# RATING PREDICTIONS FROM REVIEWS GIVEN TO PRODUCTS IN ONLINE MARKETS

CPSC-6300 Applied Data Science

Spring 2023

Checkpoint 2

Dmitry Lukyanov
Sejal Bansal
Adithya Ravi
Shareef Shaik

## Codebase

https://github.com/dlcpsc6300/project

## Dataset flaws

During EDA it was discovered that the dataset is

- highly unbalanced in ratings
- highly unbalanced in aspects presented of reviews

To address the issue it was decided to generate an artificial data set of reviews with ChatGPT API, using gpt-3.5-turbo model. In order to reflect the variety in possible aspects combinations in real reviews, the next distribution was designed:

- single-aspect reviews (1 aspect) = 220 items
    - each aspect type = 55 items
- multi-aspect reviews (3 aspects) = 60 items
    - each aspect of reviews is missing = 220 items
- multi-aspect reviews (4 aspects) = 900 items

In sum - 1340 items.
For each subset there is approximately equal distribution among rates from 1 to 5. The final set contains approximately equal distribution among rates from 0 to 5 where 0 marks missing aspects.

However, due to the imperfection of ChatGPT it was impossible to achieve stable results for the whole dataset. E.g., quite regularly when the model was told to generate a review with the product aspect for a product that would be rated as 4 out of 5, it generated a review that didn't contain any flaws for the product and contained statements that the product is perfect. Thus,

while the dataset became more balanced, such irregularities affected the model performance negatively. The model gpt-4 was tried, but produced reviews with the same issues.
As due to the training approach that was used for training and that is described below, it was not possible to guarantee that there was no data leakage, for the validation 444 new reviews with the same distribution were generated.

## Model choice

As the project requires determining ratings among several predefined options, the task will be considered as classification. During EDA it was determined that the main predictors are the text and the title of reviews that makes some models, such as K-Nearest Neighbours, Decision Trees, various regression and so on, not the optimal choice. The text reviews and the purpose of the project itself have several qualities affect model choice for classification, so the next list of requirements was composed:

- Ability to handle non-linearity of boundaries between classes. As texts can widely vary, linear borders can be not sufficient for effective classification.
- Ability to handle feature selection. This is important in aspect based sentiment analysis where identifying the most important features of the text can help to more accurately classify the sentiment.
- Robustness to noise. Texts can contain multiple errors and outliers not all of which can be eliminated during the preprocessing stage that can affect the model.
- [optional] Scalability. While for the project there is no enormous volume of data, it's better to have a model that can scale well.

There are multiple options that can meet the listed requirements, such as Random Forest, Gradient Boosting, Recurrent Neural Networks, Convolutional Neural Networks, Support Vector Machines, etc. Even with RNNs and CNNs being slightly more sensitive to mislabeled data that can be important for automatically generated reviews, all mentioned models are more or less comparable in performance for sentiment analysis. [1] Thus, considering the chronology of models' emerging and the fact that the next stage of the project should include several additional models, SVM was chosen as a baseline.

## Model performance evaluation

Several options for evaluation of the model's performance for classification were considered:

- AUC-ROC was rejected as it's beneficial for imbalanced datasets that is not the case after introducing the artificially generated dataset
- precision and recall and consequently F1-score were rejected as the main interesting metric for us is a number of correctly classified ratings in respect to the real ones

Thus, accuracy was chosen for model performance evaluation. But, as we train and use separate models for each aspect instead of an multioutput-multilabel model (see "Worklog" and "Model's training and evaluation" sections), the final metric is the average of models' accuracies for aspects.

