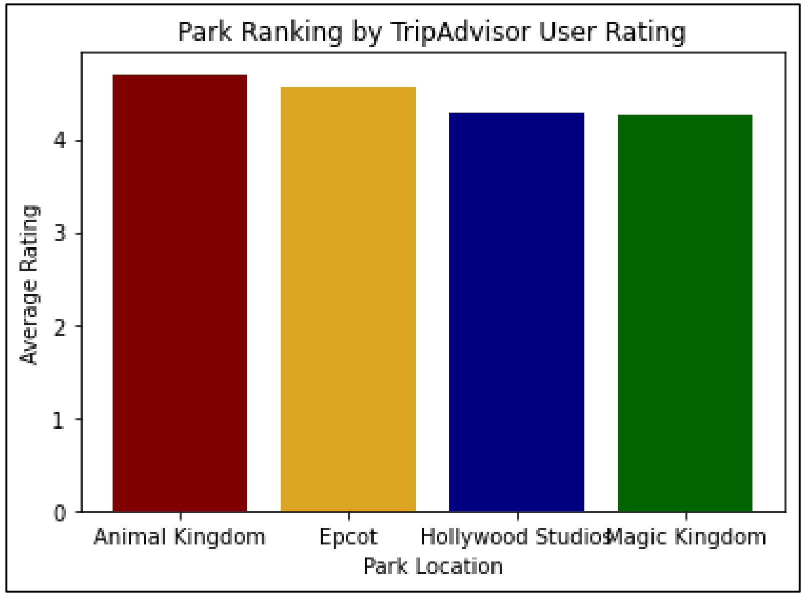
- Joined the data sets together (9,843 rows x 28 columns)
- Data set is very imbalanced – way more 4- and 5-star reviews than 1-3 star reviews

Rating	%	#
5	62%	6,081
4	24%	2,393
3	9%	902
2	2%	212
1	3%	255



ANALYSIS

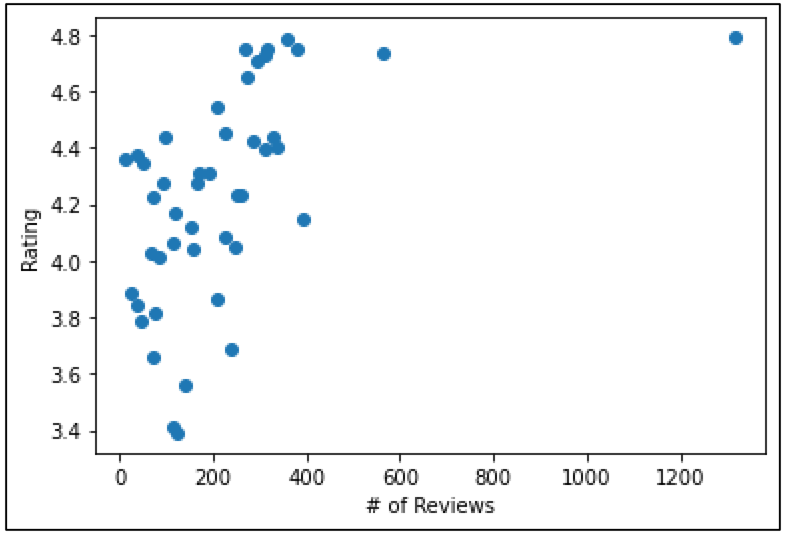
- Each park is very similar in terms of rating distributions
- Magic Kingdom has more reviews than the other parks, but it also has more rides – the number of reviews generally aligns with the number of rides



	min_rating	max_rating	mean_rating	median_rating	average_rating	review_count
park_location						
Magic Kingdom	1.0	5.0	4.265196	5.0	4.276716	4080
Animal Kingdom	1.0	5.0	4.569139	5.0	4.696236	2683
Epcot	1.0	5.0	4.283798	5.0	4.132511	1864
Hollywood Studios	1.0	5.0	4.700658	5.0	4.768092	1216

ANALYSIS

- Avatar: Flight of Passage has way more reviews than any other ride, but its rating distributions are reflective of the overall data set (mostly positive reviews)

