Research Article

Evaluating Sentiment Analysis Models on Historical Texts about Black Americans

Vishnu Athreya

Archbishop Mitty High School, 5000 Mitty Way, San Jose, California, USA

ABSTRACT

In this research study, four sentiment analysis models were trained on a dataset consisting of primary source documents about the experiences of Black Americans in the twentieth century, specifically in regards to their migration to the northern United States and racially oppressive legislation in the South. Each model used a different algorithm for sentiment analysis: Multinomial Naive Bayes, support vector machine (SVM), Generated Pre-Trained Transformer-2 (GPT-2), and Bidirectional Encoder Representations from Transformers (BERT). The goal was to determine which algorithm was best able to classify a 20th-century document about Black Americans as having either positive or negative outlook on their experiences. The results of this research, coupled with future, more advanced studies on such algorithmic capabilities, can allow for a more streamlined, objective, and accurate approach to categorizing historical documents, enabling historians to analyze them to generate insights and support arguments with greater speed and efficiency. Among the four algorithms, BERT achieved the highest accuracy rate (100%), followed by SVM and GPT (97%), and Multinomial Naive Bayes had the lowest accuracy rate (95%). However, the imbalanced nature of the dataset in terms of the ratio of positive to negative documents raises concerns about the algorithms being more likely to identify documents as positive. Also, the seemingly overwhelming accuracy of BERT signals that overfitting may have artificially skewed the results.

Keywords: Sentiment Analysis; African American Migration Narratives; 20th-Century Document Classification; NLP Models for Historical Texts; BERT Fine-Tuning on Primary Sources; Generated Pre-Trained Transformer-2; Multinomial Naive Bayes; Support Vector Machine

INTRODUCTION

Sentiment analysis is extremely potent in that it can be used to computationally determine the notions expressed in human-generated media on a large-scale and in an objective capacity. There are multiple applications for this in the humanities. For example, a survey on sentiment analysis methods discusses its use in analyzing modern opinions on literature, determining trends in the stock market, and performing market research to create better advertisements (9, 10).

Among these applications, leveraging sentiment analysis in processing historical documents is a particularly important use case. Historical documents are rank with the biases of those who wrote them; hence, an inherent part of historical document analysis

Corresponding author: Vishnu Athreya, E-mail: sendtovishnu@gmail.com.

Received April 30, 2025**; Accepted** June 14, 2025
https://doi.org/10.70251/HYJR2348.33147154

is determining the way the author feels about the topic of discussion in order to understand how that affects the validity or the emphases in the document. Historians have to do this very often, and this almost always entails sorting documents based on the sentiments expressed in them (7). Most of the history that is taught in schools today comes from the analysis historians have performed on such documents. Sentiment analysis could allow historians to to quickly sift through large numbers of documents, including reports written by journalists of the time as well as the perspectives of "boots-on-the-ground" individuals, and determine their biases. This, coupled with the fact that a sentiment analysis model will perform its respective algorithm in a predictable way regardless of the document which it is fed, could serve to create a more objective standard for determining the sentiments of documents en masse.

An article on Medium written by Tania Afzal (3) talks about the strengths of BERT and GPT (specifically ChatGPT) in accomplishing certain tasks (1). For example, it makes the contention that BERT is very good at sentiment analysis whereas GPT specializes in generating responses to queries that seem like they are written by humans. However, Afzal's results are general and may not necessarily provide a comprehensive view on how these models can be used to accomplish specific tasks, such as the categorization of historical documents. This research study endeavors to bridge this gap so that historians are better informed on what models they should use to assist their work with documents. In the specific dataset used for this study, many of the texts determined to contain positive sentiments are letters written by Black Americans to newspapers about their desire to migrate to the North, and many of the texts determined to have negative sentiments are newspaper articles reflecting on the actions of African American citizens in the South. Hence, the results of this particularly study would be most useful to historians looking to use sentiment analysis to gauge documents about 20th-century Black Americans who underwent similar experiences.

