DSAA6100 Movie Review Sentiment Classification

Egor Bodrov, Gleb Onore, Evgeniy Dubskiy

May 2025

Abstract

This project addresses the problem of automatic sentiment classification of movie reviews using modern NLP models. We compare several transformer-based approaches on the Kaggle. The goal is to achieve high accuracy in binary sentiment classification (positive/negative) and analyze the impact of model choice and preprocessing.

Our project: `https://github.com/EgorBodrov/ods-nlp/tree/main`.

1 Introduction

Sentiment analysis of movie reviews is a classic NLP task with practical applications in recommendation systems, social media monitoring, and customer feedback analysis. The challenge lies in handling informal language, sarcasm, and short texts. We focus on leveraging recent advances in transformer architectures (DistilBERT, DeBERTa v3) and compare their performance on a real-world dataset.

1.1 Team

Egor Bodrov responsibilities:

- Exploratory Data Analysis
- Research for baseline models
- Training architectures on dataset
- Writing the report
- Implementation of data annotation approach

Gleb Onore responsibilities:

- Submitting to competition
- Building hypotheses

- Research for baseline models
- Training architectures on dataset
- Hyperparameters selection
- Writing the report

Evgeniy Dubskiy responsibilities:

- Research for baseline models
- Training architectures on dataset
- Writing the report
- Competition research

2 Related Work

- **Sentiment Analysis of Movie Reviews: A New Feature Selection Method**
  This paper proposes a novel hybrid feature selection method for sentiment analysis of movie reviews, combining statistical approaches (like chi-square) with semantic similarity (WordNet-based). The method aims to enhance classification accuracy by selecting more informative and sentiment-relevant features, and it shows improved performance over traditional approaches on standard movie review datasets.

- **Sentiment Analysis of Movie Reviews Using BERT**
  This paper fine-tunes BERT (plus Bi-LSTM) on standard movie-review datasets for binary classification. The model achieves state-of-the-art accuracy and also proposes a heuristic to compute overall sentiment polarity across multiple reviews.

- **A Comparative Analysis of Transformer-Based Models for Sentiment Classification**
  This paper evaluates multiple transformer models — including BERT, RoBERTa, ELECTRA, and others — on various sentiment analysis datasets. The study highlights model performance trade-offs, showing that RoBERTa and DeBERTa often outperform BERT in both accuracy and robustness.

- **Fine-Tuning Pre-trained Transformers for Sentiment Analysis**
  The study investigates how fine-tuning pre-trained transformer models (BERT, RoBERTa, ELECTRA) affects sentiment classification. It emphasizes the importance of task-specific adaptation and demonstrates that careful fine-tuning can close the gap between lighter and heavier models.

- **Distilled Transformers for Sentiment Classification on Resource-Constrained Devices**
  This paper explores the use of distilled transformer models like MobileBERT and TinyBERT for efficient sentiment classification. It compares their trade-offs in latency, size, and accuracy, offering insights into deploying NLP models in real-time or embedded environments.

3 Model Description

Model Description

Our final solution is based on the **DeBERTaV3** architecture fine-tuned for sentiment classification. **DeBERTaV3** improves upon earlier transformer models by incorporating *disentangled attention mechanisms* and *enhanced position encodings*, allowing for better semantic representation of natural language inputs, especially in complex or nuanced sentiment tasks such as movie or game reviews.

We utilize the `AutoTokenizer` and `AutoModelForSequenceClassification` classes from the HuggingFace Transformers library. The tokenizer is responsible for converting raw input text into token IDs understandable by the model. It also manages truncation and subword tokenization.

We use the pretrained DeBERTaV3 tokenizer to split and encode sentences while maintaining semantic boundaries. `AutoModelForSequenceClassification` wraps DeBERTaV3 with a classification head — a feedforward layer mapping the `[CLS]` token output to sentiment classes.

We minimize the cross-entropy loss between predicted class distributions and true labels:

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)\,,$$

$$\begin{aligned}
\mathcal{L} &\quad \text{— loss function} \\
C &\quad \text{— number of classes} \\
y_i \in \{0,1\} &\quad \text{(true label, one-hot)} \\
\hat{y}_i \in [0,1] &\quad \text{(predicted probability)} \\
\log(\hat{y}_i) &\quad \text{— log of predicted probability}
\end{aligned}$$

At the heart of the DeBERTaV3 model lies the Scaled Dot-Product Attention mechanism:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V\,.$$

$$
\begin{aligned}
Q \in \mathbb{R}^{n \times d_k} &\quad \text{— query matrix} \\
K \in \mathbb{R}^{n \times d_k} &\quad \text{— key matrix} \\
V \in \mathbb{R}^{n \times d_v} &\quad \text{— value matrix} \\
\sqrt{d_k} &\quad \text{— scaling factor (dimension of keys)} \\
\text{softmax} &\quad \text{— converts scores into probabilities}
\end{aligned}
$$

DeBERTaV3's disentangled attention decouples content and positional information, improving the model's ability to generalize across varied review structures and languages.

4 Dataset

4.1 Source and Access

- **Dataset:** Kaggle dsaa-6100-movie-review-sentiment-classification
- **License:** For academic use, available via Kaggle competition page.
- **Files:**
  - `movie_reviews.csv` (train): columns `Id`, `text`, `label` (0 = negative, 1 = positive)
  - `test_data.csv` (test): columns `Id`, `text`

4.2 Statistics & EDA

- **Train size:** 40,000 reviews
- **Test size:** 10,000 reviews
- **Class balance:**
  - Positive (1): 50.1%
  - Negative (0): 49.9%
- **Text characteristics:**
  - Length distribution: 100–500 characters (majority)
  - Average words per review: ∼100 words
  - Duplicate reviews present: ∼2.5% of total reviews
- See `experiments/EDA.ipynb` for detailed visualizations:
  - Class distribution plots
  - Text length histograms
  - Word clouds for frequent terms
  - Comparison of positive vs negative review lengths

4.3 Data Annotation

Our team implemented an LLM-based agent to enrich the dataset by generating additional samples based on the original training data. This approach extended the dataset with 1,000 synthetic samples per class. All generated samples were unique and semantically relevant.

