CONTEXT-AWARE SENTIMENT ANALYSIS FOR NEURODIVERGENT DISCOURSE: COMPARING GPT-4 AND TRADITIONAL MODELS ON TWITTER

Annie Cui [1], Han Tun (Henry) Oo [2]

[1] Orange County School of the Arts, 1010 N Main St, Santa Ana, CA 92701
[2] California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

This research project investigates the effectiveness of sentiment analysis tools on tweets discussing neurodivergent individuals, particularly those with autism. Traditional models like TextBlob often lack the contextual awareness needed to interpret subtle or emotionally complex content. To address this, we developed a system comparing TextBlob and GPT-4 using both classification and regression-based evaluation [1]. A dataset of 100 tweets was analyzed. In the first experiment, GPT-4 achieved a macro F1-score of 0.61, outperforming TextBlob's 0.58, with both models reaching 62% accuracy. In the second experiment, which evaluated polarity scoring, GPT-4 achieved a MAE of 0.604, RMSE of 0.766, and a correlation of 0.479, compared to TextBlob's MAE of 0.650, RMSE of 0.778, and correlation of 0.394. These results confirm that GPT-4 provides more accurate and context-sensitive sentiment predictions [2]. This system improves upon prior lexicon-only approaches by combining classification and polarity scoring to offer a comprehensive, real-world analysis of sentiment in neurodivergent conversations.

KEYWORDS

Sentiment analysis, GPT-4, TextBlob, Autism, Neurodivergent discourse, Twitter data, Natural language processing, Polarity score, Classification metrics, Context-aware AI

1. INTRODUCTION

The problem we are trying to solve is the marginalization of neurodivergent groups online. The internet has significantly amplified the mistreatment of neurodivergent individuals [3]. Viral memes, reaction videos, and various TikToks or Instagram reels have repeatedly mocked traits commonly associated with autism or tics. Over 60% of autistic youth report being bullied, with many saying this bullying continues in online spaces. This is important because online spaces are becoming increasingly essential for communication, education, and socializing. However, for neurodivergent people (like those with autism, OCD, or Tourette's), these spaces can be hostile [4]. They're often laughed at, misunderstood, or excluded in comment sections and social media trends. In 2021, TikTok had to remove several hashtags mocking people with Tourette's, as faking tics and stimming became a harmful trend. This behavior damages self-esteem, mental health, and can limit access to support. For youth especially, digital rejection may lead to real-world consequences such as anxiety, depression, and disengagement in school or hobbies. Studies have shown that autistic individuals are four times more likely to attempt suicide than the general population, with cyberbullying being a major contributing factor. Many people still believe

David C. Wyld et al. (Eds): NLMLT, SEAU, ITCS, CST – 2025
pp. 57-67, 2025. CS & IT - CSCP 2025 DOI: 10.5121/csit.2025.151606

neurodivergent people are “faking it” for attention, or view accommodations or stimming as “weird” and “disruptive” due to ignorance or lack of understanding. Neurodivergent youth and adults face worsening mental health in the future and fewer online support communities [5]. Families may struggle with shame or hiding/disregarding diagnoses. Future workplaces lose out on diverse thinking and innovation if neurodiverse individuals are discouraged from joining conversations. As a whole, society risks becoming less inclusive and empathetic if online bullying continues unchecked.

Yazid et al. used a dashboard powered by lexicon-based sentiment analysis to track public discourse on autism. While visually informative, it lacked context awareness and did not explore model evaluation. Our system improves on this by evaluating accuracy and nuance through GPT-4 and TextBlob comparison. Venkit and Wilson showed that many models exhibit negative sentiment bias toward disability-related terms. While they identified bias, they did not evaluate model accuracy on real-world data [6]. Our project uses both classification and regression testing to examine model behavior and performance on real neurodivergent-related tweets. Malebary and Abulfaraj used a stacking ensemble to improve tweet sentiment classification. Their method combines lexicon and machine learning techniques but lacks modern language models. Our system includes GPT-4 and compares it to traditional tools using multiple evaluation metrics, offering deeper insights into sentiment accuracy and contextual sensitivity.

