Judging a book by its cover
(COL774: Machine Learning)
Prof. Parag Singla

Tushar Verma (2022AIY7514)
ScAI, IIT Delhi

1 Introduction

The project involves utilizing a book cover dataset that includes both images of book covers and their corresponding titles to develop a machine learning model capable of predicting the genre of a book. The primary objective is to create a model that accurately classifies books into predefined genre categories based on visual and textual features.

2 Dataset

The dataset [1] consists of a folder named `Images`, which contains all the required images. The image names are read from the corresponding CSV files, and the corresponding images are retrieved from the `Images` folder. The goal of the model is to generate the `comp_test_y.csv` file in the same format as `train_y.csv` and `non_comp_test_y.csv`.

2.1 Files

The following files are part of the dataset:

- `train_x.csv` - the training set with columns `Image_Name`, `Title`
- `train_y.csv` - the training set with columns `Genre label`
- `non_comp_test_x.csv` - the non-competitive test set with columns `Image_Name`, `Title`
- `non_comp_test_y.csv` - the non-competitive set with columns `Genre label`
- `comp_test_x.csv` - the competitive test set with columns `Image_Name`, `Title`

The ultimate goal is to predict the `comp_test_y.csv` file.

2.2 Genre Labels and Corresponding Names

These genre names provide additional context about the data. Ultimately, the model's objective is to predict the genre labels.

Genre Label	Genre Names
0	Information Technology
1	Crafts and Hobbies
2	Romance
3	Comics
4	Bibles
5	Medicine
6	Engineering
7	Parenting
8	Reference
9	Health & Fitness
10	Self-help
11	Sports
12	Maths and Science
13	History
14	Politics
15	Calendars
16	Law
17	Religion
18	Test Preparation
19	Biographies
20	Humor
21	Young Adult
22	Cookbooks
23	Business
24	Sci-fi
25	Children's books
26	Photography
27	Literature
28	Travel
29	Mystery

Table 1: Table lists the genre labels along with their corresponding names

3 Model

The ResBert model integrates BERT and ResNet18 to facilitate cross-modal interactions between text and image features, utilizing key design choices to enhance performance. The BERT module leverages a pretrained 'bert-base-cased' model to process text inputs into embeddings, while the ResNet18 module modifies a pretrained ResNet18 by removing the final layers and adding a Conv2d layer (3x3 kernel, 768 output channels) to align image features with BERT's embedding dimension. A central feature is the cross-attention mechanism (nn.MultiheadAttention), which enables BERT's text embeddings (as query) to attend to the ResNet-derived image features (keys and values), fostering effective multimodal feature interaction for improved classification.

Module	Layer	Output Shape
BERT Module	Pretrained BERT ('bert-base-cased')	(batch, 61, 768)
ResNet18 Module	Pretrained ResNet18 (modifications: Conv2d 3x3)	(batch, 49, 768)
Res_Bert Model	Cross-Attention (8 heads) Adaptive Pooling	(batch, 61, 768) (batch, 768)
Classifier	Linear (256), ReLU, Dropout Linear (30)	(batch, 256) (batch, 30)

Table 2: Concise Architecture of the Res_Bert Model

4 Results

Class	Precision	Recall	F1-Score	Support
0	0.95	0.97	0.96	1134
1	0.94	0.94	0.94	1155
2	0.93	0.97	0.95	1148
3	0.96	0.95	0.96	1116
4	0.84	0.96	0.90	1133
5	0.93	0.92	0.92	1153
6	0.93	0.91	0.92	1132
7	0.95	0.91	0.93	1132
8	0.93	0.91	0.92	1149
9	0.92	0.88	0.90	1144
10	0.86	0.94	0.90	1129
11	0.95	0.93	0.94	1135
12	0.93	0.91	0.92	1175
13	0.91	0.98	0.94	1107
14	0.93	0.85	0.89	1098
15	0.98	0.98	0.98	1149
16	0.97	0.93	0.95	1136
17	0.97	0.82	0.89	1121
18	0.96	0.97	0.96	1146
19	0.91	0.92	0.92	1156
20	0.94	0.90	0.92	1149
21	0.90	0.84	0.87	1100
22	0.94	0.99	0.96	1116
23	0.91	0.95	0.93	1158
24	0.96	0.88	0.92	1172
25	0.83	0.97	0.89	1132
26	0.94	0.90	0.92	1135
27	0.87	0.83	0.85	1168
28	0.96	0.97	0.96	1169
29	0.88	0.97	0.92	1153
accuracy	0.92			34200
macro avg	0.93	0.92	0.92	34200
weighted avg	0.93	0.92	0.92	34200