METHOD AND MATERIALS

Dataset

The Historical Sentiment Analysis Dataset was compiled and used in this analysis, consisting of textual passages from historical documents dated between the early 1900s to the mid-20th century (6). Each of the 200 entries in the dataset contain the following attributes:

- Passage: A textual excerpt from the historical source.
- Date: The date when the passage was written or spoken.
- Author: The author of the passage, if known.
- Source: The original source URL or reference.
- Label: The sentiment label associated with the passage, categorized as either Negative or Positive.

Pre-processing

Preprocessing of the text data was crucial before feeding it into the machine learning models. The following steps, as shown in Figure 1 were taken:

1. Text Cleaning: Removal of punctuation, special characters, and URLs.
2. Tokenization: Splitting the text into individual words or tokens.
3. Lowercasing: Converting all text to lowercase to maintain uniformity.
4. Stopword Removal: Eliminating commonly used stopwords (e.g., "the", "and") that do not carry sentiment information.
5. Lemmatization: Reducing words to their base form (e.g., "running" to "run") to standardize the vocabulary.

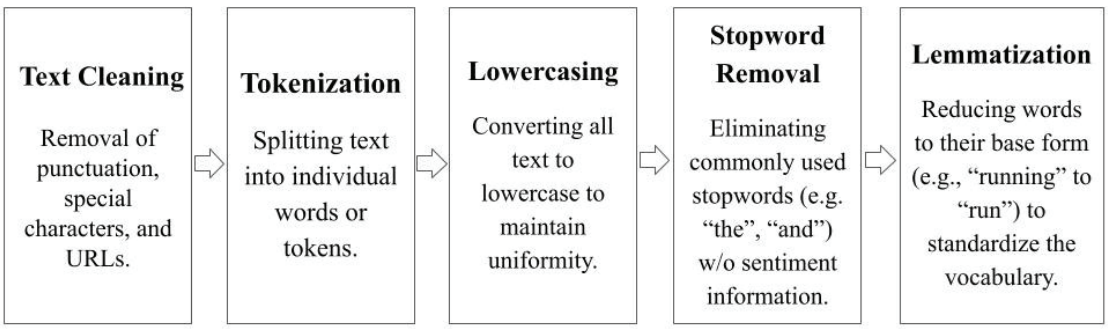

Figure 1. Pre-Processing Process.

Feature Extraction

For classical machine learning models, features were extracted using a Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer (2, 5), which transformed the textual passages into a numerical representation. This approach quantifies the importance of words in the corpus while reducing the impact of common but less informative terms. TF-IDF Vectorizer Equation:

$$W_{x,y} = tf_{x,y} \times \log \frac{N}{df_x} \quad \text{(Equation 1)}$$

where $tf_{x,y}$ is the frequency of x in y, df_x is the number of documents containing x, and N is the total number of documents.

Experimental Setup

Four models were used (4), two classical machine learning models (Naive Bayes and Support Vector Machine) and two deep learning models (Bidirectional Encoder Representations from Transformers and Generative Pre-trained Transformer-2). The dataset was split into training and testing sets using an 80-20 ratio, respectively, using a random split. All models scored 100% on training metrics: Precision, recall, F1-score, and accuracy were calculated for each class (Negative, Positive) and overall.:

1. Naive Bayes:
    - A Gaussian Naive Bayes classifier was used to predict sentiment labels based on the TF-IDF features.
2. SVM:
    - A Support Vector Machine with a linear kernel was used for classification. The model's hyper-parameters were tuned using cross-validation.
3. BERT:
    - The pre-trained BERT base model was fine-tuned on the sentiment classification task. The model weights were initialized from the bert-base-uncased checkpoint, with the classifier head fine-tuned on the dataset.
4. GPT-2:
    - A pre-trained GPT-2 model was used for sequence classification, fine-tuned similarly to BERT on the task of predicting sentiment labels.

The classical models relied on the TF-IDF feature representation of the text, while the deep learning models directly utilized the raw text passages. These are four of the most commonly used models for sentiment analysis and are quintessential representative examples of their respective types of machine learning algorithms.

Metrics

Precision (Equation 2, recall (Equation 3), and accuracy (4) were used to evaluate model performance:

$$Precision = \frac{TP}{TP + FP} \quad \text{(Equation 2)}$$

$$Recau = \frac{TP}{TP + FN} \quad \text{(Equation 3)}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad \text{(Equation 4)}$$

where TP is true positives, TN is true negatives, FP false positives, and FN is false negatives.