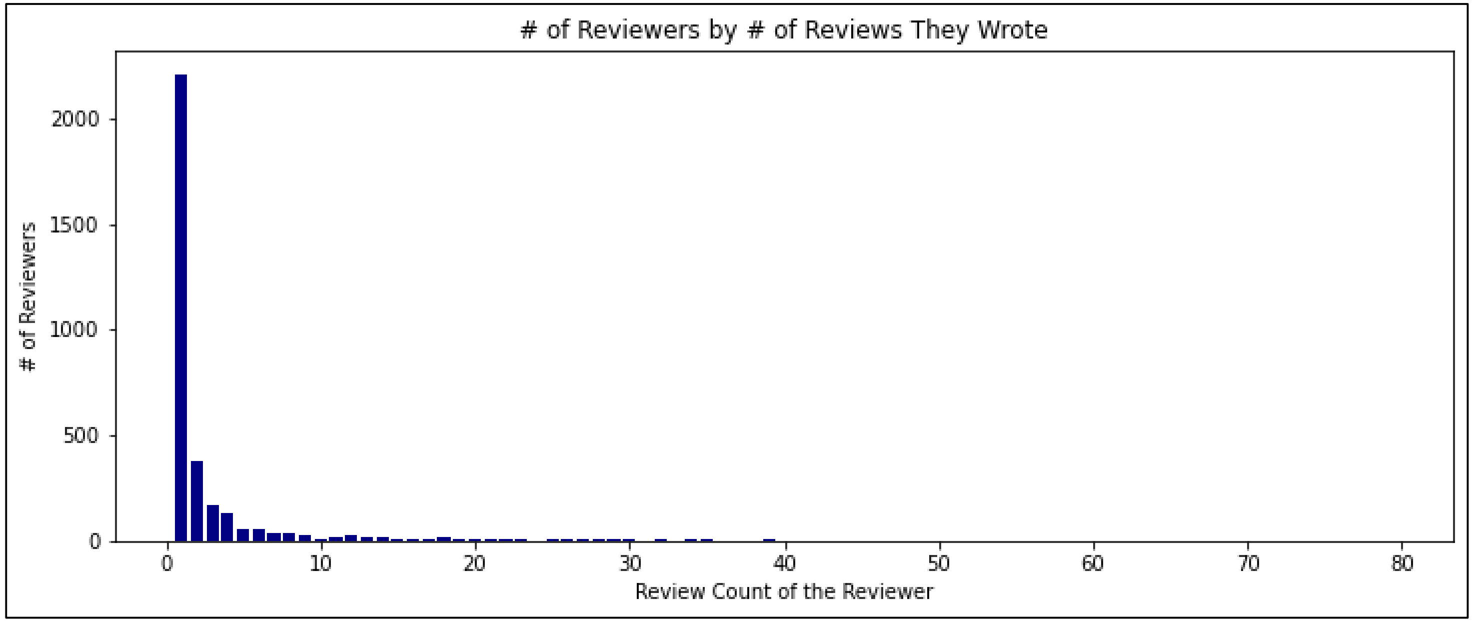


ride	min_rating	max_rating	mean_rating	median_rating	average_rating	review_count
Avatar Flight of Passage	1.0	5.0	4.790274	5.0	5.0	1316
Soarin'	1.0	5.0	4.734982	5.0	4.5	566
Seven Dwarfs Mine Train	1.0	5.0	4.150895	5.0	4.5	391
The Twilight Zone Tower of Terror	1.0	5.0	4.748031	5.0	5.0	381
Expedition Everest	1.0	5.0	4.787115	5.0	5.0	357
Haunted Mansion	1.0	5.0	4.402367	5.0	4.5	338
Space Mountain	1.0	5.0	4.440729	5.0	4.5	329
Toy Story Midway Mania	1.0	5.0	4.748408	5.0	4.5	314
Pirates of the Caribbean	1.0	5.0	4.395498	5.0	4.5	311
Kilimanjaro Safaris	1.0	5.0	4.725806	5.0	4.5	310

