PREDICTING THE POPULARITY OF ONLINE NEWS ARTICLES

Project Guide:

- *Prof. Vidhya K*

Project Members:

- *Akash Pravinkumar Bhatt*
- *Asmita Dileep Ghoderao*
- *Nicholas Lee D'Souza*
- *Saket P Shinde*
- *Swapnil Prabhakar Wagh*

BACKGROUND & OBJECTIVES

PROBLEM STATEMENT

OBJECTIVE:

- To analyse and predict the popularity of online news articles based on
  - Shares
  - Data Channel / LDA category

OUTCOME:

- Commercial, as it would benefit news agencies.
- Better understanding of the news articles generated.
- Find ways to maximize news article popularity.

BASIC DATASET DESCRIPTION

DATASET DESCRIPTION

- **Dataset:** Online News Popularity Prediction
- **Data Source:** UCI ML Repository
- **Dataset information:** News articles published by Mashable over 2 years
- **Number of attributes:** 61 (58 predictive, 2 non-predictive and 1 target)
- **Number of records:** 39644
- **Dependent Variable:** Number of shares

DATASET DESCRIPTION

ASPECTS	ATTRIBUTES	ASPECTS	ATTRIBUTES
Words (float)	Number of words of the title/content, Average word length, Rate of unique/non-stop words of contents	**Keywords (float)**	Number of keywords, Worst/best/average keywords (shares)
Links (integer)	Number of links, Number of links to other articles in Mashable	**Article category (boolean)**	Mashable data channels (bus, socmed, tech, world, lifestyle, entertainment)
Digital Media (integer)	Number of images/videos	**NLP (float)**	Closeness to five LDA topics, Title/Text polarity/subjectivity, Rate and polarity of positive/negative words, Absolute subjectivity/polarity level
Publication Time (boolean)	Day of the week/weekend	**Target (integer)**	Number of shares at Mashable

EXPLORATORY DATA ANALYSIS

OBSERVATIONS

- Dataset is clean, no missing values
- Categorical variables are already one-hot encoded
- Variables are not highly correlated with target variable

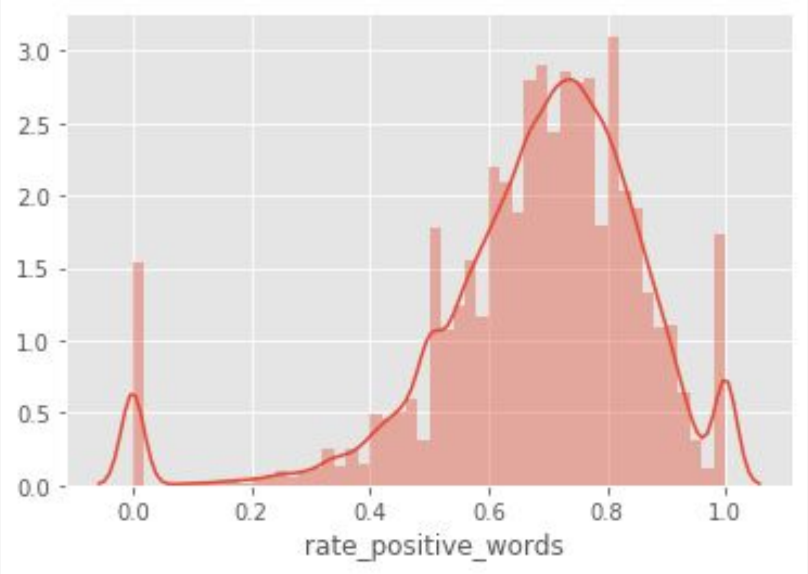
data_channel_is_bus	data_channel_is_socmed
0	0
1	0
1	0
0	0
0	0
0	0
0	0
0	0
0	0

d) Checking null values

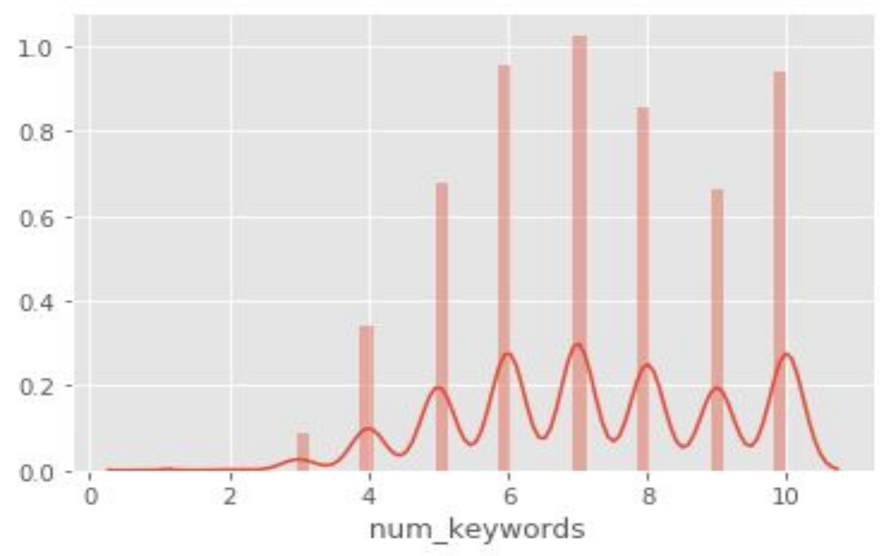
```
In [10]: ndf.isnull().sum()

Out[10]: n_tokens_title                 0
         n_tokens_content               0
         n_unique_tokens                0
         n_non_stop_words               0
         n_non_stop_unique_tokens       0
         num_hrefs                      0
         num_self_hrefs                 0
         num_imgs                       0
         num_videos                     0
```

	shares
shares	1.000000
kw_avg_avg	0.110413
LDA_03	0.083771
kw_max_avg	0.064306
self_reference_avg_sharess	0.057789
self_reference_min_shares	0.055958
self_reference_max_shares	0.047115
num_hrefs	0.045404
kw_avg_max	0.044686
kw_min_avg	0.039551

UNIVARIATE ANALYSIS



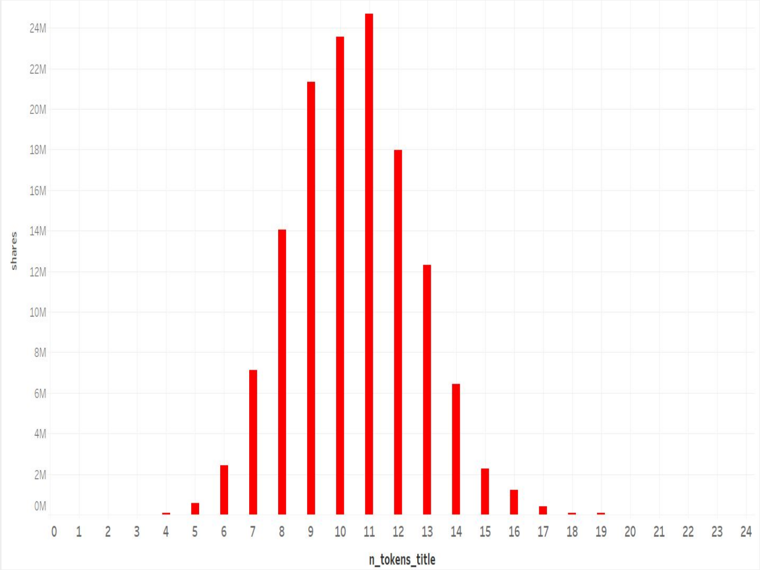
- There is a high degree of skewness (right skewed) for each numerical variable including target (shares).

