How can we distinct Iron Man and Batman

Project 3

Problem Statement

As for some lazy reddit's fandom of Iron Man that knew about Billionaire create armour suit and drive luxury car

They want to distinct between Iron man and other Heroes that Billionaire too and also wear a suit but in black

Batman

Problem Statement

So we decide to prove our model by crawl from these 2 subreddit post and train out model

https://www.reddit.com/r/ironman/

https://www.reddit.com/r/batman/

We expected that this model can distinct Ironman out of Batman forum that make us easily add it to our collections to easier to check iron man updates

Data Collection

We crawl to subreddit forum via `.json` format

By target sampling for each subreddit post is 1,000 posts

After collect data we get raw posts

- ironman 1019 posts
- batman for 1012 posts

Data Cleaning and EDA

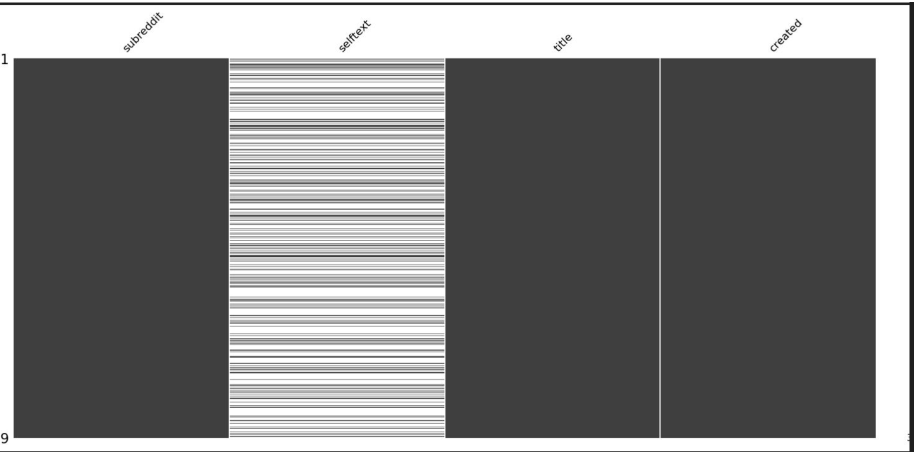
After explore raw data from subreddit 116 columns

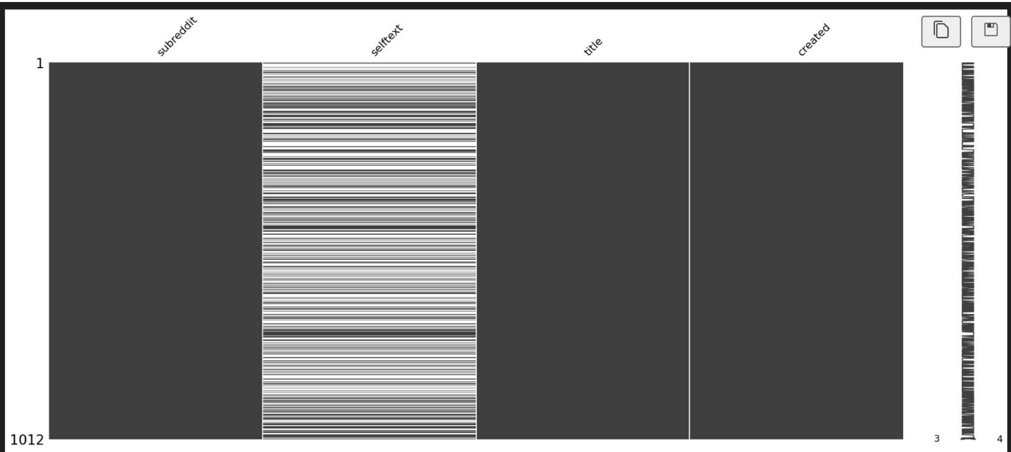
With scope to NLP I decided to use these columns to modeling

- Subreddit -> to be target result
- Title -> post’s title
- Selftext -> post’s content

