Capstone Project:
By analyzing the earning call transcripts, would classification model provide value for speculative investors to trade next day in the automobile companies?

Bill Yu
May 17, 2019

24Slides

Agenda

1. Dataset
2. Roadmap

- EDA

2. Model Summary: Coefficient and features of importance

3. Conclusion

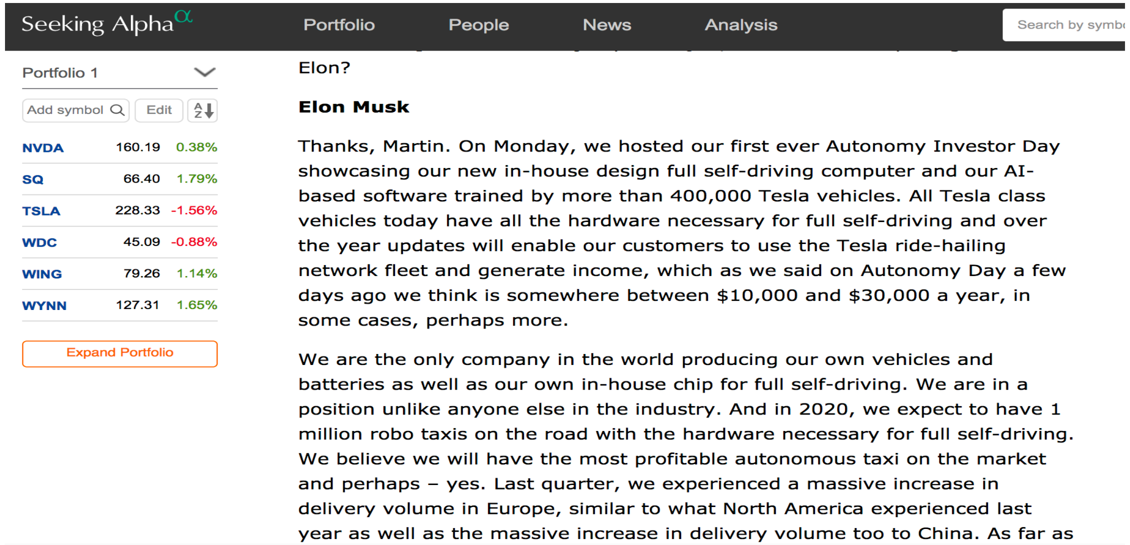
Dataset: Seekingalpha and Yahoo Finance

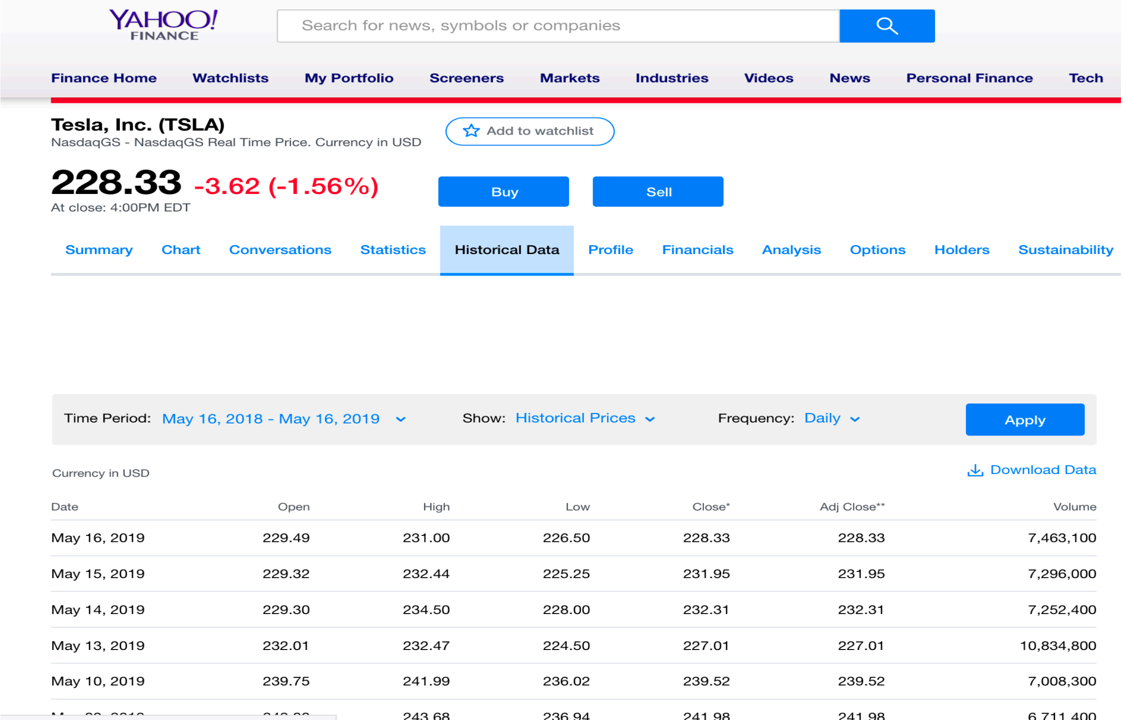
The quarterly conference call held by the IPO companies, followed by the release of their earning per share.

