Predicting yelp Reviews

Varun Uppala and Gwynie Dunlevy

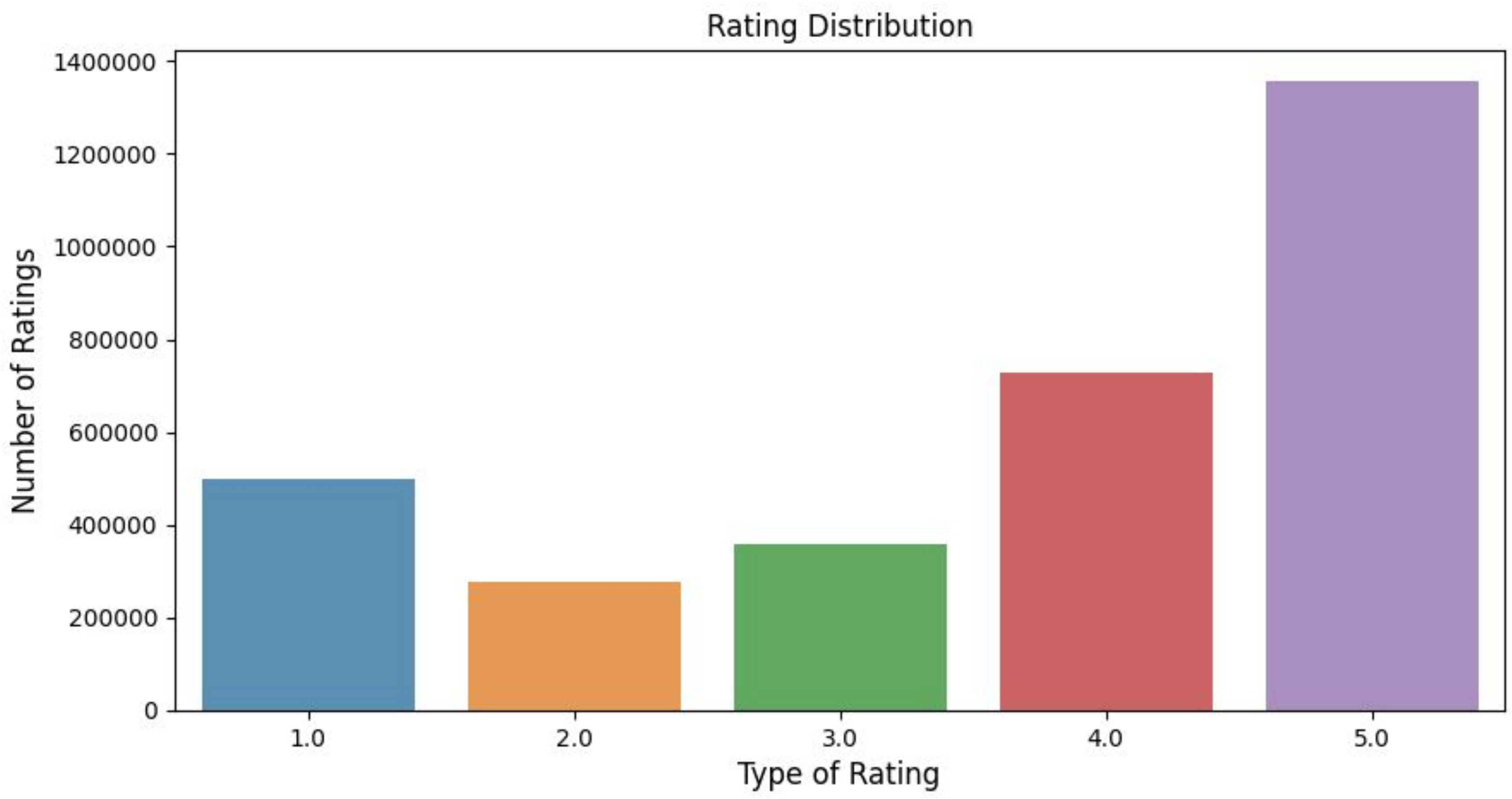
Problem Statement: Is taking the text from a review, accurate enough to find the star rating?

Recap: Our Plan

- Data only from restaurants in Florida
- Preprocessing
- Splitting up star reviews into their own CSV (1,2,3,4,5)
- Creating Unigrams and Bigrams
- Baseline (logistic regression, SVM, and SVM w/ lemmatization)
- Sentiment
- LDA to predict the star rating
- BONUS: Summarization of reviews

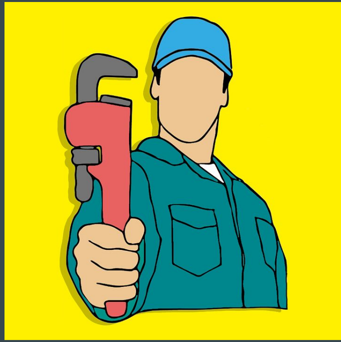
Dependencies Used

- Pandas
- Numpy
- Sklearn
- Scikit-learn
- Textblob
- Gensim
- PyLDAvis
- Json
- Collections
- Nltk
- Argparse
- Seaborn
- Matplotlib
- Contractions
- Afinn
- Vadersentiment
- Math
- String
- Sumy
- Warnings



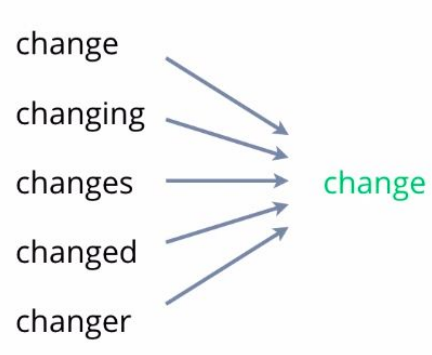
Rating Distribution
1400000
1200000
1000000
800000
600000
400000
200000
0
Number of Ratings
1.0
2.0
3.0
4.0
5.0
Type of Rating