## Data cleaning and preprocessing

- html tags are removed
- emails are removed
- URLs are removed
- accented letter are replaced with standard versions
- emojis are removed
- special symbols are removed
- excessive spaces are removed
- words contractions are replaced with full forms
- grammar is fixed
- letters lowercased
- punctuation is removed
- words are lemmatized
- stopwords are removed
- titles and texts were merged
- reviews were vectorized with TF-IDF vectorizer

## Model training, evaluation and prediction

The volume of reviews was split into training and test sets several times in order to avoid a potential situation where a specific split affects model training. For each split for each aspect a separate model was trained with hyperparameters tuning with random grid search and K-fold cross-validation with 5 folds. Each time the test set was used for model performance evaluation. In that way for each split the optimal model was trained for each aspect. For each aspect the mean accuracy was calculated for all trained models and all aspects.
The achieved mean accuracy for all trained models for the test set

| Product | Delivery | Seller | Marketplace | Overall |
|---|---|---|---|---|
| 37.44% | 38.47% | 39.26% | 33.61% | 37.60% |

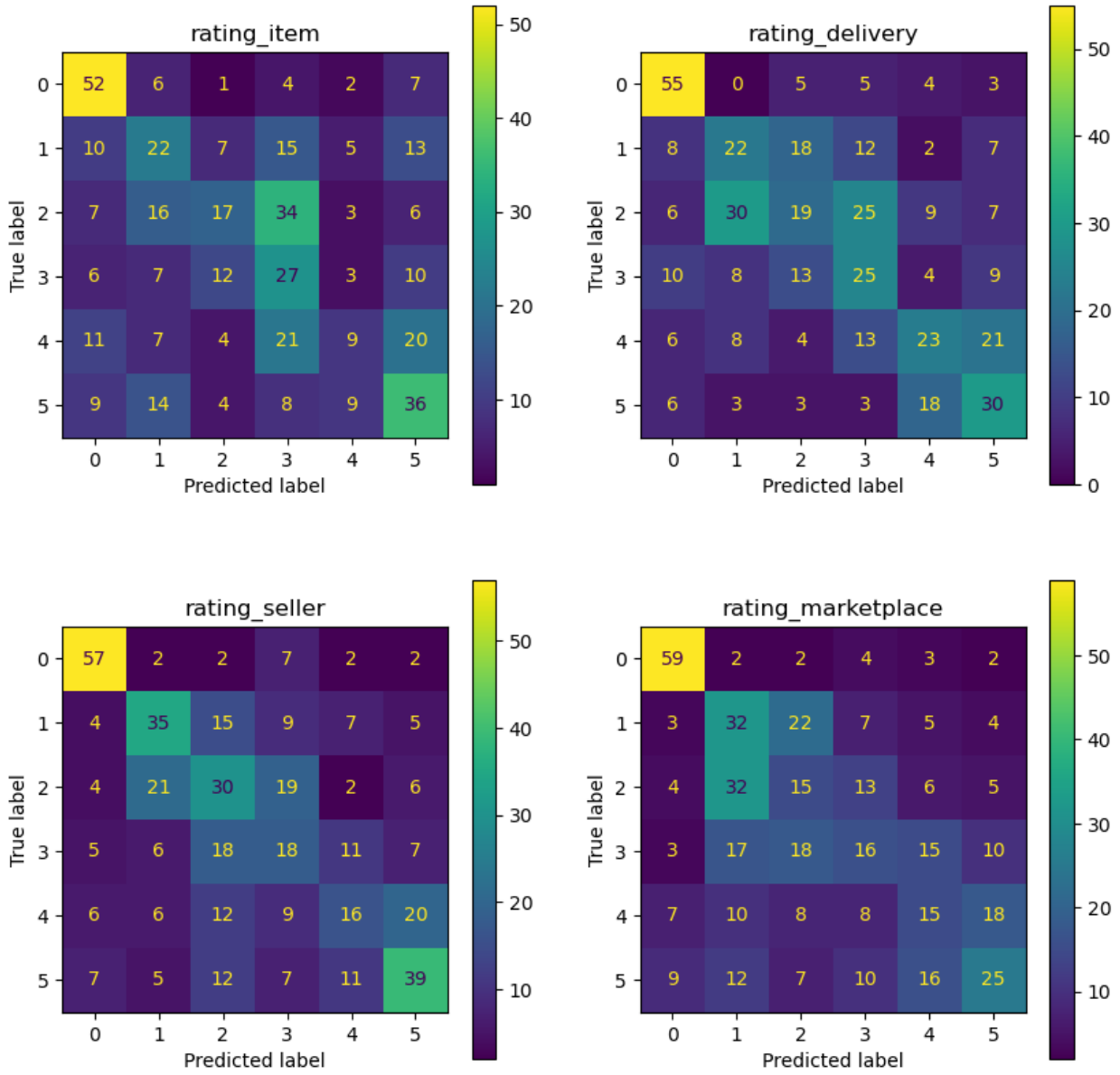
Table 1. Mean accuracy of all models on the test set