RESULTS

Although BERT performed the best out of the four models, all of the models were extremely accurate in their labeling of the historical documents fed to them from the dataset. The precision of the models in identifying positive sentiments ranged from 0.93 to 1.00 and in identifying negative sentiments was ubiquitously 1.00 (Table 1). The recall of the models in identifying positive sentiments was ubiquitously 1.00 and in identifying negative sentiments ranged from 0.83 to 1.00 (Table 2). Surprisingly, although GPT is a more complex algorithm than those run on SVMs, the SVM actually performed on the same level as GPT in terms of its ability to accurately recognize what

Table 1. Summary of model precision

Model Architecture	Precision	
	Positive	Negative
Naive Bayes	0.93	1.00
SVM	0.97	1.00
BERT	1.00	1.00
GPT-2	0.97	1.00

Table 2. Summary of model recall

Model Architecture	Recall	
	Positive	Negative
Naive Bayes	1.00	0.83
SVM	1.00	0.92
BERT	1.00	1.00
GPT-2	1.00	0.92

label a certain document should be given. This is likely due to the fact that this project was not suited to GPT's strengths. GPT specializes in generating human-like responses (3). Although generating text is useful in other applications, that particular skill does not have as much weight when simply attempting to label a document based on the sentiments it expresses. Hence, although GPT's algorithmic complexity still makes it a very accurate algorithm for sentiment identification, it is still not as potent for sentiment analysis as BERT, which is fundamentally designed to interpret text data. SVMs, like BERT, are also designed for this task, but are simpler, therefore, they balance themselves out with GPT's weaknesses in performing this particular task, making their accuracies average out to be about the same. Multinomial Naive Bayes, while also being a less complex algorithm made for sentiment analysis, was less accurate than an SVM in predicting the sentiment of the documents given. This may be for one of two reasons (or a combination of the two). The first reason is that SVMs tend to perform very well when the classes (or labels) of the given dataset are very distinct and separated from one another (1). While this is also true to an extent for Multinomial Naive Bayes, it has a particularly significant effect when dealing with SVMs. The second reason is that while both the traditional machine learning algorithms used in this project are fairly simple algorithms compared to deep learning algorithms like GPT and BERT, SVMs are still slightly more complex than Multinomial Naive Bayes in terms of the methods by which they analyze text. In fact, one could argue that although BERT yielded the most accurate results out of the four models, when incorporating algorithmic efficiency as a factor in deciding which model performed the task in the most potent manner, the SVM was the best balance between runtime and accuracy in that it is a relatively simple model that still yielded very accurate results.

The dataset fed to the models was skewed in that there was an imbalance in the representative sentiment labels given to the documents. Approximately 65% of the original dataset was determined to be positive and approximately 35% was determined to be negative (6). This may have affected the results in that the models would have had a tendency to report a sentiment as positive due to the higher representation of positive sentiments in the original dataset. However, despite the imbalance of sentiments in the original dataset, the models still performed with outstanding accuracy, indicating that the models all are very robust in their ability to discern sentiments accurately (feature learning) regardless of the flaws of the dataset they are working off of. However, another explanation is that positive documents were overrepresented in the testing set as well, which could help explain the high accuracy results yielded by the algorithms, as they were all predisposed to being able to easily identify positive documents.

Since these models were trained on a singular dataset, the results could have been skewed due to overfitting. This is reflected in the fact that many of the models exhibited 100% accuracy (Table 3). This could have caused the incredibly accurate results given by the BERT model, as although the models were not trained on the test data, the test data was still taken from the dataset on which the models were trained. Overfitting could be tested for by testing the BERT model on data from a different dataset to ensure that the BERT model's accuracy is not confined to data from a singular dataset (8).

Table 3. Summary of model accuracy

Model Architecture	Accuracy (%)
Naive Bayes	95
SVM	97
BERT	100
GPT-2	97

CONCLUSION

These results highlight the accuracy of all four model architectures on sentiment analysis. However, BERT exhibited the highest accuracy, with a supposed 100% accuracy. While this indicates BERT's superior capability in performing sentiment analysis, it also suggests that BERT may have been overfit to the training data. Overall, this work establishes BERT as the most potent of these four very commonly used sentiment analysis algorithms, paving the way for it to be the standard to be used in document analysis in the future. This work could also be extended to explore sentiment analysis in a variety of fields in the humanities to determine if BERT still outperforms other models on sentiment analysis tasks.