I propose the creation of a platform that will empower neurodivergent voices through education, advocacy, and storytelling. This community-driven platform will encourage online cultures to shift toward empathy and inclusion. This solution addresses the marginalization of neurodivergent individuals online by providing a safe space for them to share their stories, learn to self-advocate, and participate in public activities. The platform includes moderators to reduce bullying, educational resources to combat misconceptions, and campaigns to partner with online influencers and/or organizations to normalize neurodiversity. My solution tackles both the root of the problem (ignorance and misrepresentation) and its consequences (bullying and isolation). This platform aims to humanize neurodivergent individuals. Many current solutions rely solely on content moderation or reporting systems, which fail to detect subtle forms of ableism the majority of the time. Unlike those solutions, this method is proactive and centered on the neurodivergent community. It doesn’t just filter the bad, it builds an informed and inclusive culture.

Two experiments were conducted to evaluate the sentiment analysis performance of TextBlob and GPT-4. The first experiment compared the predicted sentiment labels (positive, neutral, negative) of both models against the ground truth. GPT-4 achieved higher macro-averaged precision, recall, and F1-score, indicating better classification performance, particularly for detecting negative and neutral tweets. The second experiment assessed numerical polarity predictions using Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Pearson correlation. GPT-4 again outperformed TextBlob, showing lower error rates and stronger correlation with human-labeled sentiment. These results confirmed that GPT-4's contextual understanding allows it to capture more subtle and accurate sentiment than rule-based tools. The experiments also demonstrated the value of using both classification and regression metrics when evaluating sentiment systems. GPT-4 consistently performed better across both tasks, reinforcing its potential for analyzing sensitive, emotion-rich content such as tweets about autism and neurodivergent communities.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Building Safe Peer Mentorship for Neurodivergent Communities

A major component of my program could be a peer mentorship system where neurodivergent individuals can support each other while navigating online spaces. Some potential issues I'd have to consider include ensuring that mentors are properly trained, preventing misinformation from spreading, and making sure the environment is supportive. Mentors may unintentionally give harmful advice, project their own experiences too heavily, or feel emotionally overwhelmed by their responsibility. Mentees, on the other hand, could feel invalidated or discouraged by this program if the dynamic feels unbalanced. To address this issue, I could use training modules (designed with the help of neurodivergent professionals, therapists, etc.) that cover boundaries, active listening, and trauma-informed support. I could also implement optional supervision, regular check-ins with feedback, and light moderation to make sure the environment remains healthy. Mentorship should uplift both parties, so safeguards would be essential.

2.2. Fostering Authenticity: A Trust-First Approach to Program Launch

Another major challenge I may face is a lack of engagement from neurodivergent individuals when I first launch my program. Many users from these communities have experienced being bullied, silenced, or misunderstood by certain programs claiming to help them. This can create hesitation, distrust, or emotional fatigue. Others may hesitate to participate in fear of being judged or retraumatized. To address this problem, I could find trusted ambassadors within the neurodivergent community to co-launch the platform and build credibility. This would make the program feel more authentic. I could also build a beta version of this platform which is feedback-driven with a smaller group of neurodivergent creators, allowing them to shape the experience and feel invested in the success of the platform.

2.3. Protecting Neurodivergent Spaces from Neurotypical Dominance

Lastly, there is a risk that neurotypical users may begin to dominate the platform in ways that shift the focus away from the inclusion of neurodivergent individuals. They may unintentionally derail conversations, give harmful advice, or turn the space into a performative allyship showcase used for self-promotion rather than a safe space. To combat this, I could set clear boundaries in place for all users, explaining the expectations of the community up front. I could also use tiered permissions where neurodivergent voices are prioritized, or create private spaces specifically for neurodivergent users to protect their agency and prioritize their comfort. Features such as neurodivergent-only forums or content filters could protect users from exhaustion and re-center the platform on its original purpose.

