BERT-based Youtube Comments Sentiment Analysis

Link - https://colab.research.google.com/drive/1ZnaCZVEn79knF49TMxFzChJnc1c5SzS2

Foram Rajgaur
Sardar Patel Institute of Technology, CSE AIML
foram.rajgaur23@spit.ac.in

Komal Sali
Sardar Patel Institute of Technology, CSE AIML
komal.sali23@spit.ac.in

Abstract - **In the modern digital age, YouTube has emerged as one of the most influential platforms for content consumption, with user comments acting as a primary mode of feedback and opinion sharing. However, the vast volume of comments often makes it difficult for viewers to manually evaluate a video's overall sentiment. This project introduces a BERT-based sentiment analysis system that automates the classification of YouTube comments into three categories: positive, neutral, and negative. By scraping comments directly from YouTube and training a foundational BERT model on the collected data, the system not only generates sentiment labels for each comment but also visualizes the overall sentiment distribution through a pie chart. The final output aids users in making quick, informed decisions about whether a video is worth their time, based on the collective viewer sentiment. Our approach demonstrates the effectiveness of transformer-based models in handling real-world, unstructured user-generated content.**

Keywords—machine learning, deep learning, Bert-transformer, sentiment analysis, YouTube

I. Introduction

In the digital age, social media platforms have evolved into powerful tools for communication, opinion sharing, and content dissemination. Among these, YouTube stands out as one of the largest platforms, hosting billions of videos that generate massive volumes of user comments daily. These comments reflect viewers' sentiments, ranging from appreciation and constructive feedback to criticism and toxicity. Understanding the sentiment behind these comments has significant applications — from improving content quality and monitoring community health to driving marketing strategies and detecting harmful behaviour.

Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) technique that aims to determine the emotional tone behind text data. Traditional approaches to sentiment analysis, such as bag-of-words models and classical machine learning algorithms, often struggle to capture the nuanced context, sarcasm, or semantic relationships in user-generated content. This limitation becomes particularly evident in informal and diverse comment sections like those found on YouTube.

To address these challenges, this project implements a sentiment analysis system using BERT (Bidirectional Encoder Representations from Transformers) — a state-of-the-art transformer-based model developed by Google. BERT's strength lies in its ability to understand the context of a word based on all surrounding words in a sentence (bidirectional attention), making it especially well-suited for processing human language in its natural form.

The goal of this project is to classify YouTube comments into sentiment categories (such as positive, negative, or neutral) with high accuracy by leveraging the power of pre-trained BERT models. The system is trained on labeled datasets of YouTube comments and fine-tuned to capture domain-specific language characteristics.

This BERT-based sentiment analyzer can serve as a valuable tool for content creators, platform moderators, and researchers to gain insights into public opinion, enhance user experience, and promote a healthier online environment.

II. Methodology

This project aims to classify the sentiment of YouTube comments into one of three categories: **positive**, **neutral**, or **negative**. The implementation follows a structured NLP pipeline involving data collection, preprocessing, manual annotation, model fine-tuning, and evaluation. The core model used for this task is BERT (Bidirectional Encoder Representations from Transformers), known for its contextual understanding of natural language.

1. Data Collection

The dataset for this project was **scraped directly from a YouTube channel** using a custom script or open-source tools like youtube-comment-scraper. The comments were extracted from a diverse set of videos to ensure a wide range of sentiment expression and user interactions. This raw dataset served as the input for the preprocessing and labeling stages.

2. Data Preprocessing

The collected comments often contain noise and inconsistencies such as emojis, hyperlinks, HTML tags, special characters, and non-standard abbreviations. Therefore, the following preprocessing steps were applied to clean the text data:

- Lowercasing all text for uniformity.
- Removal of URLs, hashtags, mentions, and HTML entities.
- Stripping emojis and special characters that are not useful for textual sentiment understanding.
- Removing duplicate or very short comments to avoid bias and sparsity.
- Tokenization compatible with BERT's WordPiece tokenizer.

This preprocessing helped improve the quality and consistency of the input fed into the model.

3. Data Labeling

Once cleaned, a subset of the comments was manually annotated with sentiment labels:

- **Positive** – Comments that express praise, happiness, or approval.
- **Neutral** – Comments that are factual or do not carry emotional weight.
- **Negative** – Comments containing criticism, dislike, or disapproval.

