**Explanation**

1. **Load Dataset**: We use the 20 Newsgroups dataset, which is preloaded in scikit-learn.
2. **Data Preprocessing**:
   * **Tokenization**: Split text into tokens.
   * **Stopwords Removal**: Remove common words that do not contribute meaningful information.
   * **Filtering**: Keep only alphabetic words.
3. **Vectorization**: Convert text into numerical features using TF-IDF (Term Frequency-Inverse Document Frequency).
4. **Split Dataset**: Split the dataset into training and testing sets.
5. **Train Model**: Train a Naive Bayes classifier (MultinomialNB) on the training data.
6. **Predict**: Predict labels for the test data.
7. **Evaluate**: Calculate accuracy and generate a classification report including precision, recall, and F1-score for each class.

**Explanation of Changes**

1. **Model Import**: Import LogisticRegression from sklearn.linear\_model.
2. **Train Model**: Create an instance of LogisticRegression and fit it to the training data. Note that max\_iter=1000 is specified to ensure convergence, especially for large datasets.
3. **Evaluation**: The evaluation metrics (accuracy and classification report) remain the same.

**Summary**

* **Logistic Regression**: A common and effective classification algorithm used for binary and multi-class classification tasks.
* **TF-IDF Vectorization**: Converts text into numerical features.
* **Evaluation Metrics**: Includes accuracy, precision, recall, and F1-score.