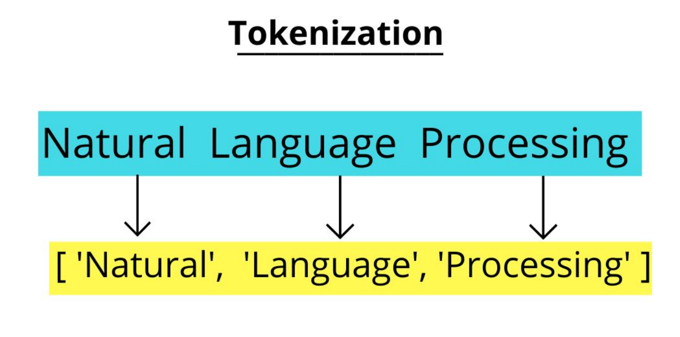
Feature Engineering



aren't - are not	I'm - I am	that's - that is

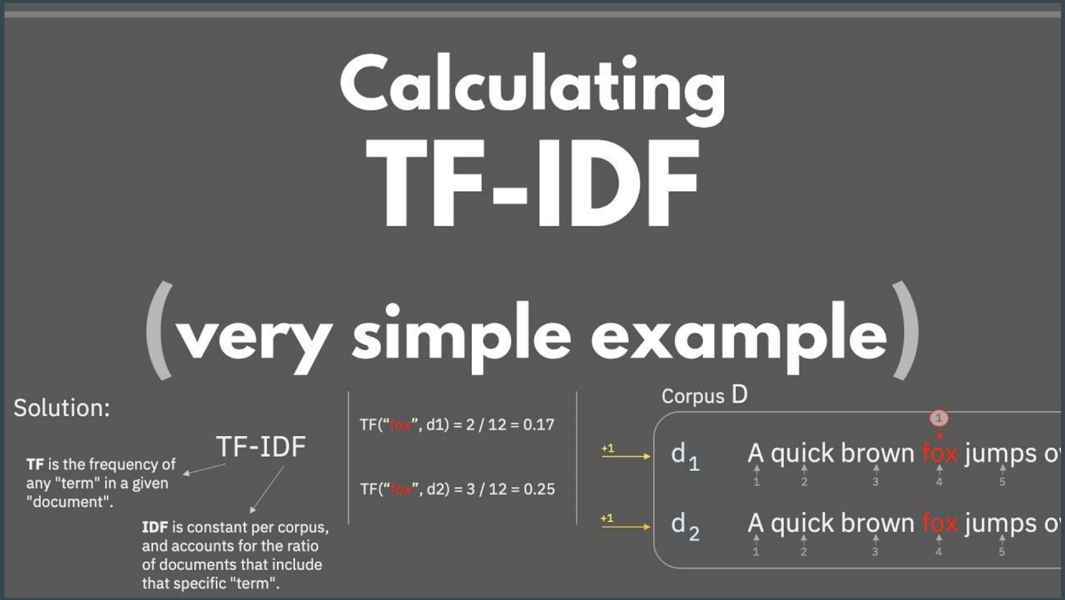
Raw	Lowercased
Canada CanadA CANADA	canada
TOMCAT Tomcat toMcat	tomcat





TF - IDF

Reflect how important a word is to a document in a collection or corpus.



Combinations For Prediction

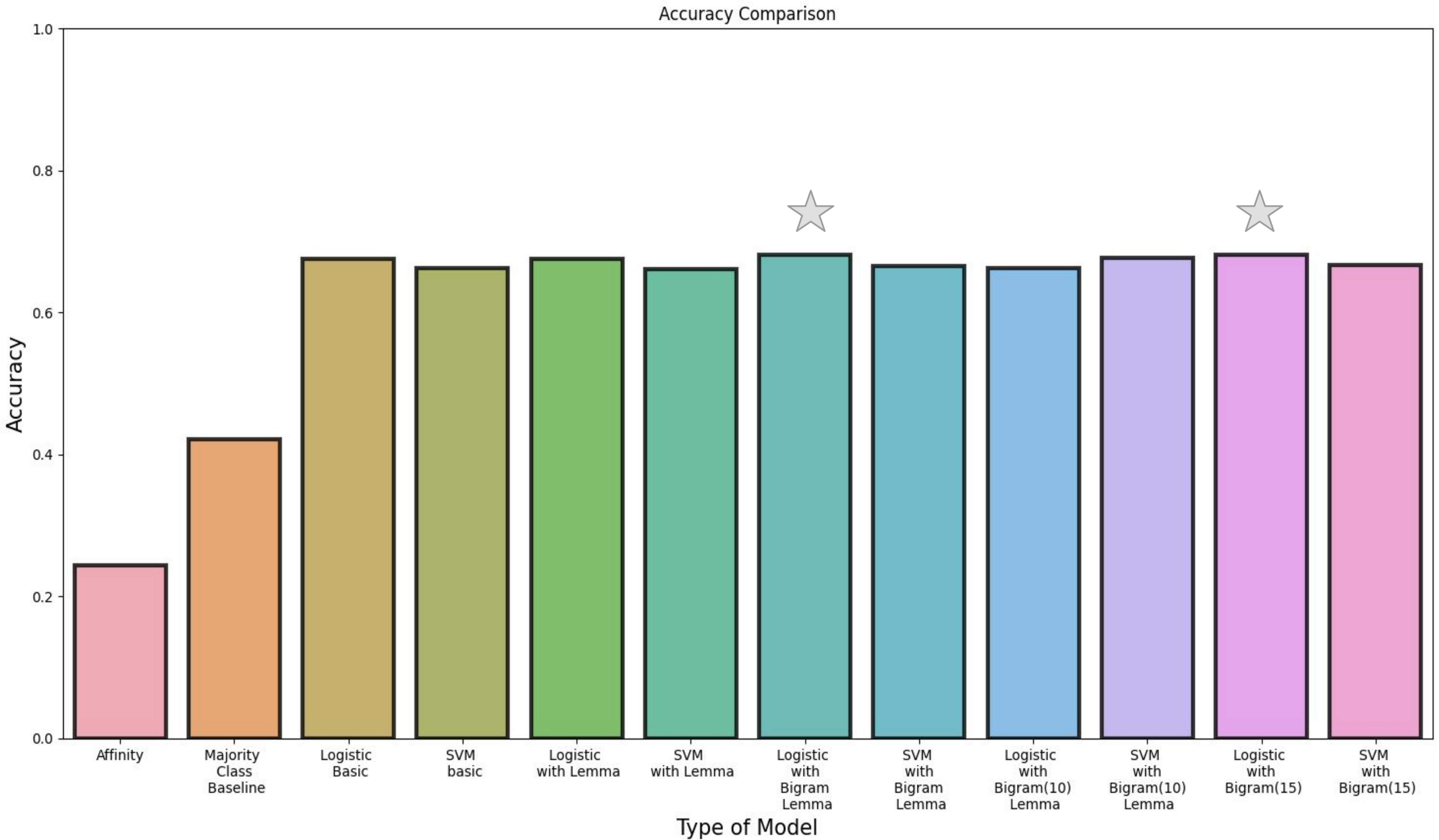
Models Used : Logistic Regression , Support Vector Machine

Number of Features :