Small Observations: 67 rows

Duration: Year 2014-2019

Companies: Ford Motor, Fiat Chrysler, General Motors, Tesla.





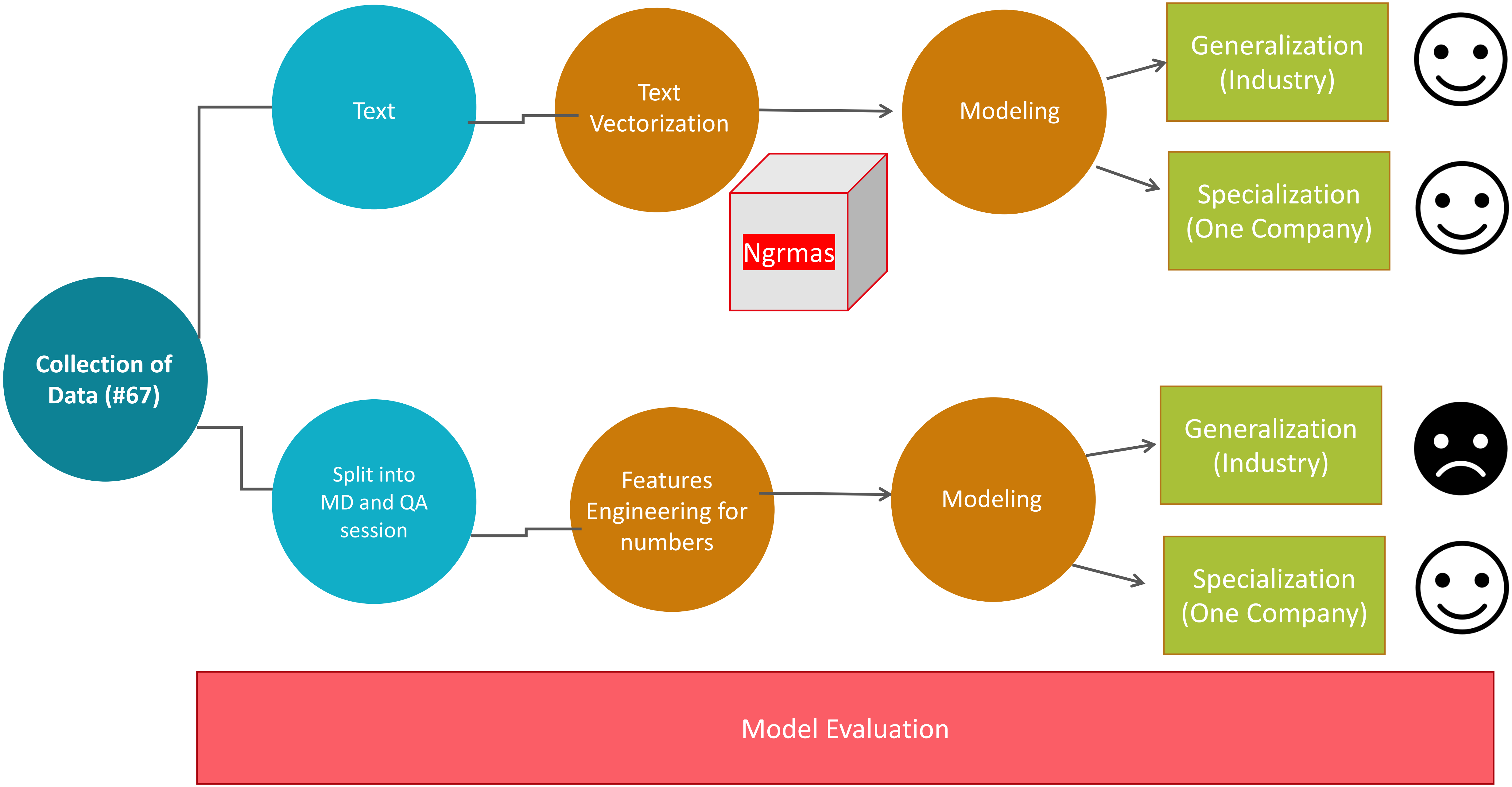
Data

	transcripts
0	Ford Motor (F) Q3 2018 Results - Earnings Call...
1	Ford Motor (F) Mark Fields on Q1 2016 Results ...
2	Tesla Motors (TSLA) Elon Reeve Musk on Q2 2015...
3	Tesla (TSLA) Q3 2017 Results - Earnings Call T...
4	General Motors (GM) Q3 2017 Results - Earnings...
5	General Motors (GM) Q4 2016 Results - Earnings...
6	Tesla (TSLA) Q4 2016 Results - Earnings Call T...
7	Ford Motor (F) Q2 2016 Results - Earnings Call...
8	Ford Motor's (F) CEO Mark Fields on Q3 2015 Re...

