ADC Project

Yelp Dataset

Group composed by:
Paulo Portela
Luís Henriques
Miguel Veloso
Karim Kousa

Outline

1. Data Understanding
2. Exploratory Data Analysis (EDA)
3. Choice of a specific city
4. Social Network Analysis (SNA)
5. Recommender System (RS)
6. Natural Language Processing (NLP)
7. Time Series (TS)
8. Conclusions

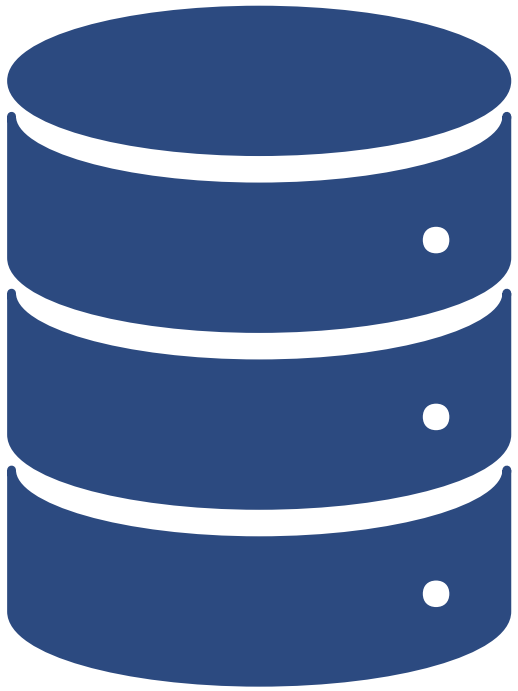
Objective Selection

Social Collaborative Filtering

Incorporate social network data into collaborative filtering algorithms to enhance recommendations

Sentiment Analysis Over Time

Implement NLP to predict sentiment in reviews and track its evolution over time



1. Data Understanding

1. Data Understanding

BUSINESS

14 COLUMNS, 150346 ROWS

REVIEWS

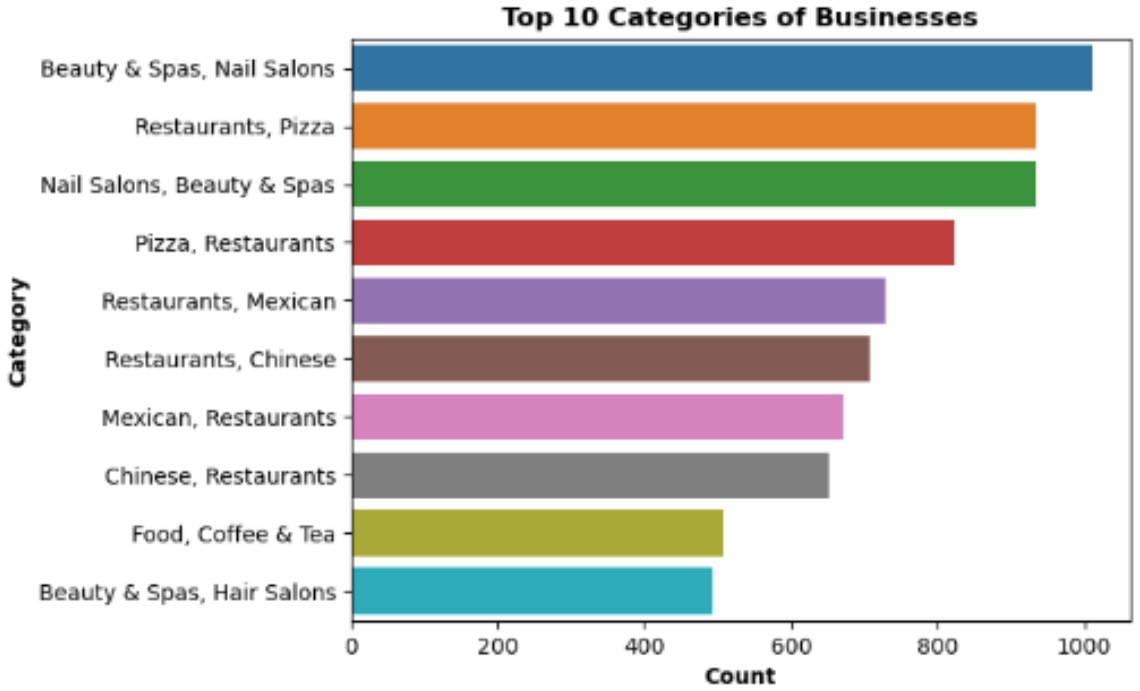
9 COLUMNS, 6990280 ROWS

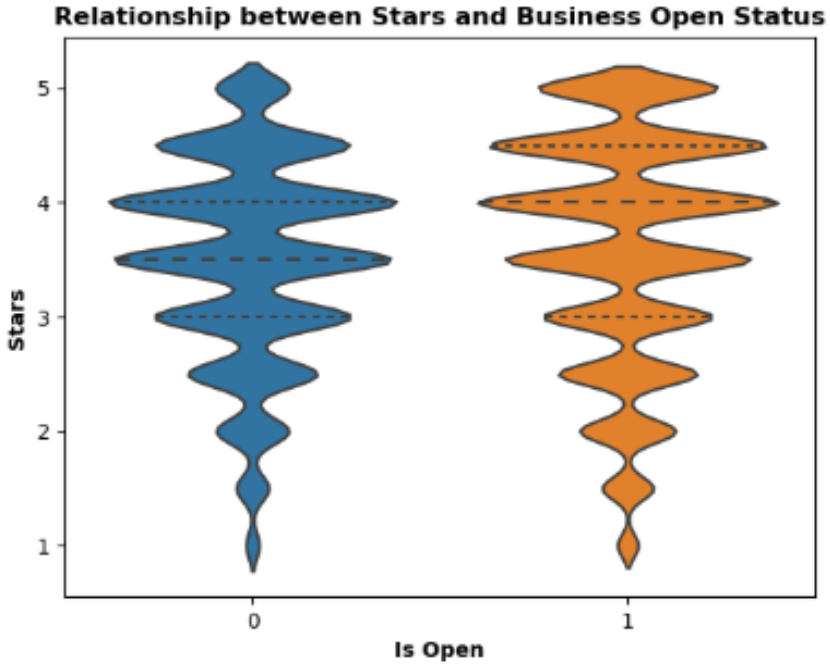
USERS

22 COLUMNS, 1987897 ROWS

2. Exploratory Data Analysis

2. Exploratory Data Analysis (EDA) - Business

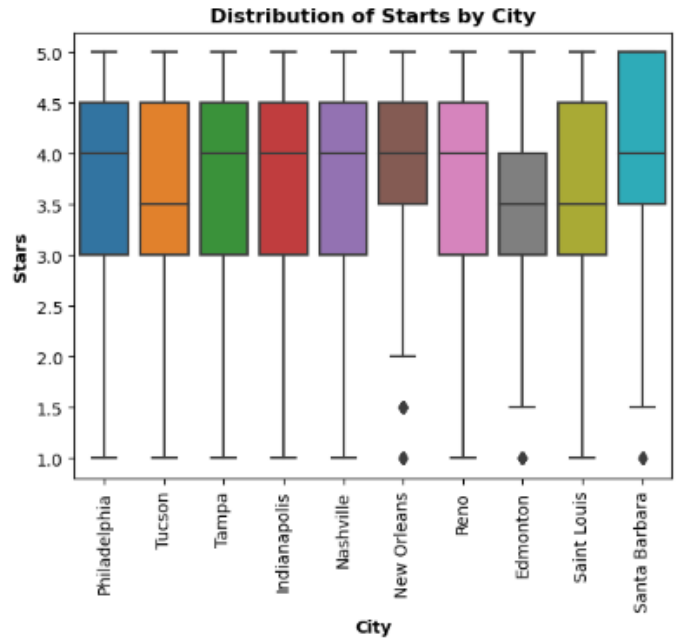




- Identification of key recurring words in business categories.
- There appears to be a correlation between the quality of reviews and the longevity of a business;

- Businesses with lower star ratings are more likely to face closure or cessation of activity;

2. Exploratory Data Analysis (EDA) - Users and Reviews



- Significant drop-in overall review activity during the pandemic;
- Variation in user behavior, preferences, and sentiments during the pandemic;
- Number of users registration decreasing after 2015.

- Limited diversity in reviews and businesses in certain cities;
- Potential impact on trends and patterns due to incomplete dataset;
- Challenges in extracting meaningful insights due to sparse data.