As the accuracy is calculated as a mean for multiple models, it's not possible to build a confusion matrix for this specific case, and it will be addressed further in the document.
As due to the approach that was used for training it's not possible to guarantee there is no data leakage for a specific model if a random sample of data is taken for validation and prediction after training is completed, 444 were generated separately as a validation set and included into fitting of the vectorizer that was used for reviews encoding, but not used for model training, were used for model validation.
To evaluate and validate models, the noted validation set was used. All trained models were combined into an ensemble with equal weights that allowed to increase the overall accuracy. The achieved accuracy for the ensemble on the validation set

| Product | Delivery | Seller | Marketplace | Overall |
|---|---|---|---|---|
| 36.71% | 39.19% | 43.92% | 36.49% | 39.08% |

Table 2. Accuracy of the ensemble on the validation set

The confusion matrices for the ensemble are

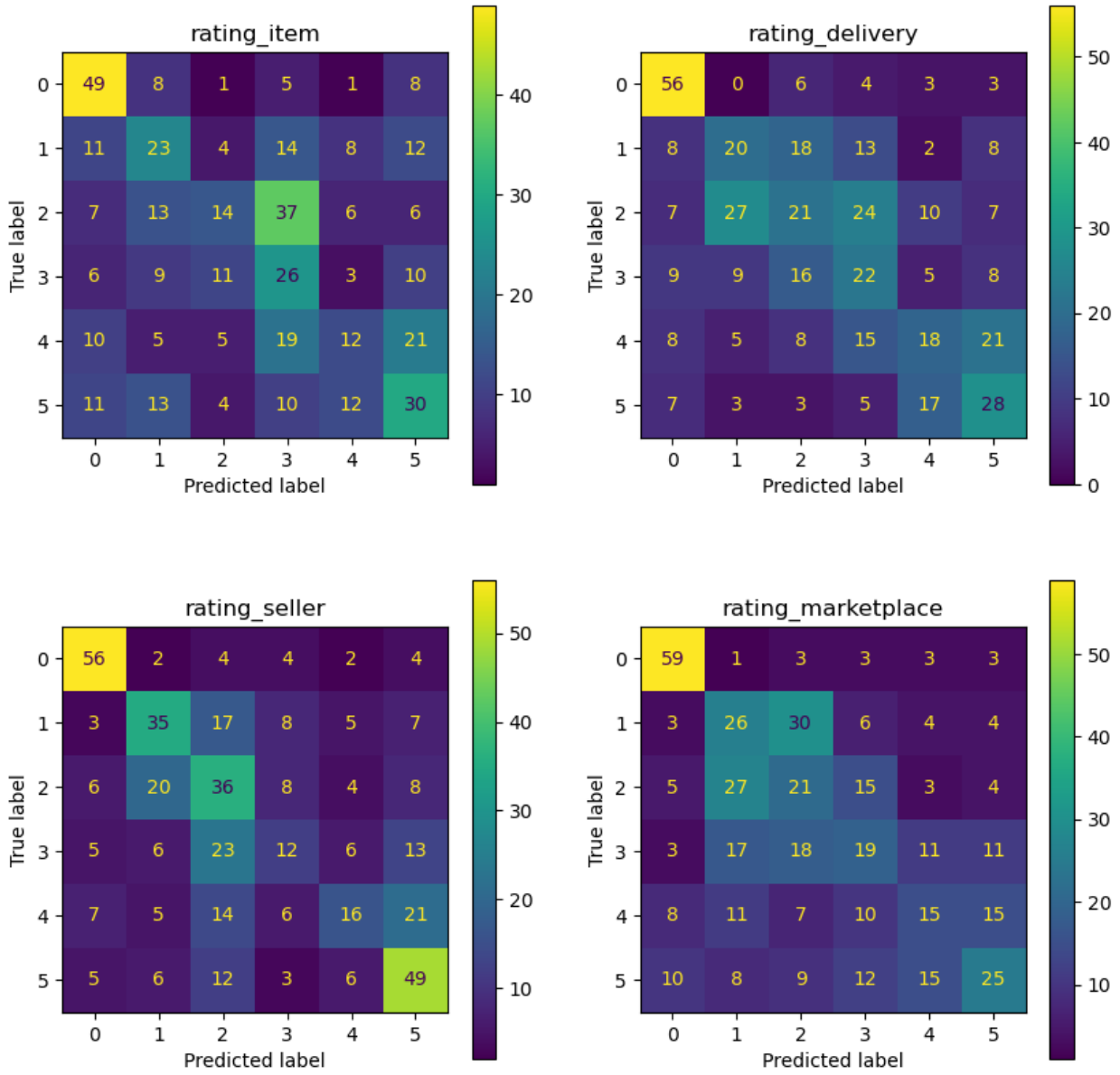



As in one case the accuracy is calculated as a mean for a set of models and in another using the ensemble, it's not possible to evaluate each specific model in detail. To address this issue the best models per aspect were taken. The achieved accuracy for such models on the validation set

| Product | Delivery | Seller | Marketplace | Overall |
|---|---|---|---|---|
| 34.69% | 37.16% | 45.95% | 37.16% | 38.74% |

Table 3. Accuracy of the best models on the validation set