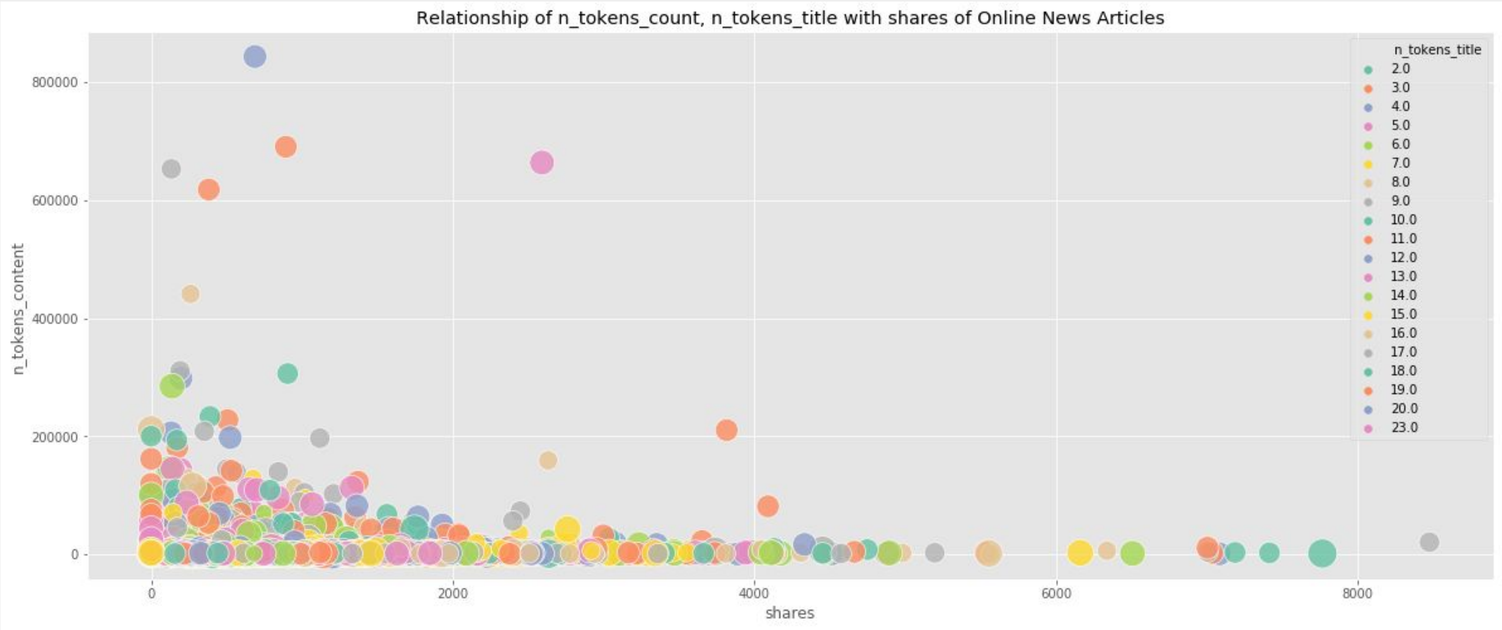


- Large number of categorical variables present as the categorical columns in the dataset given were one-hot encoded.

BIVARIATE ANALYSIS

- The shares variable (target) has data distributed discreetly at the ends of the distribution.
- The data seems to be heteroscedastic when tested using regplot.





STATISTICAL ANALYSIS

STATISTICAL TESTS

Dependent Variable	Independent Variable	Statistical Test Applied
Categorical	Numerical	Mann Whitney U test
Categorical	Categorical	Chi-square test

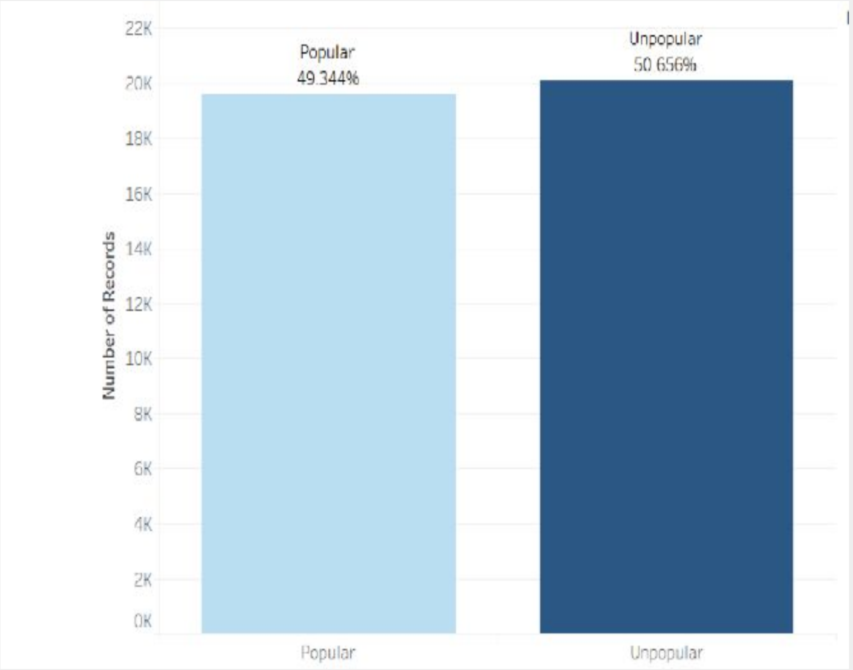
```
stat,p,df,exp= chi2_contingency(pd.crosstab(ndf['data_ch'],ndf['class']).values)
print(p)
```

```
def man_test(arr):
    for i in arr:
            cl0 = ndf[ndf['class']==0][i]
            cl1 = ndf[ndf['class']==1][i]
            t, p = mannwhitneyu(cl0,cl1)
            pval.append(p)
    return pval
```

CLASS IMBALANCE

CLASS IMBALANCE

BINARY CLASS

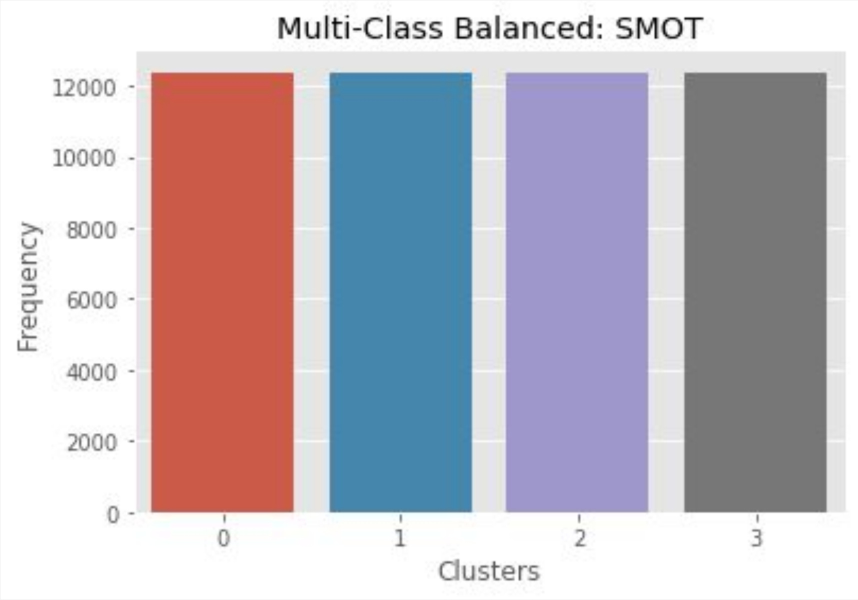


MULTI CLASS

CLASS IMBALANCE

MULTI CLASS IMBALANCE

MULTI CLASS BALANCED USING SMOT



SCALING, TRANSFORMATION, OUTLIER ANALYSIS

SCALING, TRANSFORMATION, OUTLIER ANALYSIS

- Most algorithms require scaling as a prerequisite for faster algorithm computations, so we have used scaling for all the machine learning algorithms used for binary and multi classification.

- Outlier detection (boxplot, strip plot) showed that data had extreme values, however those values were important for performing further analysis and modelling.

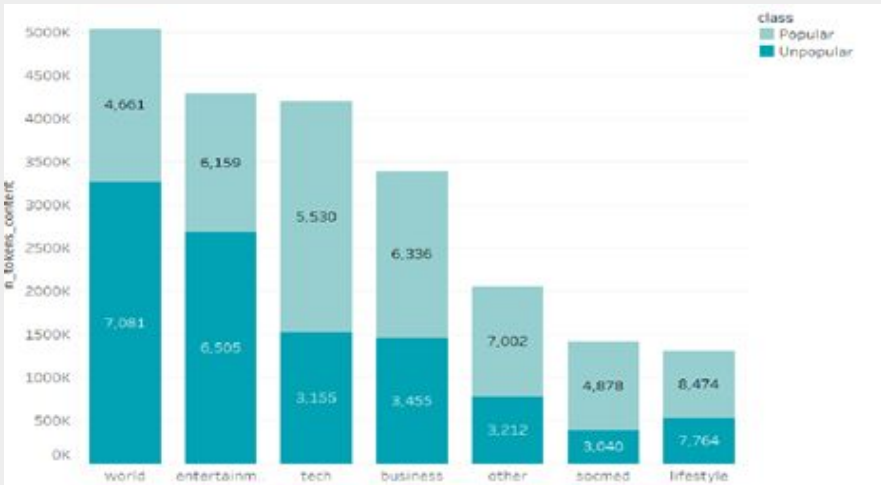
- Extreme values were not removed.