Table 3: Classification Report Training Data

4.1 Performance on Training Data

Accuracy: The overall accuracy of the model is 92%. This indicates that the model correctly classifies 92% of the instances, showing excellent performance across the training set.

4.2 Class-Level Insights

- **Class 15:** Achieves outstanding performance with a precision of 0.98 and recall of 0.98, demonstrating that the model is highly reliable in identifying this class, with very few misclassifications.
- **Class 13:** Also performs excellently, with precision of 0.91 and recall of 0.98, indicating that this class is well identified, with very few false negatives.
- **Class 0:** Shows very strong performance with a precision of 0.95 and recall of 0.97, which highlights that the model is highly accurate and reliable for this class.
- **Class 4:** Despite having lower precision of 0.84, it maintains a high recall of 0.96, suggesting that the model is good at identifying most of the instances of this class, but with a slightly higher rate of false positives.
- **Class 17:** Exhibits a slightly lower performance with precision of 0.97 and recall of 0.82, which indicates that the model tends to correctly identify this class, but some false negatives may occur.
- **Class 25:** Shows relatively lower performance with a precision of 0.83 and recall of 0.97. While the model is able to identify most instances, the precision indicates that there is a higher number of false positives compared to other classes.

4.3 Averages

- **Macro Average:** The macro average for precision, recall, and F1-score are all around 0.93, 0.92, and 0.92, respectively. This suggests that the model performs consistently well across all classes, treating each class equally without being biased by class imbalance.
- **Weighted Average:** The weighted averages for precision, recall, and F1-score are also approximately 0.93, 0.92, and 0.92, respectively. This indicates that the model's performance is similar across different classes, with the larger classes having a slight impact on the overall averages.

Class	Precision	Recall	F1-Score	Support
0	0.70	0.73	0.72	162
1	0.61	0.71	0.66	189
2	0.57	0.60	0.59	184
3	0.79	0.62	0.70	216
4	0.51	0.72	0.59	184
5	0.64	0.69	0.66	150
6	0.65	0.65	0.65	209
7	0.61	0.60	0.60	168
8	0.58	0.53	0.55	198
9	0.58	0.50	0.54	199
10	0.51	0.58	0.54	217
11	0.72	0.61	0.66	185
12	0.56	0.56	0.56	186
13	0.52	0.66	0.58	203
14	0.55	0.38	0.45	177
15	0.97	0.95	0.96	194
16	0.80	0.72	0.76	193
17	0.75	0.46	0.57	198
18	0.82	0.83	0.83	175
19	0.49	0.49	0.49	202
20	0.49	0.43	0.45	190
21	0.47	0.44	0.45	204
22	0.79	0.88	0.83	212
23	0.59	0.64	0.62	188
24	0.70	0.46	0.56	158
25	0.48	0.67	0.56	200
26	0.55	0.49	0.52	198
27	0.38	0.44	0.41	197
28	0.68	0.70	0.69	165
29	0.65	0.71	0.68	199
accuracy			0.61	5700
macro avg	0.62	0.61	0.61	5700
weighted avg	0.62	0.61	0.61	5700

Table 4: Classification Report Testing Data

4.4 Performance on Testing Data

Accuracy: The overall accuracy of the model is 61%. This indicates that the model correctly classifies around 61% of the instances. While not very high, this result shows that the model performs decently across the test set.