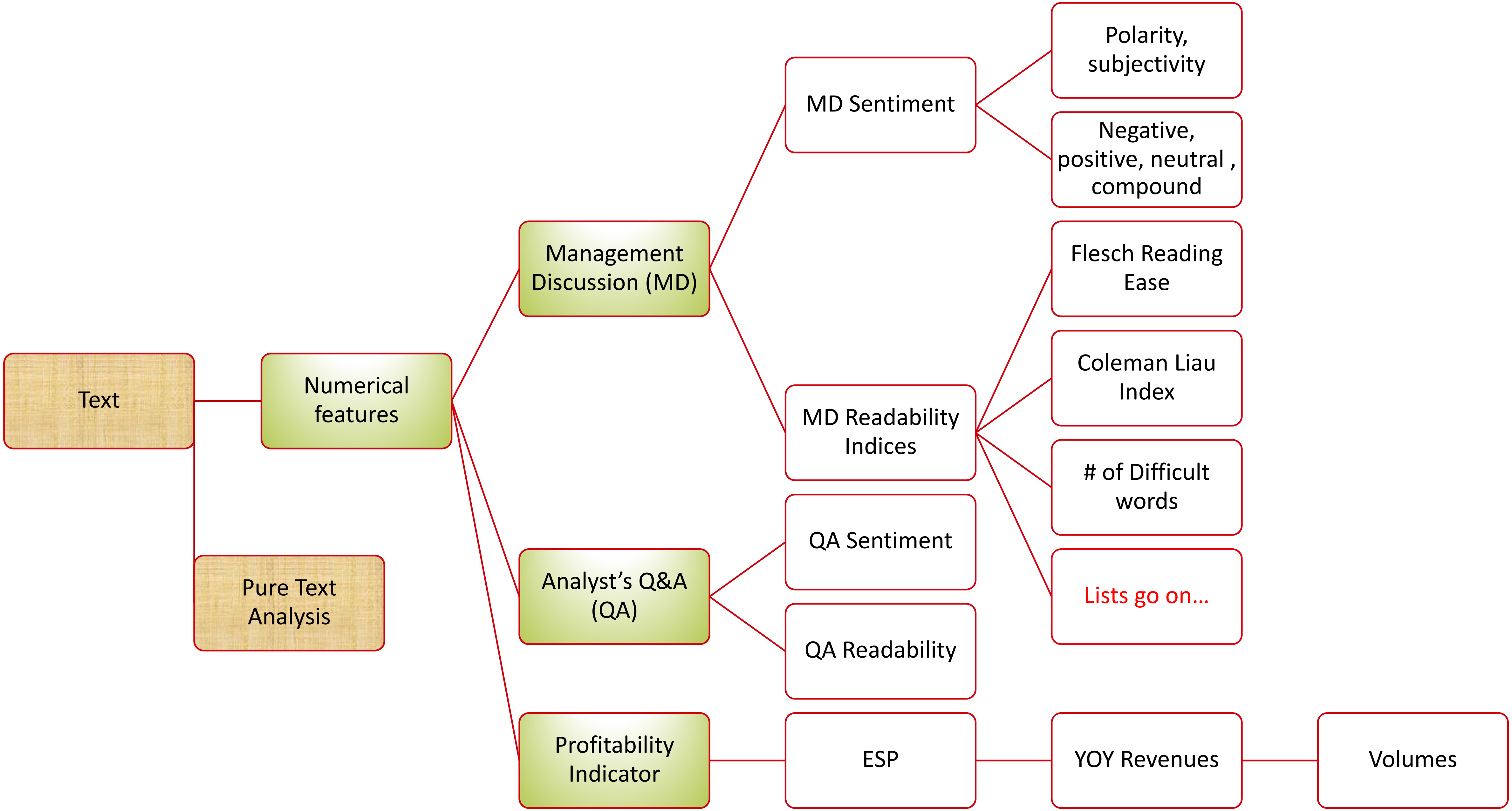
"I believe North America results demonstrate the early evidence that our fitness actions and commitment to focus on higher return opportunities are now taking hold and this is driving a more resilient business model. Within North America results this quarter, we achieved a $1 billion mix improvement from our strong product line focus, including more F-150s and more Super Duties that had record transaction prices. .."

