Emotion Detection from Text Using Deep Learning and Local Language Models

YITONG GU, UC San Diego, The USA

This project explores emotion detection from text using both traditional deep learning and local large language models (LLMs). I first replicate a CNN-LSTM model based on prior research for classifying emotions in short texts, achieving accuracy comparable to the original study. To extend this work, I evaluate the zero-shot emotion classification capability of local LLMs running on Ollama, such as Mistral. By prompting the LLM to classify emotions from the same dataset, I compare its predictions against ground truth labels and calculate both overall and per-class accuracy. My findings highlight the strengths and limitations of LLMs in understanding emotional nuance without fine-tuning. The full codebase is available at https://github.com/YitongGu/Emotion-Detection-in-Text.

CCS Concepts: • **Computing methodologies → Natural language processing**; *Supervised learning by classification*; Neural networks.

Additional Key Words and Phrases: Emotion detection, Text classification, Deep learning, LLMs, NLP

ACM Reference Format:
Yitong Gu. 2025. Emotion Detection from Text Using Deep Learning and Local Language Models. 1, 1 (June 2025), 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Motivation

1.1 General Background

Emotion detection in text is a growing area of interest within natural language processing (NLP), as emotions play a central role in human communication, influencing decision-making, perception, and interaction. Accurately identifying emotions expressed in text is essential for applications ranging from sentiment analysis and mental health monitoring to customer feedback interpretation and empathetic human-computer interaction [Kusal et al. 2022; Plaza-del Arco et al. 2024].

Recent surveys highlight both the importance and complexity of modeling emotions in language. Plaza-del-Arco et al. [Plaza-del Arco et al. 2024] emphasize the growing interest in emotion analysis, while noting challenges such as inconsistent labeling schemes and the need for interdisciplinary perspectives. Similarly, Kusal et al. [Kusal et al. 2022] review text-based emotion detection techniques and outline practical challenges in achieving reliable emotion classification.

Despite improvements in deep learning and pre-trained language models, emotion detection remains difficult due to the subtlety of emotional cues, cultural variability, and ambiguity in expressions. This motivates the exploration of both deep learning architectures and large language models (LLMs) for this task.

1.2 Private and localized approach

In this project, we replicate a CNN-LSTM-based deep learning approach and compare it to local LLM-based emotion classification using prompt-based inference with models like Mistral deployed through Ollama. This comparison aims to assess the tradeoffs in accuracy, flexibility, feasibility and runtime of emotion detection models in resource-constrained environments.

As emotion-aware systems become increasingly integrated into everyday applications—particularly those dealing with sensitive user input such as mental health, personal relationships, or internal reflections—privacy becomes a fundamental design consideration. Many users are justifiably hesitant to share emotionally charged or vulnerable text with remote servers or cloud-hosted models. In such contexts, sending user data to external APIs for inference not only raises ethical questions, but may also violate privacy regulations or user expectations.

To address this, localized deployment of language models offers a compelling solution. Running large language models (LLMs) entirely on-device allows emotional detection and dialogue generation to occur without transmitting data off the user's machine. This architecture preserves user confidentiality and empowers applications to offer emotion-aware functionality without compromising trust.

Recent advancements in efficient model architectures and optimization frameworks have made it feasible to run models like Mistral, LLaMA 2, or TinyLLaMA locally through tools such as Ollama. These models can perform zero-shot emotion classification and generate conversational responses in real time, entirely offline. By combining emotional understanding with local execution, such systems enable a new generation of private, context-aware AI companions.

This private and localized approach is especially well-suited for scenarios involving journaling, therapeutic chatbots, and wellness assistants, where emotional expression is both rich and deeply personal. By ensuring that emotional analysis is performed locally, developers can build systems that are not only intelligent but also respectful of user boundaries and data sovereignty.

2 Related Work

2.1 Machine Learning based emotion detection

2.1.1 A survey of Emotion Detection. This survey offers an in-depth analysis of various approaches to emotion recognition in text, specifically evaluating their effectiveness on established datasets such as the International Survey on Emotion Antecedents and Reactions (ISEAR) and EmoBank. The findings indicate that deep learning models significantly outperform traditional statistical methods and lexicon-based approaches, particularly when it comes to identifying

Author's Contact Information: Yitong Gu, UC San Diego, La Jolla, The USA, yig048@ucsd.edu.

ACM XXXX-XXXX/2025/6-ART
https://doi.org/10.1145/nnnnnnn.nnnnnnn

complex and nuanced emotional categories. Despite this clear advantage, it's important to note that the survey exclusively examines methodologies developed prior to 2020. Consequently, while the results suggest that deep learning techniques hold substantial promise for enhancing emotion recognition tasks, they also highlight a gap in the exploration of more recent advancements that could further improve accuracy and understanding in this field. [Patel 2025].