4.5 Class-Level Insights

- **Class 15:** Shows outstanding performance with a precision of 0.97 and a recall of 0.95. This suggests that the model is highly reliable in identifying this class and

performs exceptionally well overall, making it the best-performing class.

- **Class 18:** Also performs very well with a precision of 0.82 and a recall of 0.83, which demonstrates that the model does a good job identifying this class with high accuracy and recall.
- **Class 0:** Has a solid precision of 0.70 and recall of 0.73, which indicates that the model is fairly reliable for this class and performs well with a reasonable trade-off between precision and recall.
- **Class 14:** Exhibits poor performance with a precision of 0.55 and a recall of 0.38. This indicates a low level of correct predictions and a high number of false positives and false negatives, suggesting potential issues with data imbalance or misclassification.
- **Class 27:** Has a particularly low performance with precision of 0.38 and recall of 0.44. This indicates that the model struggles significantly to correctly identify this class, likely due to limited data or challenges in distinguishing it from others.

4.6 Averages

- **Macro Average:** The macro average precision, recall, and F1-score are all around 0.62, suggesting an overall balanced performance across the classes with no major outliers affecting the results.
- **Weighted Average:** The weighted averages for precision, recall, and F1-score are also approximately 0.61. This indicates that the model's performance is consistent across different classes, though the influence of the larger classes slightly affects the overall average.

5 Conclusion

5.1 Overall Accuracy

The model achieves 92% accuracy on the training data, which is an excellent result, indicating that the model has learned to correctly classify most instances. However, the testing accuracy is 61%, which is significantly lower, suggesting that the model may be overfitting to the training data and not generalizing well to unseen data.

5.2 Class-Level Performance

On the training data, the model performs exceptionally well across most classes, with several classes exhibiting high precision and recall values, such as Class 15 (precision: 0.98, recall: 0.98) and Class 13 (precision: 0.91, recall: 0.98). These classes indicate that the model is highly capable of correctly identifying instances with minimal misclassification.

In contrast, on the testing data, the model's performance varies more widely. While some classes like Class 15 maintain excellent precision and recall (precision: 0.97, recall: 0.95), others, such as Class 4 (precision: 0.51, recall: 0.72) and Class 19 (precision: 0.49, recall: 0.49), show poor results. This discrepancy points to potential challenges with

the model's ability to generalize effectively, particularly for underrepresented or more complex classes.

5.3 Overfitting Concerns

The significant gap between the training and testing accuracy (92% vs 61%) strongly suggests overfitting. The model is performing exceptionally well on the training data but struggles to generalize to the testing data, where the accuracy drops significantly. This overfitting issue could be due to an overly complex model, insufficient regularization, or a need for more diverse training data.

6 Future Considerations

6.1 Model Generalization

Although the data is perfectly balanced, the model's performance on the testing set suggests that it may still face challenges in generalizing to unseen data. To address this, we plan to explore additional regularization techniques, such as dropout, weight decay, or data augmentation, to enhance the model's ability to generalize better. Furthermore, evaluating the model on more diverse and challenging test data could help identify specific areas of weakness and improve its robustness.

6.2 Model Architecture

Currently, the model is based on a pre-trained ResNet, which is trained on the ImageNet dataset. While this provides a strong starting point, we believe that the use of a model specifically tailored to the book cover dataset may yield better feature extraction and improve overall performance. To this end, we plan to experiment with generative models such as a Beta Variational Autoencoder (Beta-VAE) or similar architectures. These models would allow us to extract features that are more aligned with the unique characteristics of the book cover dataset, potentially leading to more discriminative and meaningful representations for classification.

6.3 Optimization of Hyperparameters

Future work could also involve further optimization of hyperparameters, such as learning rate, batch size, and the number of layers used in the model. Techniques such as grid search or random search could be used to systematically explore the parameter space and improve model performance. Additionally, more advanced optimization methods like Bayesian optimization could provide more efficient ways to find optimal hyperparameters.

References

[1] V. B. Chirag Mohapatra, "Judging a book by its cover," 2022.