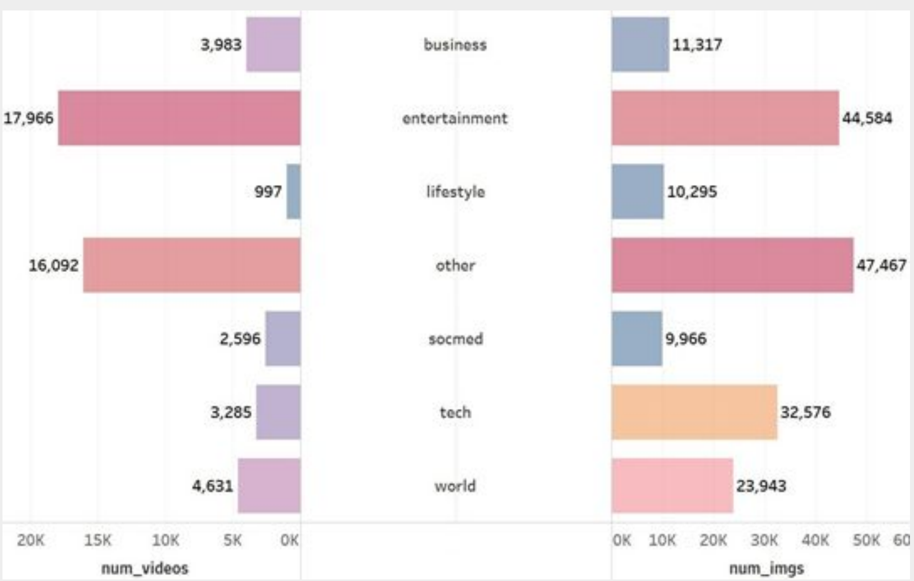
- Transforming the data to make the distributions mostly normal did not hold good for the dataset.

- Hence, the original dataset was used for analysis and modelling.

OBSERVED INSIGHTS

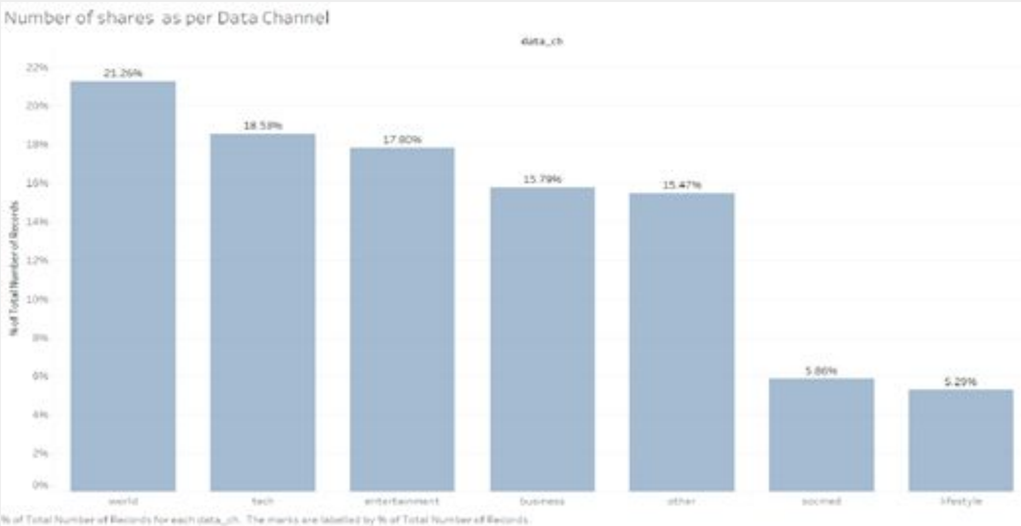
WORDS / DIGITAL MEDIA





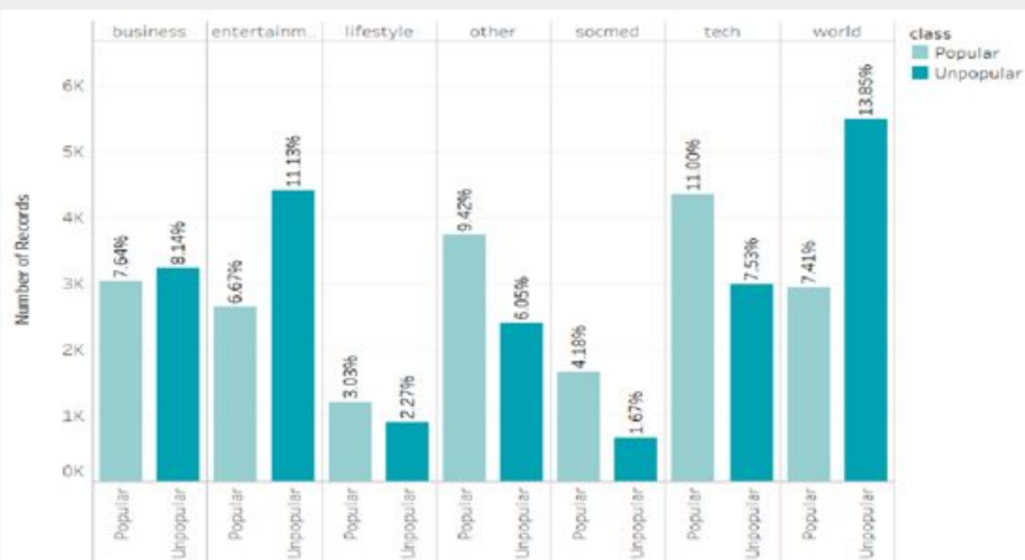
- Short articles (382-2591 words) have maximum shares
- About 101 articles do not have any textual content/images/videos
- Entertainment channel has large number of Videos and Images shared.

ARTICLE CATEGORY



Number of Articles Published

- World - highest number of articles
- Lifestyle/Social Media- lowest number of articles

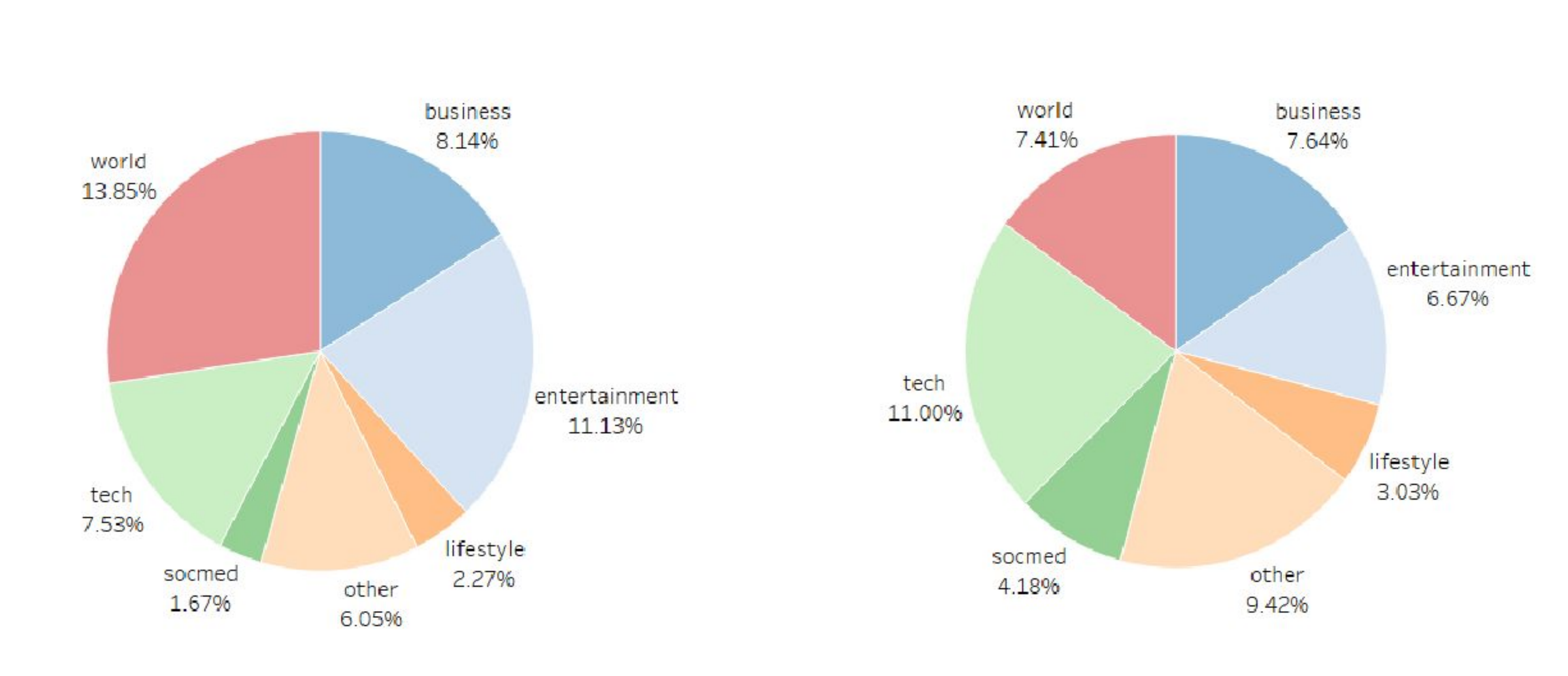


Popularity Percent

- Social Media - 70%
- World - 36%

Hence, There is no relation with the number of articles published by data channels and popularity percentage