Data Cleaning and EDA

Check null information and found out that some rows got `selftext` null but still included information that help modeling





Data Cleaning and EDA

So I decide to merge two columns `title` and `selftext` into new column `content`

Data Cleaning and EDA

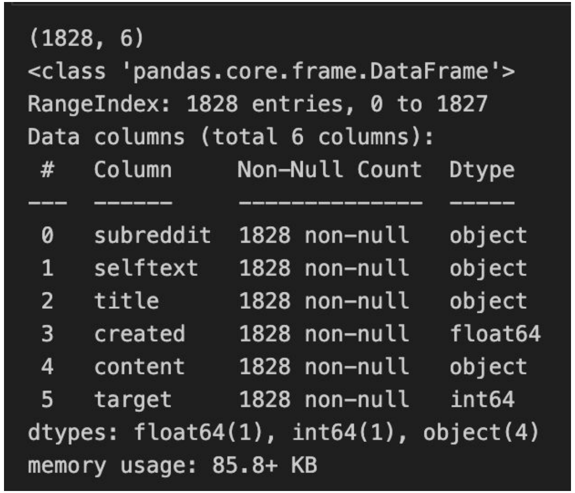
For fix leaked information we remove words `bat man` from batman data `content` and also `iron man` from ironman data `content`

And Remove Duplicate data for each data source

Convert `Subreddit` column into `Target`

with binary value

After that merge 2 this data into one data frame



```
(1828, 6)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1828 entries, 0 to 1827
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   subreddit  1828 non-null   object 
 1   selftext   1828 non-null   object 
 2   title      1828 non-null   object 
 3   created    1828 non-null   float64
 4   content    1828 non-null   object 
 5   target     1828 non-null   int64  
dtypes: float64(1), int64(1), object(4)
memory usage: 85.8+ KB
```

Modeling

For NLP we need to do 2 things

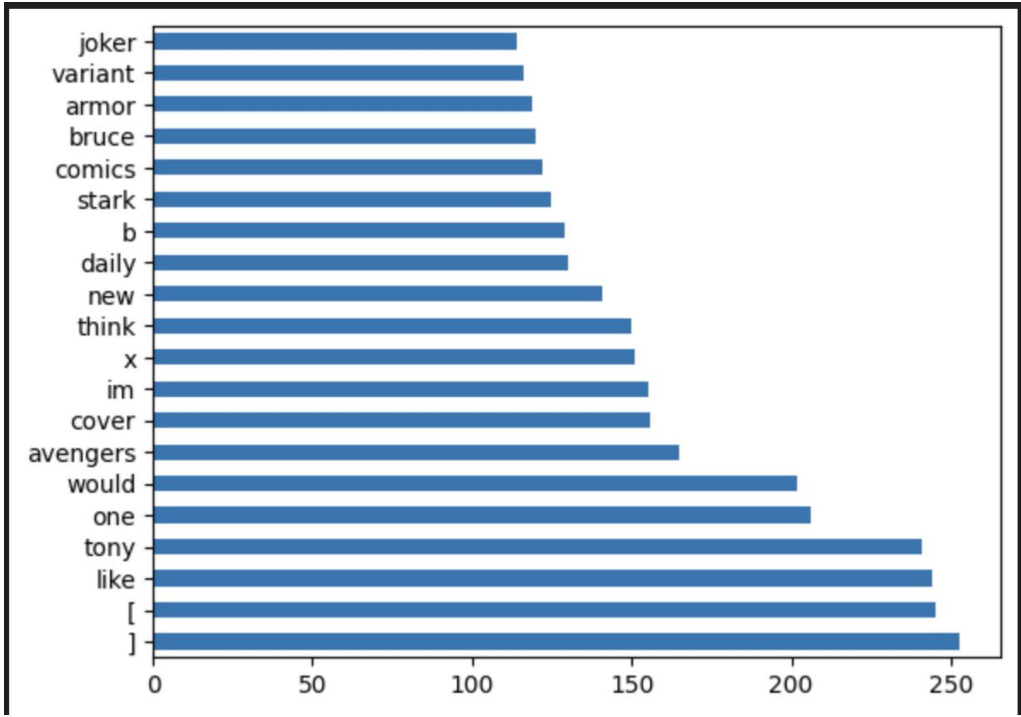
1. Words tokenize
2. Classifier

Modeling - tokenize

I decide to custom tokenize and found out that ...