3. SOLUTION

Our program is designed to evaluate and compare sentiment analysis tools in detecting the sentiment of tweets related to neurodivergent individuals, particularly autism-related discourse. The three major components of this system are: data collection and cleaning, sentiment analysis models (GPT-4 and TextBlob), and evaluation with visualization [7].

The system flow begins with a dataset of 100 tweets sampled from a larger corpus on autism-related discussions. These tweets are preprocessed by removing usernames, hashtags, URLs, and extra whitespace to ensure cleaner inputs for analysis. After cleaning, each tweet is analyzed using two sentiment analysis tools: TextBlob and OpenAI's GPT-4.

TextBlob is a rule-based, lexicon-based sentiment tool, while GPT-4 uses generative AI and contextual understanding via a prompt instructing it to return both a polarity score (-1 to 1) and a label ("positive", "neutral", or "negative") [8]. These results are stored in a CSV file for analysis.

The final component of the system evaluates the outputs of each model against the labeled ground truth sentiment using both classification (accuracy, precision, recall, F1) and regression metrics (MAE, RMSE, correlation). Two experiments were conducted: one focusing on label agreement and the other on polarity score accuracy. Matplotlib was used to visualize the results, showing grouped bar charts and metric comparisons.

Overall, this system was implemented using Python, Pandas, Matplotlib, Scikit-learn, and OpenAI's GPT API, with TextBlob and GPT sentiment models as core components. This system gives insight into which model better interprets complex neurodivergent language and sentiment.