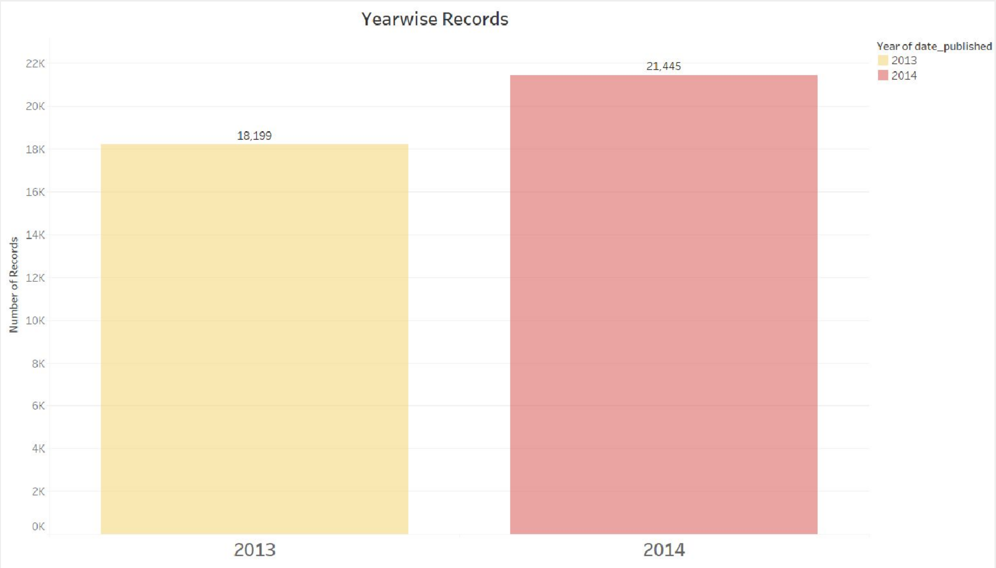
ARTICLE CATEGORY

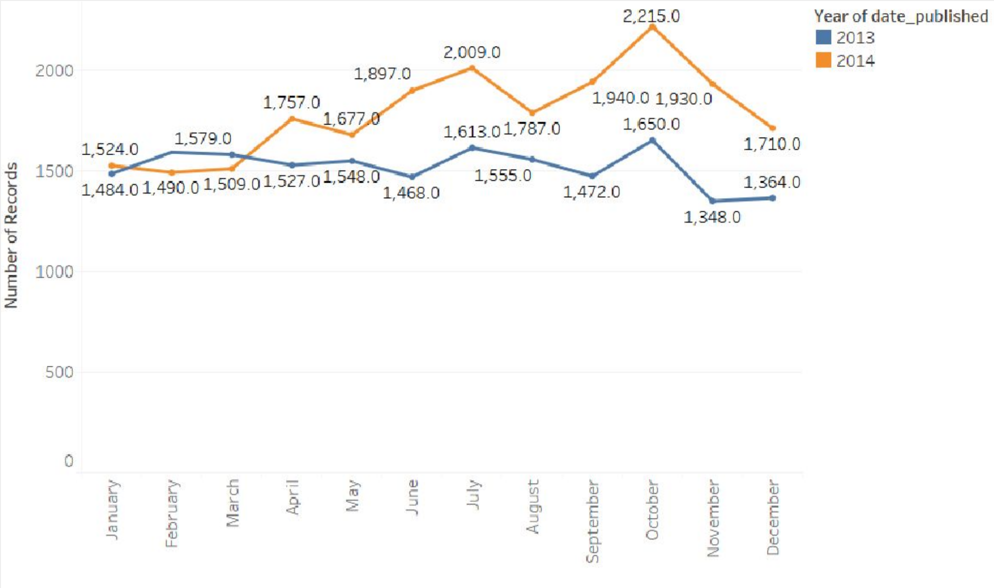


- Technology related articles contribute to high shares (23%) as well as gain popularity (27%)

- 6134 articles belong to a unknown data channel, have maximum shares and positive sentiments

PUBLICATION TIME





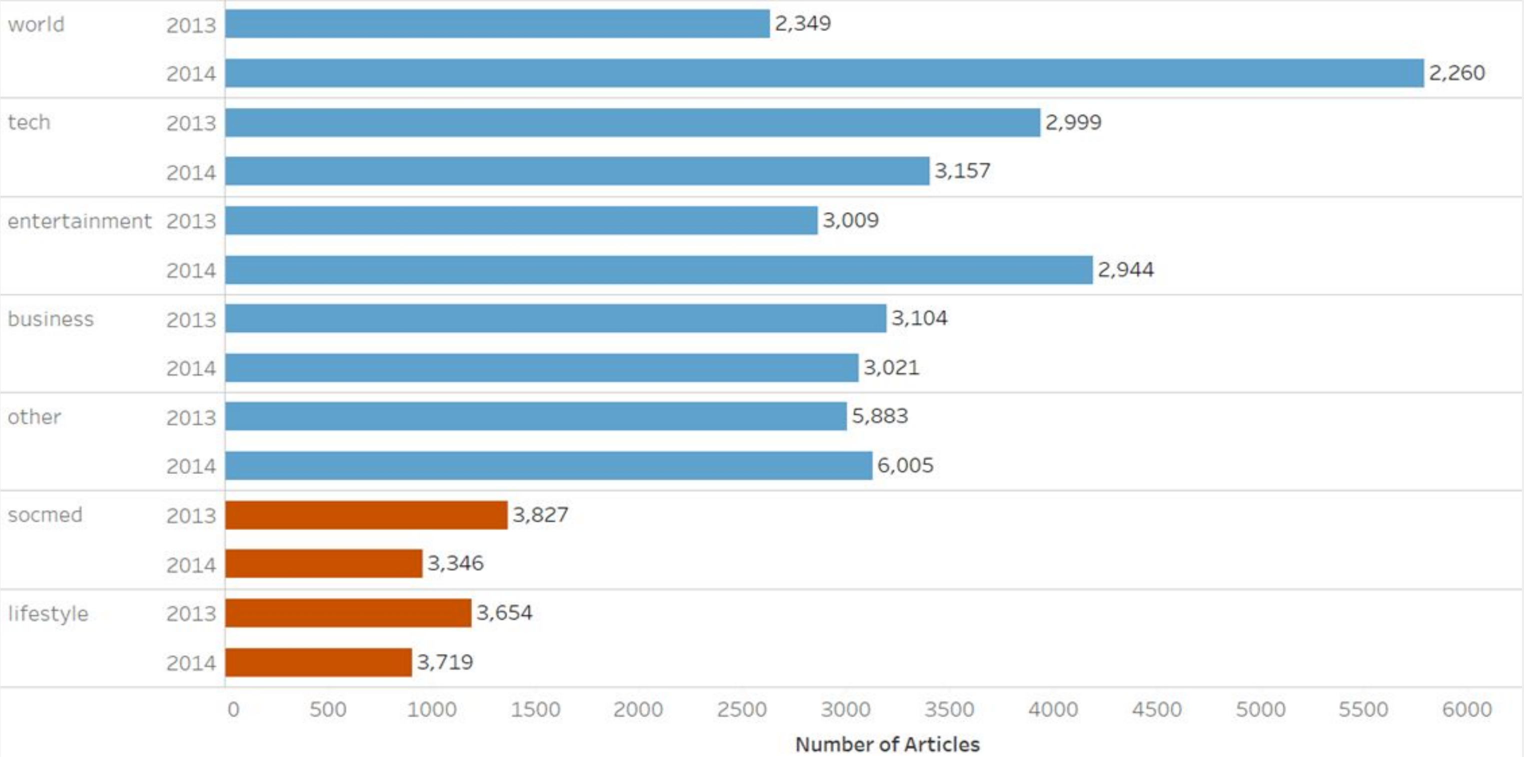
2013:

- Percent of Articles published (45.90%)
- Popularity Percent (46.61%)
- Maximum articles published in October and lowest in January