2.1.2 Exploring Text-Based Emotion Recognition Using Machine Learning. This study utilizes various classifiers, including Random Forest, Support Vector Machine (SVM), and Logistic Regression, to analyze emotional sentiments expressed in social media conversations. The research focuses on sentiments captured from Twitter, where joy and sadness were identified with the highest accuracy, suggesting that these emotions are more readily expressed and detectable in online discourse. In contrast, fear and surprise proved to be more challenging to classify accurately, potentially due to their nuanced expressions and context-dependent interpretations. Among the classifiers used, the Random Forest model demonstrated the highest overall accuracy, indicating its effectiveness in processing and categorizing the diverse emotional tones present in social media interactions. The SVM model followed closely behind, showcasing solid performance as well. This research highlights the potential of machine learning techniques in sentiment analysis within the realm of social media, paving the way for further exploration in this area [Plaza-del Arco et al. 2024].

2.1.3 Text-Based Emotion Recognition Using Deep Learning Approach. Integrating deep neural networks with traditional machine learning classifiers significantly enhances the accuracy of emotion classification tasks. This hybrid approach leverages the strengths of advanced neural architectures to capture complex patterns in data and harnesses the interpretability and efficiency of classical models. As a result, it demonstrates impressive robustness against noisy input, effectively maintaining performance where other models might falter. Notably, this methodology outperforms baseline models, highlighting its potential for real-world applications in emotion recognition. The findings are based on comprehensive experiments conducted on the EmoBank dataset, a rich resource that captures a diverse range of emotional expressions, further validating the effectiveness of this combined strategy. [Bharti et al. [n. d.]].

2.1.4 EmoTxt: A Toolkit for Emotion Recognition from Text. EmoTxt is an open-source framework designed for training and evaluating emotion classification models using textual data from software engineering. It employs machine learning techniques like Support Vector Machines (SVM) and Logistic Regression to identify and categorize emotions in software development documents, such as code comments and bug reports. EmoTxt enhances understanding of emotional dynamics in software teams, fostering better communication and collaboration. Its open-source nature invites community contributions for further research in sentiment analysis and emotion detection in technical environments [Calefato et al. 2018].

2.2 NLP methods within emotion detection

2.2.1 Emotional Detection from Text Using NLP with Prediction Probability. This study introduces a novel method that integrates machine learning, sentiment analysis, and probabilistic models to predict the probability of specific emotional states in textual data. The approach is particularly beneficial for applications in mental health monitoring and consumer feedback analysis, offering nuanced insights into subtle emotional expressions [Sr 2024].

2.2.2 Emotion Analysis in NLP: Trends, Gaps and Roadmap for Future Directions. This comprehensive survey examines 154 NLP publications over the past decade, highlighting the evolution of emotion analysis tasks, prevalent emotion frameworks, and the significance of demographic and cultural factors. The study identifies key challenges, including the lack of standardized terminology and the need for interdisciplinary research to enhance emotion modeling in NLP [del Arco et al. 2024].

2.3 LLM-based methods within emotion detection

2.3.1 GPT-4V with Emotion: A Zero-Shot Benchmark for Generalized Emotion Recognition. This study presents a comprehensive evaluation of GPT-4V's ability to perform zero-shot emotion recognition across 21 benchmark datasets, covering a wide range of tasks including visual sentiment analysis, facial expression recognition, and multimodal emotion classification. Without any task-specific fine-tuning, GPT-4V achieved an average accuracy of 66.3% across all datasets, significantly outperforming other vision-language models in zero-shot settings. Notably, on the ArtEmis dataset—which requires understanding nuanced affective content in visual art—GPT-4V reached 72.9% accuracy, and on the EMOTIC dataset, it achieved 75.1%, reflecting strong alignment between visual cues and emotion inference. The model also demonstrated competitive performance in text-prompted tasks using only visual input, highlighting its robust generalization across modalities. These results suggest that large vision-language models like GPT-4V are capable of capturing emotional context effectively without the need for fine-tuning, marking a significant advancement in zero-shot emotion understanding [Lian et al. 2024].

2.4 Conclusion

The field of emotion detection has undergone significant evolution, progressing from traditional machine learning methods to deep learning and, more recently, to large-scale language models (LLMs). While classical classifiers such as Random Forests, SVMs, and Logistic Regression have proven effective in capturing explicit emotional patterns in structured domains like social media, deep learning models have demonstrated superior capabilities in handling subtle, context-dependent expressions. Building on this progress, large-scale LLMs now represent a substantial leap forward, offering powerful generalization through zero-shot and few-shot inference without the need for extensive task-specific training. Recent studies have shown that instruction-tuned or even foundation LLMs can match or outperform earlier models on a variety of emotion detection benchmarks, especially in complex, multimodal, or low-resource scenarios. This trajectory—from DL to LLMs—underscores a broader shift in the field toward leveraging pretrained, scalable architectures, and opens promising new directions for robust and adaptable emotion recognition systems across diverse domains.