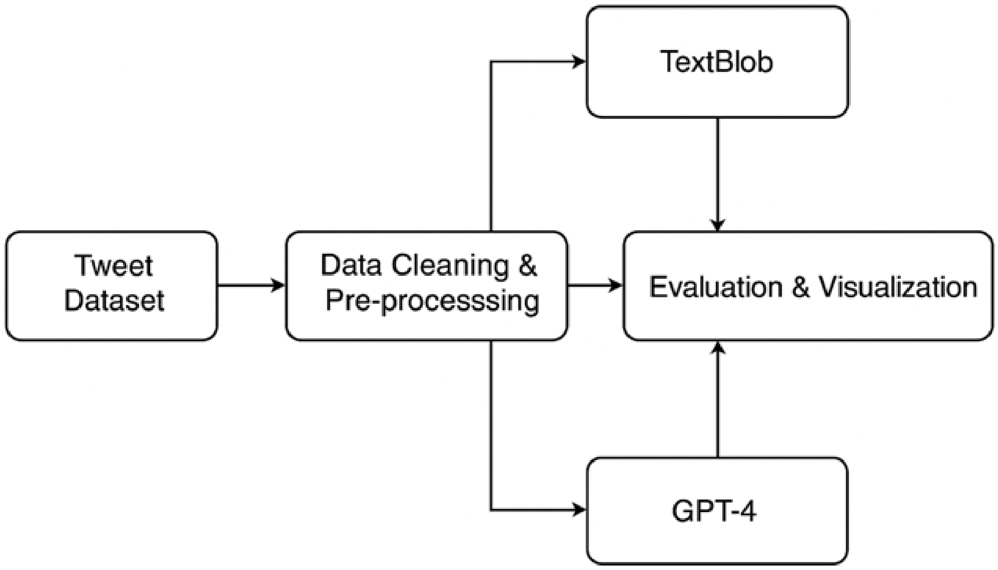

Figure 1. Overview of the solution

The first key component of the program is the sentiment classification module, which assigns each tweet a label: positive, neutral, or negative. This module is implemented using both TextBlob and GPT-4. TextBlob uses a rule-based, lexicon method that calculates polarity and classifies the text based on fixed thresholds. In contrast, GPT-4 relies on contextual language understanding through prompt-based input, where it interprets the tweet and returns both a sentiment label and a polarity score. GPT-4 is accessed through the OpenAI API [9]. This component allows a direct comparison between rule-based and generative language models in how they handle neurodivergent-related content on social media.

```python
def gpt_sentiment(text):
    prompt = (
        "Analyze the sentiment of this tweet. Return two values:\n"
        "1. A sentiment polarity score between -1 and 1.\n"
        "2. A sentiment label: Positive, Neutral, or Negative.\n"
        "Return the result as JSON with keys 'polarity' and 'label'.\n\n"
        f"Tweet: {text}"
    )

    try:
        response = client.chat.completions.create(
            model="gpt-4",
            messages=[{"role": "user", "content": prompt}],
            temperature=0,
            max_tokens=50
        )
        content = response.choices[0].message.content.strip()
        result = json.loads(content)
        return result.get("polarity", "error"), result.get("label", "error")
    except Exception as e:
        print(f"GPT Error: {e}")
        return "error", "error"
```

Figure 2. Screenshot of code 1

The gpt_sentiment function is used to analyze tweet sentiment using GPT-4. It takes a single tweet as input and formats a prompt that instructs the model to return a polarity score and a sentiment label. This prompt is sent through the OpenAI API using the chat.completions.create method, with specific parameters such as the model type, message content, and token limit. Once a response is received, the function extracts the output, which is expected in JSON format, and stores the polarity and label values.

If any error occurs, such as a formatting issue or API failure, the function catches the exception and returns an "error" flag for both values. This function is applied repeatedly to each tweet in a dataset. Each processed result is saved to a CSV file for later evaluation. This component is important for generating the GPT-4 predictions that are used in both the classification and regression comparison experiments.

The second component focuses on polarity regression analysis. Instead of classifying tweets into fixed sentiment categories, this module compares the numerical polarity scores generated by GPT-4 and TextBlob to ground truth sentiment labels that have been mapped to numeric values. This allows for a more nuanced evaluation of sentiment detection. The analysis uses metrics such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Pearson correlation. These scores reveal how closely each model's predictions align with human-annotated sentiment. The component uses Python's scikit-learn and SciPy libraries to compute these metrics and Matplotlib to visualize the comparison between the two models.

```python
# Metrics
metrics = ['MAE', 'RMSE', 'Correlation']
textblob_scores = [mae_tb, rmse_tb, corr_tb]
gpt_scores = [mae_gpt, rmse_gpt, corr_gpt]

# Plotting
x = np.arange(len(metrics))
width = 0.35

plt.figure(figsize=(8, 5))
plt.bar(x - width / 2, textblob_scores, width, label='TextBlob', color='orange')
plt.bar(x + width / 2, gpt_scores, width, label='GPT-4', color='green')

plt.xticks(x, metrics)
plt.ylabel('Score')
plt.title('TextBlob vs GPT-4: MAE, RMSE, Correlation')
plt.ylim(0, 1.1)
plt.legend()
plt.grid(axis='y', linestyle='--', alpha=0.6)
plt.tight_layout()
plt.savefig("model_polarity_score_comparison.png")
```

Figure 3. Screenshot of code 2

This section of code generates a visual comparison between the GPT-4 and TextBlob models by evaluating their performance using three regression metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Pearson correlation. The variables textblob_scores and gpt_scores store the respective metric values for each model. The metrics list defines the metric names used as x-axis labels.

The code sets up a horizontal bar plot using Matplotlib [10]. It creates a grouped bar chart where TextBlob scores are plotted slightly to the left and GPT-4 scores slightly to the right of each metric's center position on the x-axis. The bars are color-coded to visually distinguish between the two models. The chart includes axis labels, a title, a legend, and gridlines to improve readability.

The plot is saved to a file named model_polarity_score_comparison.png and used to interpret which model performs better across the selected metrics. This code is executed after all sentiment scores have been computed and is key to visualizing model accuracy in numerical sentiment prediction.