3. Choice of a specific city

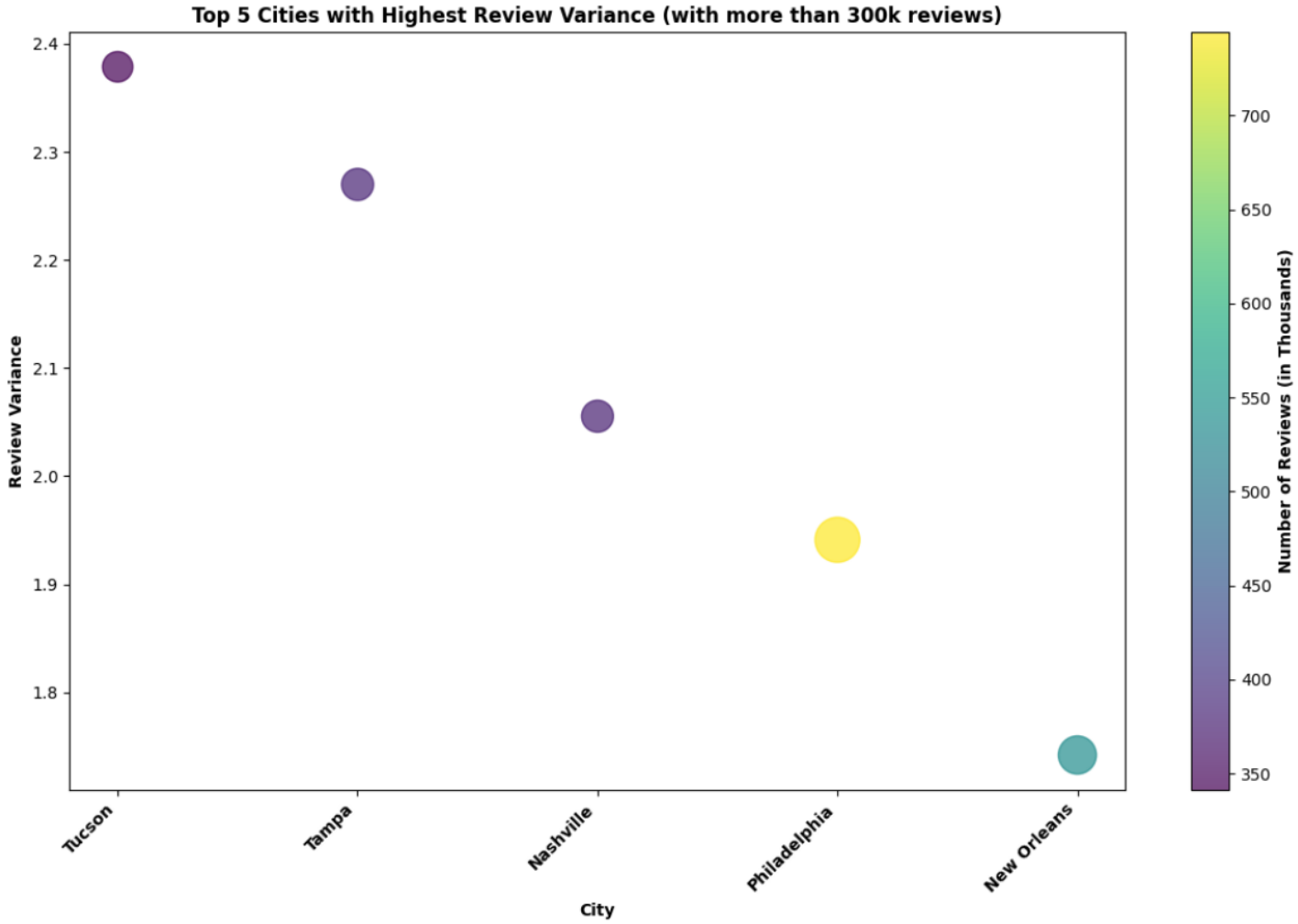
3. Choice of a specific city - Tucson

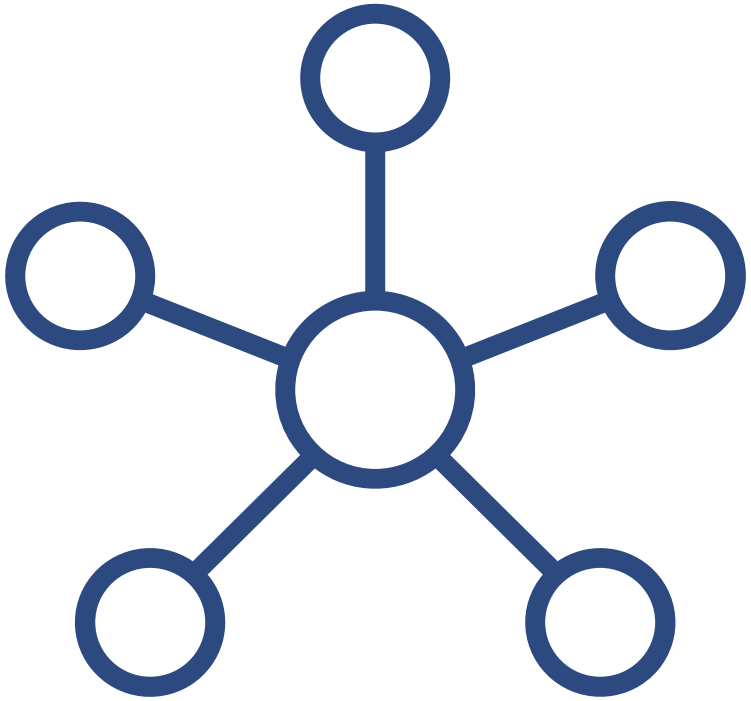
Goal

- **Higher variation on the reviews**
- **Higher number of reviews**

Addicionally at least 25% of the reviews have 2 stars or less for **Tucson**, which shows more lower reviews than all other cities, with a first quartile of 3 stars.
We also filtered the business that are open.

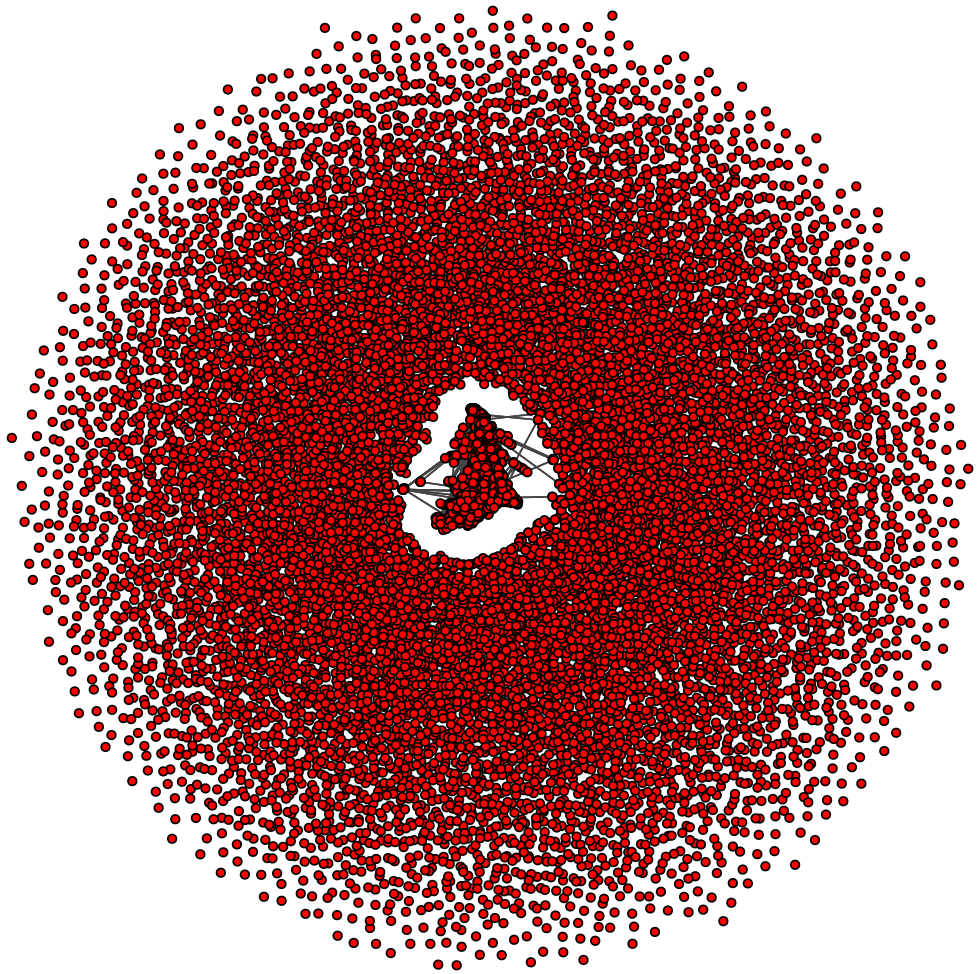
TUCSON





4. Social Network Analysis

4. SNA – Social Network



Friendships are restricted to connections between users who share Tucson as their City;
To gauge the strength of connections, we've developed a weighted approach combining insights from various functions.