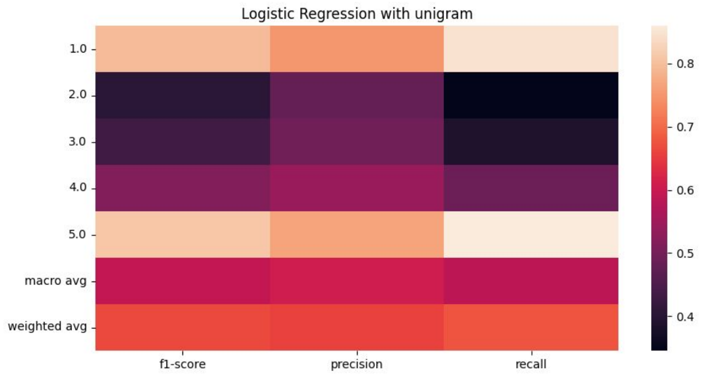
Features	Number
Unigram	372,201
Unigram with Lemma	356,040
Unigram and Bigram Lemma with 10 min counts	456,127
Unigram and Bigram Lemma with 15 min counts	422,389
Unigram and Bigram with 15 min counts	433,420

Evaluation

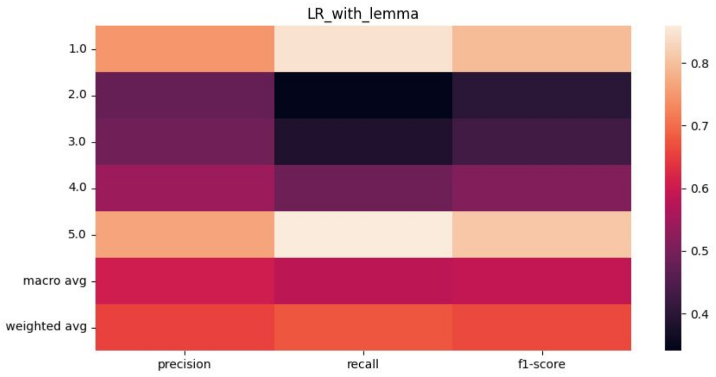
- Classification Report
- F1 score, Precision , Recall for each class
- Accuracy
- Macro Average
- Weighted Average
- Create Heatmaps for easier understanding
- Confusion Matrix for the Best SVM and Best LR model.


Accuracy Comparison
Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
Affinity
Majority Class Baseline
Logistic Basic
SVM basic
Logistic with Lemma
SVM with Lemma
Logistic with Bigram Lemma
SVM with Bigram Lemma
Logistic with Bigram(10) Lemma
SVM with Bigram(10) Lemma
Logistic with Bigram(15)
SVM with Bigram(15)
Type of Model

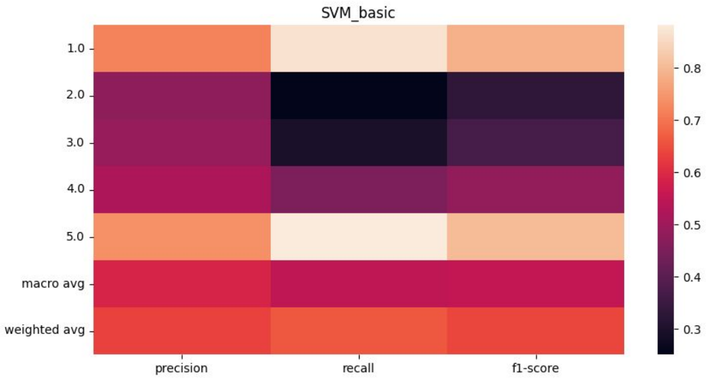
Baselines



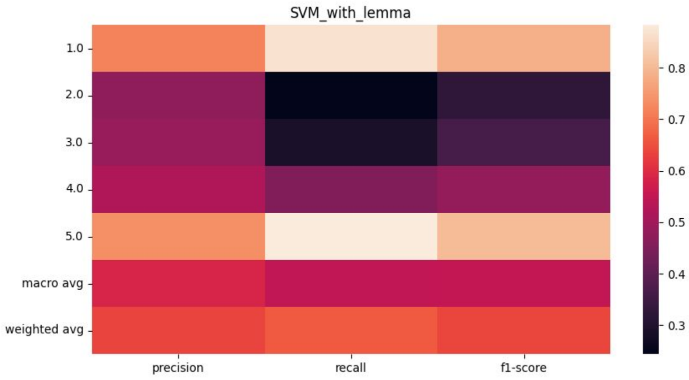
Logistic Regression with unigram
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
f1-score
precision
recall
0.8
0.7
0.6
0.5
0.4



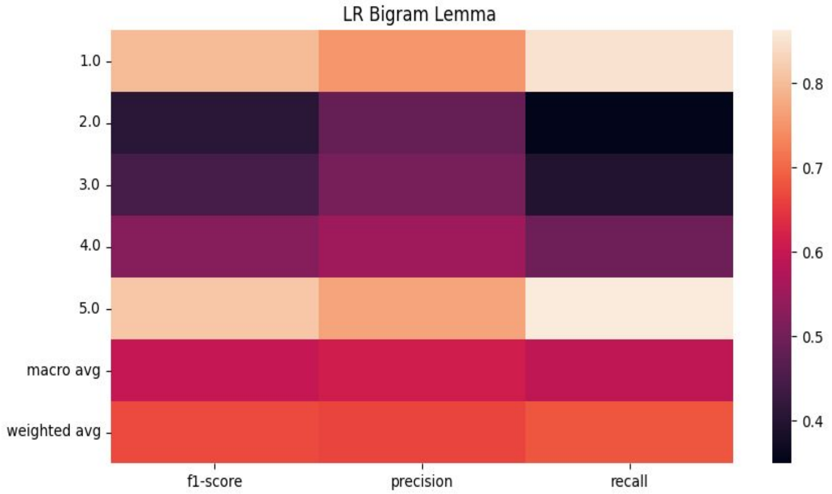
LR_with_lemma
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
precision
recall
f1-score
0.8
0.7
0.6
0.5
0.4