3 Project Aim

3.1 Replicating Existing Paper

One of the aims of this project is to replicate and evaluate the methodology presented in the paper "Detection of Emotion by Text Analysis Using Machine Learning" [Machová et al. 2023], which introduces a deep learning-based approach to textual emotion classification. It presents a hybrid deep learning model combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to classify emotions in text. The authors focus on six basic emotions—joy, sadness, fear, anger, love, and surprise—and train their model on a labeled dataset derived from social media posts and dialogue. Their approach demonstrates strong performance, indicating the model's ability to effectively capture both local word patterns and sequential emotional context. This study serves as a valuable baseline for emotion detection and provides insights into the integration of neural architectures for affective computing.

Specifically, the project seeks to reproduce the reported performance of the CNN-LSTM model on a labeled emotion dataset and assess its effectiveness across multiple emotion categories. One of the key reasons for selecting this study is that the proposed architecture is lightweight and computationally efficient, making it well-suited for local inference on resource-constrained systems. By replicating this model, we aim to validate the reproducibility of the original findings, gain deeper insights into the design and behavior of hybrid neural architectures for emotion detection, and establish a reliable performance baseline. This will also enable a meaningful comparison with more recent methods, including local large language model (LLM)-based approaches, to evaluate trade-offs between speed, accuracy, and deployment feasibility in real-world applications.

3.2 Exploring existing LLMs

This section of the project delves into the feasibility of utilizing existing large language models (LLMs) for the purpose of emotion classification tasks, with a specific focus on eliminating the need for additional fine-tuning or reliance on external API-based services. The primary objective is to assess the performance of pre-trained, locally hosted LLMs in accurately detecting emotions when prompted with tailored input.

The results obtained from these models will then be rigorously evaluated against traditional machine learning techniques and advanced deep learning methods. By conducting this comparison, we aim to highlight the advantages and limitations of employing local LLMs for emotion detection, considering factors such as accuracy, processing time, and resource efficiency. This investigation seeks to contribute valuable insights into the potential of LLMs in the realm of emotion analysis while also providing a benchmark for future research and applications in this evolving field.

3.2.1 Reducing dependencies on training data. Traditional supervised emotion classifiers typically rely on extensive, labeled datasets, which necessitate significant time and resources for data collection and model training. This process not only requires expertise in emotion classification but also demands ongoing maintenance and updating of the model as new data becomes available. In contrast, large language models (LLMs)—particularly those optimized for instruction-following—present a promising alternative. These models have the potential to generalize across a variety of tasks with minimal input, thanks to advanced prompt engineering techniques.

The primary objective of this project is to evaluate the zero-shot learning capabilities of LLMs in the context of emotion classification. By leveraging these capabilities, we aim to minimize the dependence on specialized, domain-specific labeled datasets and labor-intensive training pipelines. The outcomes of this research could pave the way for more efficient, scalable approaches to emotion recognition, ultimately enhancing the integration of emotion-aware systems in various applications.

3.2.2 Reducing dependencies on API based inference. While cloud-based large language model (LLM) APIs, such as those offered by OpenAI and Anthropic, demonstrate remarkable capabilities, they come with significant challenges. These include ongoing subscription costs, potential data privacy issues, and a reliance on stable internet connectivity, which can hinder their usability in certain contexts.

In response to these challenges, this project focuses on the feasibility of running open-weight LLMs locally, utilizing frameworks like Ollama, which allows for efficient and secure inference without the constraints of cloud services. Our primary objective is to evaluate whether these locally deployed models can match or even surpass the performance of their cloud counterparts in tasks specifically related to emotion detection.

By maintaining full offline functionality and ensuring user control over the data processed, we aim to explore the advantages of local models in terms of privacy, cost-effectiveness, and accessibility. Ultimately, this project seeks to illuminate the potential of open-weight LLMs as a viable alternative for applications that require sensitivity to both emotional nuances and user data security.

4 Methodology

4.1 Dataset

Since the work we are to replicate is trained on a specific dataset, we are using this dataset for evaluations for both the CNN-LSTM model and LLM methods. The model is trained and evaluated on a publicly available labeled dataset introduced in the original study, consisting of 34792 text samples. Each sample is annotated with one of the emotion categories. The detailed distribution is attached in Table 1.

The texts utilized in this dataset are sourced from a wide array of online platforms, including social media posts, dialogue transcripts, and various forms of user-generated content. This ensures a rich diversity in language usage, styles of communication, and emotional expressions, reflecting the varied ways individuals convey their feelings and thoughts in digital spaces.