The confusion matrices for the best models are




For both the ensemble and the best models from the confusion matrices it's visible that the models are capable of grasping the general attitude of the reviews, but struggles to recognize a specific rating.

As it's not possible to give a detailed description for the ensemble, there are the models descriptions for the best models per aspect

Model description for product

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.52 | 0.68 | 0.59 | 72 |
| 1 | 0.32 | 0.32 | 0.32 | 72 |
| 2 | 0.36 | 0.17 | 0.23 | 83 |
| 3 | 0.23 | 0.40 | 0.30 | 65 |
| 4 | 0.29 | 0.17 | 0.21 | 72 |
| 5 | 0.34 | 0.38 | 0.36 | 80 |
| accuracy | | | 0.35 | 444 |
| macro avg | 0.34 | 0.35 | 0.33 | 444 |
| weighted avg | 0.35 | 0.35 | 0.33 | 444 |

Model description for delivery

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.59 | 0.78 | 0.67 | 72 |
| 1 | 0.31 | 0.29 | 0.30 | 69 |
| 2 | 0.29 | 0.22 | 0.25 | 96 |
| 3 | 0.27 | 0.32 | 0.29 | 69 |
| 4 | 0.33 | 0.24 | 0.28 | 75 |
| 5 | 0.37 | 0.44 | 0.41 | 63 |
| accuracy | | | 0.37 | 444 |
| macro avg | 0.36 | 0.38 | 0.37 | 444 |
| weighted avg | 0.36 | 0.37 | 0.36 | 444 |

Model description for seller

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.68 | 0.78 | 0.73 | 72 |
| 1 | 0.47 | 0.47 | 0.47 | 75 |
| 2 | 0.34 | 0.44 | 0.38 | 82 |
| 3 | 0.29 | 0.18 | 0.23 | 65 |
| 4 | 0.41 | 0.23 | 0.30 | 69 |
| 5 | 0.48 | 0.60 | 0.54 | 81 |
| accuracy | | | 0.46 | 444 |
| macro avg | 0.45 | 0.45 | 0.44 | 444 |
| weighted avg | 0.45 | 0.46 | 0.44 | 444 |