ANALYSIS

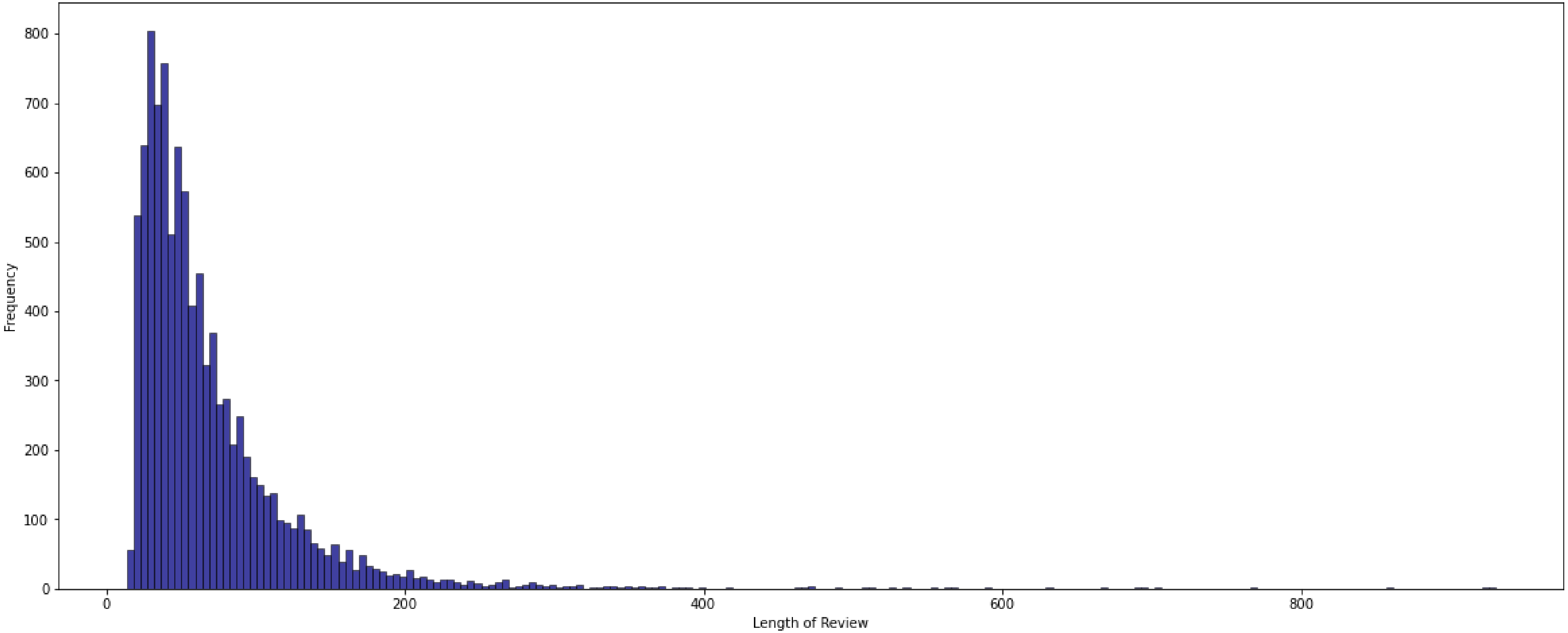
- Average reviewer leaves 3 reviews or less
- Mean is 3, median is 1, max is 79
- Top 200 of the 3,226 reviewers (6%) left 4,171 of the 9,843 total reviews (42%)

review_count	reviewer_count
1	2211
2	382
3	167
4	127
5	57
6	55
8	40
7	34
9	28
12	23
13	19
11	18