To prepare this corpus for deep learning models, a thorough preprocessing pipeline is employed. This includes tokenization, which breaks down the text into manageable pieces or tokens; stop-word removal, which eliminates common words that may not contribute

Table 1. Distribution of Emotion Labels in the Dataset

Emotion	Count
Joy	11,045
Sadness	6,722
Fear	5,410
Anger	4,297
Surprise	4,062
Neutral	2,254
Disgust	856
Shame	146
Total	**34,792**

significantly to the overall meaning; and sequence padding, which standardizes the lengths of the text sequences to facilitate efficient processing of data.

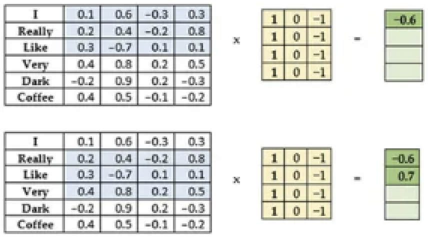
The resulting dataset serves as a balanced and comprehensive benchmark, enabling the evaluation and comparison of both traditional machine learning classifiers and advanced neural network-based emotion classifiers. By offering insights into the effectiveness and accuracy of different modeling approaches, this corpus plays a crucial role in advancing research and development in emotion recognition technology.

As shown in Table 1, the dataset exhibits a notable class imbalance, with certain emotions such as joy and sadness being heavily overrepresented, while others like shame and disgust occur far less frequently. This uneven distribution poses a risk of model bias during training, where the classifier may disproportionately favor majority classes while underperforming on minority emotions. Such imbalance can negatively affect the model's generalization and fairness, particularly in applications requiring sensitivity to less commonly expressed emotional states. Therefore, addressing this issue—through techniques such as data augmentation, class weighting, or resampling—may be necessary for future work to ensure balanced and robust performance.

4.2 Replicating CNN-LSTM

The model implemented in the original study combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks into a hybrid architecture known as CNN-LSTM. In this structure, CNN layers are first used to extract local spatial patterns from sequences of word embeddings, capturing short-term dependencies and phrase-level features. These representations are then passed into LSTM layers, which model the temporal dynamics and contextual relationships across the entire input sequence. This hybrid approach leverages the strengths of both architectures—CNN for efficient feature extraction and LSTM for sequence modeling—making it well-suited for emotion detection tasks in text.

Convolutional Neural Networks (CNNs) are a class of deep learning models that apply convolution operations using learnable filters to extract local patterns from input data. Originally developed for image processing, CNNs have been adapted for text analysis using 1D convolutions, where filters slide over word embeddings to detect features such as n-gram patterns. As shown in Figure 1, these filters operate along the sequence dimension while spanning the entire embedding space, enabling the model to capture local dependencies in textual input. Prior to processing, input text is transformed into vector representations, allowing numerical operations suitable for neural computation.



I	0.1	0.6	−0.3	0.3
Really	0.2	0.4	−0.2	0.8
Like	0.3	−0.7	0.1	0.1
Very	0.4	0.8	0.2	0.5
Dark	−0.2	0.9	0.2	−0.3
Coffee	0.4	0.5	−0.1	−0.2

x

1	0	−1
1	0	−1
1	0	−1
1	0	−1

=

−0.6

I	0.1	0.6	−0.3	0.3
Really	0.2	0.4	−0.2	0.8
Like	0.3	−0.7	0.1	0.1
Very	0.4	0.8	0.2	0.5
Dark	−0.2	0.9	0.2	−0.3
Coffee	0.4	0.5	−0.1	−0.2

x

1	0	−1
1	0	−1
1	0	−1
1	0	−1

=

−0.6
0.7

Fig. 1. CNN for text processing [Machová et al. 2023]

Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, have proven highly effective in emotion analysis tasks due to their ability to model sequential dependencies in text. Unlike traditional machine learning methods that treat text as unordered "bags of words," LSTMs process input as a sequence, preserving contextual relationships between words. LSTM networks address the vanishing gradient problem common in standard RNNs by incorporating gated memory cells that retain relevant information over longer sequences. As shown in Figure 2, the architecture consists of repeating LSTM blocks that regulate information flow through input, forget, and output gates. This design enables the model to capture long-range dependencies, making it suitable for emotion detection in complex textual inputs. Additionally, the attention mechanism can be integrated to further enhance the model's ability to focus on emotionally salient parts of longer sentences.

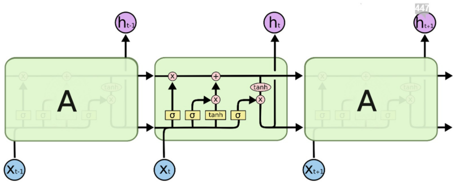


Fig. 2. LSTM structure [Machová et al. 2023]

4.2.1 With & Without data cleaning. Text cleaning is a crucial preprocessing step in natural language processing (NLP) that transforms raw textual input into a cleaner and more structured format suitable for analysis by machine learning models. As illustrated in Table 2, the original text samples often contain noise such as usernames (e.g., "@Iluvmiassantos"), excessive punctuation, irregular capitalization, or irrelevant characters. These elements may introduce bias or reduce the model's ability to generalize. The cleaning process typically includes removing mentions and URLs, converting text to lowercase, eliminating unnecessary punctuation, and optionally applying stemming or lemmatization. The primary purpose of text cleaning is to standardize the input, reduce sparsity in textual features, and retain only semantically meaningful content that contributes to accurate emotion classification. This step is particularly important for deep learning models, which are sensitive to inconsistencies in input representation.