REFERENCES

1. Comparing Naïve Bayes and SVM for Text Classification. Available from: https://www.baeldung.com/cs/naive-bayes-vs-svm (accessed 2025-05-19).
2. TF-IDF in NLP (Term Frequency Inverse Document Frequency). Available from: https://medium.com/@abhishe

kjainindore24/tf-idf-in-nlp-term-frequency-inverse-document-frequency-e05b65932f1d (accessed 2024-10-15).
3. Comparing BERT and Chat-GPT: Understanding the Differences in NLP Models. Available from: https://medium.com/@taniaafzal/comparing-bert-and-chatgpt-understanding-the-differences-in-nlp-models-afb78e436105 (accessed 2024-09-22).
4. T. Abdullah and A. Ahmet. Deep learning in sentiment analysis: Recent architectures. *ACM Computing Surveys*. 2022; 55 (8): 1–37. https://doi.org/10.1145/3548772
5. M. Z. Asghar, A. Khan, S. Ahmad, and F. M. Kundi. A review of feature extraction in sen- timent analysis. *Journal of Basic and Applied Scientific Research*. 2014; 4 (3): 181–186.
6. V. Athreya. Black Americans in the 20th Century - Historical Sentiment Analysis Dataset. *Zenodo*. 2025. (accessed 2024-09-22).
7. P. Constantopoulos, M. Doerr, M. Theodoridou, and M. Tzobanakis. Historical Documents As Monuments And As Sources. *Institute of Com- puter Science Foundation for Research and Tech- nology, Hellas*. 2002; 31: 205–208.
8. H. H. Do and P. Prasad. Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review. *Expert Systems with Applications*. 2019; 118: 272–299. https://doi.org/10.1016/j.eswa.2018.10.003
9. Y. Mao. Sentiment analysis methods, applications, and challenges: A systematic literature review. *Journal of King Saud University*. 2024; 36 (4): 1319–1578. https://doi.org/10.1016/j.jksuci.2024.102048
10. W. Mayur, A. Chandara Sekhara Rao, and C. Kulkarni. A survey on sentiment analysis methods, applications, and challenges. *Spring Nature Link*. 2022; 55: 5731–5780. https://doi.org/10.1007/s10462-022-10144-1

APPENDIX A: CONFUSION MATRICES AND ENUMERATED RESULTS

1. Naive Bayes
   - Precision: 1.00 for Negative, 0.93 for Positive.
   - Recall: 0.83 for Negative, 1.00 for Positive.
   - Accuracy: 95%.

2.SVM
   - Precision: 1.00 for Negative, 0.97 for Positive.
   - Recall: 0.92 for Negative, 1.00 for Positive.
   - Accuracy: 97%.