2014:

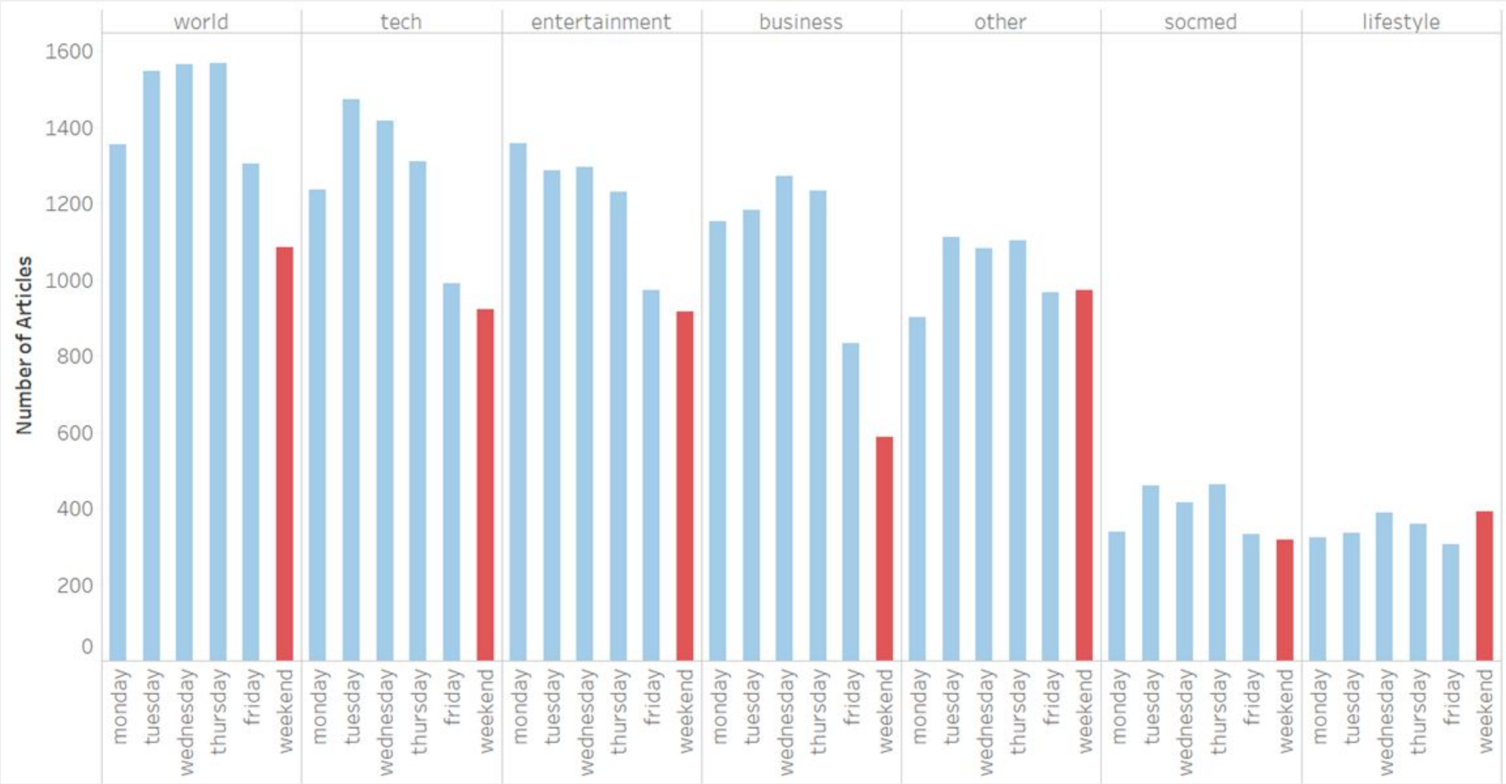
- Percentage of Articles published (54.10%)
- Popularity Percent (53.62%)
- Maximum articles published in October and lowest in January

PUBLISHED ARTICLES BASED ON DATA CHANNEL IN 2013, 2014



- Social media and lifestyle data channels by far contribute the least number of articles published. However, the world data channel contributes the most number of articles in the year 2014 and tech data channel in 2013.
- Increase in the number of articles published for a particular data channel does not contribute towards increase in number of shares or popularity.

ARTICLES PUBLISHED BASED ON THE WEEKDAYS, DATA CHANNEL



- On weekdays, in general more number of articles are published as compared to weekends irrespective of data channel.
- Social Media and Lifestyle data channel show a decreased number of articles published as compared to other data channels.

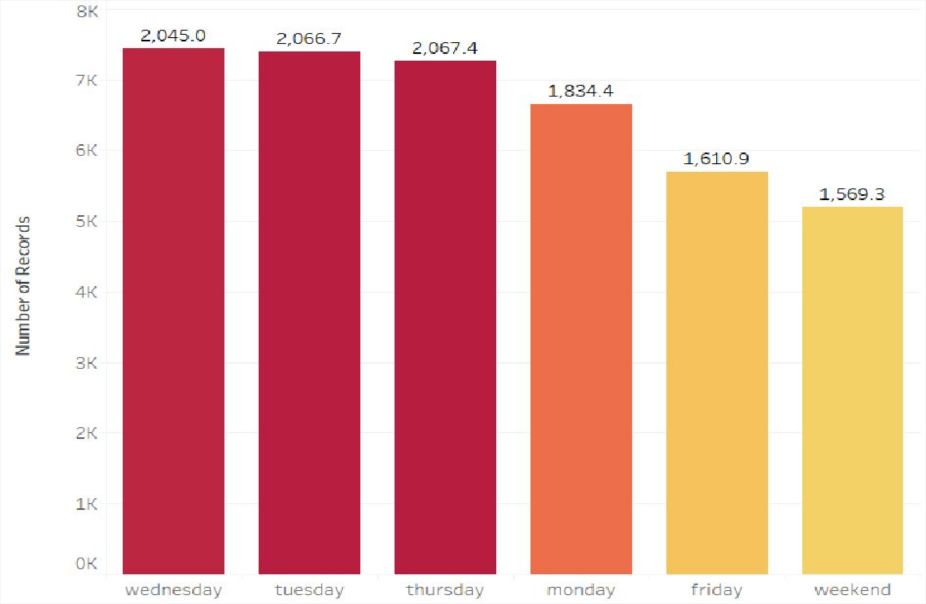
ARTICLES PUBLISHED BASED ON THE WEEKDAYS, DATA CHANNEL

topic	data_ch	monday	tuesday	wednesday	thursday	friday	weekend
LDA_00	Business	975.0	1,005.0	1,075.0	1,038.0	669.0	536.0
LDA_01	Entertainment	730.0	661.0	670.0	655.0	517.0	399.0
LDA_02	world	1,120.0	1,306.0	1,342.0	1,313.0	1,122.0	927.0
LDA_03	other	805.0	977.0	966.0	984.0	865.0	700.0
LDA_04	Technology	1,087.0	1,311.0	1,260.0	1,168.0	878.0	806.0

- Maximum articles published are from LDA_02 (World) and LDA_04 (Technology) on tuesday, wednesday and thursday.

- Minimum articles published on weekends especially from LDA_00 (Business) and LDA_01 (Entertainment).

SUBJECTIVITY/POLARITY



Sentiment Type	Articles
Positive	89%
Negative	8%
Neutral	3%

- 71% of news articles are factual rather than opinion based.
- Most articles overall published have positive sentiments.
- Articles published on weekdays are more subjective than those published nearing weekends.
- Most articles that have positive content may not have positive titles

POSITIVE / NEGATIVE WORDS

- LDA_01 (Entertainment) has least number for both positive and negative words
- LDA_04 (Technology) has most number of positive words
- LDA_02 (World) has most number of negative words

CHALLENGES BASED ON EDA

CHALLENGES BASED ON (EDA)

The challenges faced are as follows:

- As we had received the statistics of the news article data rather than the original data containing news article information it was challenging to understand the underlying meaning of each attribute.

- The data has large number of extreme values.