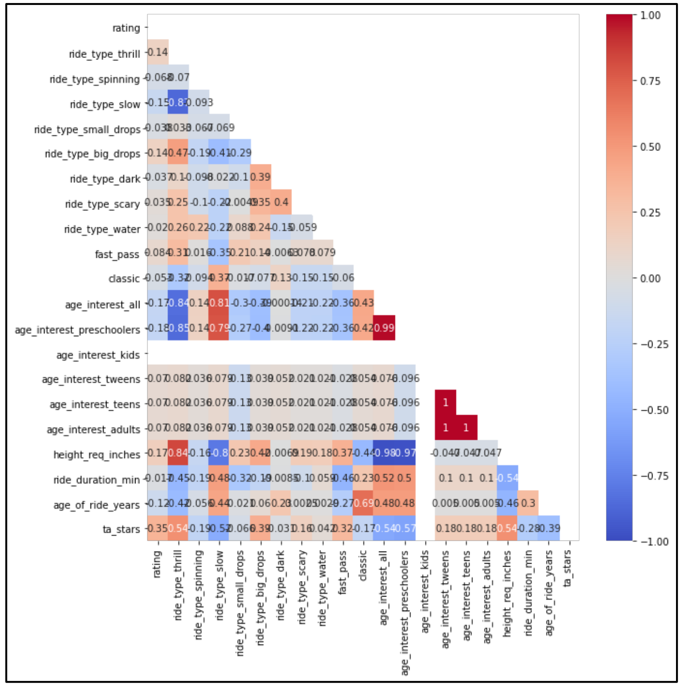
ANALYSIS

- Most reviews are 200 words or less



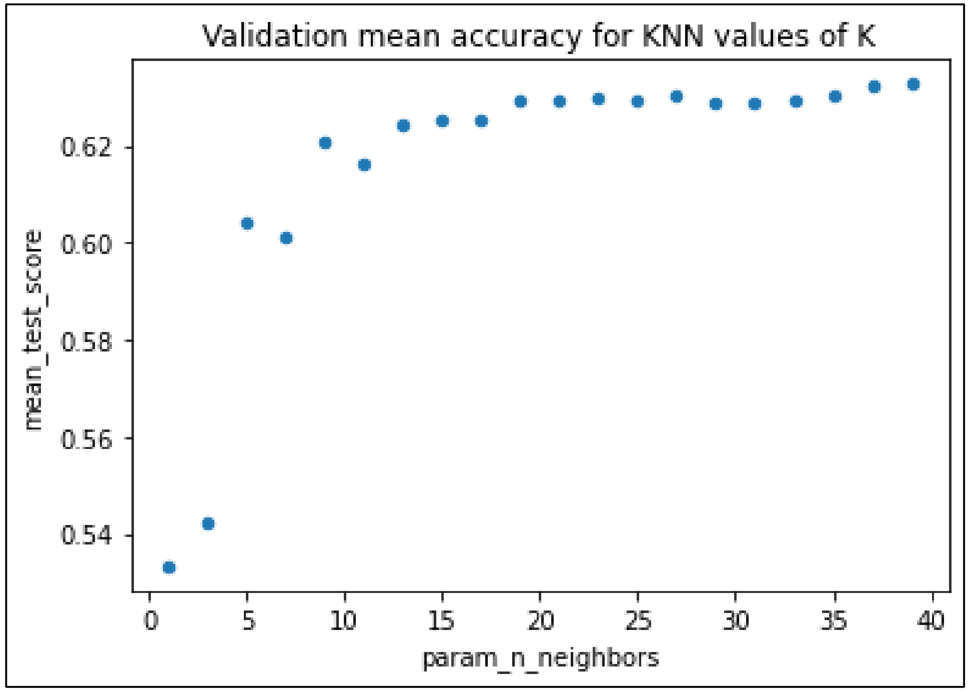
MODELING

- Adjusted which columns I used or dropped depending on the model (dropped to 17 columns for linear/lasso/ridge regressions – 34 after one-hot encoding)
- After adding/removing columns, trying different levels of regularization (including with grid search), could not get a linear regression model based on ride characteristics to explain more than 10% variance



MODELING

- Predicting ratings can be treated as either a regression or classification problem (https://towardsdatascience.com/1-to-5-star-ratings-classification-or-regression-b0462708a4df)
- Tried K nearest neighbors, random forests, and random forests with gradient boosting, but couldn't come up with a model that did better than the null model (guessing all 5-stars would be 62% accurate)