From Q3 Ford Motor Transcripts

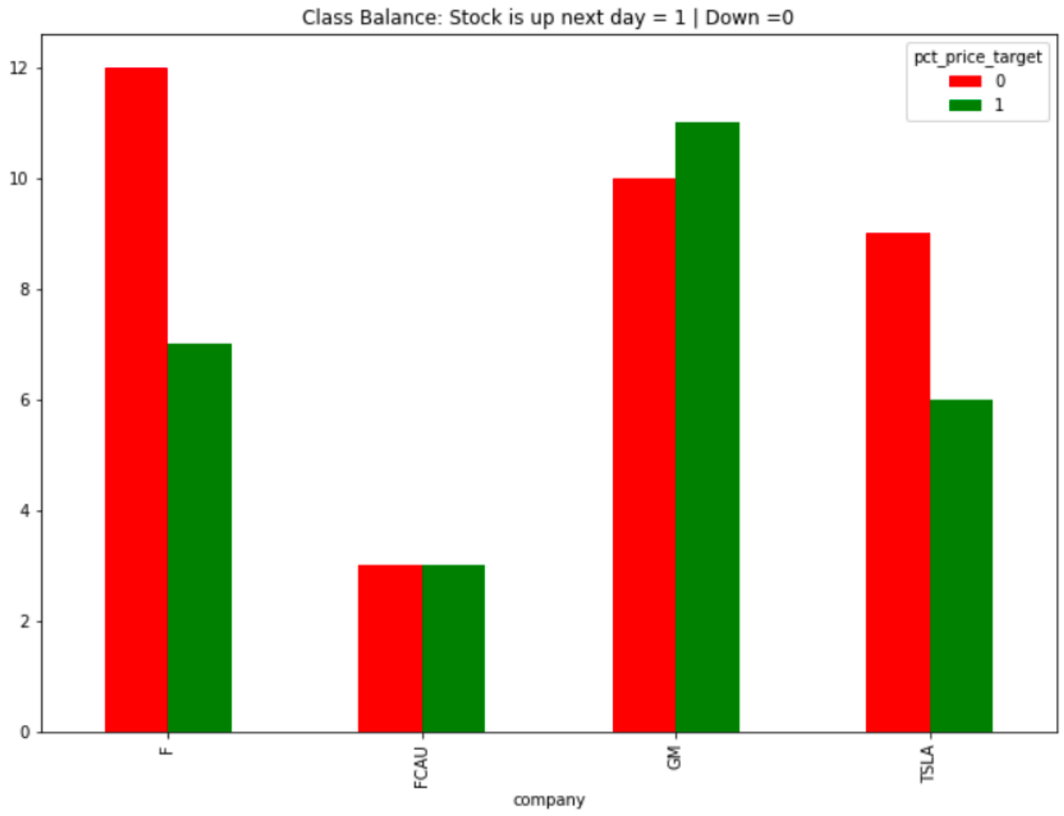
Roadmap for Modeling and Evaluation

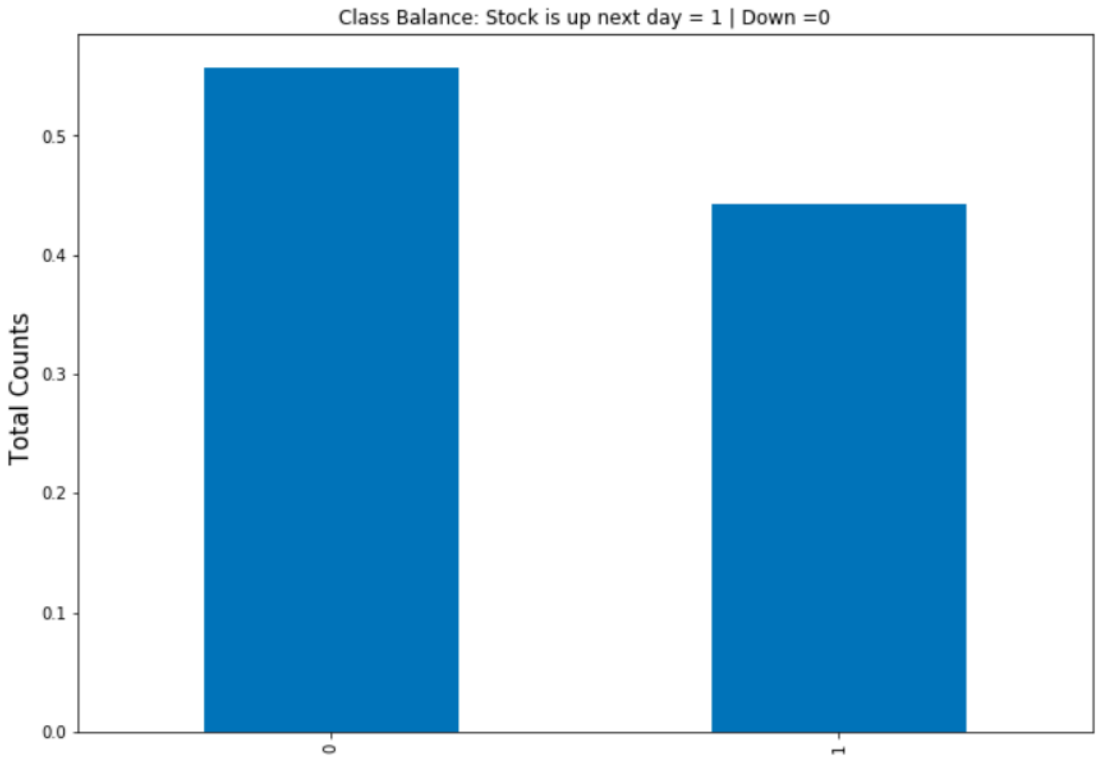


Dataset: 63 features are added



Dataset: Target Variable Y





- Target value Y is defined (Open Price – Closed Price) /Closed Price
- When the percentage change is less than 0%, it is 0 (down), or 1 (up)
- Make 0 as a baseline: 0.55

