## Exploring Naive Bayes and SVM for Sentiment Analysis

This report investigates the performance of Naive Bayes and Support Vector Machines (SVM) for sentiment analysis of text data.

### 1. Data Acquisition and Preprocessing

* **Dataset:** A dataset containing labeled text reviews (positive/negative sentiment) will be obtained from a reliable source. Examples include movie reviews, product reviews, or social media sentiment datasets.
* **Preprocessing:** The text data will undergo the following preprocessing steps:
  + **Tokenization:** Splitting the text into individual words or phrases (tokens).
  + **Stop word removal:** Removing common words that don't contribute much to sentiment (e.g., "the", "a", "is").
  + **Text normalization:** Lowercasing text, handling punctuation, and potentially stemming/lemmatization (converting words to their root form).
  + **Feature engineering:** Converting the preprocessed text into numerical features suitable for the models. This might involve techniques like:
    - **Bag-of-Words (BoW):** Representing each review as a frequency vector of words appearing in the document.
    - **TF-IDF:** Assigning weights to words based on their frequency in the document and rarity across the corpus, emphasizing informative words.

### 2. Model Implementation

* **Naive Bayes:** A Naive Bayes classifier will be implemented. This probabilistic model assumes independence between features (words) and calculates the probability of a document belonging to a specific sentiment class (positive or negative) based on the word occurrences.
* **Support Vector Machine (SVM):** An SVM model will be trained for sentiment classification. SVMs aim to find a hyperplane in the feature space that best separates the positive and negative sentiment reviews. Here, features represent the preprocessed text data (e.g., BoW or TF-IDF vectors).

**Training-Testing Split:** The preprocessed data will be split into training and testing sets. The training set will be used to train the models, and the testing set will be used to evaluate their performance on unseen data.

### 3. Model Evaluation

The performance of both models will be evaluated on the testing set using metrics commonly used for text classification:

* **Accuracy:** Proportion of correctly classified reviews (positive or negative).
* **Precision:** Ratio of correctly predicted positive reviews to the total predicted positive reviews.
* **Recall:** Ratio of correctly identified positive reviews to the total actual positive reviews.
* **F1-score:** Harmonic mean of precision and recall, combining both metrics into a single measure.

Additionally, we might consider:

* **Confusion Matrix:** Visualizing the distribution of correct and incorrect predictions for each sentiment class.

### 4. Comparison and Discussion

By comparing the performance metrics of both models on the testing set, we can determine which model is more effective for sentiment analysis of the specific dataset.

* **Naive Bayes:** Naive Bayes is a simple and efficient classifier that can perform well when the feature independence assumption holds true (words are conditionally independent given the sentiment). It can be particularly efficient for large datasets.
* **SVM:** SVMs are powerful learners capable of handling non-linear relationships between features, potentially leading to higher accuracy in some cases. However, they can be more computationally expensive to train than Naive Bayes.

**Choosing the Best Model:**

The choice between Naive Bayes and SVM depends on several factors:

* **Dataset characteristics:** If the feature independence assumption in Naive Bayes is likely to hold, it might be a good choice for its simplicity and efficiency.
* **Performance trade-off:** If maximizing accuracy is crucial, and the dataset is complex, SVM might be a better option despite its higher computational cost.
* **Interpretability:** While Naive Bayes offers some level of interpretability by analyzing word probabilities, SVMs can be less interpretable due to the complex decision boundaries learned in the feature space.

### 5. Conclusion

This report explored the performance of Naive Bayes and SVM for sentiment analysis. By evaluating them on a text dataset, we can identify the model that best suits the specific task considering factors like accuracy, interpretability, and computational efficiency.