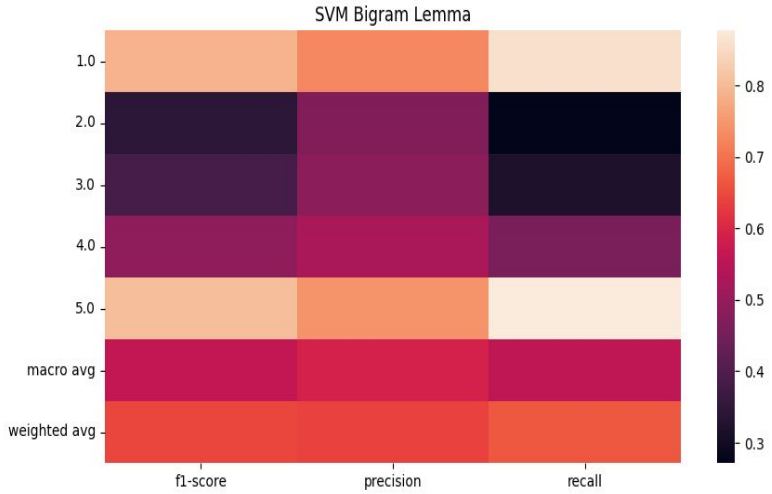


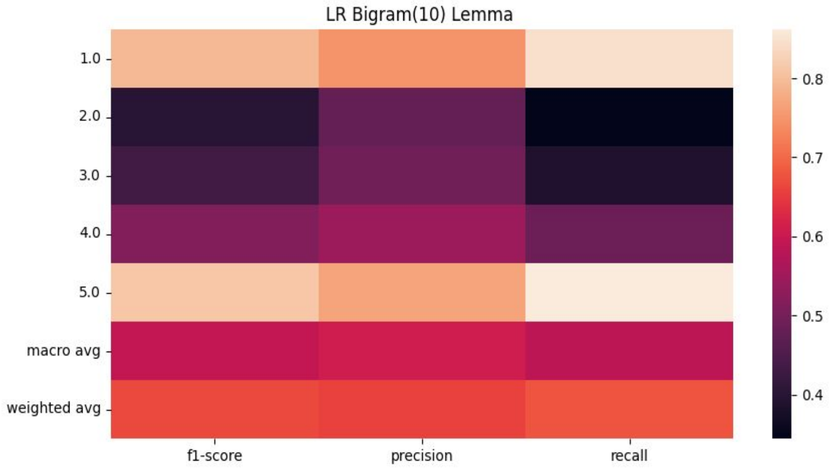
SVM_basic
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
precision
recall
f1-score
0.8
0.7
0.6
0.5
0.4
0.3

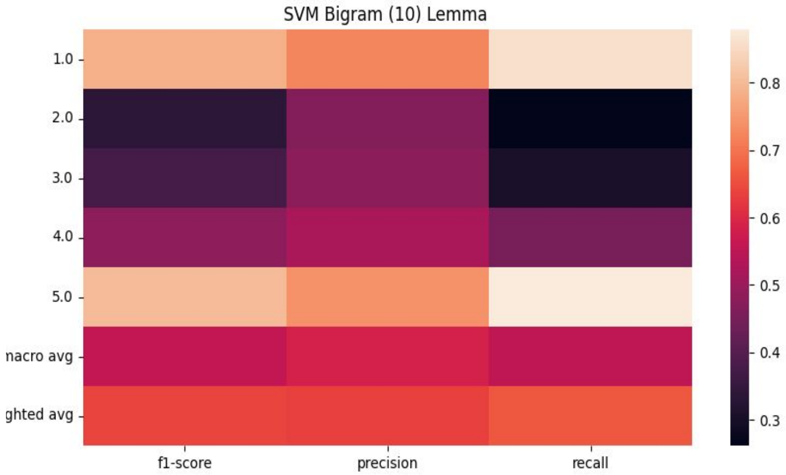


SVM_with_lemma
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
precision
recall
f1-score
0.8
0.7
0.6
0.5
0.4
0.3

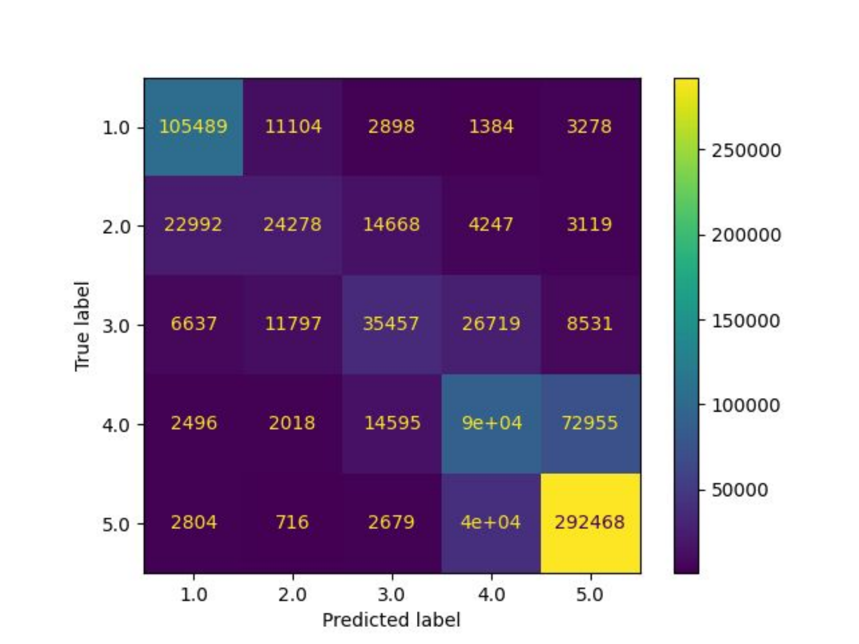

LR Bigram Lemma
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
f1-score
precision
recall
0.8
0.7
0.6
0.5
0.4

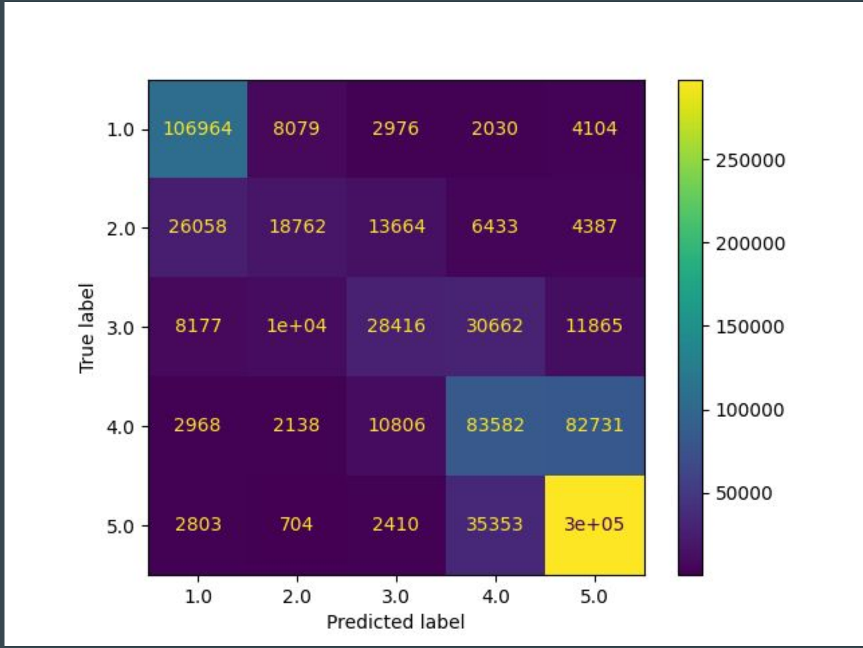

SVM Bigram Lemma
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
f1-score
precision
recall
0.8
0.7
0.6
0.5
0.4
0.3