The third major component of this system is the data cleaning and preprocessing step. This component prepares the tweets for sentiment analysis by removing elements such as usernames, hashtags, URLs, and extra whitespace. Clean data ensures that both GPT-4 and TextBlob receive accurate and relevant input without noise or distractions. Tweets are often messy, containing irrelevant metadata, so cleaning is essential for improving model performance and consistency. The system uses Python's re (regular expressions) and pandas libraries to implement this process. This component runs once, before any sentiment analysis or evaluation steps, and directly impacts the reliability of the final results.

```python
def clean_text(text):
    text = re.sub(r'@\w+', '', str(text))         # Remove @mentions
    text = re.sub(r'#\w+', '', text)              # Remove hashtags
    text = re.sub(r'http\S+', '', text)           # Remove URLs
    text = re.sub(r'\s+', ' ', text).strip()      # Remove extra spaces
    return text

sample_df['cleaned_tweet'] = sample_df['Tweet text'].apply(clean_text)
```

Figure 4. Screenshot of code 3

This function, named clean_text, is designed to prepare raw tweet data for analysis by removing unwanted elements commonly found in social media posts. These elements include user mentions (e.g., @username), hashtags (e.g., #autism), URLs, and excessive whitespace. The re.sub function from Python's regular expression module is used to match and replace these patterns.

The function begins by removing any mentions using the pattern @\w+, followed by hashtag removal using #\w+. Next, it eliminates URLs using http\S+, which matches any link beginning with "http". Finally, it removes any repeated whitespace and trims the result. The cleaned text is then applied to the dataset using pandas.DataFrame.apply, which updates the new cleaned_tweet column.

This code runs before any sentiment analysis takes place. Its role is critical because cleaner input leads to more reliable output from both GPT-4 and TextBlob. Tweets that contain irrelevant tokens can easily confuse models or dilute the accuracy of polarity and label predictions.

4. EXPERIMENT

4.1. Experiment 1

A key blind spot in our system is whether the sentiment labels generated by GPT-4 and TextBlob accurately reflect the intended tone of tweets discussing neurodivergent topics. Misclassifying sentiment could lead to misleading analysis or conclusions, especially when evaluating sensitive social content. It is essential that the models identify negative, neutral, and positive tones correctly to ensure reliability in future applications.

To evaluate sentiment classification accuracy, we used a labeled dataset of 100 tweets containing true sentiment annotations: negative, neutral, or positive. These tweets were analyzed using both GPT-4 and TextBlob. Each model predicted a sentiment label, which was compared against the ground truth using scikit-learn'sclassification_report. This generated precision, recall, and F1-score for each label, along with macro and weighted averages. The macro average was chosen as a key metric since it gives equal weight to all three sentiment categories.

