1. Summarize the results of your experiments for Problem 2. Are your results "good" or "bad", and importantly, how do you determine that?

   The Naive Bayes classifier achieved 43% accuracy, facing difficulties in accurately predicting political parties from their speeches as input. Precision, recall, and F1-score metrics vary across different parties, highlighting the challenges in classifying speeches with respect to political party. The dominant parties showed better performance, while others faced difficulties. Overall, these factors collectively suggest that the results can be regarded as less favorable or "bad" due to the model's struggles in handling the complexities of the dataset. Class imbalance impacted generalization caused misclassifications. Addressing imbalances and exploring alternative models are to be considered for improvement.

2. In Problem 4, you implemented undersampling. How did your results change compared to Problem 2? How would "oversampling" have looked like for this task?

   Implementing undersampling resulted in a slight drop in accuracy to 41%. While precision, recall, and F1-score for each party exhibited changes, indicating the impact of class balancing, achieving balance affected the model's ability to predict on minority classes. The trade-off is evident in decreased precision and recall for most classes.

   If we had done oversampling, it could make more examples of the less common groups to help the model get better at understanding them, potentially enhancing the model's exposure to these classes and reduce the difficulties posed by imbalances. But, we would need to check if this really improves the model's performance.

3. Why is it important to do a hyperparameter search before drawing conclusions about the performance of a model? Why do you think it is often *not* done, anyway? Why should you never tune hyperparameters on the test set?

   Hyperparameters are like settings given to a model before it starts learning from data which are to be chosen before training begins. It is important to search for the best hyperparameters before deciding how well a model is doing. This is because these settings impact how the model learns from data and makes predictions. Doing a careful search helps ensure the model performs at its best and can handle new, unseen information well.

   While hyperparameter search is a crucial step in optimizing model performance, it may not always be done for various reasons like Computational Resources, lack of domain knowledge, lack of awareness on hyperparameter tuning, non-availability of enough data.

   Tuning hyperparameters on the test set can lead to overfitting, where the model becomes too familiar to the test data. It is like test data becomes training data. Using a separate validation set or cross-validation helps mitigate this risk.