LR Bigram(10) Lemma
1.0
2.0
3.0
4.0
5.0
macro avg
weighted avg
f1-score
precision
recall
0.8
0.7
0.6
0.5
0.4


SVM Bigram (10) Lemma
1.0
2.0
3.0
4.0
5.0
nacro avg
ghted avg
f1-score
precision
recall
0.8
0.7
0.6
0.5
0.4
0.3

Error Analysis

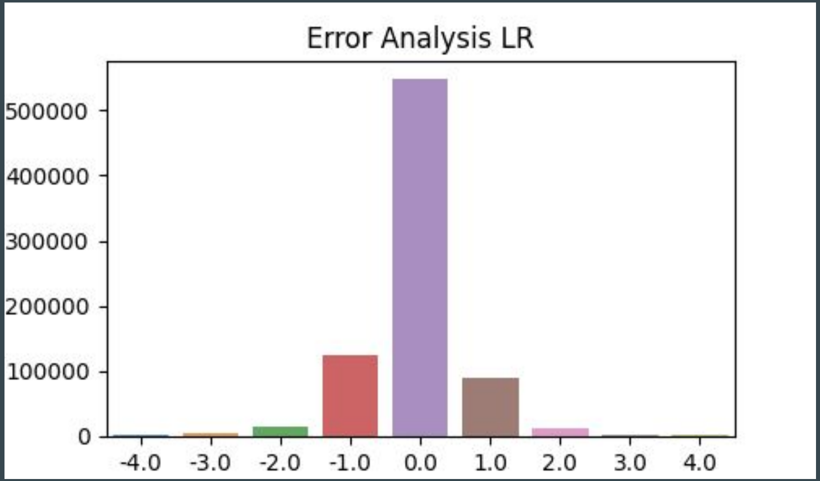


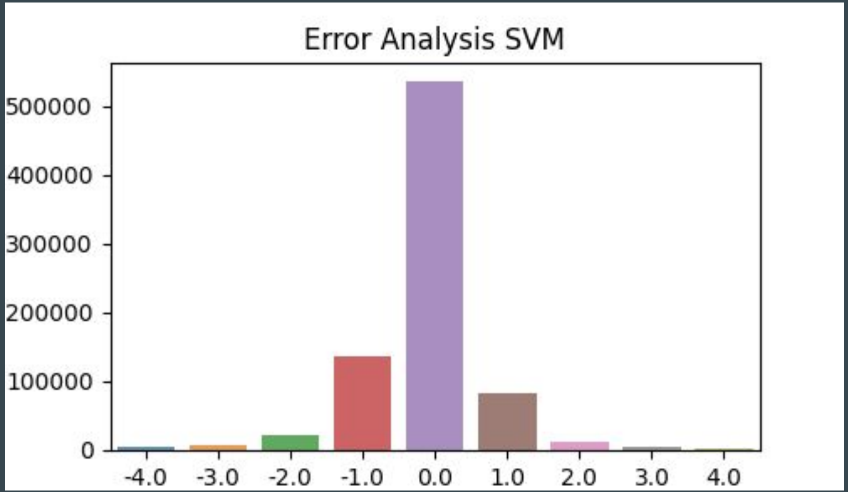


Continued ..

Shows the difference between Original label and Predicted Label.

Useful to check the deviation of the results..



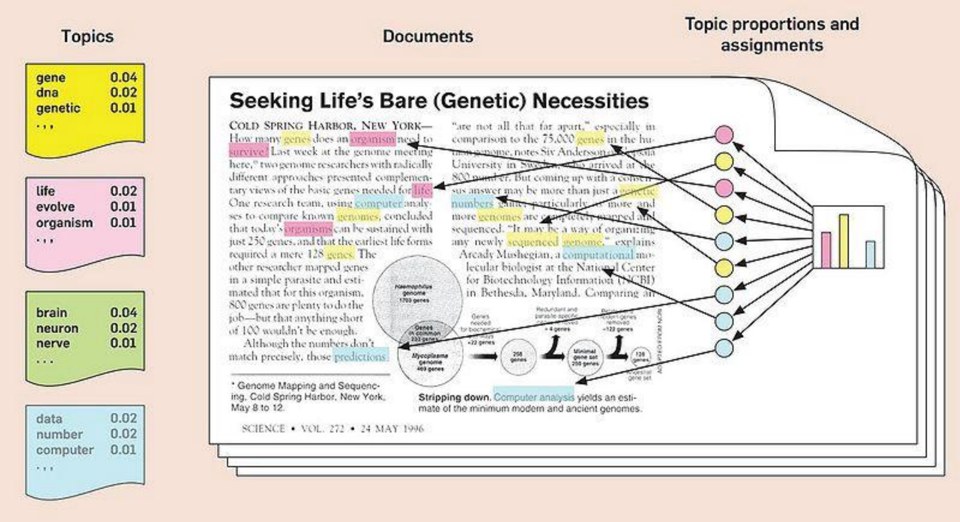


Latent Dirichlet Allocation

Hypothesizes that each document is a mixture of a small number of topics and that each word's presence is attributable to one of the document's topics.

To discover topics in a collection of documents, and then automatically classify any individual document within the collection in terms of how "relevant" it is to each of the discovered topics

Topic Labelling to be done by the user.



Time Taken for LDA

Parameters

Stars Dataset:

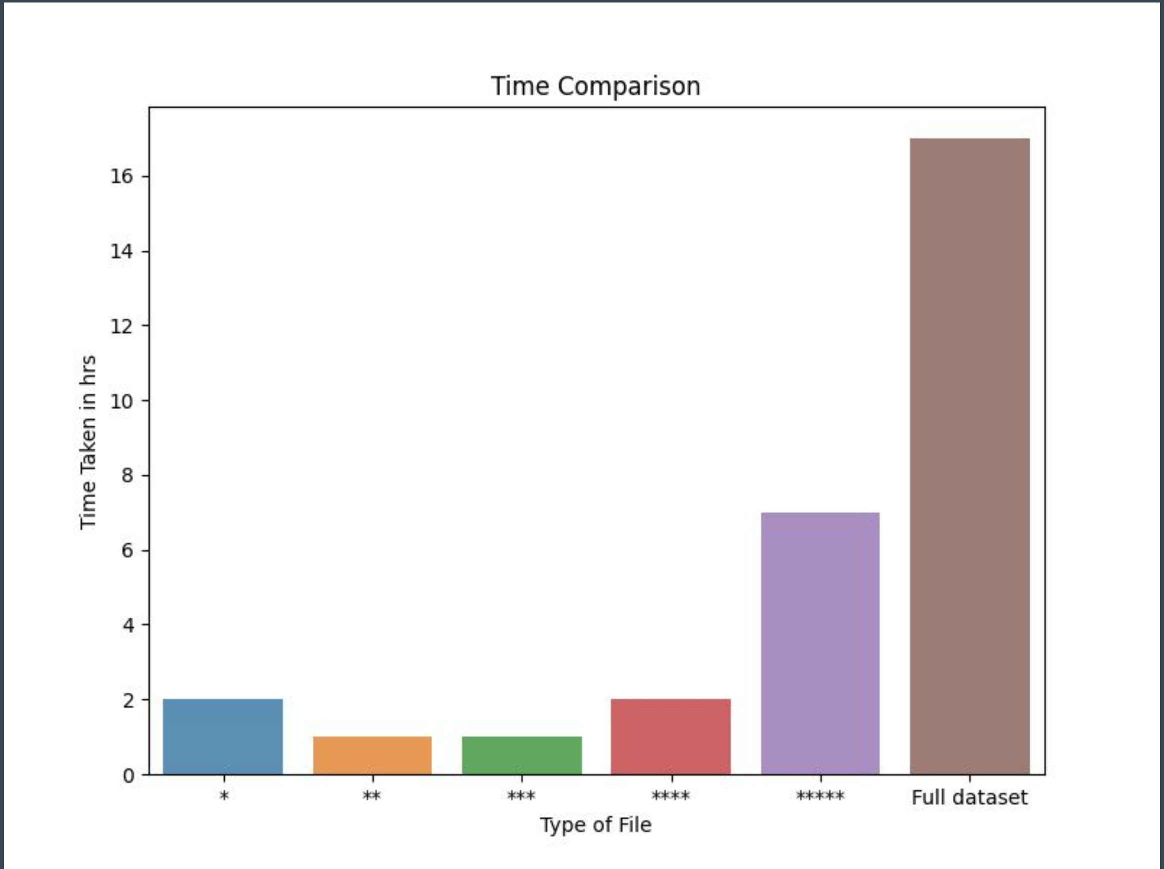
Num Topics : 20

Passes : 10

Full Dataset:

Num Topics : 15

Passes : 7



LDA Results

Example : Full Dataset

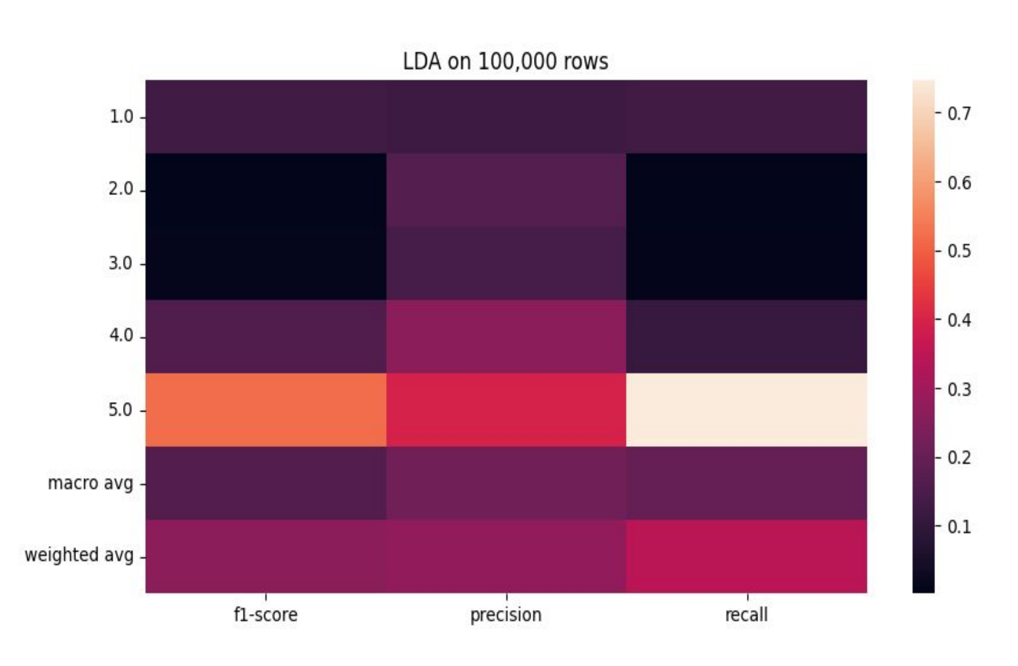
[(0, '0.049*"bar" + 0.045*"room" + 0.037*"night" + 0.032*"line" + 0.031*"inside" ' '+ 0.026*"open" + 0.026*"outside" + 0.024*"fun" + 0.023*"hotel" + ' '0.021*"walk"'),

(1, '0.123*"shop" + 0.119*"ok" + 0.113*"coffee" + 0.072*"mean" + 0.059*"others" ' '+ 0.055*"tea" + 0.044*"grab" + 0.043*"eye" + 0.039*"remember" + ' '0.037*"texture"'),

(2, '0.135*"kid" + 0.091*"cool" + 0.073*"brunch" + 0.055*"incredible" + ' '0.054*"ride" + 0.052*"average" + 0.047*"visiting" + 0.042*"lol" + ' '0.039*"sad" + 0.036*"mushroom"'),

Prediction Using LDA

- Rows used : 100,000
- Topics used : 15
- Accuracy : 34%
- Split 75 - 25
- Example array
  - [0.28,0.27 ....,0.15, len of review]
  - Use standard scaler
  - Fit to Logistic Regression Model



Conclusion: Predicting Ratings

- Logistic with Lemmatized Bigram and Logistic with Bigram did the best.
- Logistic Performed better than SVM in all cases.
- SVM did better in classes with better support.

Bonus: Summarization

- Summarizing Yelp restaurant data from Nevada
- Gold summarizations were the Tips
  - No Florida tips
  - Getting the tips to match with the reviews
- Scored with ROUGE
  - ROUGE-1 : unigrams
  - ROUGE-2 : bigrams
  - ROUGE-L : longest common subsequence

Summarization

- Tool: Sumy
  - Simple library and command line utility for extracting summaries
  - Each Sumy method is summarizing the review to 1 sentence
- Methods:
  - Stop Word Removal
  - Lex Rank
  - LSA
  - Luhn

Summarization: Stop Word Removal

- By removing the stop words from the sentence, we created summaries that did not stay grammatically correct.