Key Aspects

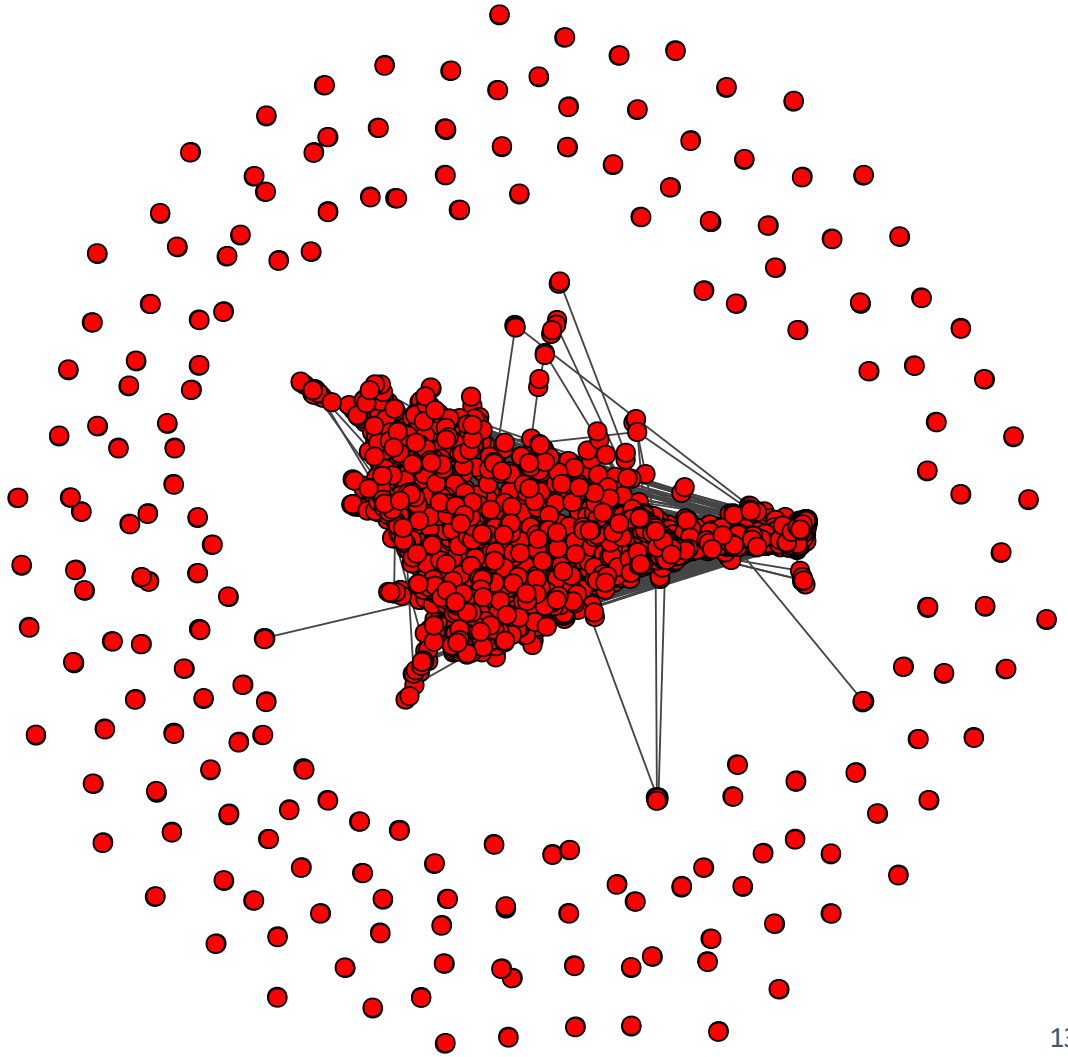
1. **Elite Status Influence:**
   - The EliteUsers function aids us in determining the 'elite' status of user1_id, offering a valuable metric for connection weight.
2. **Fan Contribution:**
   - We factor in the influence of user1_id's fans by contributing a portion of their count, scaled for relevance, divided by 100.
3. **Friendship Confirmation:**
   - The Friends function is instrumental in confirming whether user2_id is part of the friend list of user1_id, adding another layer to connection weight.

4. SNA – Social Network

The following content were removed:

1. **Nodes without connection**
2. **Nodes with redundant connections**

Density	Closeness (Range)	Betweenness (Range)	Homophily	Average path length (directed)
0.0011	0.125741 to 1	0 to 7.104133e+06	-0.072	4.019



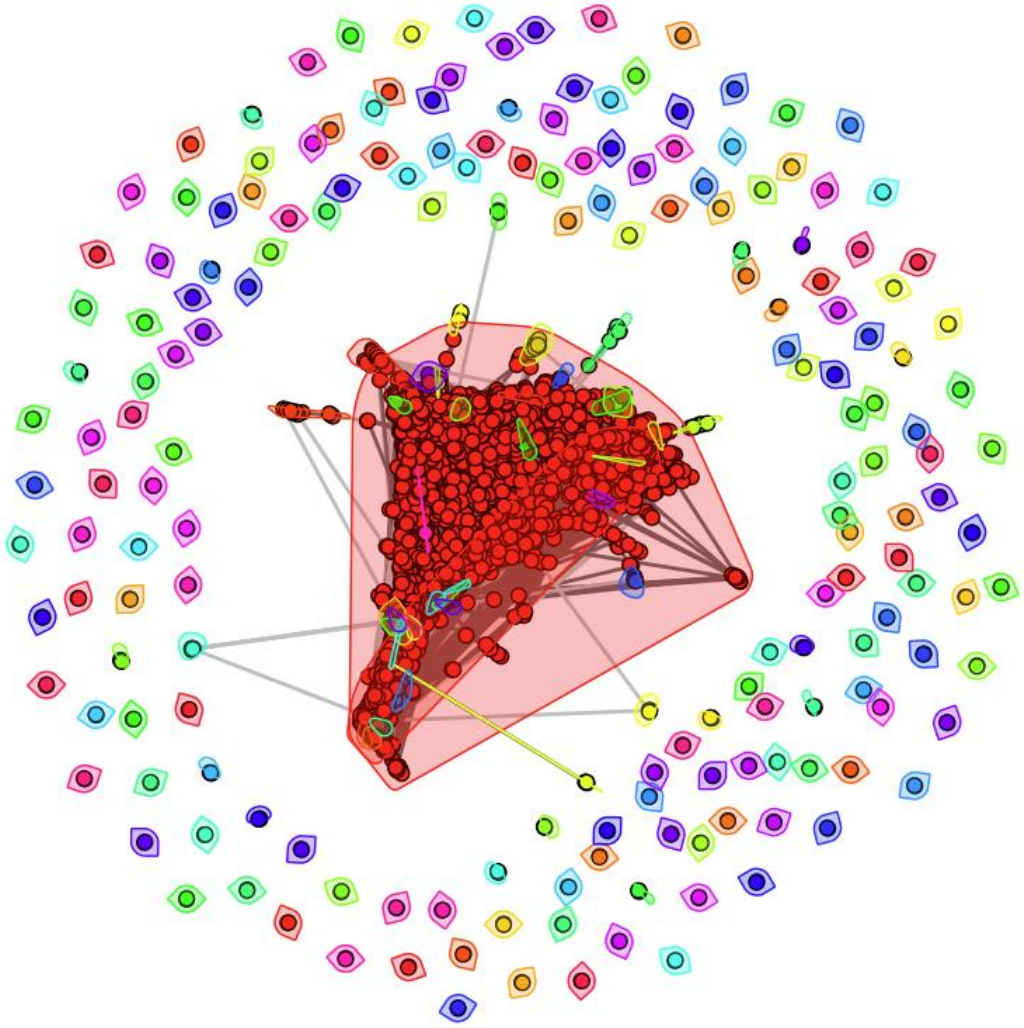
4. SNA – Communities detection

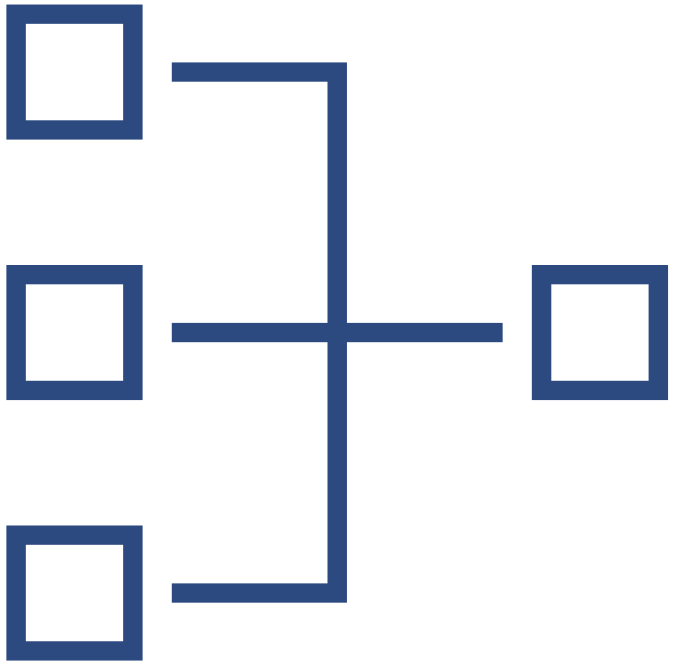
Community detection based on edge betweenness (Newman-Girvan) with label propagation

- Low sparsity (0.59%) in datasets hinders effective community detection;
- Accurate communities are crucial for understanding complex relationships;
- Multiple communities - **3 strong communities**
- Elite users use their influence to build strong communities, considering they build the strongest links (high weights)

224 communities identified

- *1st position - 4371 users*
- *2nd position – 1259 users*
- *3rd position– 368 users*
- *224th position – 2 users*





5. Recommender System

5. Recommendation System (RS)

Developed the inaugural **Recommendation System (RS)** utilizing SNA-based features for users, a hybrid recommender system.
The system has been customized to provide personalized recommendations by analyzing user ratings, while feeding it with user features extracted from SNA, for example communities, closeness, betweenness, etc. This functionality allows for the suggestion of content closely aligned with the preferences and interests of users within those same communities.