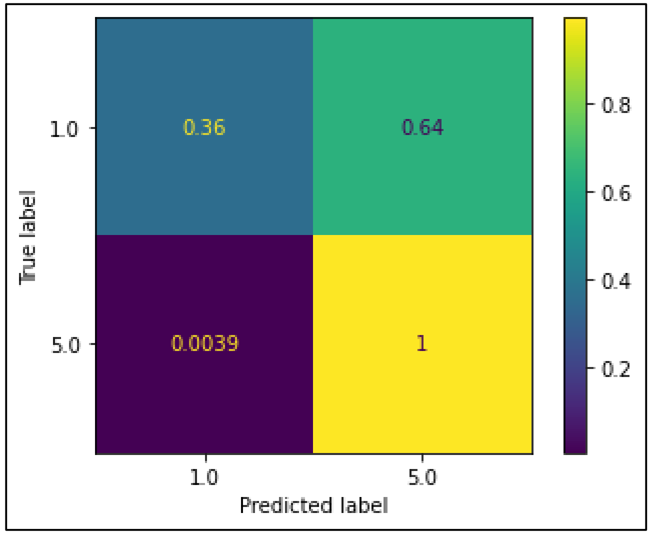
MODELING

- Determined that perhaps the ride attribute data wouldn't work, and wanted to try looking at the textual data instead
- Used natural language processing to tokenize the body of the review text and use only those tokens to predict the rating (dropped the ride characteristic features)
- Also decided to simplify prediction from multi-classification, to just a binary 1- or 5-star prediction

MODELING

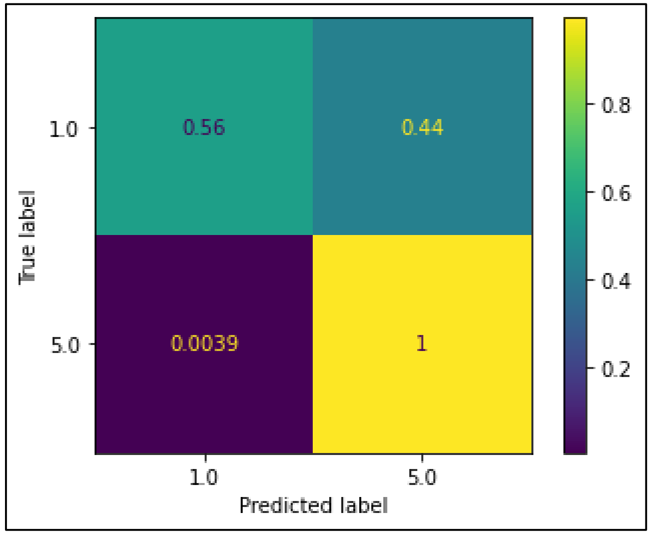
(1) Logistic Regression Grid Search Pipeline with stop words excluded

Precision: 79%
Recall: 36%
F-1 score: 49%



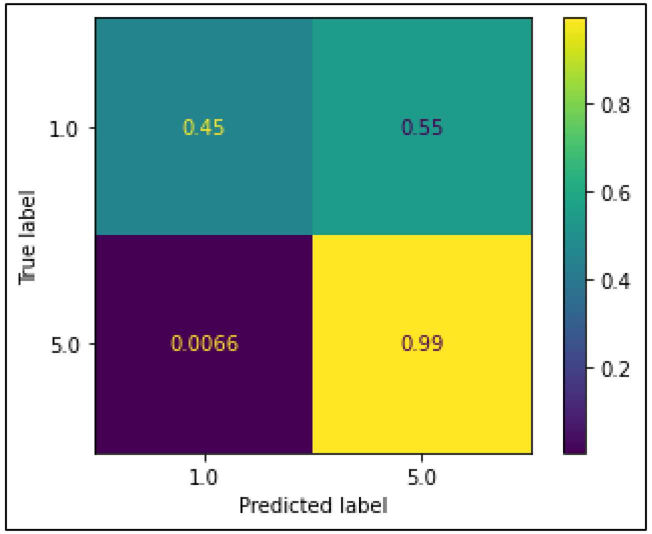
(2) Mulitnomal NB Grid Search Pipeline with stop words included