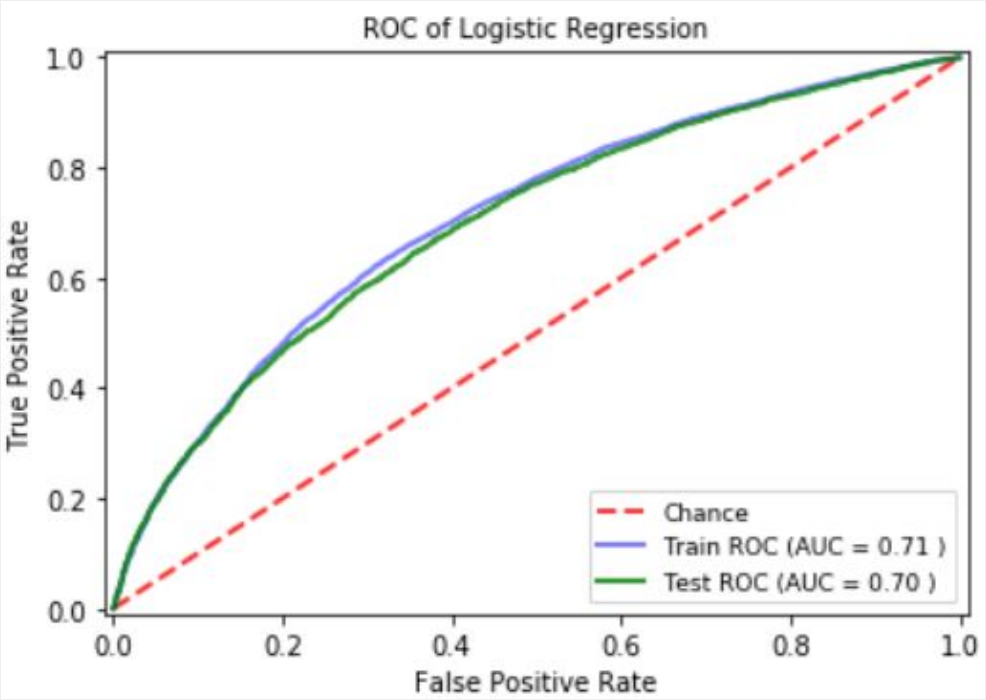
- Weak correlation with target (shares)

- Heteroscedastic data relationship

BINARY CLASSIFICATION

MODELS & METRICS

BASE MODEL



MODELS

- Logistic Regression
- Decision Tree
- Random Forest
- AdaBoost
- Gradient Boost
- Support Vector Machine

METRICS

- ROC AUC Score
- Accuracy

Classification Report:

	precision	recall	f1-score	support
0	0.65	0.66	0.65	6072
1	0.64	0.62	0.63	5822
accuracy			0.64	11894
macro avg	0.64	0.64	0.64	11894
weighted avg	0.64	0.64	0.64	11894

PARAMETER TUNING

RandomizedSearchCV used for hyperparameter tuning as it is not that computationally expensive

```
In [17]: dtrscvall = DecisionTreeClassifier(random_state = 0)

         params = {'max_depth': sp_randint(3,5),
                   'min_samples_split': sp_randint(2,7),
                   'min_samples_leaf': sp_randint(2,5),
                   'criterion': ['gini', 'entropy']}

         rand_search = RandomizedSearchCV(dtrscvall, param_distributions = params, cv = 5, random_state = 1)

         rand_search.fit(X,y)

         print(rand_search.best_params_)
```

- 1st params = {'max_depth': sp_randint(3,58), 'min_samples_split': sp_randint(2,50), 'min_samples_leaf': sp_randint(2,50), 'criterion': ['gini', 'entropy']}

- 2nd params = {'max_depth': sp_randint(3,50), 'min_samples_split': sp_randint(2,47), 'min_samples_leaf': sp_randint(2,48), 'criterion': ['gini', 'entropy']}

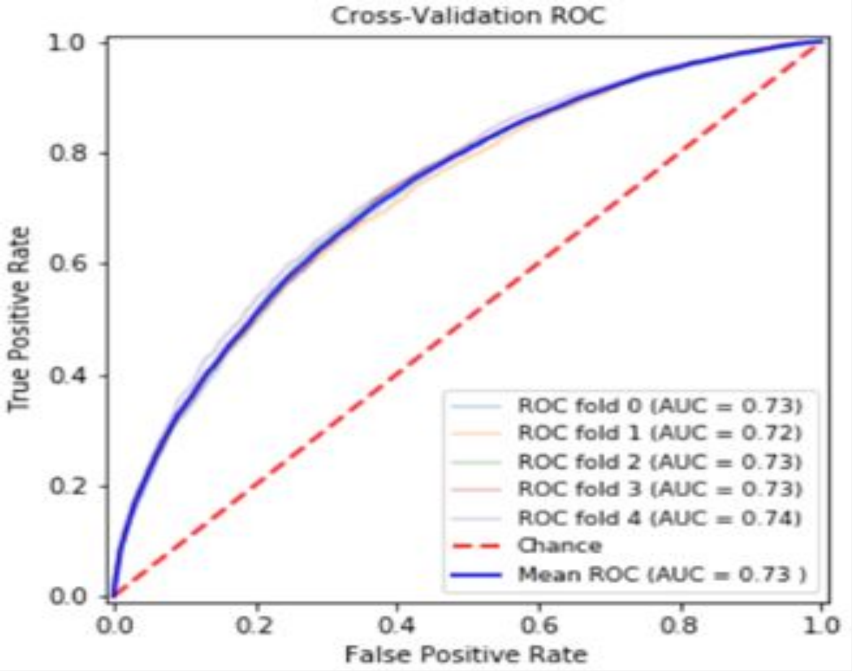
RESULTS

	ROC AUC		ACCURACY	
MODELS	Train	Test	Train	Test
Logistic Regression (Base Model)	70.65	69.75	65.61	64.31
RFE + Logistic Regression	69.19	68.43	64.45	62.95
Decision Tree	65.82	64.98	62.89	62.42
Tuned Decision Tree (20)	70.74	68.31	65.05	63.09
RFE + Decision Tree	65.68	64.98	62.88	62.15
RFE + Decision Tree (Tuned)	70.76	68.28	65.11	63.09
Random Forest	70.13	69.29	64.71	64.25
Tuned Random Forest (20)	72.25	69.55	66.28	64.23
RFE + Random Forest	70.68	69.70	64.84	64.56
RFE + Random Forest (Tuned)	73.07	69.99	67.20	64.57
Support Vector Machine	69.99	68.99	64.49	63.31
Boosted RFE + DT	72.94	69.88	66.79	64.27
Boosted RFE + RF	74.49	71.84	67.87	65.79
Gradient Boost	71.67	70.23	65.88	64.99
Boosted RFE + Support Vector Machine	68.03	67.14	60.48	59.71
Boosted RFE + DT (Tuned)	73.31	70.00	66.93	64.32
Boosted RFE + RF (Tuned)	**75.36**	**72.01**	**68.65**	**65.87**
Gradient Boost (Tuned)	75.05	71.87	68.35	65.98
Boosted RFE + DT (Tuned)/Boosted RFE + RF (Tuned)/ Gradient Boost (Tuned)	75.14	71.90	68.33	65.92
Boosted RFE + RF (Tuned)/Gradient Boost (Tuned)	75.10	71.90	68.33	66.01

FINAL MODEL

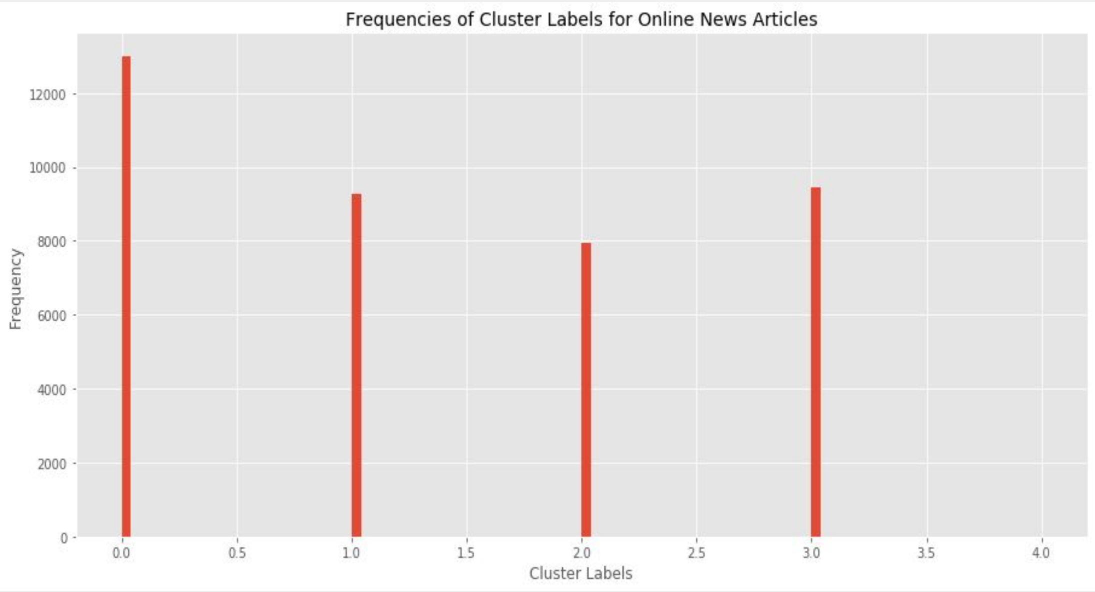
	After 5-fold Cross Validation	
MODELS	**ROC AUC**	**ACCURACY**
Boosted RFE + RF (Tuned)	**73**	**66.77**
Gradient Boost(Tuned)	72.76	66.72
Boosted RFE + RF (Tuned)/Gradient Boost(Tuned)	72.81	66.71

