```
import pandas as pd
import json

# 讀取數據
tweets =
pd.read_json('/kaggle/input/dm-2024-isa-5810-lab-2-homework/tweets_DM.
json', lines=True)
emotion =
pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/emotion.csv
')
data_id =
pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/data_identi
fication.csv')

# 檢查數據
print(tweets.head())
print(emotion.head())
print(data_id.head())
```

```
   _score          _index
_source  \
0     391  hashtag_tweets  {'tweet': {'hashtags': ['Snapchat'],
'tweet_id...
1     433  hashtag_tweets  {'tweet': {'hashtags': ['freepress',
'TrumpLeg...
2     232  hashtag_tweets  {'tweet': {'hashtags': ['bibleverse'],
'tweet_...
3     376  hashtag_tweets  {'tweet': {'hashtags': [], 'tweet_id':
'0x1cd5...
4     989  hashtag_tweets  {'tweet': {'hashtags': [], 'tweet_id':
'0x2de2...

             _crawldate   _type
0  2015-05-23 11:42:47  tweets
1  2016-01-28 04:52:09  tweets
2  2017-12-25 04:39:20  tweets
3  2016-01-24 23:53:05  tweets
4  2016-01-08 17:18:59  tweets
   tweet_id       emotion
0  0x3140b1       sadness
1  0x368b73       disgust
2  0x296183  anticipation
3  0x2bd6e1           joy
4  0x2ee1dd  anticipation
   tweet_id identification
0  0x28cc61           test
1  0x29e452          train
2  0x2b3819          train
3  0x2db41f           test
4  0x2a2acc          train
```

```python
# 檢查tweets 的列
print("Columns in tweets:", tweets.columns)

# 檢查emotion 的列
print("Columns in emotion:", emotion.columns)

# 檢查data_id 的列
print("Columns in data_id:", data_id.columns)

# 確保所有表的'tweet_id' 列名一致
tweets.rename(columns=lambda x: x.strip(), inplace=True)  # 去除空格
emotion.rename(columns=lambda x: x.strip(), inplace=True)
data_id.rename(columns=lambda x: x.strip(), inplace=True)

# 如果有具體列名錯誤，例如'Tweet_ID' 或'tweetId'
tweets.rename(columns={'Tweet_ID': 'tweet_id'}, inplace=True)
emotion.rename(columns={'Tweet_ID': 'tweet_id'}, inplace=True)
data_id.rename(columns={'Tweet_ID': 'tweet_id'}, inplace=True)

# 檢查文件頭部是否正確
print(pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/emotion.csv').head())
print(pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/data_identification.csv').head())

# 確保文件讀取時正確處理編碼
emotion = pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/emotion.csv', encoding='utf-8')
data_id = pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/data_identification.csv', encoding='utf-8')
```

```
Columns in tweets: Index(['_score', '_index', '_source', '_crawldate', '_type'], dtype='object')
Columns in emotion: Index(['tweet_id', 'emotion'], dtype='object')
Columns in data_id: Index(['tweet_id', 'identification'], dtype='object')
   tweet_id       emotion
0  0x3140b1       sadness
1  0x368b73       disgust
2  0x296183  anticipation
3  0x2bd6e1           joy
4  0x2ee1dd  anticipation
   tweet_id identification
0  0x28cc61           test
1  0x29e452          train
2  0x2b3819          train
3  0x2db41f           test
4  0x2a2acc          train
```

```python
# 匯入必要套件
import pandas as pd
import json

# 讀取JSON 檔案並檢查結構
tweets = []
with
open('/kaggle/input/dm-2024-isa-5810-lab-2-homework/tweets_DM.json',
'r') as f:
    for i, line in enumerate(f):
        if i < 5:  # 檢查前5 行的JSON 結構
            print(json.loads(line))  # 打印JSON 結構
        tweets.append(json.loads(line))

# 將JSON 資料轉為DataFrame
tweets_df = pd.DataFrame(tweets)

# 提取`_source` 中的`tweet` 資料
tweets_df = pd.json_normalize(tweets_df['_source'])

# 檢查展平後的資料
print("Flattened tweets data preview:")
print(tweets_df.head())

# 重命名列以匹配其他表格
tweets_df.rename(columns={'tweet.tweet_id': 'tweet_id', 'tweet.text':
'tweet_text'}, inplace=True)

