

Predict movie rating by user,
given his text review, date
and movie characteristics.

BABAYAN GAYANE

GROUP 143

Feature extraction

- ▶ TF-IDF of text of review
- ▶ TF-IDF of summary review
- ▶ Token occurrences in text of review
- ▶ Token occurrences in summary review

Used packages and methods

- `sklearn.feature_extraction.text.TfidfVectorizer`, `CountVectorizer`

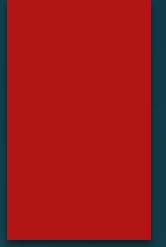
Feature extraction

Parameter tuning

N-gram range:

- ▶ Extracting only unigrams
- ▶ Extracting unigrams and bigrams
- ▶ Extracting unigrams, bigrams, trigrams

Feature extraction



Parameter tuning

- ▶ Keeping/ignoring terms with very low document frequency
- ▶ Keeping/ignoring terms with very high document frequency
- ▶ Keeping/ignoring stopwords

For texts I tried the following parameters: *min df* = 1 . . . 5,

- ▶ *max df* = 0.7 . . . 1.0, *stopwords* = "English" or None

For summaries I tried the following parameters: *min df* = 1 . . . 10,

- ▶ *max df* = 0.7 . . . 1.0, *stopwords* = "English" or None

Feature extraction

Best Parameters

- ▶ N-gram range: Unigrams and bigrams
 - ▶ Trigrams didn't improve the rating because dataset wasn't large enough.
- ▶ For text:
 - ▶ $\min df = 5$, $\max df = 0.9$, stopwords = "english"
- ▶ For summary:
 - ▶ $\min df = 5$, $\max df = 0.9$, stopwords = "english"

Feature selection

I tried the following algorithms

- ▶ Logistic Regression with L1 penalty
- ▶ Logistic Regression with L2 penalty
- ▶ Randomized Logistic Regression

Results

Features in final model

- ▶ text, summary tf-idf
- ▶ text, summary token occurrences (min_df=5, max_df=0.9)
- ▶ Logistic Regression with l2-penalty and $C = 1.0$
- ▶ RMSE is 0.81323 on public dataset, 0.81914 on private.