In achieving this, the **LightFM** library and **SVD algorithm** were employed, as well as others like UBCF, and IBCF. Subsequently, their outcomes were compared to assess their respective performance in the recommendation system. However, LightFM doesn't give predictions like SVD, rather it gives scores that tell how likely it is for a user to interact with a certain business (probabilities). Therefore, these models can't be compared using evaluation metrics. So, the only way to do it is by rankings.

Using the test dataset to validate the business id recommendations – SVD proved to have better results, overall.

Business Recommendations for User 9539		
Ground truth	**LightFM**	**SVD**
219	87	**219**
	107	111
	173	20
	156	159
	111	82

Business Recommendations for User 2172		
Ground truth	**LightFM**	**SVD**
1	107	**1**
	200	240
	113	79
	1	25
	4	4

User 9539	
Models	Rankscore
SVD	1.0
LightFM	0.0

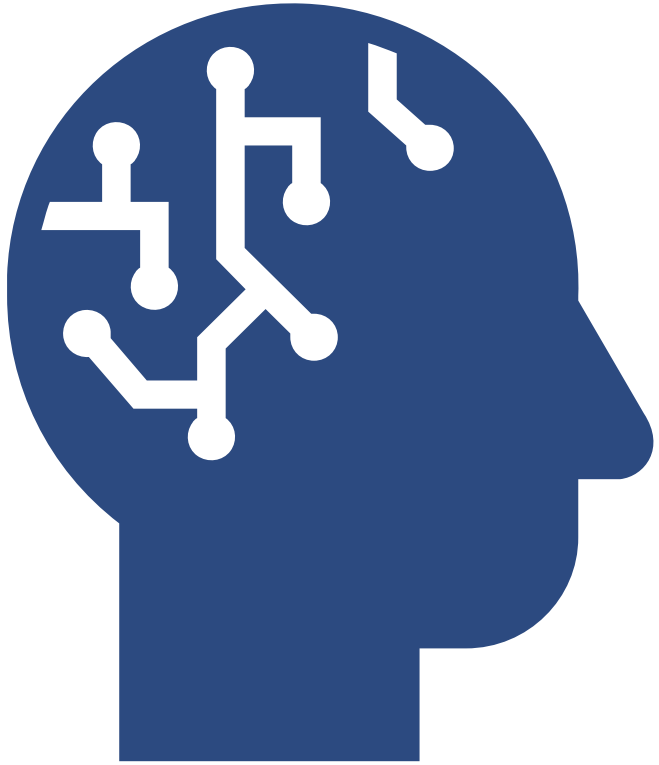
User 2172	
Models	Rankscore
SVD	1.0
LightFM	0.354

5. Recommendation System (RS)

Even though we can't compare evaluation metrics between Surprise and LightFM models, we can still compare Surprise models with themselves. So, regarding these models, we have a random recommender, a user-based collaborative filtering recommender (UBCF), an item-based collaborative filtering recommender (IBCF), and two matrix factorization models (SVD and SVD++)

Looking at the results, we can see that the matrix factorization models deliver the lowest RMSE, and between those the standard SVD model shows the lowest RMSE of them all.

Models	RMSE
Random recommender	1,9006
UBCF	1,4035
IBCF	1,4545
SVD	1,2748
SVD++	1,3147



6. Natural Language Processing (NLP)

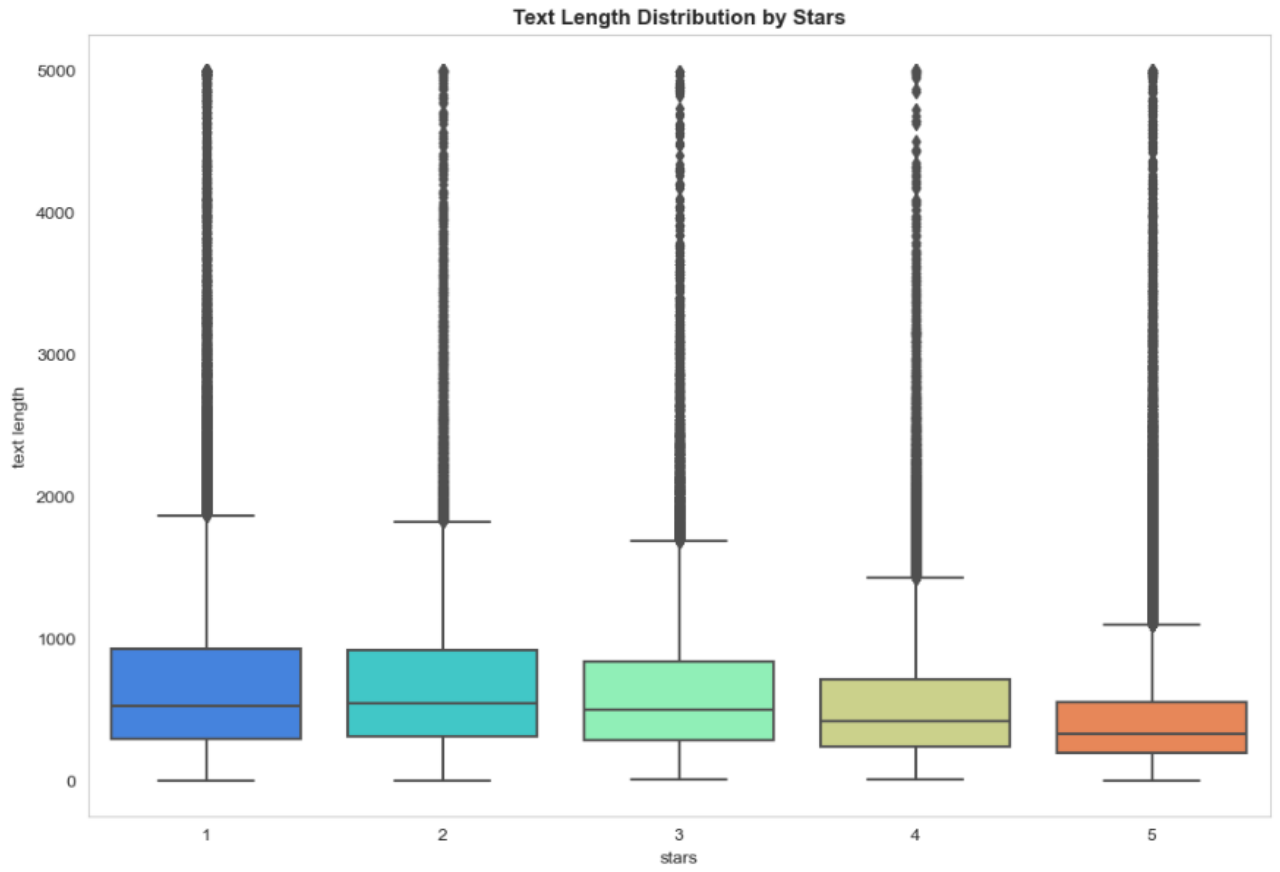
6. Natural Language Processing (NLP)

1. EDA

- Distribution of reviews and Stars, text length analysis, etc ...

2. Data Cleaning

- Non-alpha characters removal with a regular expression;
- Removal of wrong content such as "&amp";
- Removal of Non-English reviews (langdetect);
- Convert all the words to lowercase – Ensured that words like "Amazing" "AMAZING" and "amazing" are all represented in the same way.



