Project Documentation: DistilBERT for Monetary Policy Speech Classification

The project uses several user-defined functions that needs to be documented for readability.

```python
def plot_classes(df, column_name):
    """
    Plots the class distribution of a specified column in a DataFrame.

    Parameters:
    - df: The input DataFrame
    - column_name: The column whose distribution is to be plotted
    """
    # Function body would go here
```

```python

def flat_accuracy(preds, labels):
    """
    Computes the accuracy of predictions against the true labels.

    Parameters:
    - preds (numpy.ndarray): A 2D numpy array containing the prediction scores.
    - labels (numpy.ndarray): A 1D numpy array containing the true class labels.

    Returns:
    - float: The computed accuracy of the
      predictions.
    """
    pred_flat = np.argmax(preds, axis=1).
      flatten()
    labels_flat = labels.flatten()
    return np.sum(pred_flat == labels_flat)
       / len(labels_flat)
```

```
def format_time(elapsed):
    '''
    Takes a time in seconds and returns a
      string hh:mm:ss
    '''
```

```
def classify_sentiment(text):
    """
    Classifies the sentiment of a given
      text as either 'Positive' or '
      Negative' using a pre-trained model.

    Parameters:
    - text (str): The input text string to
      be classified.

    Returns:
    - str: 'Positive' if the sentiment of
      the text is positive, otherwise '
      Negative’.

    Note:
    Assumes that a pre-trained model named
      ’model’ and its associated tokenizer
      ’tokenizer’ are already loaded.
    """
```