COMP30027 Report

1. Introduction

The goal of this research is to develop and assess supervised machine learning models that can effectively capture the relationships between book features and the corresponding ratings to predict book ratings based on their attributes. The problem involves multi-class classification.

The data used for this project is collected from Goodreads, a platform that allows users to search, rate, and review books. The dataset contains 9 book features such as name, authors, publish year, publish month, publish day, publisher, language, page numbers, and description. The dataset is divided into a training set and a test set, where the training set contains 23063 observations of the book features and the rating label. The test set contains 5767 observations of the book features, without labels. The target variable has three possible levels: 3, 4, or 5.

The models that will be analysed in this project are Linear SVM (LSVM) and Logistic Regression (LR). By evaluating the classifier's performance, the aim is to adjust the hyperparameters to get the highest accuracy possible for each classifier.

2. Methodology

The language feature is excluded from the analysis as there are too many missing values, and inaccurately imputing them may lead to biased analysis. As for the publisher feature, missing values are treated as a separate category by imputing them with a default value, 'Unknown' to retain the missing value, while still preserving the information. The count vectorizer used to pre-process the dataset produced the text features (name, authors, publisher, and description), which are a sparse matrix of count vectors that represent the words. The sparse matrices are then combined with the numeric features (130504 features in total). Then, the training dataset is separated using the holdout method with stratified sampling to ensure that the proportion of each class is preserved in both the training and test datasets.

Feature selection is then performed only on datasets used on LR model. While feature selection reduces noise and optimizes the overall likelihood of the data for LR, the removed features might contain relevant information for LSVM to find the best hyperplane that separates the classes.

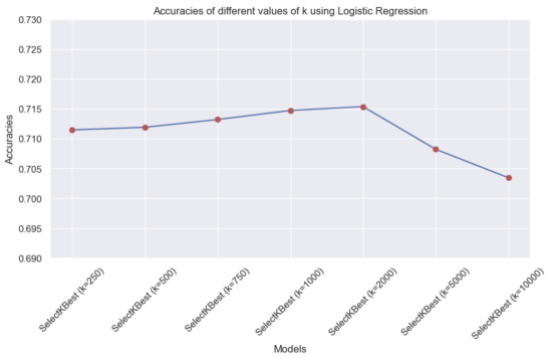

Figure 1- Line plot of accuracies of different values of k using Logistic Regression

Using the selectKbest function with the chi-square method, the best 1000 features are selected to be in the training and test datasets used on LR.

After that, feature scaling is performed only on datasets used on LSVM model. This is because scaling the features before performing LSVM can help find the best hyperplane that separates the classes.

On the other hand, LR does not require the data to have a linear relationship between the features and the target variable, thus, normalization may distort the data, making it harder to capture the underlying patterns.

3. Data Exploration

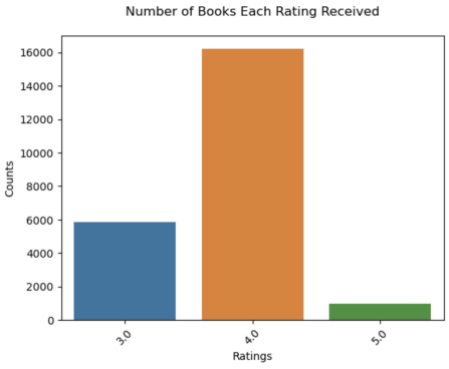

Figure 2- Bar plot of number of books each rating

received.

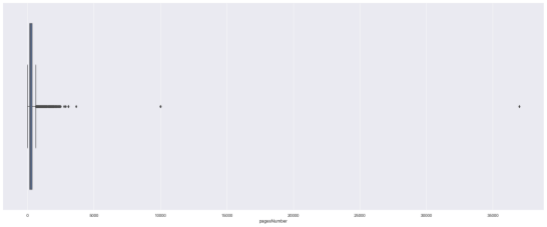

Figure 3- Box plot of books according to number of pages number (outlier detection)

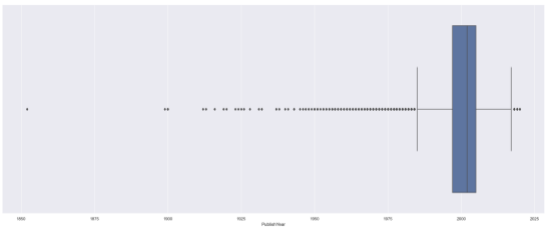

Figure 4- Box plot of books according to publish year (outlier detection)