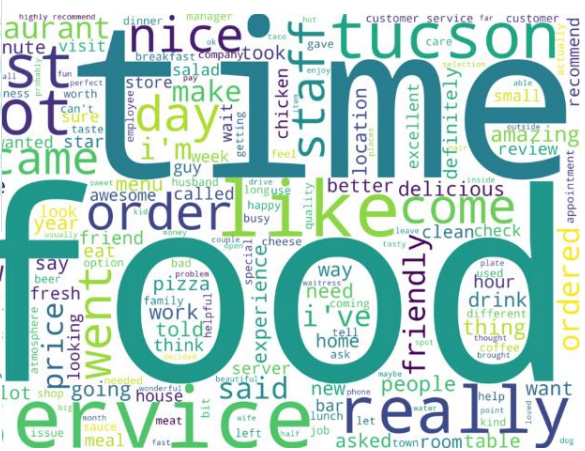
6. Natural Language Processing (NLP)

3. Text Mining

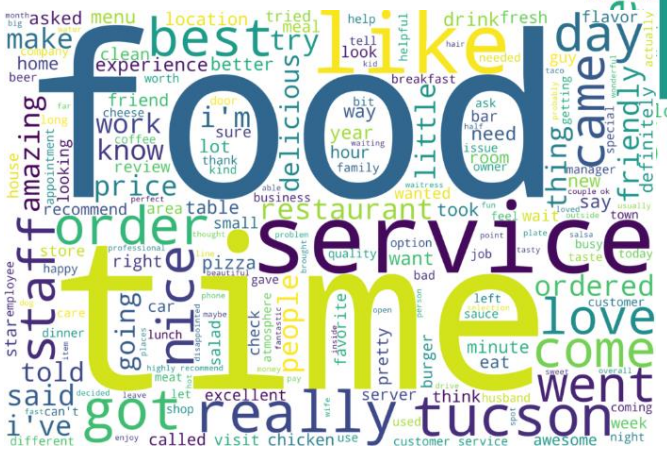
- Definition of the StopWords - NLTK + Scikit-Learn
- Word Cloud - Applied for all the reviews and for:
  - Positive Reviews (5 and 4 Stars)
  - Neutral Reviews (3 Stars)
  - Negative Reviews (2 and 1 Stars)

Positive Reviews

Neutral Reviews



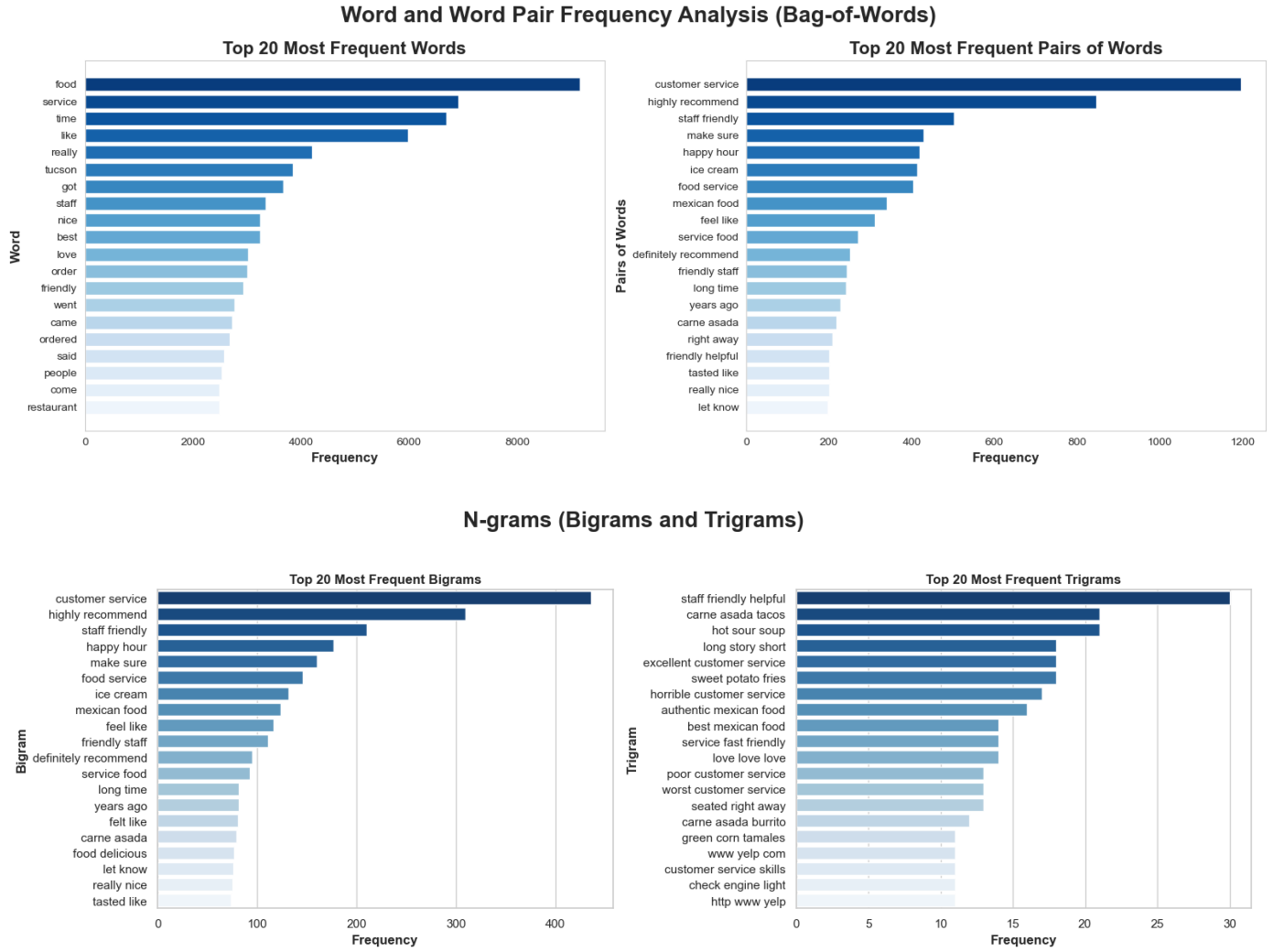
Negative Reviews



6. Natural Language Processing (NLP)

4. Tokenization and Vectorization

- Bag-of-Words
  - Stemming
  - Lemmatization
- N-grams and CountVectorizer
- Word2Vec
- Spacy
- Regular Expressions
- Multi-word expressions (MWE)



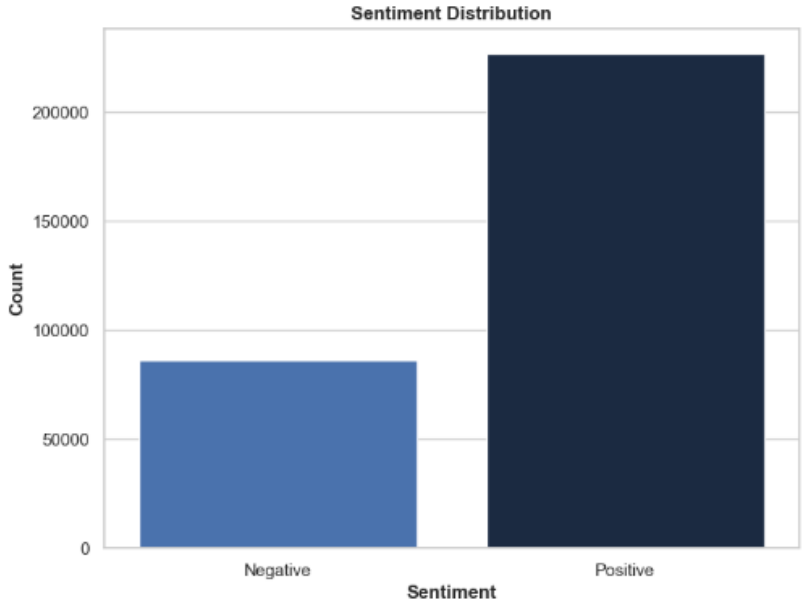
6. Natural Language Processing (NLP)

5. Sentiment Analysis Evaluation – labeling and train & test split

- Creation of the target label, called "sentiment" considering the following:
  - Reviews with 1 and 2 Stars were considered negative -> Label "-1"
  - Reviews with 4 and 5 Stars were considered positive -> Label "1"
  - Reviews with 3 Stars were not considered
- More than the double of the reviews were considered as positives