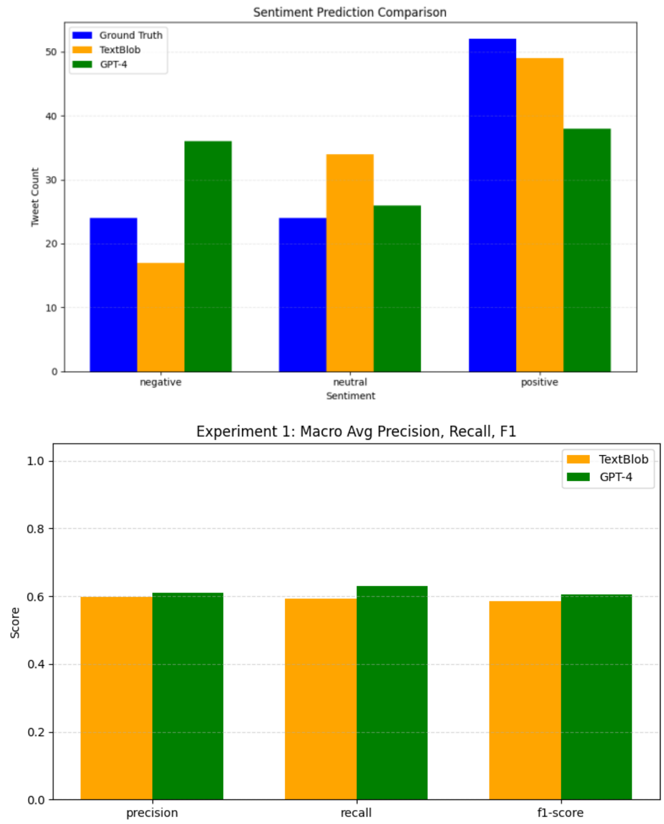

Figure 5. Figure of experiment 1

The classification report reveals that GPT-4 slightly outperformed TextBlob in overall macro-averaged metrics. GPT-4 achieved a macro average F1-score of 0.61, while TextBlob scored 0.58. The recall score was higher for GPT-4 at 0.63, compared to 0.59 for TextBlob, indicating better sensitivity in detecting all sentiment categories. However, TextBlob had a slightly better precision score on positive sentiment (0.73 vs 0.82 recall in GPT but lower precision at 0.82).

The accuracy of both models was identical at 62 percent, but the distribution of correct predictions varied. TextBlob performed better at identifying positive sentiment, while GPT-4 was stronger at detecting negative and neutral tones. This outcome confirms that GPT-4 has better contextual understanding, especially with more subtle or emotionally nuanced text, which is common in conversations about autism and neurodiversity.

One interesting result was TextBlob’s relatively strong recall for neutral sentiments, which suggests it tends to classify tweets more conservatively. GPT-4, while stronger overall, could still benefit from refinement in precision, especially for negative posts.

4.2. Experiment 2

A major blind spot in sentiment analysis is how accurately a model can assign a numerical polarity score that reflects the true emotional intensity of a statement. Misjudging sentiment strength can misrepresent a tweet's tone, especially in discussions about neurodivergent experiences where emotion is often subtle or complex. Accurate polarity scoring is crucial when building emotionally aware applications or tools.

To evaluate polarity score accuracy, we used the same set of 100 labeled tweets from Experiment A. Each tweet was mapped to a numeric sentiment label: -1 for negative, 0 for neutral, and 1 for positive. The GPT-4 and TextBlob models both produced polarity scores between -1 and 1 for each tweet. We then measured the difference between predicted scores and the ground truth using three metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Pearson Correlation Coefficient. These metrics were computed using scikit-learn and SciPy.

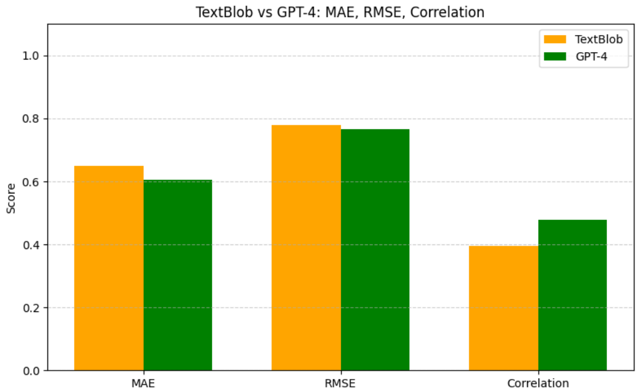

Figure 6. Figure of experiment 2

The results show that GPT-4 produced more accurate polarity predictions than TextBlob across all three metrics. GPT-4 achieved a MAE of 0.604, RMSE of 0.766, and a correlation of 0.479, compared to TextBlob's MAE of 0.650, RMSE of 0.778, and correlation of 0.394. This indicates that GPT-4's predictions were closer to the true sentiment scores and better aligned with human annotations.

While the difference in RMSE between the two models was small, the improvement in correlation suggests that GPT-4 better tracks the general sentiment trend across tweets. TextBlob, being lexicon-based, struggles with more complex or context-dependent emotional cues. On the other hand, GPT-4 can interpret nuance and sarcasm more effectively through its language modeling capabilities.