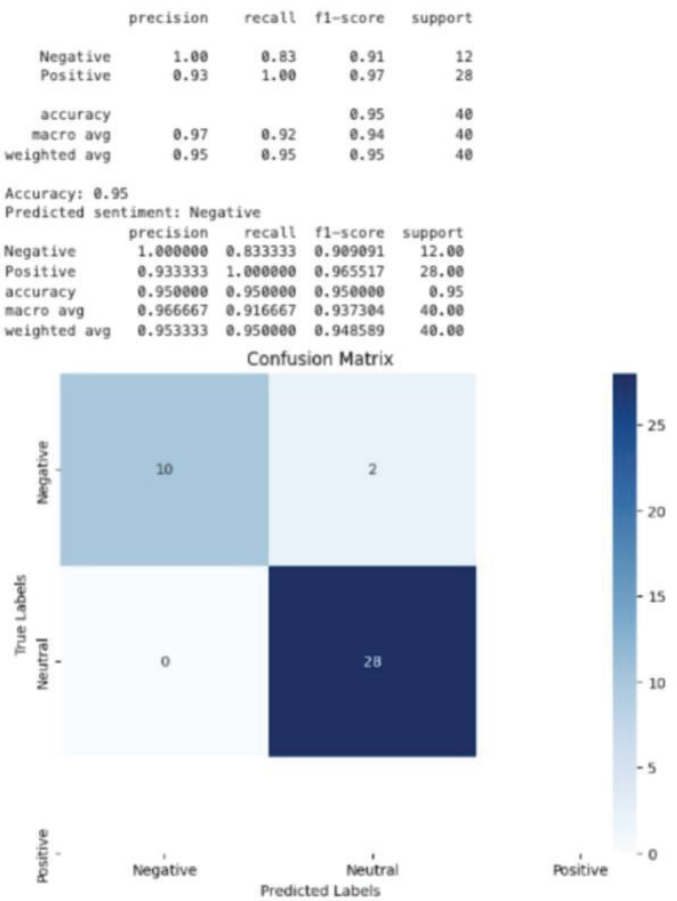

Figure 2. Naive Bayes Confusion Matrix & Classification Report.

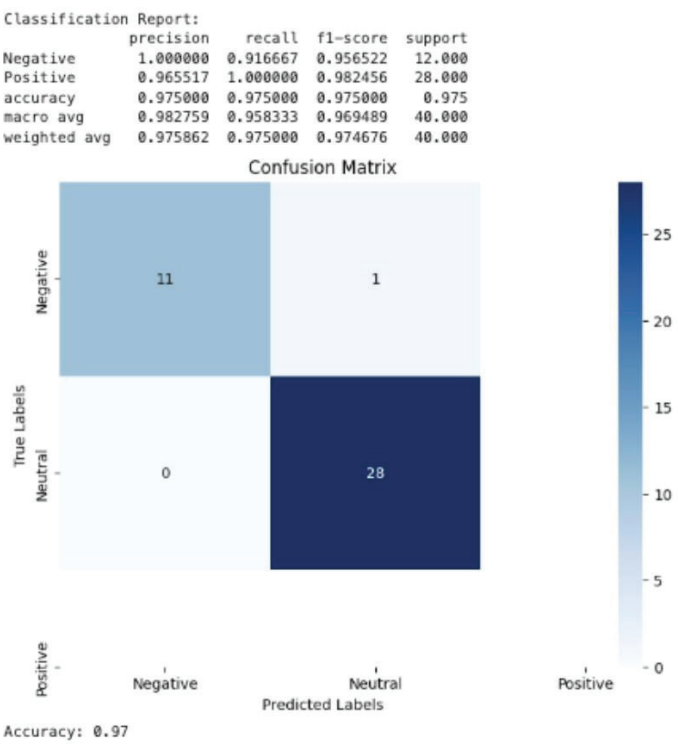

Figure 3. SVM Confusion Matrix & Classification Report.

3. BERT
   - Precision, Recall, F1-score, and Accuracy: 100%.

4. GPT-2
   - Precision: 1.00 for Negative, 0.97 for Positive.
   - Recall: 0.92 for Negative, 1.00 for Positive.
   - Accuracy: 97%