The final solution was trained on this augmented dataset, leading to improved generalization and robustness.

5 Experiments

5.1 Metrics

- **Primary metric:** Accuracy
- **Rationale:** Dataset is balanced, accuracy is the competition metric

5.2 Experiment Setup

- **Data split:** 90% train / 10% validation (stratified)
- **Training parameters:**
  - Batch size: 16
  - Max sequence length: 256 tokens
  - Optimizer: AdamW (lr = 2e-5)
  - Epochs: 3–4 with early stopping
  - Loss: Cross-entropy
- **Hardware:** GPU (CUDA if available)
- **Model selection:** Best checkpoint by validation accuracy

5.3 Experiments

1. **Zero-shot Baselines**
   - DistilBERT (no fine-tuning): `sarahai/movie-sentiment-analysis`
   - DistilBERT SST-2: Pre-trained on Stanford Sentiment Treebank
   - DistilBERT Amazon: Pre-trained on Amazon reviews
2. **DeBERTa v3 IMDB (dfurman/deberta-v3-base-imdb)**
   - Pre-trained on IMDB reviews
   - Fine-tuned on our dataset
   - Implementation: `experiments/deberta-v3-base-imdb.ipynb`

3. **DistilBERT Fine-tuned**
    - Base: `distilbert-base-uncased`
    - Custom fine-tuning on movie reviews
- Implementation: `experiments/distilbert`$_f$*inetuned.ipynb*
4. **AlBert Fine-tuned**
    - Base: `albert-base-v2`
    - Custom fine-tuning on movie reviews
    - Implementation: `experiments/albert-finetuned.ipynb`
5. **XLM Roberta Fine-tuned**
    - Base: `xlm-roberta-base`
    - Custom fine-tuning on movie reviews
    - Implementation: `experiments/xml-bert-finetuned.ipynb`
6. **DeBertaV3 pretrain on IMDB**
    - Base: `dfurman/deberta-v3-base-imdb`
    - Custom fine-tuning on movie reviews
    - Implementation: `experiments/deberta-v3-base-imdb.ipynb`
7. **ELECTRA Fine-tuned**
    - Base: `google/electra-base-discriminator`
    - Custom fine-tuning on movie reviews
    - Implementation: `experiments/electra-finetuned.ipynb`
8. **MobileBert Fine-tuned**
    - Base: `nreimers/MiniLM-L6-H384-uncased`
    - Custom fine-tuning on movie reviews
    - Implementation: `experiments/mobile-bert-finetuned.ipynb`

6 Results

This section now showcases the performance of each model. The table below summarizes the performance of each model on the movie review sentiment analysis task:

As shown in the table below, the XLM-Roberta model achieved the highest validation accuracy of 0.92, indicating its strong capability in multilingual sentiment understanding. It was followed closely by DeBERTa v3 (IMDB) at 0.90, and both DistilBERT (fine-tuned) and ELECTRA models at 0.89. The

base DeBERTa v3 and DistilBERT (pretrained) also performed well, scoring 0.89 and 0.85 respectively. In contrast, the MobileBERT model showed significantly lower performance (0.55), highlighting its limitations in this particular task. Overall, transformer models with task-specific fine-tuning consistently outperformed zero-shot baselines.

Model	**Validation Accuracy**	**Responsible**
DistilBERT (pretrained, no FT)	∼0.85	Egor B.
DistilBERT (fine-tuned)	∼0.89	Egor B.
DistilBERT (SST-2, zero-shot)	∼0.83	Egor B.
DistilBERT (Amazon, zero-shot)	∼0.81	Gleb O.
DeBERTa v3 (fine-tuned, IMDB)	∼0.90	Gleb O.
DeBERTa v3 (fine-tuned, base)	∼0.89	Gleb O.
XLM (fine-tuned, base)	∼0.92	Egor B.
AlBert (fine-tuned, base)	∼0.80	Evgeniy D.
ELECTRA (fine-tuned, base)	∼0.89	Evgeniy D.
MobileBERT (fine-tuned, base)	∼0.55	Evgeniy D.

Table 1. Experiments results.

Below you can find an example of classification results

Review	**Label**
What can possibly said about this movie other than viewer beware Christmas Evil should come with a w...	0
I have a problem with the movie snobs who consider Americans to be uncouth semi literates unable to...	0
This movie tries hard but completely lacks the fun of the 1960s TV series that I am sure people do r...	1
John Wayne Albert Dekker compete for oil rights on Indian territory and for the attention of Martha...	0
First of all let me underline that Im not a great fan of political correctness In fact I like satire...	0

Table 2: Examples of reviews with sentiment labels

7 Conclusion

The Kaggle competition focuses on sentiment analysis of movie reviews. Most reviews consist of lengthy sentences, often exceeding 200 words. The primary evaluation metric is binary accuracy. DeBERTa, with its disentangled attention mechanism, is a powerful model well-suited for this challenge. Nonetheless, comparing its performance against various baseline models is essential to identify the optimal solution. In summary, although DeBERTa shows strong potential for this task, thorough benchmarking with baseline models is necessary to select

the best-performing model for sentiment classification in this competition. We compared several transformer-based models for movie review sentiment classification. Fine-tuned DeBERTa v3 on extended dataset performed the best results on competition.