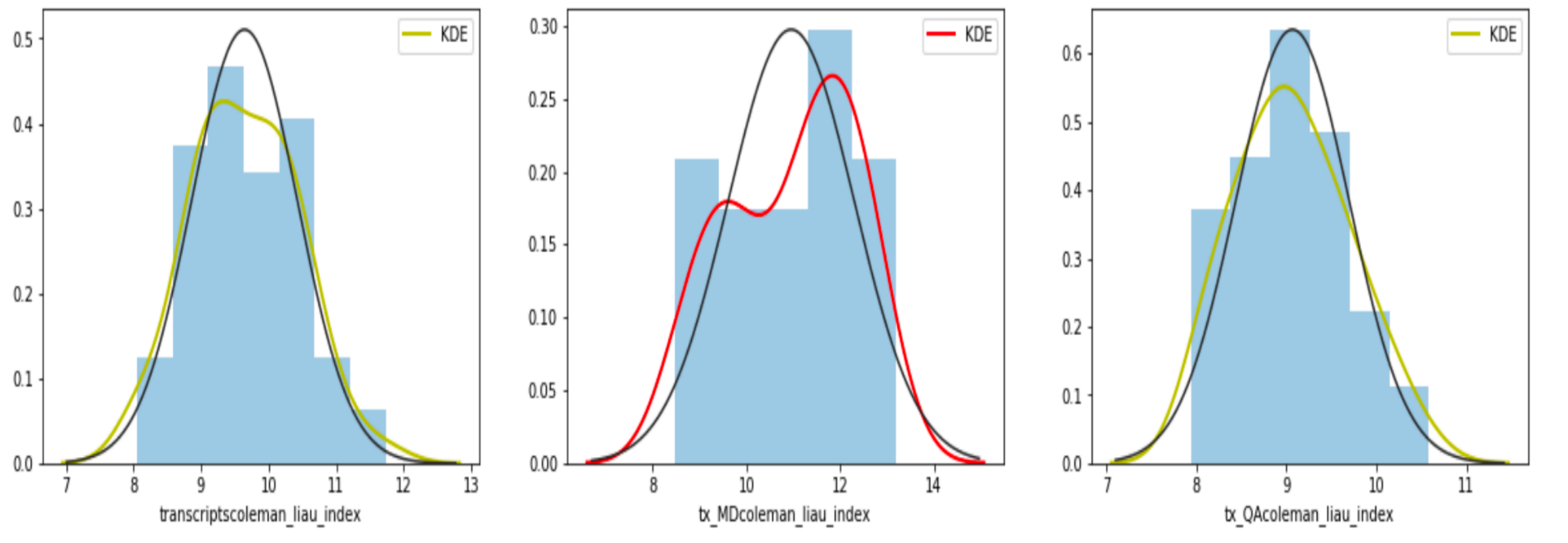
Dataset: X Variables

tx_QA_avg_sentence_length	tx_QAtextstat.lexicon_count	tx_QAflesch_reading_ease	tx_QAflesch_kincaid_grade	tx_QAdifficult_words	tx_QAlinse
28.5	6529	67.93	10.9	499	
21.7	7177	66.37	9.4	572	
25.5	8225	62.51	10.9	653	
17.3	5255	70.84	7.7	476	
23.2	5454	64.85	10.0	435	

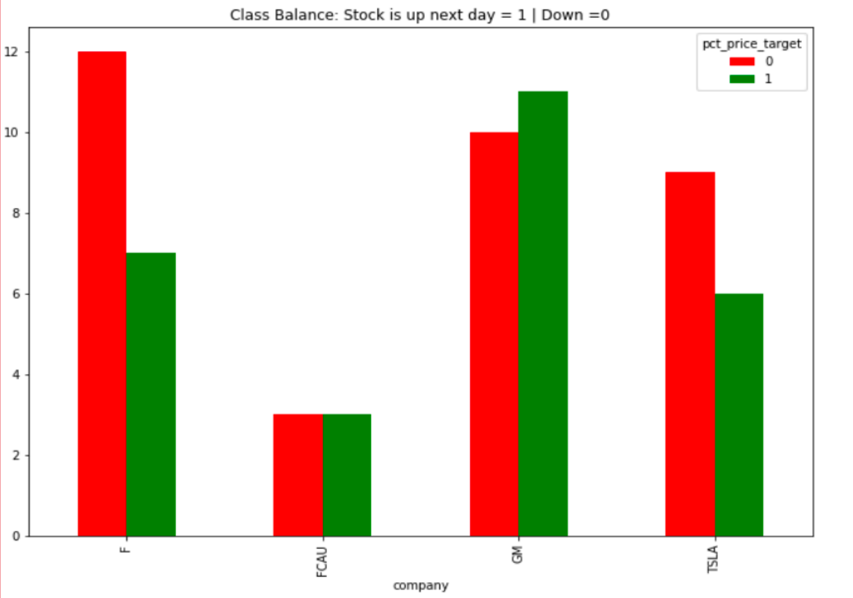
EDA: Univariate Analysis

Univariate Observation in readability:

Firms with a high readability index tend to miss the target, suggesting a possibility that management team tries to increase their language complexity to avoid taking some issues.



		transcriptscoleman_liau_index		
		mean	median	std
company	pct_price_target			
F	0	9.424167	9.465	0.575807
	1	9.338571	9.290	0.428386
FCAU	0	9.486667	9.530	0.237978
	1	9.426667	9.230	0.439356
GM	0	10.528000	10.480	0.374724
	1	10.409091	10.330	0.518757
TSLA	0	8.975556	8.820	0.462523
	1	8.750000	8.820	0.604649



Note: Coleman index: Public School textbooks readability. Score is tantamount to the required nth grade education for readers

