# Rotten Tomatoes Movies Rating Prediction

0

You must be logged in to download the datasets

More about this project

*Meta: Predict Rotten Tomatoes labels via feature-based and text-driven classification. Python, Scikit-learn, Random Forest, NLP.*

## Assignment

In this project, you are given large datasets from Rotten Tomatoes - a popular online review aggregator for film and television. Your task is to build a high performing classification algorithm to predict whether a particular movie on Rotten Tomatoes is labeled as 'Rotten', 'Fresh', or 'Certified-Fresh'.

## Data Description

There are 2 datasets

1. `rotten_tomatoes_movies.csv` - contains basic information about each movie listed on Rotten Tomatoes; each row represents one movie;
2. `rotten_tomatoes_critic_reviews_50k.tsv` - contains 50.000 individual reviews by Rotten Tomatoes critics; each row represents one review corresponding to a movie;

`rotten_tomatoes_movies` dataset contains the following columns:

- `rotten_tomatoes_link` - movie ID
- `movie_title` - title of the movie as displayed on the Rotten Tomatoes website
- `movie_info` - brief description of the movie
- `critics_consensus` - comment from Rotten Tomatoes
- `content_rating` - category based on the movie suitability for audience
- `genres` - movie genres separated by commes, if multiple
- `directors` - name of director(s)
- `authors` - name of author(s)
- `actors` - name of actors
- `original_release_date` - date in which the movie has been released in theatres, in YYY-MM-DD format
- `streaming_release_date` - date in which the movie has been released on streaming platforms, in YYY-MM-DD format
- `runtime` - duration of the movie in minutes
- `production_company` - name of a studio/company that produced the movie
- `tomatometer_status` - a label assgined by Rotten Tomatoes: "Fresh", "Certified-Fresh" or "Rotten"; **this is the target variables in this challenge**
- `tomatometer_rating` - percentage of positive critic ratings
- `tomatometer_count` - critic ratings counted for the calculation of the tomatomer status
- `audience_status` - a label assgined based on user ratings: "Spilled" or "Upright"
- `audience_rating` - percentage of positive user ratings
- `audience_count` - user ratings counted for the calculation of the audience status

- `tomatometer_top_critics_count` - number of ratings by top critics
- `tomatometer_fresh_critics_count` - number of critic ratings labeled "Fresh"
- `tomatometer_rotten_critics_count` - - number of critic ratings labeled "Rotten"

`rotten_tomatoes_critic_reviews_50k` dataset contains the following columns:

- `rotten_tomatoes_link` - movie ID
- `critic_name` - name of critic who rated the movie
- `top_critic` - boolean value that clarifies whether the critic is a top critic or not
- `publisher_name` - name of the publisher for which the critic works
- `review_type` - was the review labeled "Fresh" or "Rotten"?
- `review_score` - review score provided by the critic
- `review_date` - date of the review in YYYY-MM-DD format
- `review_content` - text of the review

## Practicalities

Define, train and evaluate a predictive model that takes as the input the data provided. You may want to split the data into training, testing and validation sets, according to your discretion. Do not use external data for this project. You may use any algorithm of your choice or compare multiple models.

Make sure that the solution reflects your entire thought process - it is more important how the code is structured rather than the final metrics. You are expected to spend no more than 3 hours working on this project.