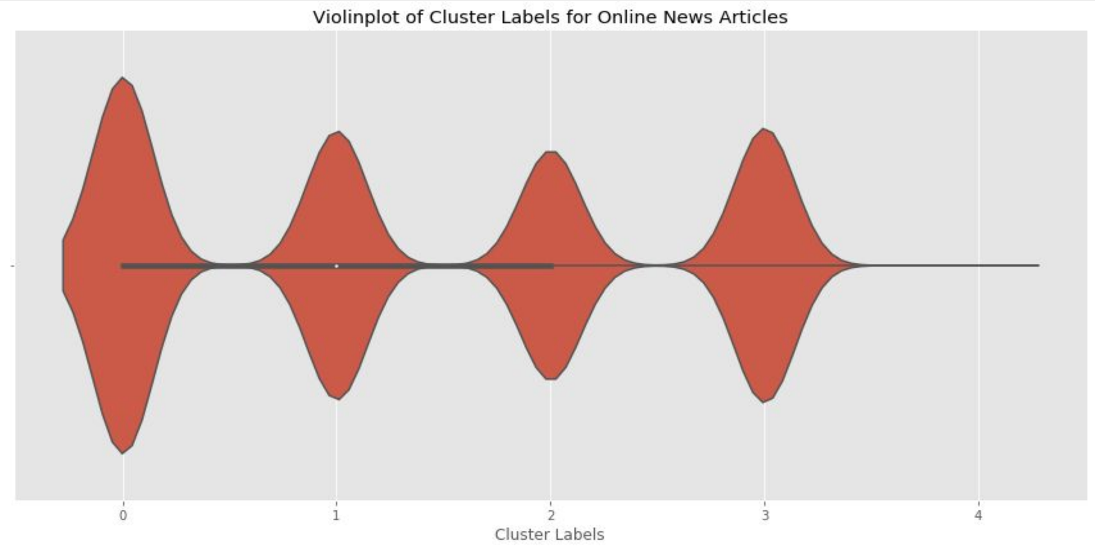
The Random Forest model with AdaBoost and hyperparameter tuning is the best model with ROC AUC score (73 %) and Accuracy (66%).



MULTICLASS CLASSIFICATION

CLUSTER ANALYSIS





CLUSTERS

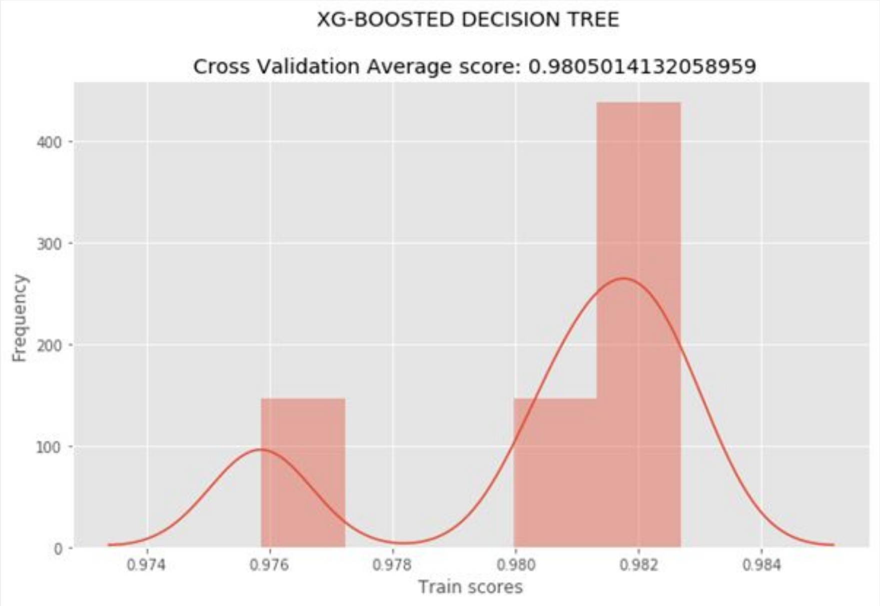
- **Cluster 0** (Entertainment, LDA_03)
- **Cluster 1** (World, LDA_02)
- **Cluster 2** (Business, LDA_00)
- **Cluster 3** (Tech, LDA_02)
- **Cluster 4** (No known LDA, data channel)

RESULTS

APPROACH	FEATURE ENGINEERING	EVALUATION METRICS	BASE MODEL		BAGGING		BOOSTING			
							ADA-BOOST		GRADIENT-BOOST	XG-BOOST
		MODEL	Decision Tree	Random Forest	Decision Tree	Random Forest	Decision Tree	Random Forest	Decision Tree	Decision Tree
KMEANS + SCALED DATASET (SMOT) (60+1 features)	ALL FEATURES	TRAIN SCORE	0.96	0.98	0.97	0.98	0.98	0.98	0.97	0.99
		TEST SCORE	0.93	0.96	0.96	0.96	0.96	0.97	0.96	0.97
		KAPPA COHEN TRAIN SCORE	0.95	0.97	0.97	0.97	0.97	0.98	0.97	0.98
		KAPPA COHEN TEST SCORE	0.89	0.93	0.93	0.93	0.93	0.95	0.94	0.96
PCA + KMEANS + SCALED DATASET (60+1 features)	ALL FEATURES	TRAIN SCORE	0.95	0.96	0.96	0.96	0.96	0.97	0.97	0.98
		TEST SCORE	0.94	0.96	0.95	0.96	0.96	0.97	0.97	0.97
		KAPPA COHEN TRAIN SCORE	0.93	0.95	0.94	0.95	0.95	0.96	0.96	0.97
		KAPPA COHEN TEST SCORE	0.92	0.95	0.94	0.95	0.94	0.95	0.95	0.97
PCA + KMEANS + SCALED DATASET (60+1 features)	ALL FEATURES	CROSS VALIDATION SCORE	0.95	0.96	0.96	0.96	0.96	0.97	0.97	0.98
PCA + KMEANS + PCA COMPONENTS (38+1 features)	PCA	TRAIN SCORE	0.92	0.96	0.96	0.96	0.97	0.97	0.96	0.98
		TEST SCORE	0.9	0.96	0.95	0.96	0.96	0.96	0.96	0.97
		KAPPA COHEN TRAIN SCORE	0.90	0.95	0.94	0.95	0.95	0.96	0.95	0.97
		KAPPA COHEN TEST SCORE	0.87	0.94	0.93	0.95	0.95	0.95	0.95	0.97
PCA + KMEANS + SCALED DATASET (17+1 features)	FEATURE IMPORTANCE	TRAIN SCORE	0.95	0.96	0.96	0.96	0.95	0.96	0.96	0.96
		TEST SCORE	0.94	0.96	0.95	0.95	0.95	0.96	0.96	0.96
		KAPPA COHEN TRAIN SCORE	0.93	0.95	0.94	0.95	0.94	0.94	0.95	0.95
		KAPPA COHEN TEST SCORE	0.92	0.94	0.93	0.94	0.93	0.94	0.94	0.95
PCA + KMEANS + SCALED DATASET (30+1 features)	RFE	TRAIN SCORE	0.95	0.96	0.96	0.96	0.96	0.97	0.97	0.98
		TEST SCORE	0.94	0.96	0.95	0.96	0.96	0.97	0.96	0.97
		KAPPA COHEN TRAIN SCORE	0.93	0.95	0.94	0.95	0.95	0.96	0.96	0.97
		KAPPA COHEN TEST SCORE	0.92	0.95	0.94	0.95	0.95	0.95	0.95	0.96
PCA + KMEANS + SCALED DATASET (13 features)	RFE CV	TRAIN SCORE	0.94	0.95	0.95	0.95	0.95	0.95	0.95	0.96
		TEST SCORE	0.93	0.95	0.94	0.95	0.95	0.95	0.95	0.96
		KAPPA COHEN TRAIN SCORE	0.92	0.94	0.93	0.94	0.93	0.94	0.94	0.95
		KAPPA COHEN TEST SCORE	0.91	0.94	0.92	0.94	0.93	0.93	0.94	0.94
PCA + KMEANS + SCALED DATASET (3+1 features)	CORRELATION MATRIX ANALYSIS	TRAIN SCORE	0.77	0.79	0.78	0.79	0.76	0.76	0.76	0.78
		TEST SCORE	0.76	0.78	0.78	0.78	0.75	0.71	0.76	0.78
		KAPPA COHEN TRAIN SCORE	0.68	0.71	0.71	0.71	0.67	0.68	0.68	0.70
		KAPPA COHEN TEST SCORE	0.67	0.71	0.70	0.70	0.66	0.61	0.68	0.70
PCA + KMEANS + SCALED DATASET (17+1 features)	SELECT K BEST (CHI SQUARE)	TRAIN SCORE	0.95	0.95	0.95	0.95	0.95	0.96	0.96	0.96
		TEST SCORE	0.94	0.95	0.95	0.95	0.95	0.95	0.96	0.96
		KAPPA COHEN TRAIN SCORE	0.93	0.94	0.93	0.94	0.93	0.94	0.94	0.95
		KAPPA COHEN TEST SCORE	0.92	0.94	0.93	0.93	0.93	0.94	0.94	0.94
PCA + KMEANS + SCALED DATASET (17+1 features)	SELECT K BEST (MUTUAL INFORMATION)	TRAIN SCORE	0.94	0.96	0.96	0.96	0.96	0.96	0.96	0.96
		TEST SCORE	0.94	0.95	0.95	0.95	0.95	0.96	0.96	0.96
		KAPPA COHEN TRAIN SCORE	0.92	0.94	0.94	0.94	0.94	0.94	0.94	0.95
		KAPPA COHEN TEST SCORE	0.91	0.94	0.93	0.94	0.94	0.94	0.94	0.95