The slightly lower error and higher correlation confirm that GPT-4 provides a more robust polarity estimation. This experiment validates GPT-4's suitability for applications requiring finer emotional granularity, such as mental health monitoring or sensitive content analysis.

5. Related Work

A recent study titled "A Sentiment Analysis of Autism Tweets" (April 2024) introduced an Autism Tweets Visualization Dashboard that processes tweets with keywords related to autism, uses lexicon-based sentiment scoring to categorize them, and displays interactive visualizations of public attitudes [11]. Their system effectively aggregates emoji, hashtag, and text sentiment trends in real time, making public awareness and discourse more transparent. However, because it relies on lexicon-based scoring and simple thresholds for polarity, it struggles with subtle nuances, sarcasm, or context-dependent sentiments. It also ignores the categorical sentiment labels (positive/neutral/negative) that human annotators find essential. Moreover, it processes only English-language tweets using fixed dictionaries, which limits its language adaptability and contextual sensitivity. In contrast, our project leverages both GPT-4's contextual understanding and TextBlob's lexicon scores, and we compare not only their lexicon-based polarity but also classification accuracy and regression metrics against ground truth. This results in a more robust and nuanced analysis, capturing both label agreement and numerical sentiment fidelity. In this way, our work improves upon prior lexicon-only systems by incorporating dual-model evaluation, polarity regression, and a comprehensive set of metrics to better understand sentiment, particularly in sensitive, neurodiversity-related content.

Venkit and Wilson (2021) investigate bias against people with disabilities in popular sentiment and toxicity models such as TextBlob, VADER, Google Cloud NL, DistilBERT, and Toxic-BERT [12]. They introduce the Bias Identification Test in Sentiments (BITS) corpus 1,126 sentences designed to expose bias and demonstrate that all tested models show statistically significant negative sentiment bias when disability-related terms are present. This work effectively highlights harmful system behavior that can mislabel benign references to disability as negative or toxic, especially damaging in social media moderation contexts. However, BITS tests are largely template-based and do not assess model performance on actual user-generated content or nuanced emotional expressions. The study does not propose debiasing strategies, nor does it evaluate classification and regression performance of alternative models. In contrast, our work evaluates both TextBlob and GPT-4 on real-world tweets, measuring macro-averaged classification and numerical polarity accuracy against ground truth. By including GPT-4, which leverages context, we aim to minimize bias visible in lexicon-based tools and assess its effectiveness using multiple metrics, thereby extending bias analysis into practical performance assessment.

Malebary and Abulfaraj (2024) propose a stacking ensemble that merges lexicon-based and machine learning methods for tweet sentiment analysis [13]. They use feature extraction from lexicons (e.g., MPQA, SentiWordNet) combined with supervised learners, and employ a bat algorithm–optimized Elman recurrent neural network as a meta-classifier. Their approach significantly outperforms individual baselines across three standard tweet corpora. While their model improves classification accuracy and captures some contextual nuance, it still relies on manually assembled lexicon features and complex meta-optimization. It does not incorporate advanced language models capable of deep contextual understanding, nor does it evaluate regression-based polarity scoring or measure performance on neurodivergent discourse. Our project expands on this by using GPT-4 for its generative context interpretation, explicitly benchmarking against lexicon-based TextBlob on both classification and polarity regression tasks. We also include comprehensive evaluation—precision, recall, F1, MAE, RMSE, correlation—on tweets specifically about autism/neurodivergence. This allows us to combine an ensemble's strengths with modern LLM capabilities, offering deeper insight into sensitive-text performance and improving error analysis.