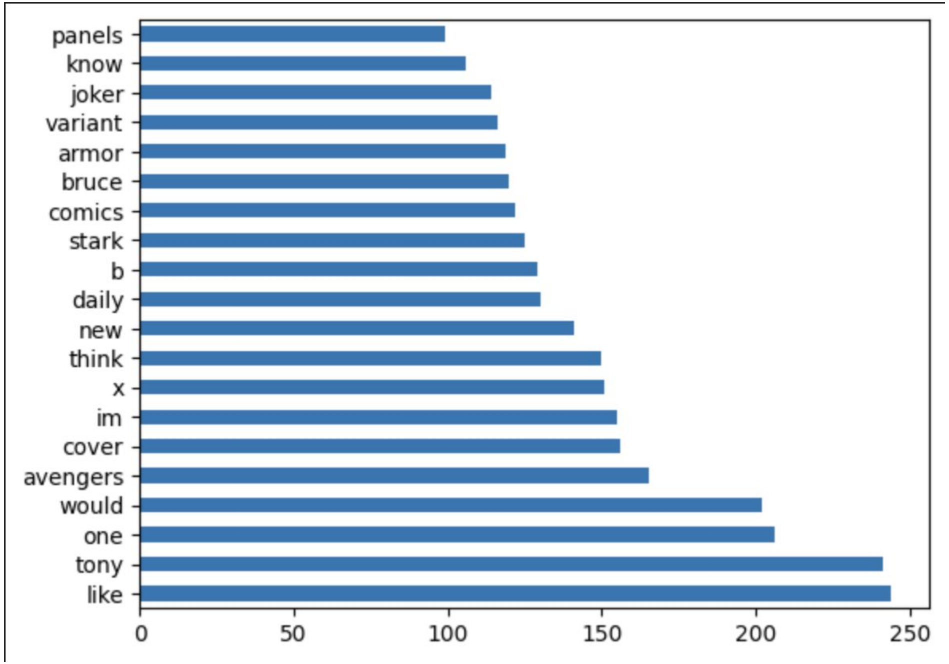
`[` , `]`

is most frequency and cannot predict any value so i decide to remove this with my custom tokenize