Table 2. Sample Entries from the Emotion Dataset

Emotion	Text	Cleaned Text
Neutral	Why ?	?
Joy	Sage Act upgrade on my to do list for tomorrow.	Sage Act upgrade list tomorrow.
Sadness	ON THE WAY TO MY HOME-GIRL BABY FUNERAL!!! MAN …	WAY HOME-GIRL BABY FUNERAL!!! MAN HATE FUNERALS…
Joy	Such an eye! The true hazel eye–and so brilliant…	eye! true hazel eye-and brilliant! Regular f…
Joy	`@Iluvmiassantos` ugh babe.. huggzzzz for u! b…	ugh babe.. huggzzzz u! babe naamazed nga ako…

4.3 Exploring Existing LLMs

This section investigates the feasibility of using pre-trained large language models (LLMs) for emotion detection via prompt-based inference. Unlike traditional supervised approaches, this method aims to reduce dependencies on labeled training data and cloud-based inference APIs by leveraging zero-shot capabilities of local LLMs.

4.3.1 Encoder-only Models. Encoder-only models such as DistilBERT are designed to understand and represent textual input with high efficiency. In this project, we utilize Hugging Face's `pipeline` interface to run `distilbert-base-uncased` for zero-shot classification, supplying the model with the full emotion label set: *joy, sadness, anger, fear, surprise, disgust, shame, neutral.* The model embeds the input and compares it with candidate labels using internal classification heads trained for natural language inference (NLI). While these models are generally smaller and faster to execute, they lack generative capabilities and are constrained to fixed output schemas.

4.3.2 Decoder-only Models. Decoder-only models like LLaMA2, Mistral, and Phi-2 operate autoregressively, predicting tokens based on previously seen context. These models are queried using carefully constructed prompts that ask the model to classify the emotion of a given text, with the constraint to respond using only one label from a predefined set. Since they do not require fine-tuning and can run locally using frameworks like Ollama, they present a compelling alternative for offline inference. The flexibility of their language generation also makes them adaptable to more nuanced or context-sensitive emotion detection tasks.

4.3.3 LLM Inference Workflow. Encoder-only and Decoder-only models are used to

Prompt Construction: For each sample, a prompt is generated in the format: `"Classify the emotion in the following sentence: ’...’. Choose from: [...]. Answer in one word."`

Model Execution:

- Encoder-only models use zero-shot classification via Hugging Face.
- Decoder-only models are executed locally using Ollama, querying each sentence and parsing the response.

Evaluation: The predicted labels are compared against the ground-truth labels from the dataset to calculate overall accuracy and per-emotion performance.

Table 3. All Models Evaluated for Emotion Detection

Model Name	Architecture	Size (B)
facebook/bart-large-mnli	Encoder-Decoder	406M
mistral	Decoder-only	7B
tinyllama	Decoder-only	1.1B
sentence-transformers/all-MiniLM-L6-v2	Encoder-only	22M
sentence-transformers/all-MiniLM-L12-v2	Encoder-only	33M
sentence-transformers/all-mpnet-base-v2	Encoder-only	110M
sentence-transformers/gtr-t5-large	Encoder-Decoder	1.2B
sentence-transformers/gtr-t5-xl	Encoder-Decoder	3.7B

5 Results

5.1 Replicating CNN-LSTM

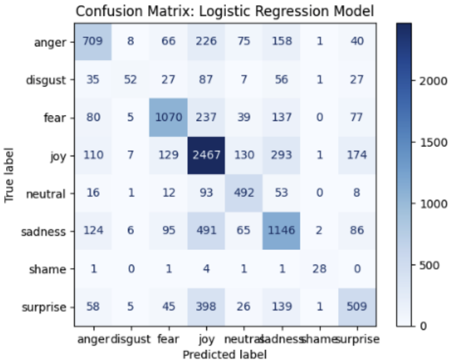
5.1.1 With data cleaning. After applying text cleaning, the logistic regression model achieved an overall accuracy of 62% on a test set of 10,438 samples. As shown in Table 4, the model performed best on well-represented classes such as joy (F1 = 0.67), fear (F1 = 0.69), and neutral (F1 = 0.65), while struggling with minority classes like disgust (F1 = 0.28) and surprise (F1 = 0.48). Notably, shame, despite having only 36 samples, was classified with a relatively high F1-score of 0.80, likely due to its distinct linguistic patterns.