Summarization: Stop Word Removal

Star Rating	Rouge	Avg. Recall	Avg. Precision	Avg. F1
1	Rouge-1	0.344128	0.161599	0.191876
1	Rouge-2	0.122024	0.0588711	0.0694595
1	Rouge-L	0.336379	0.158693	0.188255
2	Rouge-1	0.28206	0.0924457	0.114157
2	Rouge-2	0.0707654	0.0275954	0.0319964
2	Rouge-L	0.276395	0.0903676	0.111597
3	Rouge-1	0.276313	0.0770825	0. 0992213
3	Rouge-2	0.0677045	0.0222485	0.0269659
3	Rouge-L	0.268863	0.0222485	0.0968102
4	Rouge-1	0.277328	0.0787303	0.102699
4	Rouge-2	0.0749976	0.0262328	0.0322665
4	Rouge-L	0.270014	0.0767823	0.100051
5	Rouge-1	0.318215	0.12646	0.157998
5	Rouge-2	0.110099	0.0896828	0.0580722
5	Rouge-L	0.310283	0.123786	0.154469

Summarization: Lex Rank

- Unsupervised
- Text rank to find summary
- Cosine similarity and vector based algorithms
  - Find minimum cosine distance among words and store the most similar words together

Summarization: Lex Rank

Star Rating	Rouge	Avg. Recall	Avg. Precision	Avg. F1
1	Rouge-1	0.273712	0.329951	0.260945
1	Rouge-2	0.185721	0.244607	0.184616
1	Rouge-L	0.262788	0.320637	0.252057
2	Rouge-1	0.182617	0.184468	0.157282
2	Rouge-2	0.0911611	0.108333	0.0840041
2	Rouge-L	0.173225	0.175959	0.149194
3	Rouge-1	0.169536	0.166842	0.144008
3	Rouge-2	0.0703384	0.0937178	0.0687023
3	Rouge-L	0.158974	0.15928	0.136087
4	Rouge-1	0.159034	0.169039	0. 139792
4	Rouge-2	0.0719759	0.100791	0.0721567
4	Rouge-L	0.150703	0.161798	0.132959
5	Rouge-1	0.232281	0.287033	0.226138
5	Rouge-2	0.14694	0.211812	0.152118
5	Rouge-L	0.22334	0.27935	0.218789

Best over all: 1 Star reviews

Rouge-1 was the best for each category, but Rouge-L was close

Summarization: Luhn

- Scores sentences by frequency of the most important words
- Would do better with the removal of stop words

Summarization: Luhn

Star Rating	Rouge	Avg. Recall	Avg. Precision	Avg. F1
1	Rouge-1	0.2465	0.250902	0.217289
1	Rouge-2	0.141798	0.166901	0.134926
1	Rouge-L	0.231137	0.238914	0.205966
2	Rouge-1	0.181727	0.154357	0.142873
2	Rouge-2	0.0736692	0.0821042	0.0643784
2	Rouge-L	0.168409	0.144825	0.132686
3	Rouge-1	0.164893	0.131468	0.125045
3	Rouge-2	0.0515726	0.0614545	0.0473624
3	Rouge-L	0.152141	0.123549	0.116299
4	Rouge-1	0.163945	0.135262	0.126666
4	Rouge-2	0.0566547	0.0662685	0.051635
4	Rouge-L	0.152612	0.126949	0.118028
5	Rouge-1	0.211654	0.215815	0.186496
5	Rouge-2	0.10775	0.138218	0.105968
5	Rouge-L	0.199701	0.206563	0.177075

Best over all: 1 Star reviews

Rouge-1 was the best for each category, but Rouge-L was close

Summarization: LSA

Latent Semantic Analyzer: extracts hidden semantic structures in order to summarize

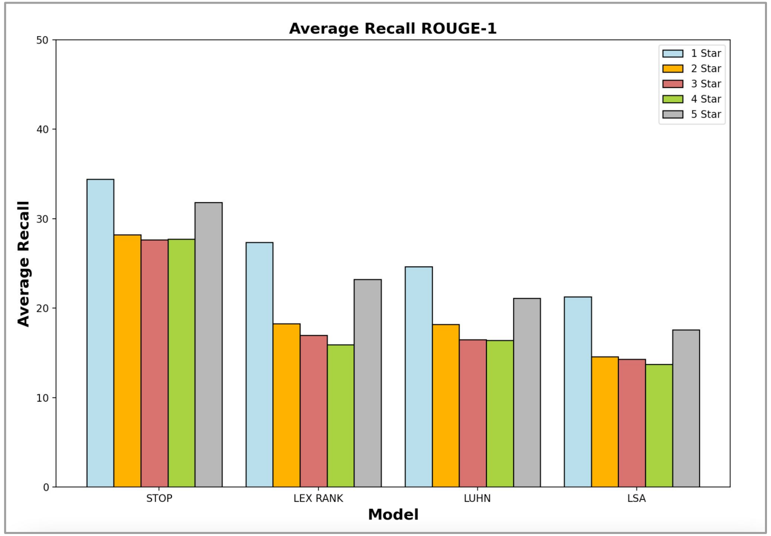
- Unsupervised
- Reduces dimensionality of original text data
- Finds relations between terms