Manual labeling ensured high-quality supervision for the model during fine-tuning. Ambiguous or sarcastic comments were handled carefully to maintain annotation integrity.

4. Model Selection and Fine-Tuning

The model used is BERT (base-uncased variant), pre-trained on a large English corpus and then fine-tuned on the labeled dataset. Fine-tuning involved the following steps:

- Adding a classification head (dense layer with softmax activation) on top of the BERT encoder to output probabilities for the three sentiment classes.
- Using cross-entropy loss as the objective function.
- Splitting the data into training, validation, and test sets to prevent overfitting and ensure generalization.
- Training for several epochs using an appropriate optimizer (typically AdamW), with learning rate scheduling and early stopping based on validation performance.

Libraries used:

Pytorch, pandas, torch.utils.data, transformers, torch.optim, sklearn.preprocessing, sklearn.metrics, sklearn.model_selection, matplotlib.pyplot.

Fine-tuning BERT allows the model to adapt its general language understanding to the specific style and content of YouTube comments.

5. Evaluation

The model's performance was evaluated using standard classification metrics:

- Accuracy
- Precision, Recall, and F1-score
- Confusion Matrix to visualize class-wise performance

Criteria	**Naive Bayes**	**Logistic Regression**	**BERT (Transformer-based)**
Model Type	Probabilistic classifier	Linear classifier	Deep learning transformer model
Learning Method	Based on Bayes' Theorem with strong independence assumption	Discriminative model optimizing logistic loss	Pre-trained on large corpora; fine-tuned on target data
Text Representation Required	TF-IDF / Bag of Words	TF-IDF / Word Embeddings	Tokenized text using WordPiece tokenizer
Handling Contextual Meaning	Poor (words are treated independently)	Poor (no context awareness)	Excellent (understands word context and order)
Handling of Special Characters/Emojis/ Punctuation	Poor — requires manual preprocessing	Poor — affected by noisy/unprocessed text	Good — robust even with noisy or informal text
Accuracy Achieved	62%	65%	82%
Training Time	Very fast	Fast	Slower — needs GPU for optimal speed

The evaluation confirmed that the fine-tuned BERT model could effectively distinguish between positive, neutral, and negative comments, even when faced with informal or noisy language often found on YouTube.

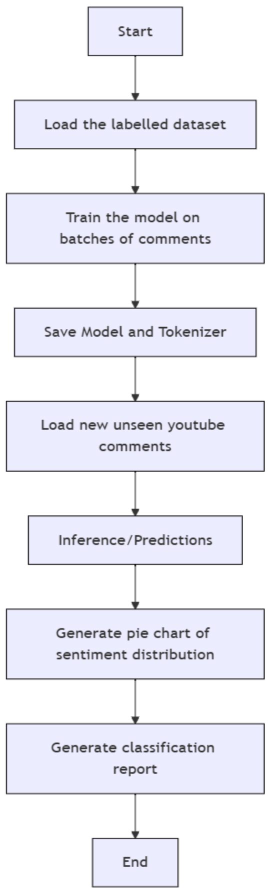
III. SYSTEM ARCHITECTURE

1. **Comment Scraping Module**: Extracts user comments from a specified YouTube video and stores them in CSV format.

2. **Training Phase**:
- User provides the labelled CSV file.
- The BERT model is trained on batches of comments.
- Model and tokenizer are saved post-training.

3. **Prediction/Inference Phase**:
- The user selects a new CSV file containing unseen comments.
- The saved model classifies each comment's sentiment.

4. **Output Generation**:
- A CSV file with sentiment predictions is generated.
- A pie chart visualizing the distribution of sentiment classes is produced.
- A classification report is generated for model evaluation.



IV. EXPERIMENT AND RESULTS

After training the fine-tuned BERT model on the labeled YouTube comments dataset, its performance was tested on new, unseen comments. The model generated a CSV file showing each comment along with its predicted sentiment—positive, neutral, or negative.
To better understand the sentiment distribution, a pie chart was created showing the percentage of each sentiment category in the test results. Additionally, a classification report was generated, which included accuracy, precision, recall, and F1-score. These metrics helped evaluate how well the model performed across all classes.