Table 4. Classification Report for Precision, Recall, and F1-Score per Emotion Category with data cleaning

Emotion	Precision	Recall	F1-Score	Accuracy
Anger	0.63	0.55	0.59	0.55
Disgust	0.62	0.18	0.28	0.18
Fear	0.74	0.65	0.69	0.65
Joy	0.62	0.75	0.67	0.75
Neutral	0.59	0.73	0.65	0.73
Sadness	0.58	0.57	0.57	0.57
Shame	0.82	0.78	0.80	0.78
Surprise	0.55	0.43	0.48	0.43
Accuracy	0.62 (overall on 10,438 samples)			
Macro Avg	0.64	0.58	0.59	
Weighted Avg	0.62	0.62	0.61	

The confusion matrix (Figure 3) indicates that most misclassifications occurred between emotionally similar classes, such as sadness being confused with neutral, and anger overlapping with fear and sadness. The macro-averaged F1-score of 0.59 reflects moderate balance across all emotion classes, while the weighted average F1-score of 0.61 suggests slightly better overall performance due to the dominance of majority classes.

These results highlight that data cleaning improves consistency and classification accuracy, though further improvement may require techniques such as class balancing or more expressive models.



	anger	disgust	fear	joy	neutral	sadness	shame	surprise
anger	709	8	66	226	75	158	1	40
disgust	35	52	27	87	7	56	1	27
fear	80	5	1070	237	39	137	0	77
joy	110	7	129	2467	130	293	1	174
neutral	16	1	12	93	492	53	0	8
sadness	124	6	95	491	65	1146	2	86
shame	1	0	1	4	1	1	28	0
surprise	58	5	45	398	26	139	1	509

Fig. 3. Confusion Matrix with data cleaning

5.1.2 *Without data cleaning.*

When the same logistic regression model was evaluated without applying text cleaning, it achieved a slightly higher overall accuracy of 63% (Table 5), similar to the cleaned version. The performance metrics remained largely consistent across major emotion categories, with joy (F1 = 0.70), fear (F1 = 0.70), and neutral (F1 = 0.68) showing the strongest results.

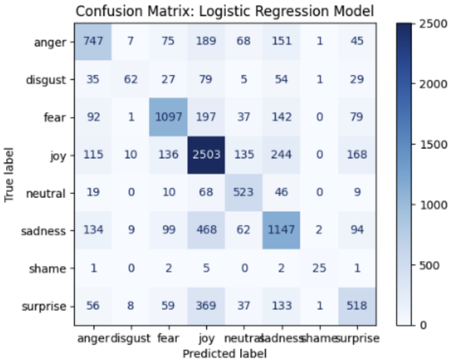
Minor classes such as disgust and surprise remained challenging, with F1-scores of 0.32 and 0.49, respectively—nearly identical to the cleaned setup. Interestingly, shame, again a very low-frequency class, was predicted with high F1-score (0.76), though slightly lower than with cleaning.

Table 5. Classification Report for Precision, Recall, and F1-Score per Emotion Category without data cleaning

Emotion	Precision	Recall	F1-Score	Accuracy
Anger	0.62	0.58	0.60	0.58
Disgust	0.64	0.21	0.32	0.21
Fear	0.73	0.67	0.70	0.67
Joy	0.65	0.76	0.70	0.76
Neutral	0.60	0.77	0.68	0.77
Sadness	0.60	0.57	0.58	0.57
Shame	0.83	0.69	0.76	0.69
Surprise	0.55	0.44	0.49	0.44
Accuracy	0.63 (overall on 10,438 samples)			
Macro Avg	0.65	0.59	0.60	
Weighted Avg	0.63	0.63	0.63	

The confusion matrix (Figure 4) reflects similar misclassification patterns as before, with overlapping between sadness, neutral, and anger categories. Overall, the benefit of data cleaning appears marginal in this case for logistic regression, with only minor changes in macro and weighted averages.

These results suggest that while data cleaning improves consistency and interpretability of input, its impact on model performance for linear classifiers may be limited—particularly when the model already has sufficient robustness to textual noise. Further experiments with more complex models may better reflect the value of preprocessing.



	anger	disgust	fear	joy	neutral	sadness	shame	surprise
anger	747	7	75	189	68	151	1	45
disgust	35	62	27	79	5	54	1	29
fear	92	1	1097	197	37	142	0	79
joy	115	10	136	2503	135	244	0	168
neutral	19	0	10	68	523	46	0	9
sadness	134	9	99	468	62	1147	2	94
shame	1	0	2	5	0	2	25	1
surprise	56	8	59	369	37	133	1	518

Fig. 4. Confusion Matrix without data cleaning

5.2 Exploring existing LLMs

5.2.1 *Decoder-only Models.*

The comparison between Mistral and TinyLLaMA, shown in Table 7, reveals clear differences in their ability to perform zero-shot emotion detection. Mistral, being a larger model, demonstrates strong and consistent performance across a wide range of emotions, particularly in categories that require nuanced contextual understanding. TinyLLaMA, on the other hand, struggles to generalize effectively, showing limited capability in distinguishing between emotion classes. While smaller models offer advantages in terms of efficiency and resource usage, these results suggest that they may lack the representational power needed for subtle language tasks like emotion classification. Overall, the analysis highlights the importance of model scale in achieving reliable performance in zero-shot NLP applications.