6. CONCLUSIONS

While the current system provides a strong comparison between GPT-4 and TextBlob in sentiment analysis, there are still notable limitations. First, the dataset is relatively small, consisting of only 100 tweets. A larger and more diverse dataset would provide better generalization and reduce sampling bias. Second, the sentiment labels used as ground truth are human-assigned, but subjective differences in interpretation may affect consistency. Additionally, the GPT-4 sentiment API can be expensive or rate-limited, which may prevent real-time scalability for larger applications [14]. The current system also does not support multilingual analysis, limiting its use to English tweets. To improve the project, we would expand the dataset, including inter-rater agreement for sentiment labels, and test additional models like RoBERTa or finetuned BERT. We could also introduce debiasing techniques to evaluate fairness and create a dashboard that visualizes sentiment trends across time, which would make the system more useful for real-time advocacy and policy monitoring.

This project demonstrates that GPT-4 outperforms traditional sentiment tools like TextBlob when analyzing emotionally complex tweets about autism. By evaluating both classification and regression outputs, we gain deeper insights into how AI interprets neurodivergent discourse [15]. Our findings suggest GPT-4 offers a more context-aware and accurate approach for sentiment detection.

REFERENCES

[1] Özel, Mustafa, and Özlem Çetinkaya Bozkurt. "Sentiment analysis on GPT-4 with comparative models using Twitter data." Acta Infologica 8.1 (2024): 23-33.

[2] Steinert, Rick, and Saskia Altmann. "Linking microblogging sentiments to stock price movement: An application of GPT-4." arXiv preprint arXiv:2308.16771 (2023).

[3] Salvatore, Sophia, Claudia White, and Stephen Podowitz-Thomas. "“Not a cookie cutter situation”: how neurodivergent students experience group work in their STEM courses." International Journal of STEM Education 11.1 (2024): 47.

[4] Heasman, Brett, and Alex Gillespie. "Neurodivergent intersubjectivity: Distinctive features of how autistic people create shared understanding." Autism 23.4 (2019): 910-921.

[5] Hotez, Emily, et al. "Developing “adulting for health”: investigating the health needs of neurodivergent emerging adults." Cureus 15.6 (2023).

[6] Liu, Fang, and Demosthenes Panagiotakos. "Real-world data: a brief review of the methods, applications, challenges and opportunities." BMC Medical Research Methodology 22.1 (2022): 287.

[7] Dang, Nhan Cach, María N. Moreno-García, and Fernando De la Prieta. "Sentiment analysis based on deep learning: A comparative study." Electronics 9.3 (2020): 483.

[8] Maaninen-Olsson, Eva, and Tomas Müllern. "A contextual understanding of projects—The importance of space and time." Scandinavian Journal of Management 25.3 (2009): 327-339.

[9] Auger, Tom, and Emma Saroyan. "Overview of the OpenAI APIs." Generative AI for Web Development: Building Web Applications Powered by OpenAI APIs and Next. js. Berkeley, CA: Apress, 2024. 87-116.

[10] Bisong, Ekaba. "Matplotlib and seaborn." Building machine learning and deep learning models on google cloud platform: A comprehensive guide for beginners. Berkeley, CA: Apress, 2019. 151-165.

[11] Awang, Norkhushaini, et al. "A Sentiment Analysis of Autism Tweets." J. Electrical Systems 20.4s (2024): 582-589.

[12] Venkit, Pranav Narayanan, and Shomir Wilson. "Identification of bias against people with disabilities in sentiment analysis and toxicity detection models." arXiv preprint arXiv:2111.13259 (2021).

[13] Malebary, Sharaf J., and Anas W. Abulfaraj. "A stacking ensemble based on lexicon and machine learning methods for the sentiment analysis of tweets." Mathematics 12.21 (2024): 3405.

[14] Maceda, Lany Laguna, et al. "Classifying sentiments on social media texts: A gpt-4 preliminary study." Proceedings of the 2023 7th International Conference on Natural Language Processing and Information Retrieval. 2023.

[15] Saunders, Pamela. "Neurodivergent rhetorics: Examining competing discourses of autism advocacy in the public sphere." Journal of Literary & Cultural Disability Studies 12.1 (2018): 1-17.