Modeling - tokenize improve

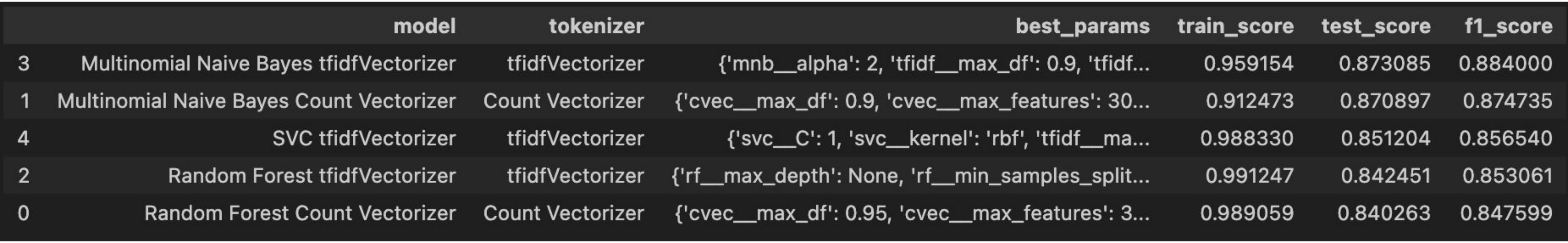
After remove `[`, `]`



Modeling

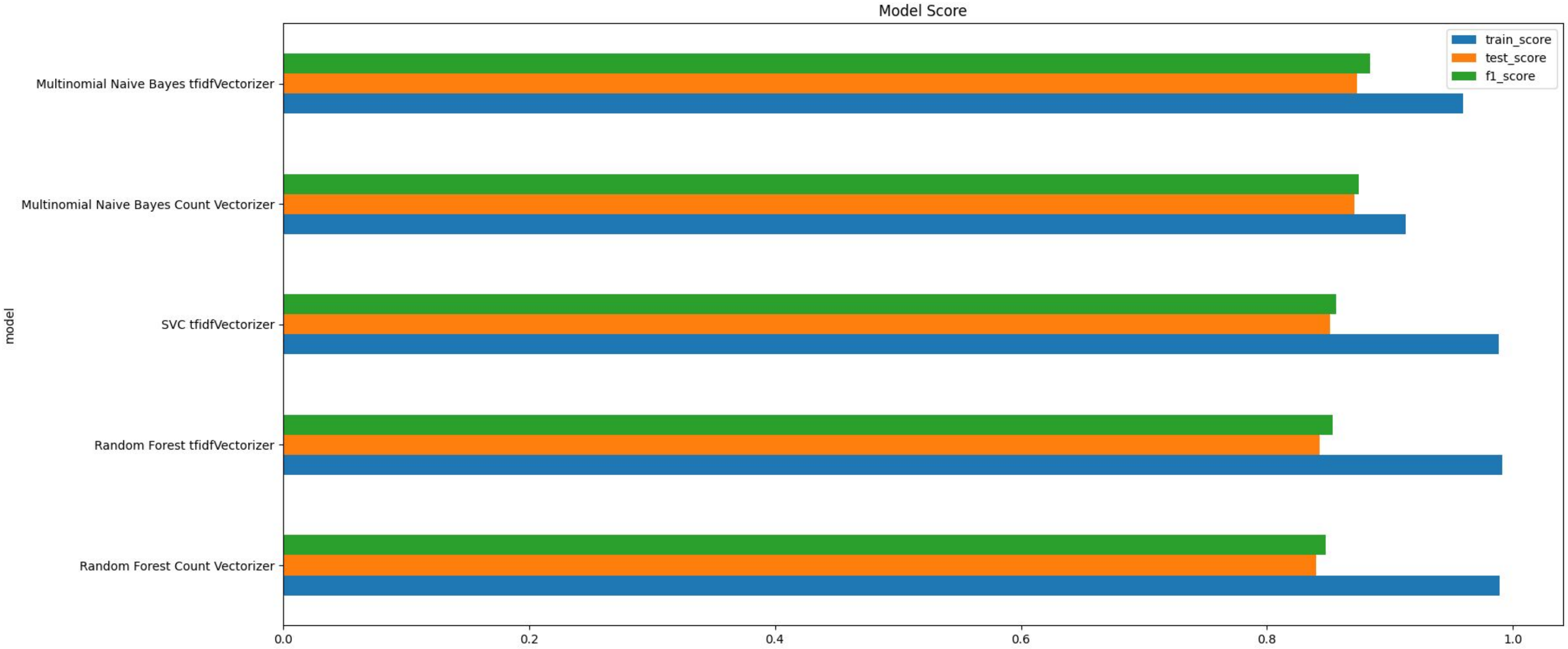
After improve tokenize I use GridSearch with

2 Vectorizer and 3 Classifier with vary parameters



	model	tokenizer	best_params	train_score	test_score	f1_score
3	Multinomial Naive Bayes tfidfVectorizer	tfidfVectorizer	{'mnb__alpha': 2, 'tfidf__max_df': 0.9, 'tfidf...	0.959154	0.873085	0.884000
1	Multinomial Naive Bayes Count Vectorizer	Count Vectorizer	{'cvec__max_df': 0.9, 'cvec__max_features': 30...	0.912473	0.870897	0.874735
4	SVC tfidfVectorizer	tfidfVectorizer	{'svc__C': 1, 'svc__kernel': 'rbf', 'tfidf__ma...	0.988330	0.851204	0.856540
2	Random Forest tfidfVectorizer	tfidfVectorizer	{'rf__max_depth': None, 'rf__min_samples_split...	0.991247	0.842451	0.853061
0	Random Forest Count Vectorizer	Count Vectorizer	{'cvec__max_df': 0.95, 'cvec__max_features': 3...	0.989059	0.840263	0.847599