EDA: Pearson Correlation

	pct_price_target
pct_price_target	1.000000
esp_target	0.185844
tx_QAflesch_reading_ease	0.137845
polarity	0.131189
tx_QA_compound	0.115045
subjectivity	0.109585
transcriptsflesch_reading_ease	0.099037
tx_QA_polarity	0.097424
tx_QA_subjectivity	0.095725
tx_MDdale_chall_readability_score	0.088516
tx_MD_subjectivity	0.079533

	pct_price_target
tx_MD_num_syl	-0.124492
transcriptsgunning_fog	-0.128670
transcripts_avg_sentence_length	-0.130711
tx_MDtextstat.lexicon_count	-0.133579
transcriptsdale_chall_readability_score	-0.133712
tx_QAautomated_readability_index	-0.134759
tx_QAgunning_fog	-0.144407
tx_QAflesch_kincaid_grade	-0.148334
tx_QA_avg_sentence_length	-0.149017
tx_QAdale_chall_readability_score	-0.165917
Volume	-0.171352

Top positive correlations are related to profit indicator and sentiment index

Top Negative corrections are likely related to readability index.

EDA: Word Cloud and Topic Modeling


```
[(0,
  '0.001*"year" + 0.001*"question" + 0.001*"think" + 0.001*"quarter" + '
  '0.001*"go" + 0.001*"thank" + 0.001*"co" + 0.001*"tesla" + 0.001*"would" + '
  '0.001*"inc"'),
 (1,
  '0.005*"year" + 0.004*"think" + 0.003*"co" + 0.003*"general_motor" + '
  '0.003*"look" + 0.003*"s" + 0.003*"quarter" + 0.002*"go" + 0.002*"expect" + '
  '0.002*"cost"'),
 (2,
  '0.004*"year" + 0.004*"think" + 0.003*"go" + 0.002*"question" + 0.002*"get" '
  '+ 0.002*"look" + 0.002*"would" + 0.002*"model" + 0.002*"quarter" + '
  '0.002*"s"'),
 (3,
  '0.003*"year" + 0.002*"quarter" + 0.002*"think" + 0.001*"go" + 0.001*"would" '
  '+ 0.001*"see" + 0.001*"thank" + 0.001*"s" + 0.001*"well" + 0.001*"inc"'),
 (4,
  '0.043*"tesla" + 0.041*"inc" + 0.020*"elon" + 0.018*"reeve_musk" + 0.014*"s" '
  '+ 0.013*"think" + 0.012*"go" + 0.010*"really" + 0.010*"be" + 0.010*"car"'),
 (5,
  '0.004*"s" + 0.004*"year" + 0.004*"think" + 0.004*"co" + '
  '0.003*"general_motor" + 0.003*"question" + 0.003*"go" + 0.003*"would" + '
  '0.003*"have" + 0.003*"look"'),
```

1. Word clouds and LDA topic model suggest that the context of the call transcript are related to date, entity, action, and person name.
2. Turnover rate of the management team may varied. Elon Mark has attended every conference call from 2014 to 2019. Hence, his name show up.

Model Evaluation Text only: Generalization and Specialization

```
sm = pd.DataFrame() # Instantiate the empty shell to hold the function
models = [logreg, knn, bg, rf, et, ada, gb, nb, svc, clf, xb,xb1,evc]

for i in models:
    sm = sm.append(model_scores(i, X_train_vect, y_train, X_test_vect, y_test))
sm
```

	model	accuracy score	cv train score	cv test score	train score	test score	train-test gap	model status	bias vs variance
0	LogisticRegression(C=1.0, class_weight=None, d...	0.6250	0.423413	0.566667	1.000000	0.6250	0.375000	overfit	high variance
0	KNeighborsClassifier(algorithm='auto', leaf_si...	0.7500	0.418849	0.377778	0.555556	0.7500	-0.194444	underfit	high variance
0	(DecisionTreeClassifier(class_weight=None, cri...	0.4375	0.555754	0.566667	0.666667	0.4375	0.229167	overfit	high variance
0	(DecisionTreeClassifier(class_weight=None, cri...	0.5000	0.509921	0.377778	1.000000	0.5000	0.500000	overfit	high variance
0	(ExtraTreeClassifier(class_weight=None, criter...	0.6250	0.468056	0.500000	1.000000	0.6250	0.375000	overfit	high variance
0	(DecisionTreeClassifier(class_weight=None, cri...	0.3750	0.467857	0.466667	1.000000	0.3750	0.625000	overfit	high variance
0	([DecisionTreeRegressor(criterion='friedman_ms...	0.6250	0.490278	0.444444	1.000000	0.6250	0.375000	overfit	high variance
0	MultinomialNB(alpha=1.0, class_prior=None, fit...	0.6875	0.398214	0.444444	1.000000	0.6875	0.312500	overfit	high variance
0	LinearSVC(C=1.0, class_weight=None, dual=True,...	0.6250	0.378770	0.566667	1.000000	0.6250	0.375000	overfit	high variance
0	SVC(C=1.0, cache_size=200, class_weight=None, ...	0.5625	0.485913	0.500000	0.911111	0.5625	0.348611	overfit	high variance
0	XGBClassifier(base_score=0.5, booster='gbtree'...	0.7500	0.404167	0.566667	1.000000	0.7500	0.250000	overfit	high variance
0	XGBClassifier(base_score=0.5, booster='gbtree'...	0.6250	0.490278	0.566667	1.000000	0.6250	0.375000	overfit	high variance
0	VotingClassifier(estimators=[('lr', LogisticRe...	0.6250	0.424802	0.500000	1.000000	0.6250	0.375000	overfit	high variance

Comment

Drawback:

Increase of N grams will change the features in the model. In this case, the higher the n-grams, the lower the score.