70% Train

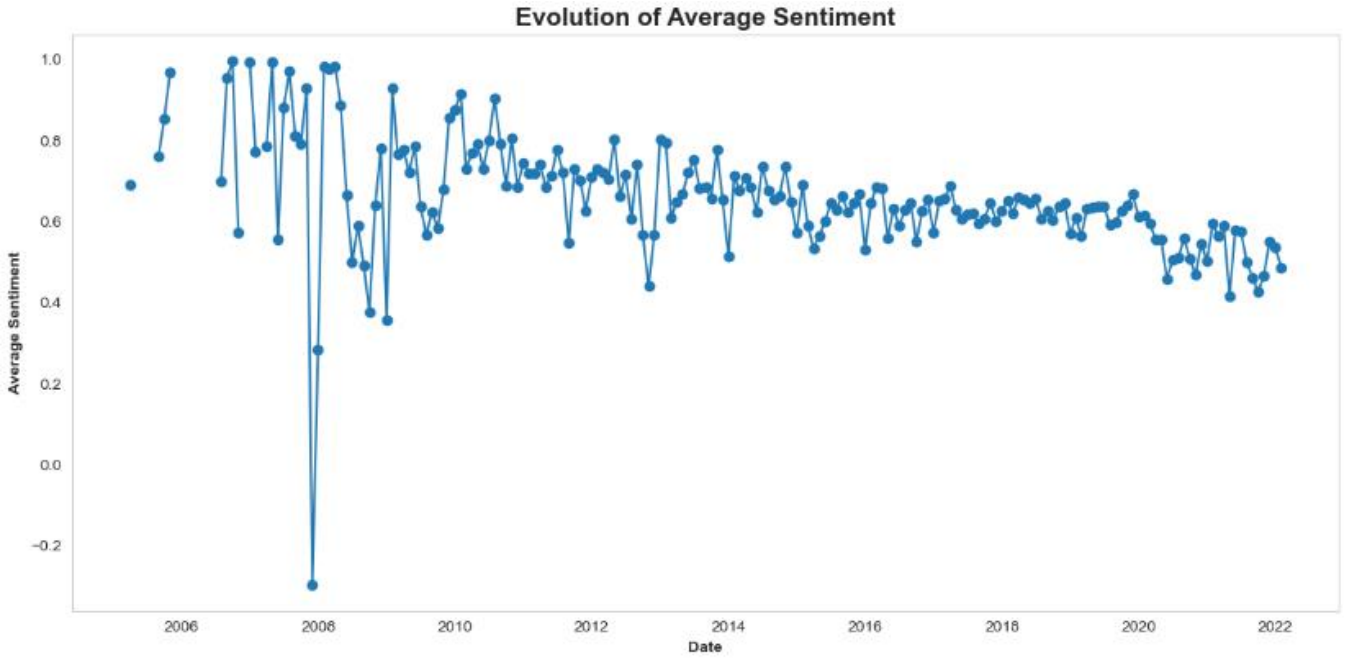
30% Test



6. Natural Language Processing (NLP)

5. Sentiment Analysis Evaluation

- Valence Aware Dictionary for Sentiment Reasoning (Vader) – Sentiment-base tool
  - A **positive** sentiment, **compound** ≥ 0.05
  - A **negative** sentiment, **compound** ≤ -0.05
  - A **neutral** sentiment, the **compound** is between -0.05 and 0.05



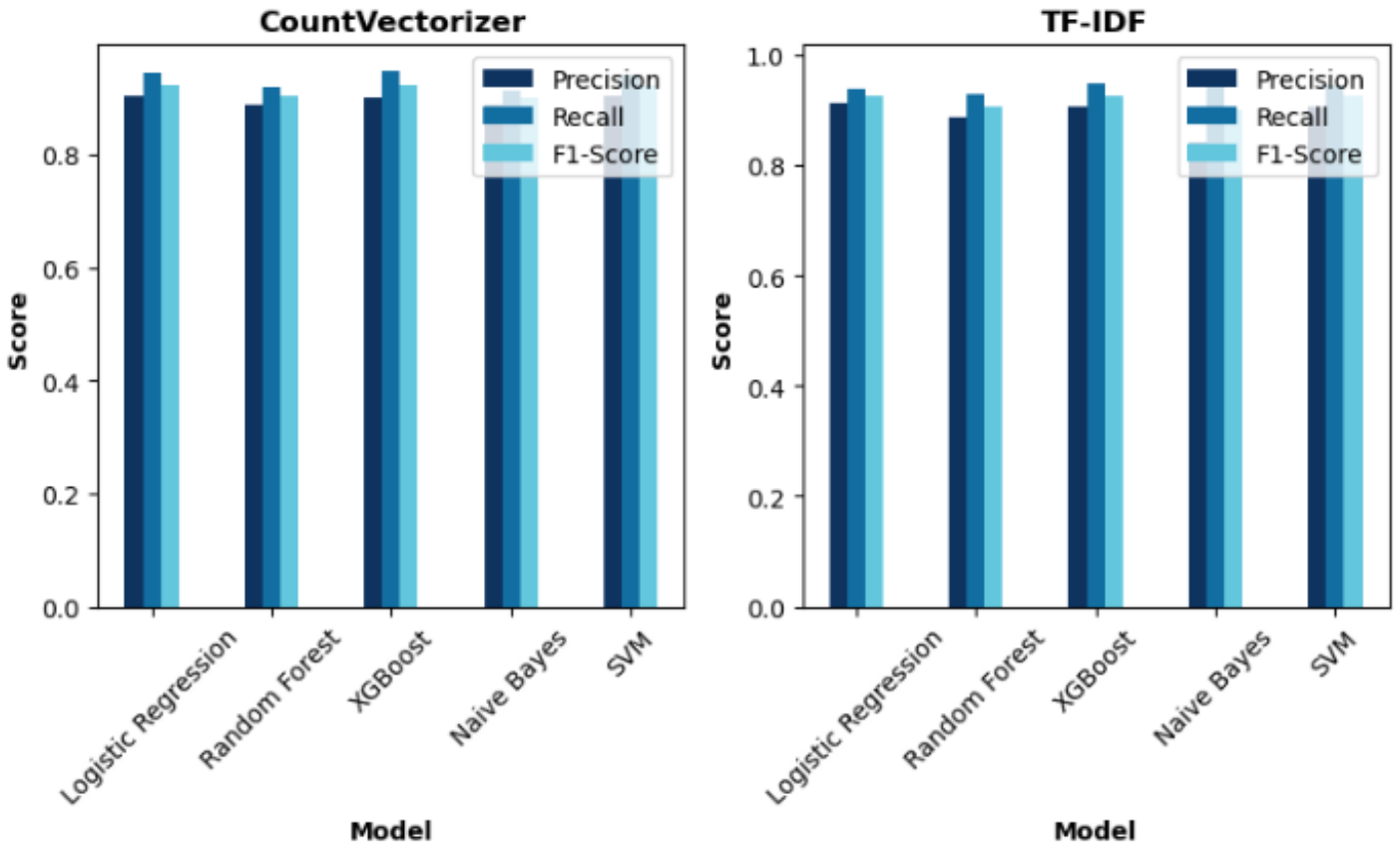
Accuracy	Precision	Recall	F1-Score
0.66	0.53	0.66	0.59

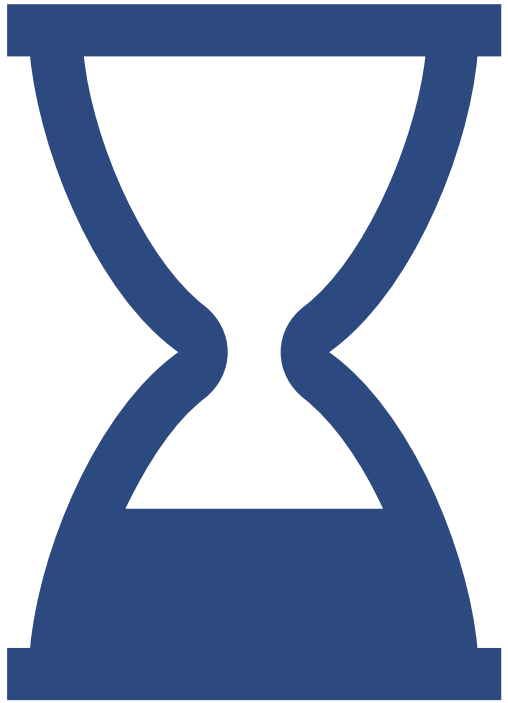
6. Natural Language Processing (NLP)

5. Sentiment Analysis Evaluation

- In the realm of text pre-processing and text representation, several approaches were employed, including:
  - **Count Vectorizer**
  - **Term Frequency-Inverse Document Frequency (TD-IDF)**
  - Word Embeddings - Word2Vec
  - Spacy

- The subsequent classification algorithms were tested for each of these approaches:
  - Logistic Regression
  - **Random Forest**
  - **XGBoost**
  - Naïve Bayes
  - Support Vector Machine (SVM)





7. Time Series (TM)

7. Problem definition

WHAT TO FORECAST?
SENTIMENT (POSITIVE OR NEGATIVE)

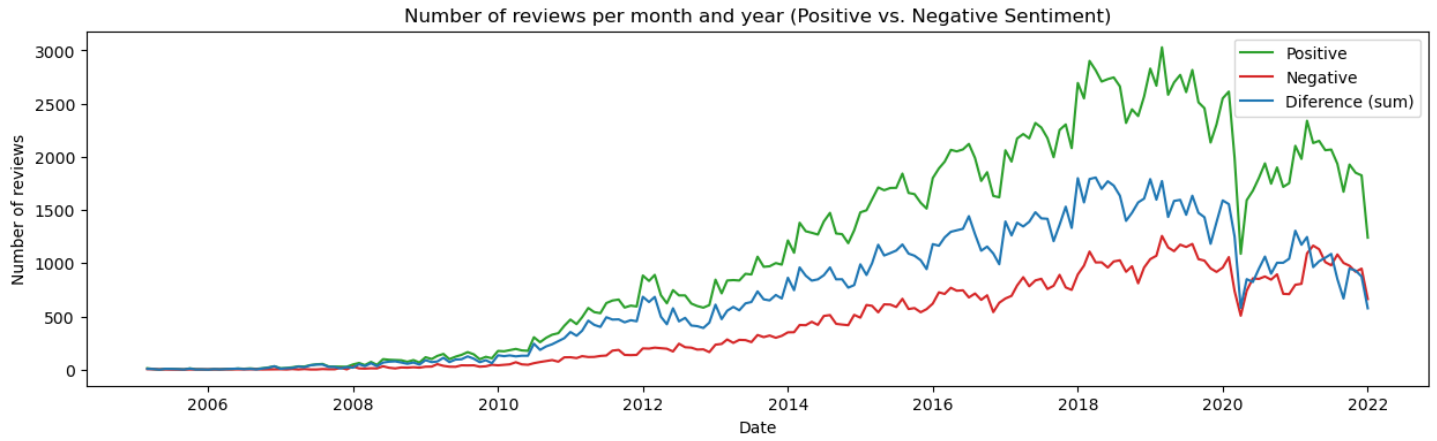
TIMEFRAME?
MONTHLY

HORIZON OF FORECASTING?
PREDICTING FOR 2021

7. Visualizations – Seasonal plots (1/2)

- To track the evolution of the review's sentiment, in Tucson, across all years:
  - Number of reviews aggregated by the combination (month, year);
  - Reviews segregated by sentiment;
  - Third time series created by subtracting the positive and negative time series.