4. Results

Linear SVM	Logistic Regression
0.7195	0.7143

Table 1- Table that compares model accuracy between Linear SVM and Logistic Regression

5. Discussion and Critical Analysis

5.1 Linear SVM

In the feature space, LSVM assumes that the classes can be separated linearly or roughly linearly. As a result, LSVM can efficiently capture trends and create reliable predictions when there are obvious patterns or linear correlations between the features of bookS and its rating labels.

This model is good at handling multiclass classification directly by identifying the best hyperplane that divides the classes in the feature space, making it especially well-suited for this dataset. To solve multiclass problems, LSVM applies one-vs-rest (OvR) method that involves training a separate binary classifier for each class, with the goal of teaching each classifier to differentiate one class from the other classes. The input sample is subjected to each binary classifier during prediction, and the class with the highest confidence or probability is selected as the predicted class.

The goal is to maximize the margin between the classes, which leads to better generalization and robustness of the model.

On implementation, the hyperparameters are set as follows:

1. C = 0.01
2. class_weight = 0.4,0.2,0.4
3. multi_class = ovr (default)

5.1.1 Hyperparameter C

The trade-off between attaining a low training error and a low classification error on new, unseen data is controlled by the hyperparameter C. A higher C value imposes a smaller margin and works to reduce misclassification, which might overfit the training dataset. On the other hand, a smaller value of C results in a model that can generalise over more datasets but also allows for a greater margin and more misclassifications in the training data.

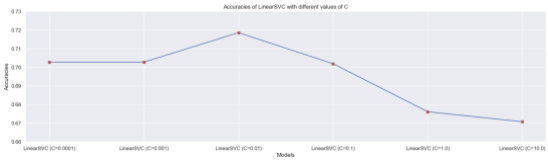

Figure 5- Line plot of Linear SVM accuracy depending on different values of C.

C=0.01 performs better than other values, giving a model accuracy of 0.7186. It implies that this number achieves a compromise between allowing certain misclassifications and reducing the training error.

5.1.2 Hyperparameter 'class_weight'

As the training dataset contains imbalanced class distribution (Figure 1), assigning specific weights to each class can help give more importance to the minority class during training.

class_weight	Accuracy
0.4, 0.1, 0.5	0.71
0.3, 0.2, 0.5	0.7182
0.4, 0.2, 0.4	0.7195

Table 2- Table of Linear SVM accuracy depending on different class_weight values.

In the training dataset, class 5.0 has the fewest samples, followed with class 3.0 and class 4.0. Thus, giving more weight to class 5.0 and 3.0 compared to class 4.0 may help improve the model performance.

After experimenting with various range of

class weights, allocating more importance to class 5.0 (40% weight) and class 3.0 (40% weight), followed by class 4.0 (20% weight), has improved the model performance, giving an accuracy of 0.7195.

5.1.3 Hyperparameter 'multi_class'

While the crammer-singer strategy aims to find a decision boundary that directly separates all classes, in the one-vs-rest (ovr) strategy, a separate binary classifier is trained for each class, treating one class as the positive class and the rest as the negative class.

multi_class	Accuracy
ovr	0.7195
crammer_singer	0.7063

Table 3- Table of Linear SVM accuracy depending on different types of multi_class.

The ovr approach gives better model performance, which suggests that the classes in the dataset can be easily distinguished by separating binary classifiers for each class, especially since the distribution of classes are imbalanced (Figure 1). The final accuracy score for the LSVM model is 0.7195.

5.2 Logistic Regression

LR is a statistical learning algorithm used for binary classification tasks. It models the relationship between a set of input features and the probability of an instance belonging to a certain class. Although LR is generally used for binary classification, it may be expanded to tackle multi-class classification issues using methods like one-vs-rest (ovr) or multinomial LR.

LR will pick one of the three classes as a pivot, then build regression models for the other two classes. After building the regression models, LR predicts the instance's class based on the class with the greatest probability score.