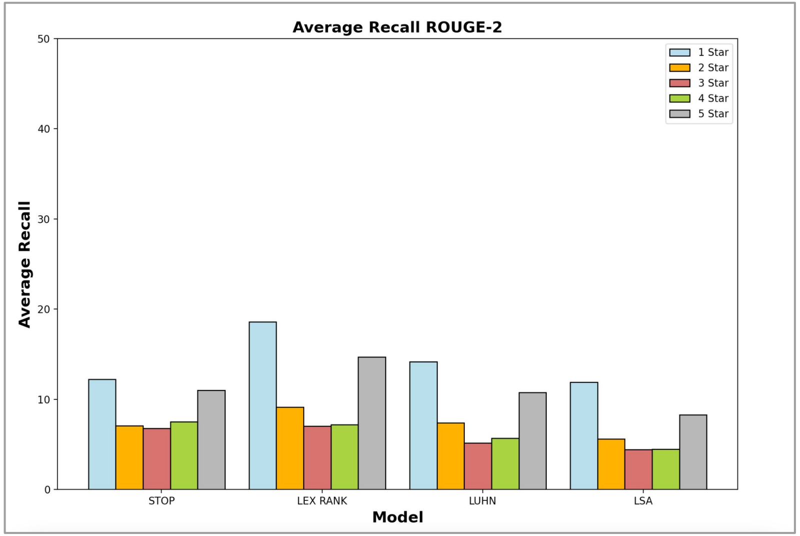
Summarization: LSA

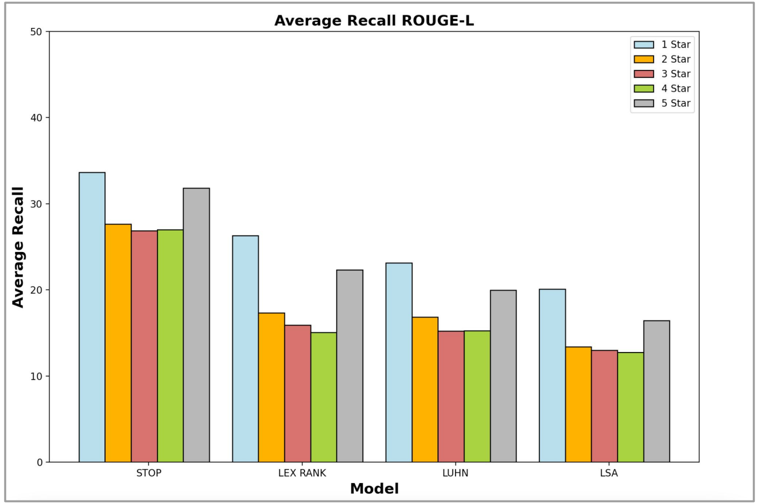
Star Rating	Rouge	Avg. Recall	Avg. Precision	Avg. F1
1	Rouge-1	0.212854	0.209993	0.188841
1	Rouge-2	0.118811	0.130333	0.110156
1	Rouge-L	0.200909	0.199288	0.178823
2	Rouge-1	0.145744	0.125731	0.120071
2	Rouge-2	0.0558964	0.0589634	0.0495379
2	Rouge-L	0.133829	0.115934	0.110142
3	Rouge-1	0.142842	0.115235	0. 112749
3	Rouge-2	0.0443532	0.0511687	0.0416062
3	Rouge-L	0.129827	0.106232	0.10296
4	Rouge-1	0.137032	0.111915	0.109443
4	Rouge-2	0.0451491	0.051056	0.0426099
4	Rouge-L	0.12734	0.104527	0.101798
5	Rouge-1	0.175683	0.157997	0.149773
5	Rouge-2	0.0828182	0.0896828	0.0769363
5	Rouge-L	0.16426	0.148555	0.140367

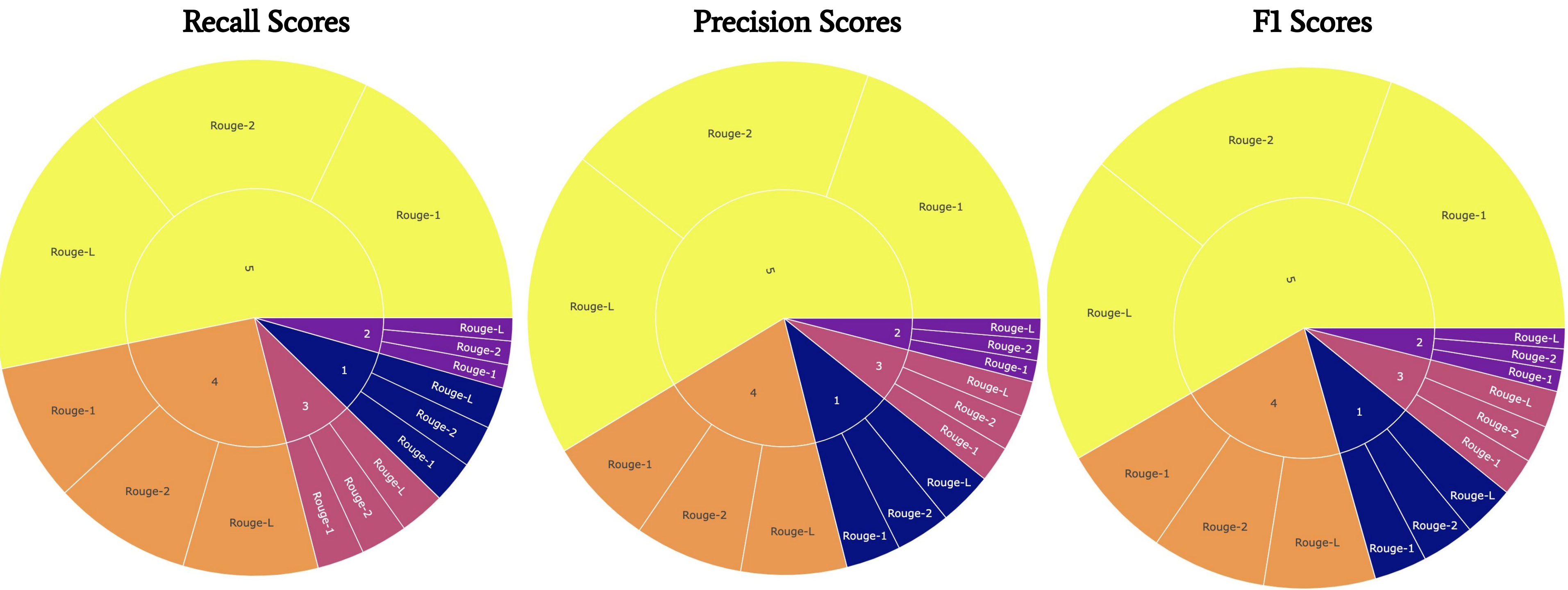
Best over all: 1 Star reviews

Rouge-1 was the best for each category, but Rouge-L was close


Average Recall ROUGE-1
Average Recall
Model
STOP
LEX RANK
LUHN
LSA
1 Star
2 Star
3 Star
4 Star
5 Star
0
10
20
30
40
50


Average Recall ROUGE-2
Average Recall
Model
STOP
LEX RANK
LUHN
LSA
1 Star
2 Star
3 Star
4 Star
5 Star
0
10
20
30
40
50


Average Recall ROUGE-L
Average Recall
Model
STOP
LEX RANK
LUHN
LSA
1 Star
2 Star
3 Star
4 Star
5 Star
0
10
20
30
40
50



Recall Scores
Precision Scores
F1 Scores
5
4
3
2
1
Rouge-1
Rouge-2
Rouge-L

Conclusion: Summarization

- Hard to compare the summarized reviews to the tips.
- Sometimes tips just added more information to the review and didn't correlate.
- There might be a correlation between the dataset size versus the rouge score.
- Removing the stop words did better than we thought it would since the other models are making more coherent sentences.

If we had more time...

- Compare a few topic modeling methods.
- Go beyond unigrams and bigrams. Find more complex features.
- Look into better methods for summarization.
- Try explore more LDA models with varied number of topics and calculate their coherence and perplexity.