Generalization:
&
Specialization

XGBoost with ngrams (1,1)	
Accuracy:	0.75
Train:	1.00
Test:	0.75

Logistic	(2,2),	(3,3)
Accuracy:	0.8	0.68
Train:	1.00	1.00
Test:	0.8,	0.37

Model Evaluation Numerical feature: Generalization and Specialization

```
score_matrix = pd.DataFrame() # Instantiate the empty shell to hold the function

models = [logreg, rf,et,ada,gb, svc, clf, xb, xb1, evc]

for i in models:
    score_matrix = score_matrix.append(model_scores(i, X_train_sc, y_train, X_test_sc, y_test))
score_matrix
```

	model	accuracy score	cv train score	cv test score	train score	test score	train-test gap	model status	bias vs variance
0	LogisticRegression(C=1.0, class_weight=None, d...	0.5625	0.335913	0.433333	0.888889	0.5625	0.326389	overfit	high variance
0	(DecisionTreeClassifier(class_weight=None, cri...	0.3750	0.469643	0.322222	0.933333	0.3750	0.558333	overfit	high variance
0	(ExtraTreeClassifier(class_weight=None, criter...	0.4375	0.530556	0.322222	1.000000	0.4375	0.562500	overfit	high variance
0	(DecisionTreeClassifier(class_weight=None, cri...	0.4375	0.266071	0.622222	1.000000	0.4375	0.562500	overfit	high variance
0	([DecisionTreeRegressor(criterion='friedman_ms...	0.3750	0.331151	0.622222	1.000000	0.3750	0.625000	overfit	high variance
0	LinearSVC(C=1.0, class_weight=None, dual=True,...	0.5000	0.401389	0.433333	1.000000	0.5000	0.500000	overfit	high variance
0	SVC(C=1.0, cache_size=200, class_weight=None, ...	0.3750	0.554365	0.622222	0.755556	0.3750	0.380556	overfit	high variance
0	XGBClassifier(base_score=0.5, booster='gbtree'...	0.3750	0.359524	0.511111	1.000000	0.3750	0.625000	overfit	high variance
0	XGBClassifier(base_score=0.5, booster='gbtree'...	0.4375	0.362500	0.511111	1.000000	0.4375	0.562500	overfit	high variance
0	VotingClassifier(estimators=[('svc', LinearSVC...	0.4375	0.310516	0.511111	1.000000	0.4375	0.562500	overfit	high variance

Comment

Takeaway:

1. Boosting is a method of converting a set of weak learners into strong learners

Drawback:

1. Model will be defeated by noisy data
2. Dataset are too small; it is not significant to generalize the data.

Generalization: all 4 automobile makers

Logistics Regression

Accuracy: 0.56

Train: 0.88

Test: 0.56

Specialization: Tesla only

Ada and XGBoost

Accuracy: 1.00

Train: 1.00

Test: 1.00

Model Summary

Scores

Text Features

Trade Off

- N grams has significant impacts on the models because of features are changing.
- Small data set with good score may suggest overfitting and not able to generalize the model

Baseline Accuracy	Naive Bayes	XGBoost
0.55	0.68	0.75

Numerical Features

Trade Off

- Generalized model will be defeated by noisy since each company has unique aspects and business operation, even though normalize the scale of the data set.
- Score could be improved, but suffers from multicollinearity by adding features

Baseline Accuracy	Logistic	Extra Tree
0.55	0.56	0.75

Baseline Accuracy (Tesla only)	Extra Tree	XGBoost
0.60	0.75	1.00

Modeling Summary

Text Features

Importance

Features of importance

- Surprisingly, management personnel and entity didn't show up in the feature of importance. However, entity and person name show up in n gram (3,3)
- Inventory, profit, profit, and future plan are the importance features

	feature	coef
751	answer	0.069094
2754	enough	0.055316
5594	pre	0.051719
1151	bit	0.048817
6293	respond	0.047789
7624	transfer	0.039171
1328	buy	0.032894
6600	seasonally	0.032449
3108	field	0.032232
7227	successful	0.031198
4207	key	0.028650

	feature	coef
81196	okay thanks	0.063167
61623	inventory level	0.062798
63711	kind get	0.054985
91903	profit margin	0.051042
129386	well obviously	0.050579
16184	billion liquidity	0.050461
117870	team really	0.049992
38598	emmanuel rosner	0.045380
131499	would something	0.040254
93355	put take	0.026317
20926	cash balance	0.026243
51497	going forward	0.023870