# 讀取其他資料表
emotion =
pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/emotion.csv
', encoding='utf-8')
data_id =
pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/data_identi
fication.csv', encoding='utf-8')

# 檢查欄位名稱並去除多餘空格
tweets_df.rename(columns=lambda x: x.strip(), inplace=True)
emotion.rename(columns=lambda x: x.strip(), inplace=True)
data_id.rename(columns=lambda x: x.strip(), inplace=True)

# 合併資料
tweets_df = tweets_df.merge(emotion, on='tweet_id', how='left')  # 合併
情緒標籤
tweets_df = tweets_df.merge(data_id, on='tweet_id', how='left')  # 合併
數據識別

# 確認結果
print("Merged data preview:")
print(tweets_df.head())
```

```
# 儲存結果到CSV（可選）
tweets_df.to_csv('merged_tweets.csv', index=False)
```

```
{'_score': 391, '_index': 'hashtag_tweets', '_source': {'tweet':
{'hashtags': ['Snapchat'], 'tweet_id': '0x376b20', 'text': 'People who
post "add me on #Snapchat" must be dehydrated. Cuz man.... that\'s
<LH>'}}, '_crawldate': '2015-05-23 11:42:47', '_type': 'tweets'}
{'_score': 433, '_index': 'hashtag_tweets', '_source': {'tweet':
{'hashtags': ['freepress', 'TrumpLegacy', 'CNN'], 'tweet_id':
'0x2d5350', 'text': '@brianklaas As we see, Trump is dangerous to
#freepress around the world. What a <LH> <LH> #TrumpLegacy.  #CNN'}},
'_crawldate': '2016-01-28 04:52:09', '_type': 'tweets'}
{'_score': 232, '_index': 'hashtag_tweets', '_source': {'tweet':
{'hashtags': ['bibleverse'], 'tweet_id': '0x28b412', 'text':
'Confident of your obedience, I write to you, knowing that you will do
even more than I ask. (Philemon 1:21) 3/4 #bibleverse <LH> <LH>'}},
'_crawldate': '2017-12-25 04:39:20', '_type': 'tweets'}
{'_score': 376, '_index': 'hashtag_tweets', '_source': {'tweet':
{'hashtags': [], 'tweet_id': '0x1cd5b0', 'text': 'Now ISSA is stalking
Tasha 😂😂😂 <LH>'}}, '_crawldate': '2016-01-24 23:53:05', '_type':
'tweets'}
{'_score': 989, '_index': 'hashtag_tweets', '_source': {'tweet':
{'hashtags': [], 'tweet_id': '0x2de201', 'text': '"Trust is not the
same as faith. A friend is someone you trust. Putting faith in anyone
is a mistake." ~ Christopher Hitchens <LH> <LH>'}}, '_crawldate':
'2016-01-08 17:18:59', '_type': 'tweets'}
Flattened tweets data preview:
                  tweet.hashtags tweet.tweet_id  \
0                     [Snapchat]       0x376b20
1  [freepress, TrumpLegacy, CNN]       0x2d5350
2                    [bibleverse]       0x28b412
3                              []       0x1cd5b0
4                              []       0x2de201

                                           tweet.text
0  People who post "add me on #Snapchat" must be ...
1  @brianklaas As we see, Trump is dangerous to #...
2  Confident of your obedience, I write to you, k...
3                Now ISSA is stalking Tasha 😂😂😂 <LH>
4  "Trust is not the same as faith. A friend is s...
Merged data preview:
                  tweet.hashtags  tweet_id  \
0                     [Snapchat]  0x376b20
1  [freepress, TrumpLegacy, CNN]  0x2d5350
2                    [bibleverse]  0x28b412
3                              []  0x1cd5b0
4                              []  0x2de201

                                          tweet_text       emotion  \
```

```
0  People who post "add me on #Snapchat" must be ...  anticipation
1  @brianklaas As we see, Trump is dangerous to #...       sadness
2  Confident of your obedience, I write to you, k...            NaN
3                Now ISSA is stalking Tasha 😭😭😭 <LH>           fear

4  "Trust is not the same as faith. A friend is s...            NaN

  identification
0          train
1          train
2           test
3          train
4           test
```