Model description for the marketplace

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.67 | 0.82 | 0.74 | 72 |
| 1 | 0.29 | 0.36 | 0.32 | 73 |
| 2 | 0.24 | 0.28 | 0.26 | 75 |
| 3 | 0.29 | 0.24 | 0.26 | 79 |
| 4 | 0.29 | 0.23 | 0.26 | 66 |
| 5 | 0.40 | 0.32 | 0.35 | 79 |
| accuracy | | | 0.37 | 444 |
| macro avg | 0.36 | 0.37 | 0.36 | 444 |
| weighted avg | 0.36 | 0.37 | 0.36 | 444 |

While sentiment analysis of the text overall is the task that was solved numerous times and applied in practice, the project consists of the combination of two related, but more complex

subtasks - aspect based sentiment analysis and sentiment analysis with more granular quantification than “positive / neutral / negative”. Particularly, non-triviality of the second subtask that was a separate task (Task 4, Subtask E) at the International Workshop on Semantic Evaluation 2016 where the results of different participating teams were widely distributed in performance was highlighted in [2]. Considering this context, the achieved accuracy, while not enough for the method to be used in practice, noticeably exceeded the expected value for random distribution that would be 16.(6)% for 6 classes. However, it’s quite probable that the result can be improved with more advanced approaches that are to be used for the next checkpoint.

# Worklog

- as the dataset is highly unbalanced and as almost all reviews contain only the product aspect, the artificial dataset of 300 reviews with variety in styles, a mood and product categories was generated with ChatGPT API
- SVM with multioutput-multilabel classifier was trained, the achieved overall accuracy was 40.1%
- SVM with multioutput-multilabel classifier with hyperparameters tuning was trained, the achieved overall accuracy was 41.17%
- the artificial dataset of reviews was regenerated with decreased variety, the number of reviews was decreased to 1200
- SVM with multioutput-multilabel classifier with hyperparamers tuning was trained, the achieved overall accuracy was 43.02%
- for the total rating instead of per-aspects ratings SVM with one classifier with hyperparameters tuning was trained, the achieved accuracy was 75%. As it didn’t meet the project requirements, the approach was not developed further
- SVM with separate classifiers with hyperparameters tuning instead of one multioutput-multilabel classifier was trained, the achieved overall accuracy was 45.1%
- the artificial dataset was regenerated with reviews’ size increased by 4 times, the achieved overall accuracy didn’t increase
- the aspects’ ratings were mapped from the scale from 1 to 5 to the scale negative (1, 2) / neutral (3) / positive (4, 5), the achieved overall accuracy increased to 61%. As it didn’t meet the project requirements, the approach was not developed further
- removing stopwords was added to preprocessing
- SVM with separate classifiers with hyperparameters tuning was trained, the achieved overall accuracy was 46.88%
- n-grams were introduced to TF-IDF vectorization, the achieved overall accuracy increased for a not tuned model, but decreased for the tuned model
- filtering for the least frequent and the most frequent words was added to TF-IDF vectorization, the achieved overall accuracy increased for a not tuned model, but decreased for the tuned model
- pre-trained corpus word2vec-google-news-300 was used instead of TF-IDF vectorization, the achieved overall accuracy decreased
- a bug causing uneven distribution among ratings was found

- after fixing the bug the overall accuracy decreased, but differences in accuracy among classes and difference in accuracy between test and validation sets disappeared. Thus the model ability to generalize increased
- for each aspect 9 models were trained and combined in a ensemble with equal weights, the achieved overall accuracy slightly increased
- the best models for the test set per aspect were found and evaluated on the validation test

# References

[1] Das, Ringki & Singh, Thoudam Doren. (2023). Multimodal Sentiment Analysis: A Survey of Methods, Trends and Challenges. ACM Computing Surveys. 10.1145/3586075.
[2] Nakov, Preslav & Ritter, A. & Rosenthal, Sara & Sebastiani, Fabrizio & Stoyanov, V.. (2016). Semeval-2016 task 4: Sentiment analysis in Twitter. Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016). 1-18.