Numerical Features

Coefficients

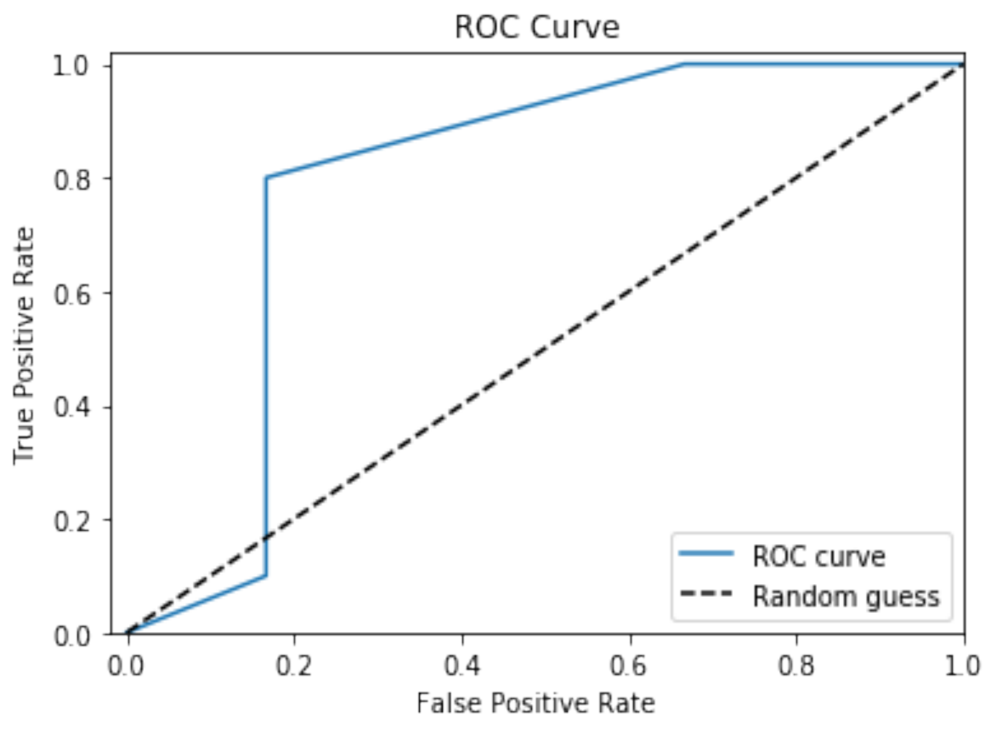
- Readability Index has a higher impact in the model

	feature	coef
43	tx_MDcoleman_liau_index	0.494497
9	tx_MD_neg	0.159128
3	pct_volume	0.144896
2	rev_total_b	0.143217
10	tx_MD_pos	0.046582
21	tx_MD_subjectivity	0.005561
17	polarity	0.005493
51	tx_QAlinsear_write_formula	0.000550
38	tx_MDflesch_kincaid_grade	0.000075
34	tx_MD_num_syl	0.000000

Model Summary: Supplemental info

Scores

ROC curve on model with the numerical features



- The separability between positive class and negative class is distinct in this model

Principal Component Analysis (PCA)

The explained variance ratio with 2 components is 0.99. It could produce the 0.75 accuracy with Extra tree model, compare to models without PCA

Confusion Matrix

```
[[5 1]
 [3 7]]
              precision    recall  f1-score   support

           0       0.62      0.83      0.71         6
           1       0.88      0.70      0.78        10

   micro avg       0.75      0.75      0.75        16
   macro avg       0.75      0.77      0.75        16
weighted avg       0.78      0.75      0.75        16
```

- Out of the 6 actual instances of market is down, 5 is predicted correctly
- Out of the 10 instances, 7 is predicted correctly.
- Class 1 (market is up) is more precise than class 0; it could due to the imbalance sample in the test set

Conclusion

1. The implication of using this model could help investors to understand of tone and language complexity of the management team and therefore suggest the market is either go up or down next day.
2. Investors could receive up or down signal from the classification model. They could compare whether these signals will be helpful to validate their intuition on next day trading when the earning call is released.
3. In general, the EDA process and model result suggest that firms with a high readability index tend to miss the target, suggesting a possibility that management team tries to increase their language complexity to avoid taking some issues.
4. Automobile industry focuses on the expansion of North America and China, Inventory, margin profit, and future plan are the importance features. There topics are the important features suggested by the XG Boost model. Generalized model will be defeated by noisy since each company has unique aspects and business operation, even though normalize the scale of the data set. However, the use of unsupervised learning such as KNN may be not applicable in this case.
5. The change of N-grams has a high impact on the model.

- Bigrams provides more information than sing grams or trigrams, without scarifying too much on accuracy and overfiting.

Limitations

Limitations

1. Due to the small data set, this model is subjected to a limited use in a few automobile manufactures.
2. Consider to change the target value to a multi-classification problem (discreate range of positive and negative value of 0-10%. 11-30%, 50% or above)
3. Create confusion matrix and investigate the probability on each misclassification

Thank You

References

- **Here is a breakdown of grade levels in the U.S.:**
- 5-6 yrs. old — Kindergarten
- 6-7 yrs. old — First Grade
- 7-8 yrs. old — Second Grade
- 8-9 yrs. old — Third Grade
- 9-10 yrs. old — Fourth Grade
- 10-11 yrs. old — Fifth Grade
- 11-12 yrs. old — Sixth Grade
- 12-13 yrs. old — Seventh Grade
- 13-14 yrs. old — Eighth Grade
- 14-15 yrs. old — Ninth Grade
- 15-16 yrs. old — Tenth Grade
- 16-17 yrs. old — Eleventh grade
- 17-18 yrs. old — Twelfth grade
- 18-22 yrs. old - College