On implementation, the hyperparameters are set as follows:

1. solver = lbfgs (default)

2. C = 0.01

3. multi_class = multinomial

4. max_iter = 1000

5.2.1 Hyperparameter 'solver'

The choice of solver affects the optimization algorithm used to estimate the model's coefficients. Different solvers use different algorithms with varying computational efficiency and convergence properties.

solver	Accuracy
lbfgs	0.7147
saga	0.7028
sag	0.7028

Table 4- Table of Logistic Regression accuracy depending on different types of solver.

The lbfgs solver outperformed the sag and saga solvers, indicating that it converged faster or found a better set of coefficients that minimized the loss function. This also means that the classes in the training dataset are separable by a relatively simple decision boundary, compared to sag and saga, which perform well when the decision boundary is more complex.

5.2.2 Hyperparameter 'C'

The hyperparameter C controls the inverse of the regularization strength, where a smaller value of C indicates stronger regularization. Regularization finds a balance between fitting the training data well and generalizing well to unseen data.

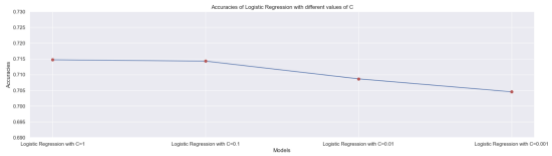

Figure 6- Line plot of Logistic Regression accuracy depending on different values of C.

Choosing C=0.1 strikes a good balance between fitting the training data and generalizing well to unseen data. This allows the model to capture the underlying patterns without memorizing noise in the training data.

A higher value of C (1) may cause the model to be more sensitive to noise or outliers, whereas a smaller value of C (0.01, 0.001) might impose excessive regularization, leading to underfitting and poor performance.

Since feature selection has been performed, some level of regularization has been incorporated into the model, thus, the C value is not as small as it would be without feature selection.

5.2.3 Hyperparameter 'multi_class'

The multinomial approach considers the correlations between different classes, whereas the ovr approach treats each class independently.

multi_class	Accuracy
multinomial	0.7143
ovr	0.7121

Table 5- Table of Logistic Regression accuracy depending on different types of multi_class.

The multinomial approach gives better model performance than the ovr approach, which suggests that the decision boundaries between classes in the dataset are not easily separable, considering that the interdependencies among classes lead to improved performance.

Thus, the better results with the multinomial approach indicate that it captures the complexity of the data more accurately and can make more reliable predictions for multi-class classification tasks.

5.2.4 Hyperparameter 'max_iter'

As the log-likelihood is concave, increasing the 'max_iter' value to 10000 allows for convergence.

5.3 Error Analysis

The model performance of LSVM is higher than LR (Table 1). This may indicate that the classes in the datasets can be separated by a linear boundary, which makes the LSVM model more effective at capturing this separation when compared to LR model that models the probability of each class.

Moreover, LSVM has higher performance is because it is less sensitive to outliers compared to LR as LSVM aims to find a decision boundary with a maximal margin. Logistic regression, on the other hand, is more influenced by outliers as it directly models the probabilities and tries to fit the data based on the maximum likelihood estimation.

```
[[ 199  974    0]
 [ 131 3110    1]
 [   5  183   10]]
```

Figure 7- Confusion matrix result using Linear SVM.

```
[[ 106 1066    1]
 [  59 3173   10]
 [   1  181   16]]
```

Figure 8- Confusion matrix result using Logistic Regression.

The confusion matrix from both model presents somewhat similar pattern. It can be observed that Class 2 (4.0) has a high number of TP and a relatively low number of FN, suggesting that the model performs well in predicting this class. Class 1 (3.0) and Class 3 (5.0) have a relatively low number of TP and a high number of FN, indicating that the model struggles to accurately predict the two classes.

This is because the provided dataset has an unbalanced distribution of classes. Since there are more examples to learn from and can better estimate the decision boundary for the majority class, the larger representation of the majority class in the training data can benefit the model in terms of learning patterns and characteristics specific to that class. The model's exposure to more examples during training allows it to perform in that class with greater accuracy.

6. Conclusion

This report reviewed the mechanism of linear SVM and logistic regression while demonstrating the impact of setting the hyperparameters of each classifier to produce a range of accuracy results using the supplied dataset. When compared to the LR model, the LSVM model exhibits superior model performance.