Table 7. Accuracy per Emotion: Mistral vs. TinyLLaMA

Emotion	Mistral (7B)	TinyLLaMA (1.1B)
Anger	0.63 (2722/4297)	0.01 (42/4297)
Disgust	0.23 (201/856)	0.28 (236/856)
Fear	0.32 (1709/5410)	0.21 (1145/5410)
Joy	0.52 (5749/11045)	0.20 (2238/11045)
Neutral	0.85 (1912/2254)	0.13 (295/2254)
Sadness	0.55 (3712/6722)	0.22 (1478/6722)
Shame	0.75 (110/146)	0.09 (13/146)
Surprise	0.12 (470/4062)	0.02 (62/4062)
Overall Accuracy	**0.62**	**0.16**
Model Size	7B	1.1B

Table 6. Accuracy per Emotion for Encoder-only Models

Emotion	MiniLM-L6 (22M)	MiniLM-L12 (33M)	MPNet (110M)	T5-Large (770M)	T5-XL (3B)
Anger	0.32	0.39	0.50	0.31	0.34
Disgust	0.33	0.35	0.38	0.31	0.38
Fear	0.51	0.50	0.43	0.34	0.32
Joy	0.40	0.44	0.45	0.31	0.27
Neutral	0.15	0.04	0.08	0.21	0.27
Sadness	0.43	0.55	0.50	0.34	0.33
Shame	0.32	0.32	0.23	0.73	0.52
Surprise	0.22	0.21	0.46	0.51	0.58
Overall Accuracy	**0.38**	**0.41**	**0.44**	**0.34**	**0.34**
Model Size	22M	33M	110M	770M	3B

However, the CNN-LSTM model significantly outperforms both Mistral and TinyLlama across almost all emotion categories. It demonstrates strong balanced accuracy, with particularly robust performance on high-frequency emotions such as joy, sadness, and fear. Its ability to capture nuanced patterns in training data enables it to maintain stable precision and recall even for more difficult classes like disgust and surprise.

In contrast, Mistral—despite being a decoder-only LLM with a 7B parameter size—achieves decent performance on neutral, anger, and shame, but its performance drops considerably for surprise and disgust. TinyLlama, a significantly smaller 1.1B model, shows consistent underperformance across nearly all emotions, with particularly poor results for anger, shame, and surprise. Its overall accuracy is noticeably lower, suggesting that model size and capacity have a direct impact on classification effectiveness in zero-shot scenarios.

These results illustrate that while LLMs can generalize without fine-tuning, their effectiveness is still substantially below that of task-specific deep learning models when applied directly to complex emotion classification. This highlights the trade-off between flexibility and task performance, and motivates future work on prompt engineering, few-shot learning, or domain-specific adaptation for LLMs.

5.2.2 Encoder-only Models. Encoder-only models, shown in Table 6 such as MiniLM and MPNet, are often regarded as well-suited for understanding sentence-level semantics due to their strong contextual encoding capabilities. However, in this evaluation, they generally underperform compared to both traditional deep learning models and larger decoder-only language models in the task of zero-shot emotion detection.

Despite their theoretical advantage in text understanding, their practical performance lags behind, especially in recognizing nuanced or less frequent emotions. This discrepancy may be attributed to their relatively small model sizes, which limit their capacity to generalize in the absence of fine-tuning. While encoder-only models remain efficient and effective for certain semantic tasks, their limitations become apparent in complex affective classification scenarios under zero-shot conditions.

6 Discussion

6.1 Pros and Cons for CNN-LSTM Methods

The CNN-LSTM model has several clear strengths when it comes to emotion detection. It performs well overall, especially when trained on a good dataset, and it runs quickly, which makes it useful for real-time applications.

However, the model depends heavily on the quality of the training data. In our testing, we noticed that it sometimes made incorrect predictions because some of the labels in the dataset seemed wrong or unclear. This shows that while CNN-LSTM can be powerful, it doesn't handle noisy or imperfect data very well, and its results can suffer when the training examples are misleading.

6.2 Pros and Cons for LLM

Compared to traditional deep learning models such as the CNN-LSTM architecture, large language models (LLMs) introduce a new paradigm in emotion detection by enabling zero-shot classification through prompt-based inference. While CNN-LSTM models require supervised training and careful preprocessing, they are optimized specifically for the task and often perform reliably on structured datasets.