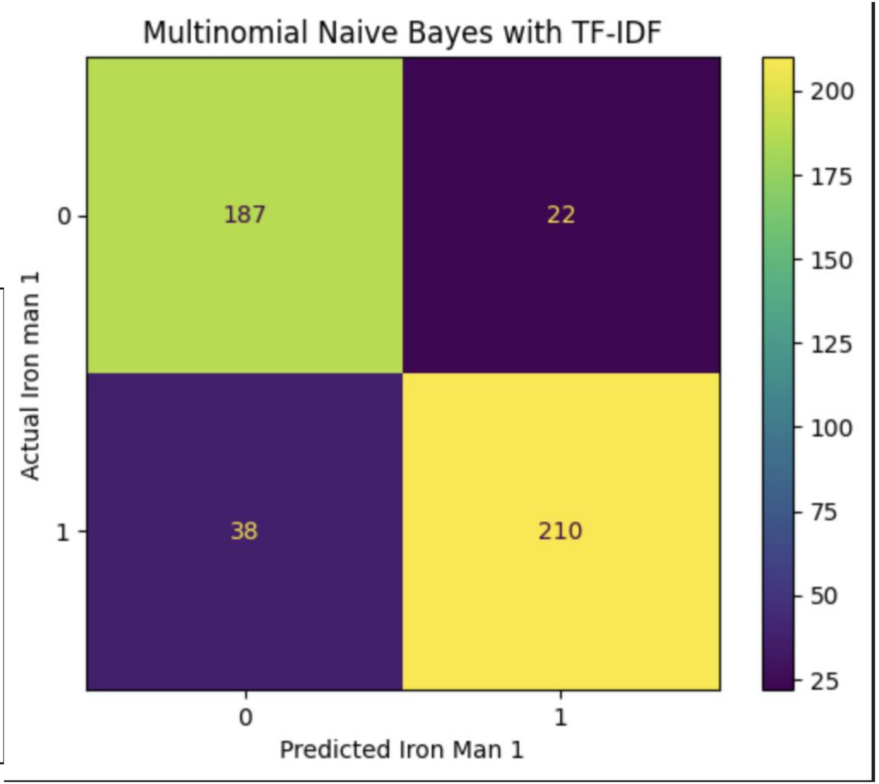
Modeling


Model Score
train_score
test_score
f1_score
Multinomial Naive Bayes tfidfVectorizer
Multinomial Naive Bayes Count Vectorizer
SVC tfidfVectorizer
Random Forest tfidfVectorizer
Random Forest Count Vectorizer
model
0.0
0.2
0.4
0.6
0.8
1.0

Evaluate

Multinomial Naive Bayes tfidfVectorizer make best f1_score

That I focus more that Precision or Recall since this modeling didn’t need to focus on any false positive or false negative that matter



Evaluate

If we dig deeper into why model predict fail

We found out some interesting content

Why so Serious?

GPU Holder

If Tom Hardy played Walter White

Evaluate

Our model predict this quote as Ironman

But it actually came from Batman

Let guess why?

Thanos #1 (feat)https://youtu.be/9FEDbhepU9I?si=0nO-8Ku5lth35sTO

RhodeyWhy did they recast rhodey the first rhodey and RDJ had alot more chemistry and was better

Evaluate

Our model predict this quote as Batman

But it actually came from Ironman

Let guess why?

Conclusion

For better score for our modeling we can improve with more weight word in each series eg. RDJ

- remove images, Meme post if we focus on NLPs
- add bags of focus words that included in IronMan universe eg. characters name, actors name, quote in specific topic we focus