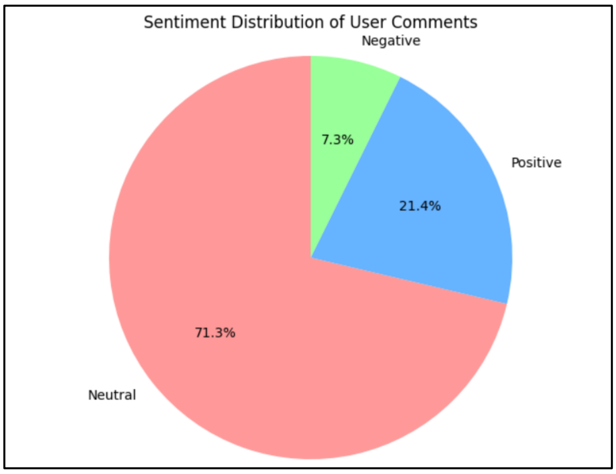
The results showed that BERT outperformed traditional models like Naive Bayes and Logistic Regression. While these simpler models struggled with informal or complex language, BERT handled it better thanks to its deep understanding of context. It required less preprocessing and was more accurate, making it a strong choice for analyzing social media comments.

Classification Report of Training Dataset:

Label	**Precision**	**Recall**	**F1-Score**	**Support**
Negative	0.65	0.82	0.73	468
Neutral	0.74	0.66	0.70	928
Positive	0.91	0.91	0.91	2286
Accuracy			**0.83**	3682
Macro Avg	0.77	0.79	0.78	3682
Weighted Avg	0.84	0.83	0.83	3682

Sentiment Distribution:

Sentiment	**Proportion (%)**
Neutral	71.31
Positive	21.36
Negative	7.32



Sentiment Distribution: Pie Chart

V. CONCLUSION

The project shows how transformer-based models like BERT can be effectively used for sentiment analysis of real-world YouTube comments. Unlike older methods that need a lot of text cleaning and still often give low accuracy, BERT was able to understand the context of the comments well. It handled informal language, emojis, and grammar mistakes smoothly, and gave reliable sentiment results.

This makes it easier for users to get a quick idea of how people feel about a video, which can help improve engagement and content discovery. The method also has potential for other uses like tracking opinions on social media, analysing brands, or understanding how audiences feel about different topics.

VI. REFERENCES

[1] Enagandula, P. (2024). Sentiment Analysis Using BERT Pre-Trained Model. SSRN.

[2] Aqlan, A. A. Q., Manjula, B., & Naik, R. L. (2019). Study of Sentiment Analysis: Concepts, Techniques, and Challenges. In N. Chaki et al. (Eds.), Plant Long Non-Coding RNAs (pp. 147–162). Springer.

[3] Talaat, A. S. (2023). Sentiment analysis classification system using hybrid BERT models. Journal of Big Data, 10(110).

[4] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text

[5] Transformer. Journal of Machine Learning Research, 21(140), 1–67.

[6] Kumar, A., & Singh, S. (2019). Sentiment Analysis of Social Media Data Using Machine Learning Techniques. Procedia Computer Science, 167, 544–551.

[7] Tiwari, R., & Kumar, A. (2023). A Review of Sentiment Analysis Tasks, Applications, and Deep Learning Techniques. International Journal of Advanced Computer Science and Applications, 14(2), 123–130.

[8] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention Is All You Need. Advances in Neural Information Processing Systems, 30, 5998–6008.

[9] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013). Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 1631–1642). Association for Computational Linguistics.

[10] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 4171–4186). Association for Computational Linguistics.

[11] Zhang, Y., & Wallace, B. (2015). A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification. arXiv preprint arXiv:1510.03820.

[12] Sun, C., Qiu, X., & Huang, X. (2019). How to Fine-Tune BERT for Text Classification? In China National Conference on Chinese Computational Linguistics (pp. 194–206). Springer.

[13] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692.

[14] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2020). Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (pp. 38–45). Association for Computational Linguistics.

[15] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Advances in Neural Information Processing Systems (Vol. 32). Curran Associates, Inc.

[16] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2020). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In International Conference on Learning Representations.

Plagiarism Report

Team Member Names:
1) Foram Rajgaur - 2023601007
2) Komal Sali - 2023601008

I hereby submit the plagiarism report for my research paper titled BERT-based Youtube Comments Sentiment Analysis. The checked plagiarism percentage is 12%, verified using PapersOwl.