In contrast, LLMs offer greater flexibility and do not require retraining for new tasks, but their performance can vary depending on prompt design, model scale, and contextual sensitivity. This trade-off illustrates a shift from task-specific modeling toward more generalized, reusable architectures. The comparison highlights that while deep learning methods provide targeted accuracy and efficiency, LLMs—especially larger ones—offer adaptability and scalability for broader NLP applications.

6.3 Future work

Although our current results did not show a clear performance advantage for small-scale large language models (LLMs) in emotion detection tasks, their potential should not be overlooked. LLMs are designed to understand and generate human-like text, and with further adaptation, they may excel in recognizing emotional cues. This is particularly important for applications such as emotionally intelligent chatbots, where detecting the user's emotional state is essential for producing empathetic and contextually appropriate responses. Future work could explore fine-tuning LLMs on emotion-specific datasets, integrating prompt engineering techniques, or designing hybrid systems that combine the interpretability of traditional models with the generative power of LLMs.

7 Conclusion

In this project, we explored the task of emotion detection from text using both traditional deep learning methods and large language models (LLMs). We began by replicating a CNN-LSTM model from existing literature, which demonstrated strong performance across most emotion categories, particularly when supported by high-quality, balanced training data. The model's fast inference time and effectiveness made it a solid baseline for comparison.

We then evaluated a variety of LLMs—including encoder-only models (e.g., MiniLM, MPNet) and decoder-only models (e.g., Mistral, TinyLLaMA)—under zero-shot settings. While LLMs offer the advantage of flexibility and do not require task-specific training, we found that small-scale LLMs generally underperformed compared to the CNN-LSTM model. Decoder-only models like Mistral showed more promise than encoder-only models, but none of the LLMs evaluated matched the accuracy or robustness of the supervised deep learning approach.

The main takeaway is that while LLMs are not yet outperforming task-specific models in this domain—especially when deployed in a zero-shot or lightly supervised fashion—they hold substantial potential for future development. Emotionally intelligent applications, such as chatbots, may benefit from integrating LLMs with stronger emotional understanding. Future work should explore fine-tuning LLMs on emotion-specific datasets, as well as hybrid architectures that combine the interpretability of traditional models with the generative strengths of LLMs [OpenAI 2025].

References

Santosh Kumar Bharti, S Varadhaganapathy, Rajeev Kumar Gupta, Prashant Kumar Shukla, Mohamed Bouye, Simon Karanja Hingaa, and Amena Mahmoud. [n. d.]. Text-Based Emotion Recognition Using Deep Learning Approach. *Computational Intelligence and Neuroscience* 2022, 1 ([n. d.]), 2645381. doi:10.1155/2022/2645381 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1155/2022/2645381

Fabio Calefato, Filippo Lanubile, and Nicole Novielli. 2018. EmoTxt: A Toolkit for Emotion Recognition from Text. arXiv:1708.03892 [cs.HC] https://arxiv.org/abs/1708.03892

Flor Miriam Plaza del Arco, Alba Curry, Amanda Cercas Curry, and Dirk Hovy. 2024. Emotion Analysis in NLP: Trends, Gaps and Roadmap for Future Directions. arXiv:2403.01222 [cs.CL] https://arxiv.org/abs/2403.01222

Nisal Kusal, Ruwan Wickramarachchi, and Chamath Keppitiyagama. 2022. Emotion Detection in Text: A Review. *arXiv preprint arXiv:2205.03235* (2022). https://arxiv.org/abs/2205.03235

Zheng Lian, Licai Sun, Haiyang Sun, Kang Chen, Zhuofan Wen, Hao Gu, Bin Liu, and Jianhua Tao. 2024. GPT-4V with Emotion: A Zero-shot Benchmark for Generalized Emotion Recognition. arXiv:2312.04293 [cs.CV] https://arxiv.org/abs/2312.04293

Kristína Machová, Martina Szabóova, Ján Paralič, and Ján Mičko. 2023. Detection of emotion by text analysis using machine learning. *Frontiers in Psychology* Volume 14 - 2023 (2023). doi:10.3389/fpsyg.2023.1190326

OpenAI. 2025. ChatGPT (June 2025 Version). https://chat.openai.com. Large language model used for assistance.

Romal Bharatkumar Patel. 2025. A survey of Emotion Detection. (March 2025). doi:10.36227/techrxiv.174285087.74645354/v1

Francisco Plaza-del Arco, Helena Gómez-Adorno, Francisco Rangel, and Paolo Rosso. 2024. Emotion Analysis in Natural Language Processing: A Multidisciplinary Review. *arXiv preprint arXiv:2403.01222* (2024). https://arxiv.org/abs/2403.01222

Ragunathan Sr. 2024. Emotional Detection from Text Using NLP (Natural Language Processing) With Prediction Probability.