Precision: 86%
Recall: 56%
F-1 score: 68%



(3) Mulitnomal NB Grid Search Pipeline with stop words excluded

Precision: 74%
Recall: 45%
F-1 score: 56%



MODELING

(1) Logistic Regression

Positive			Negative	
journey	loved	1.942935	boring	-3.667130
thrilling	amazing	1.846825	poor	-2.953340
wet	best	1.696945	waste	-2.871824
fastpass	great	1.587399	money	-2.776897
nice	classic	1.575069	worst	-2.596002
stand line	enjoyed	1.533735	breaking	-2.470273
quite	awesome	1.481636	55	-2.141151
interesting	love	1.391646	awful	-2.088805
experience	beautiful	1.299003	recover	-2.063510
kids	great ride	1.227289	skip	-1.969993
straight	moving	1.212137	needs	-1.963728
relaxing	fantastic	1.182848	disappointed	-1.961874
worth wait	fun ride	1.157767	ride large	-1.804217
wonderful	flight	1.145366	hours	-1.760666
fast	fun	1.127956	overhaul	-1.735599
miss	little	1.122434	waited hours	-1.661210
make	ages	1.096186	badly	-1.644463
queuing	best ride	1.089211	waste time	-1.643827
morning	exciting	1.071515	planning	-1.612293
incredible	saw	1.069654	horrible	-1.603740
favorite			stupid	-1.600445
			outdated	-1.596391
			uncomfortable	-1.544295

(2) Multinomial NB Alt. 1

Positive			Negative	
fast pass	the	-3.259037	torture	-15.356973
do	and	-3.951733	boring ride	-15.356973
like	it	-4.068430	waste of	-15.356973
line	ride	-4.098153	dangerous	-12.959078
this is	to	-4.146492	stupid	-12.959078
if you	you	-4.231904	wasted	-12.959078
pass	is	-4.315316	very disappointed	-12.959078
it is	this	-4.445875	for such	-12.312451
there	of	-4.457257	felt sick	-12.312451
time	in	-4.681042	breaking down	-12.312451
disney	for	-4.934960	how bad	-12.312451
great	we	-4.984389	boring and	-12.312451
wait	on	-5.066830	badly	-12.312451
can	was	-5.111990	degrees	-12.312451
your	but	-5.244806	overhaul	-12.312451
fun	that	-5.296012	to fall	-12.312451
all	are	-5.350884	recover	-12.312451
be	as	-5.370018	rip	-12.312451
in the	at	-5.381118	instructed	-12.312451
have	so	-5.417106	sick and	-12.312451

(3) Multinomial NB Alt. 2

Positive			Negative	
going	ride	-3.020367	torture	-14.279187
definitely	fast	-4.563174	wasted	-11.881291
magic	fun	-4.694459	stupid	-11.881291
different	wait	-4.776102	dangerous	-11.881291
year	great	-4.789625	speakers	-11.234664
people	disney	-4.789625	boring ride	-11.234664
favorite	time	-4.804868	badly	-11.234664
make	pass	-4.925526	instructed	-11.234664
got	line	-4.959006	felt sick	-11.234664
went	like	-4.964396	line 30	-11.234664
family	fast pass	-5.018058	recover	-11.234664
feel	long	-5.181903	puke	-11.234664
way	really	-5.207993	degrees	-11.234664
fastpass	just	-5.211447	nightmare	-11.234664
roller	rides	-5.309518	overhaul	-11.234664
attraction	love	-5.310791	panic	-11.234664
did	worth	-5.393054	rip	-11.234664
minutes	experience	-5.429673	jerking	-10.845200
kids	times	-5.451425	spun	-10.845200
amazing	best	-5.454362		

NEXT STEPS

- Might be helpful as part of a park recommendation tool in an app to help guests plan their visit
- Sentiment analysis of the most important words used to predict ratings as a way of identifying best/worst ride characteristics
- Would like to build upon the model by adding in the text from the review titles, as well as the dates, as seasonality may also impact enjoyment of different attractions – could also try a multi-classification model using natural language processing
- Could revisit a model based on ride attributes at some point with additional data/new features about the rides

QUESTIONS?