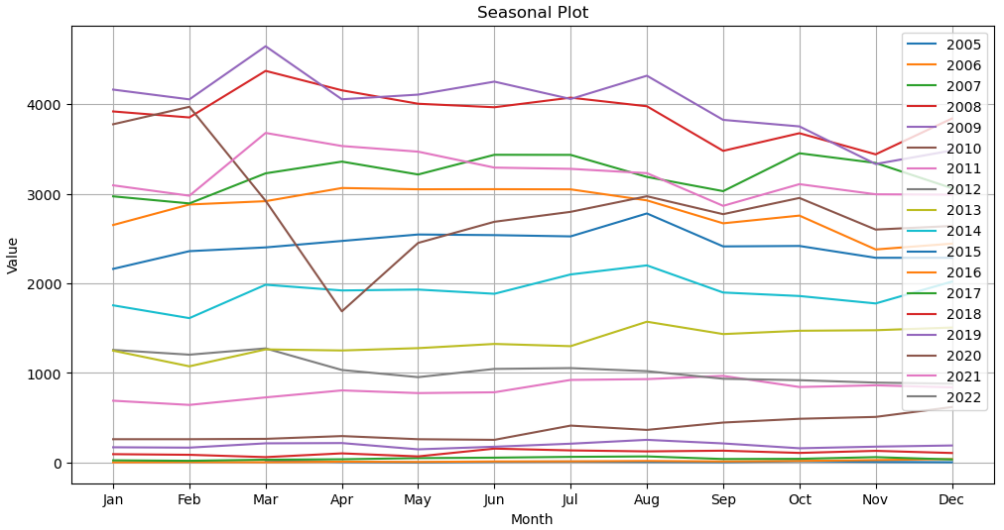
- From this visualization, several analysis can be extracted:
  - Difference series having a positive trend, until end of 2018, means that positive reviews increased in a faster rate than negative reviews;
  - An abrupt decline in reviews occurs in 2020, with the first COVID-19 confinement;
  - Clear seasonal patterns can be found across all three series;
  - After the decline, positive reviews declined while negative reviews not so much. Negative reviews now represent more than 50% of the number of positive reviews.

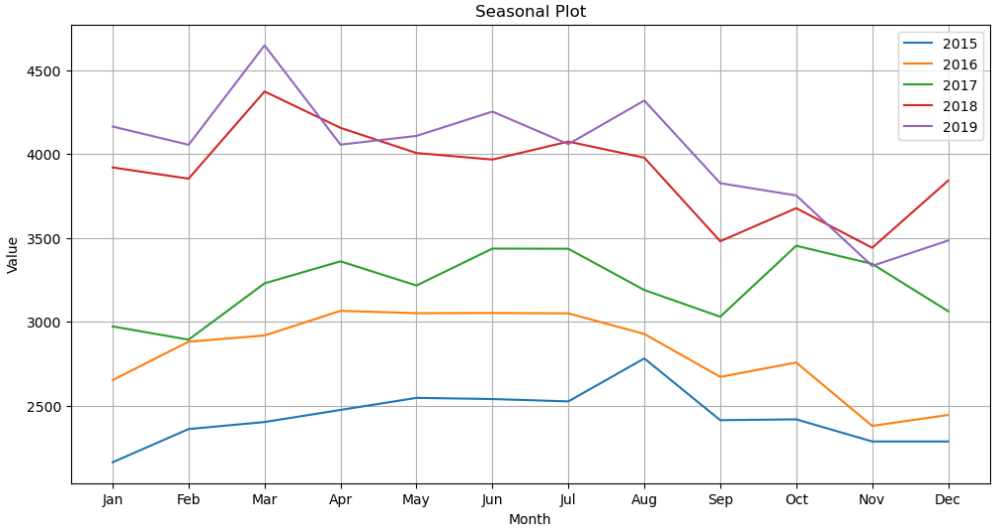


7. Visualizations – Seasonal plots (2/2)

- Another way of looking at time series is by stacking “mini-series” together, in this case each series of those corresponds to one year:
  - Once again, we can see that similar trends, as well as some seasonality;
  - The time series that stands out is the one from 2020.

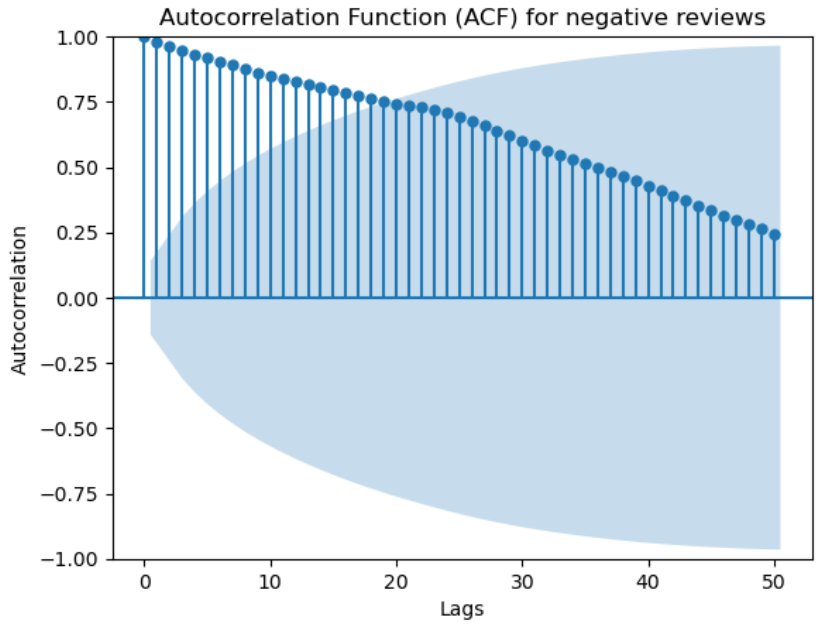
- By focusing on the most recent years prior to COVID-19:
  - Similar trends and seasonality can be seen, in general;
  - Like it was mentioned previously, the trend of the positive reviews started to decline near the end of 2018;
  - Then, in 2019, the negative trend becomes even more evident;
  - But still, the number of positive reviews increases along the years.