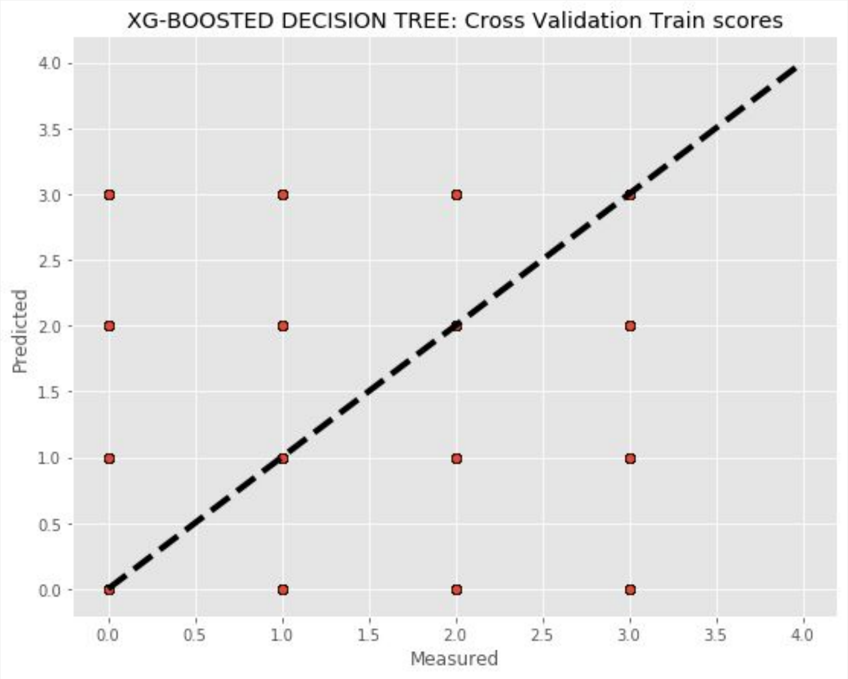
XGBOOST: Decision Tree

Multi Classification using Unsupervised approach (KMeans) with Scaled 38 (Independent) Principal Components (PC) and 1 (Dependent) features gave the best result using xg-boost for decision tree.

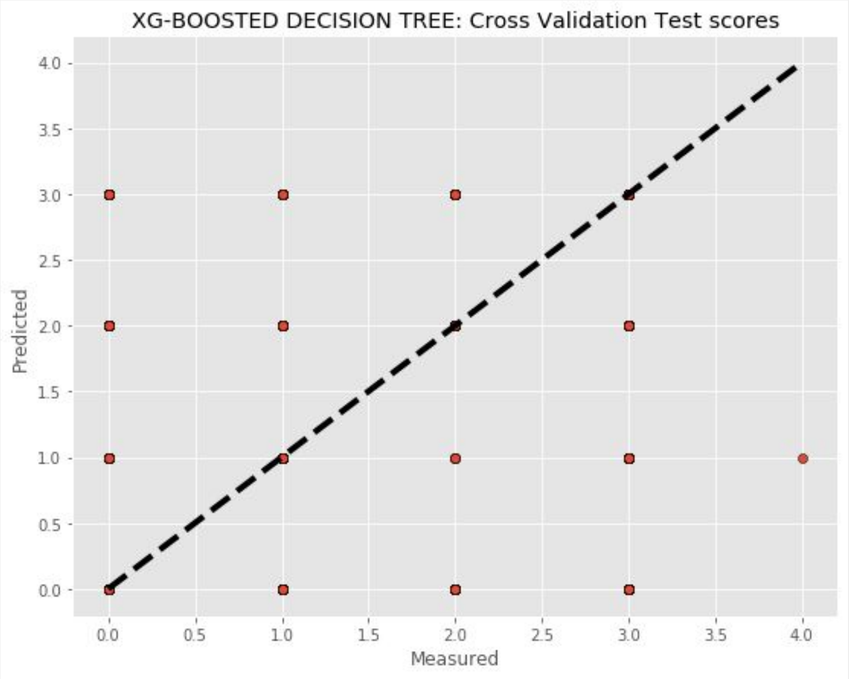


Randomised Search Cross Validation (RSCV) on Train data

RANDOM SEARCH CROSS VALIDATION



Randomised Search Cross Validation (RSCV) on Train data



Randomised Search Cross Validation (RSCV) on Test data

METRICS

	precision	recall	f1-score	support
0	0.98	0.99	0.98	9099
1	0.99	0.98	0.98	6524
2	0.98	0.98	0.98	5558
3	0.98	0.97	0.98	6569
accuracy			0.98	27750
macro avg	0.98	0.98	0.98	27750
weighted avg	0.98	0.98	0.98	27750

XG-Boosted Decision Tree Classification Report for Train data

```
[[8982   26   56   35]
 [  70 6387   34   33]
 [  68   19 5422   49]
 [  91   50   44 6384]]
```

XG-Boosted Decision Tree Confusion Matrix for Train data

	precision	recall	f1-score	support
0	0.97	0.98	0.98	3879
1	0.98	0.97	0.98	2742
2	0.97	0.97	0.97	2377
3	0.98	0.97	0.97	2895
4	0.00	0.00	0.00	1
accuracy			0.97	11894
macro avg	0.78	0.78	0.78	11894
weighted avg	0.97	0.97	0.97	11894

XG-Boosted Decision Tree Classification Report for Test data

```
[[3812   19   27   21    0]
 [  45 2660   19   18    0]
 [  33    6 2314   24    0]
 [  46   22   24 2803    0]
 [   0    1    0    0    0]]
```

XG-Boosted Decision Tree Confusion Matrix for Test data

METRICS

KAPPA COHEN SCORES

Train score: 0.972

Test score: 0.965

The f1-weighted train score of 0.98, test score of 0.97 and kappa cohen train score of 0.97 and test score of 0.97 from XG Boosted Decision Tree seemed to be the most preferred for Multi-Classification of News Articles based on category (LDA, Data Channel).

RECOMMENDATIONS

An article should satisfy the below points to gain popularity:

- Title length (7 -19 words)
- Short Articles (382-2591 words)
- No. of images and videos (0-2)
- Few number of links (3-5)
- The title/content of articles should be subjective
- Publish the articles on weekends

To increase popularity:

- Increase the number of articles related to Social media / Lifestyle
- Increase the articles published on weekends
- Include articles that have positive sentiments that can relate to people

IMPROVEMENTS

Data related to the following can be added:

- Publication time (date, month, year, timestamp)
- Keywords for each article so that Sentiment Analysis, NLP can be applied on the data to further improve insights on online news articles.
- Make known the names of the LDA topics as well as their characteristics to better understand news article categories.
- Find the polarity (positive and negative) of keywords for better insights on the news articles and enhance popularity.

THANK YOU