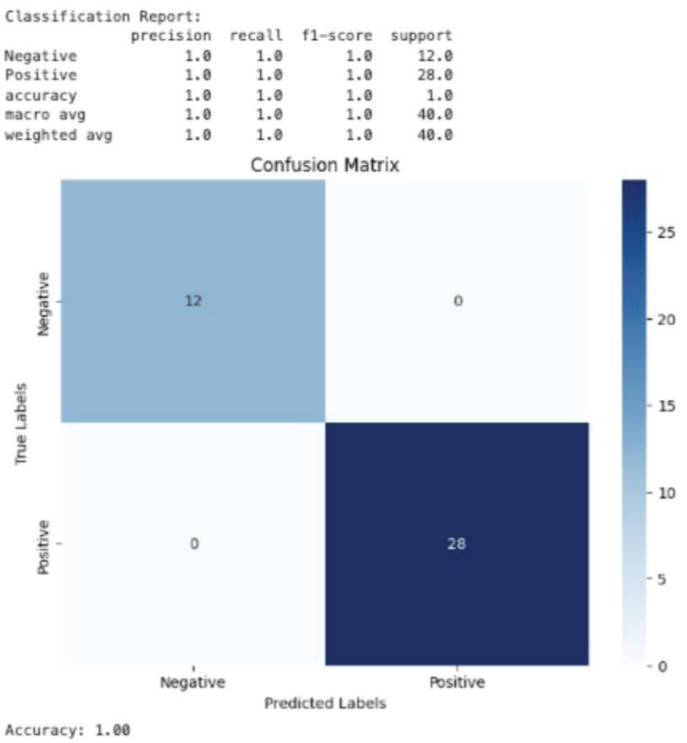

Figure 4. BERT Confusion Matrix & Classification Report.

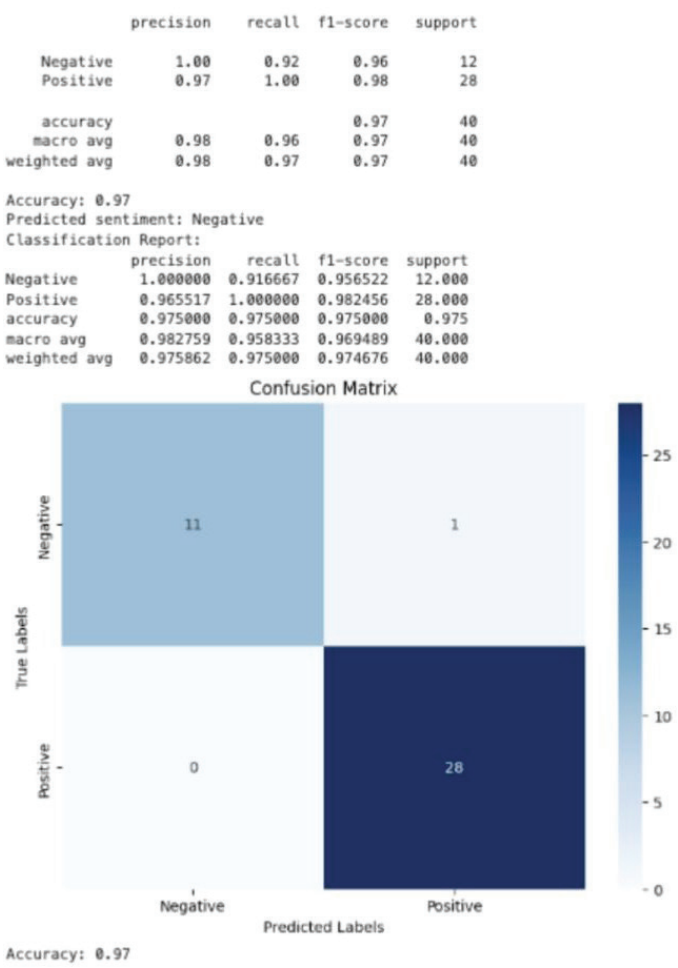

Figure 5. GPT-2 Confusion Matrix & Classification Report.

APPENDIX B: P-VALUES TO STATISTICALLY COMPARE PERFORMANCE DIFFERENCES BETWEEN THE MODELS

- Naive Bayes vs. SVM:
  - Naive Bayes correct, SVM wrong: 0
  - SVM correct, Naive Bayes wrong: 0
  - p-value: 1.0000

- Naive Bayes vs. GPT-2:
  - Naive Bayes correct, GPT-2 wrong: 0
  - GPT-2 correct, Naive Bayes wrong: 28
  - p-value: 0.0000

- Naive Bayes vs BERT
  - Naive Bayes correct, BERT wrong: 0
  - BERT correct, Naive Bayes wrong: 28
  - p-value: 0.0000

- SVM vs GPT-2
  - SVM correct, GPT-2 wrong: 0
  - GPT-2 correct, SVM wrong: 28
  - p-value: 0.0000

- SVM vs BERT
  - SVM correct, BERT wrong: 0
  - BERT correct, SVM wrong: 28
  - p-value: 0.0000

- GPT-2 vs BERT
  - GPT-2 correct, BERT wrong: 0
  - BERT correct, GPT-2 wrong: 0
  - p-value: 1.0000

- Statistical Test Used: McNemar's Test

- Number of Samples: 28