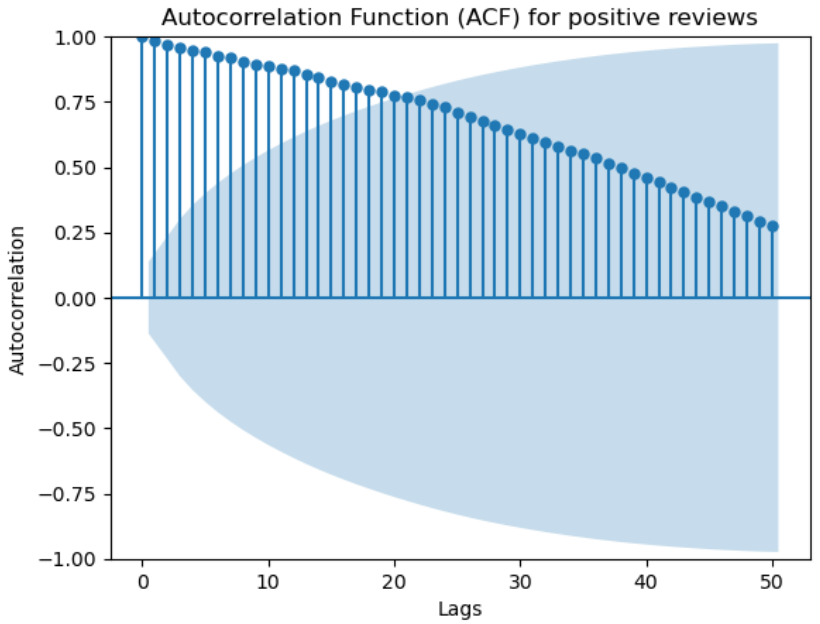




7. Visualizations – Autocorrelations

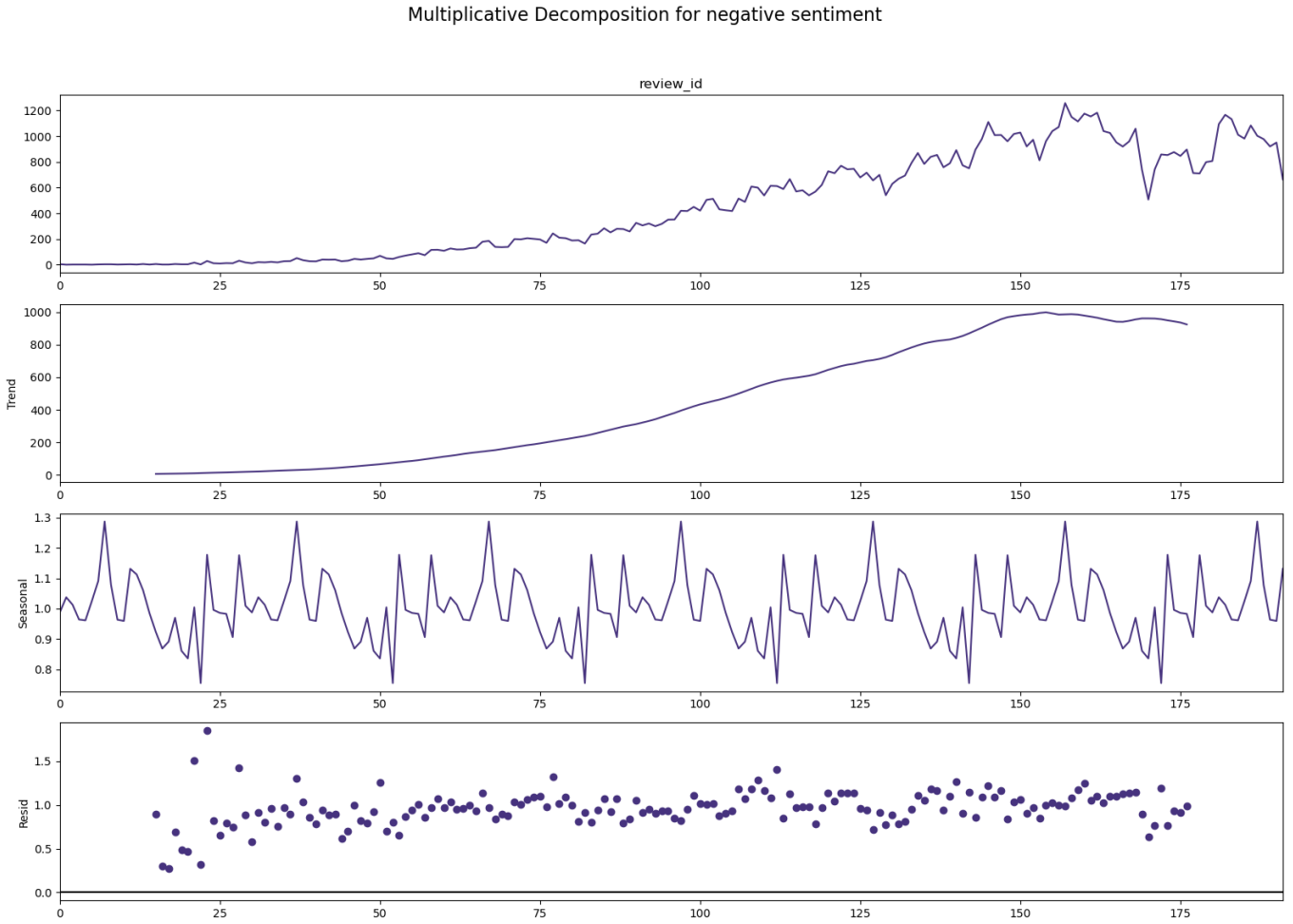
- The ACF enables the analysis on how a time-series value is dependent on previous values:
  - It appears the values of the time series, for positive and negative reviews, are more dependent on the most recent data points;
  - As the lags becomes larger, the autocorrelation decreases, proving the previous point;
  - If we extended the x-axis, we would see that the autocorrelation reaches zero, an in fact, after that becomes negative.





7. Visualizations – Feature Engineering – Decomposition

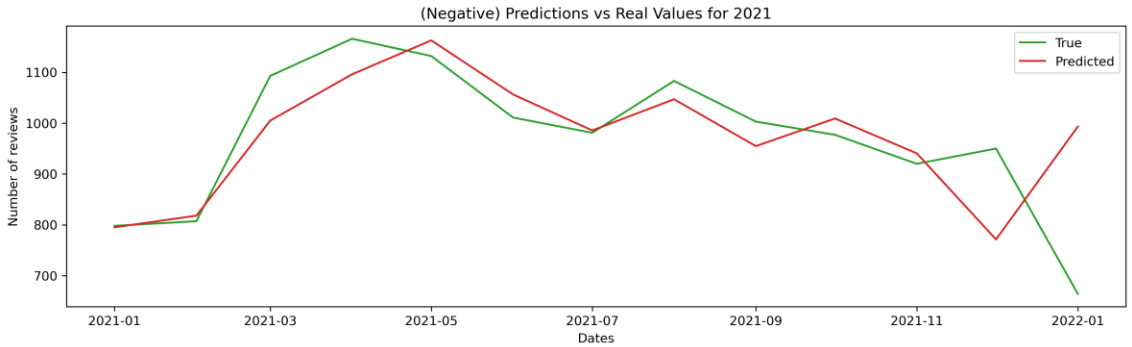
- Although decomposition is used for FE, we can also visualize its components:
  - This is one example of a series with a multiplicative decomposition, in this case the series for negative sentiment;
  - Positive trend until 2018, then nearly flattens;
  - Seasonal component can impact the series values up to 30%, either positive or in a negative manner;
  - Residuals appear to be random, staying around the value 1.

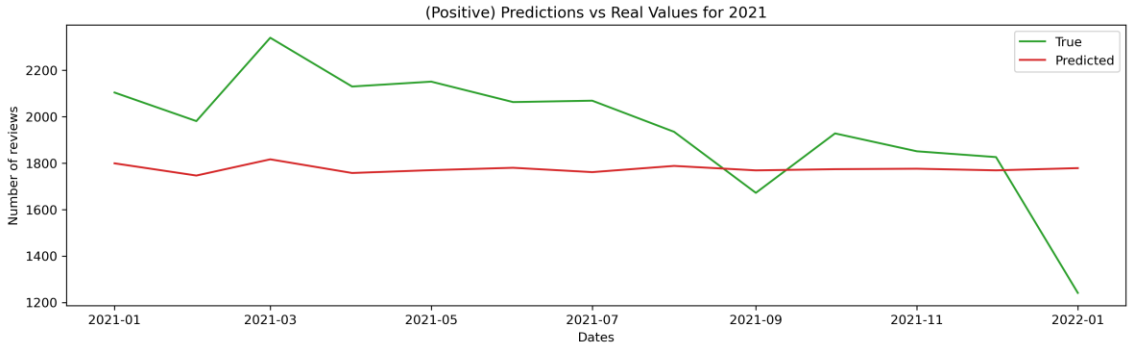


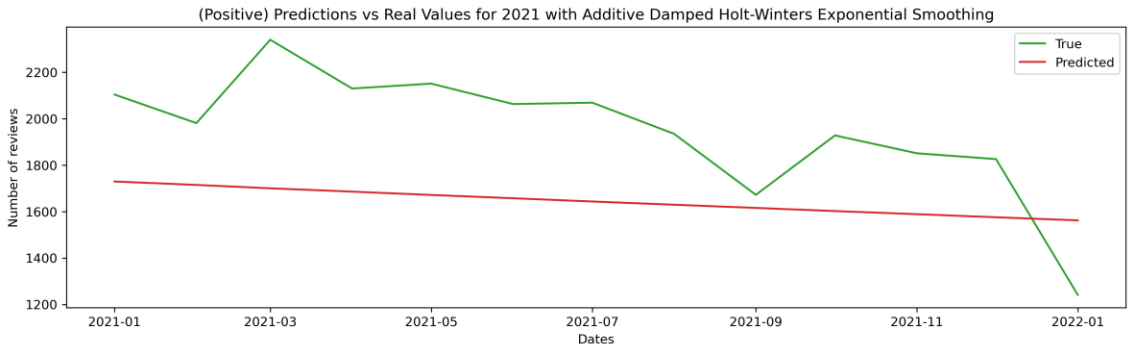
7. Visualizations – Time series models

- Besides the main goal of tracking the sentiment's evolution, baselines, ETS, ARIMA, and Regression models were implemented:
  - The plot on top corresponds to the regression model (linear regression), the second to the ARIMA model, and the third to the Damped version of the Holt-Winters model;
  - The regression model gave the best results, overall, in specific to the negative sentiments series;
  - Regarding hyperparameters, additive and multiplicative models were implemented. Damped and not damped models implemented. Alpha, beta, and gamma were also tuned a bit.

Metrics (negative)	Baseline - naive	Linear regression
RMSE	292,73	111,34
MAE	265,15	69,02
MAPE	26%	8%







8. Conclusions

8. Conclusions & Future work

- The hybrid RS seems to need more relevant SNA features to become better, and the NLP and TS components gave very interesting results, although some of them not so good, for example the Holt-Winters model;
- For future works, removing users with less then 5 friends or explore the center of the social network could improve the analysis, as well as creating a more complete and robust weight function for the connections;
- Maybe with these points above LightFM could outperform the non-hybrid matrix factorization models;
- Besides the rankings, implementing ranking metrics (like rankscore), as well as evaluation metrics for LightFM, for example AUC.