模型部分

這部分的程式碼負責定義、訓練或評估機器學習模型，包括模型選擇與超參數設定。

```
import pandas as pd
import re
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from nltk.corpus import stopwords
import nltk

# 確保下載nltk 停用詞（如果無法在線下載，提供手動路徑）
try:
    nltk.download('stopwords')
except:
    print("Unable to download stopwords. Please check the network
connection.")

# 如果網路不可用，手動提供停用詞
try:
    stop_words = set(stopwords.words('english'))
except:
    stop_words = {"a", "an", "the", "is", "in", "at", "of", "on",
"and", "to", "with", "for", "by", "that", "this", "from"}

# 確保`tweets` 是DataFrame
tweets_list = []  # 如果原本是list, 應先構造DataFrame
with
open('/kaggle/input/dm-2024-isa-5810-lab-2-homework/tweets_DM.json',
'r') as f:
    for line in f:
        tweets_list.append(json.loads(line))

# 將JSON list 轉為DataFrame
tweets_df = pd.json_normalize([tweet['_source'] for tweet in
tweets_list])
```

```
# 提取欄位
tweets_df.rename(columns={'tweet.text': 'tweet_text',
'tweet.tweet_id': 'tweet_id'}, inplace=True)

# 確認欄位
print("Columns:", tweets_df.columns)

# 預處理函數
def preprocess_text(text):
    # 刪除網址、標籤和特殊字符
    text = re.sub(r'http\S+|www\S+|@\w+|#\w+', '', text)
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    text = text.lower()  # 全部轉小寫
    text = ' '.join([word for word in text.split() if word not in
stop_words])  # 移除停用詞
    return text

# 預處理文本
tweets_df['clean_text'] =
tweets_df['tweet_text'].apply(preprocess_text)

# 加載其他表格
emotion =
pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/emotion.csv
', encoding='utf-8')
data_id =
pd.read_csv('/kaggle/input/dm-2024-isa-5810-lab-2-homework/data_identi
fication.csv', encoding='utf-8')

# 合併數據
tweets_df = tweets_df.merge(emotion, on='tweet_id', how='left')
tweets_df = tweets_df.merge(data_id, on='tweet_id', how='left')

# 分割訓練和測試數據
train_data = tweets_df[tweets_df['identification'] == 'train']
test_data = tweets_df[tweets_df['identification'] == 'test']

X_train = train_data['clean_text']
y_train = train_data['emotion']
X_test = test_data['clean_text']

# 確認分割後數據
print("Training Data:", train_data.head())
print("Test Data:", test_data.head())
```

```
[nltk_data] Downloading package stopwords to /usr/share/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Columns: Index(['tweet.hashtags', 'tweet_id', 'tweet_text'],
dtype='object')
Training Data:                   tweet.hashtags  tweet_id  \
```

```
0                      [Snapchat]  0x376b20
1  [freepress, TrumpLegacy, CNN]  0x2d5350
3                             []  0x1cd5b0
5      [authentic, LaughOutLoud]  0x1d755c
6                             []  0x2c91a8

                                          tweet_text  \
0  People who post "add me on #Snapchat" must be ...
1  @brianklaas As we see, Trump is dangerous to #...
3                Now ISSA is stalking Tasha 😩😩😩 <LH>
5  @RISKshow @TheKevinAllison Thx for the BEST TI...
6       Still waiting on those supplies Liscus. <LH>

                                          clean_text       emotion  \
0   people post add must dehydrated cuz man thats lh  anticipation
1              see trump dangerous around world lh lh       sadness
3                             issa stalking tasha lh          fear
5  thx best time tonight stories heartbreakingly ...           joy
6                    still waiting supplies liscus lh  anticipation

  identification
0          train
1          train
3          train
5          train
6          train
Test Data:                          tweet.hashtags  tweet_id  \
2                        [bibleverse]  0x28b412
4                                  []  0x2de201
9   [materialism, money, possessions]  0x218443
30               [GodsPlan, GodsWork]  0x2939d5
33                                 []  0x26289a

                                           tweet_text  \
2   Confident of your obedience, I write to you, k...
4   "Trust is not the same as faith. A friend is s...
9   When do you have enough ? When are you satisfi...
30  God woke you up, now chase the day #GodsPlan #...
33  In these tough times, who do YOU turn to as yo...

                                          clean_text emotion
identification
2   confident obedience write knowing even ask phi...     NaN
test
4   trust faith friend someone trust putting faith...     NaN
test
9               enough satisfied goal really money lh     NaN
test
30                             god woke chase day lh     NaN
test
```

```
33                      tough times turn symbol hope lh     NaN
test
```

```
# 使用TF-IDF 將文本轉為數值特徵
tfidf = TfidfVectorizer(max_features=1000)  # 選擇前1000 個最常見詞語

X_train_tfidf = tfidf.fit_transform(X_train)
X_test_tfidf = tfidf.transform(X_test)
```

模型部分
這部分的程式碼負責定義、訓練或評估機器學習模型，包括模型選擇與超參數設定。

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report

# 訓練邏輯回歸模型
model = LogisticRegression(max_iter=1000)
model.fit(X_train_tfidf, y_train)

# 預測結果
y_pred = model.predict(X_train_tfidf)

# 評估模型
print(classification_report(y_train, y_pred))
```

```
              precision    recall  f1-score   support

       anger       0.67      0.05      0.10     39867
anticipation       0.50      0.38      0.43    248935
     disgust       0.32      0.14      0.20    139101
        fear       0.64      0.16      0.26     63999
         joy       0.43      0.83      0.57    516017
     sadness       0.35      0.27      0.30    193437
    surprise       0.48      0.06      0.10     48729
       trust       0.50      0.12      0.19    205478

    accuracy                           0.43   1455563
   macro avg       0.49      0.25      0.27   1455563
weighted avg       0.45      0.43      0.38   1455563
```

模型部分
這部分的程式碼負責定義、訓練或評估機器學習模型，包括模型選擇與超參數設定。

```
# 預測情緒
test_data.loc[:, 'emotion'] = model.predict(X_test_tfidf)  # 使用.loc
修改資料

# 生成提交文件
submission = test_data[['tweet_id', 'emotion']].copy()
submission.columns = ['id', 'emotion']
```

```python
# 保存到/kaggle/working 目錄
submission.to_csv('/kaggle/working/submission.csv', index=False)


print("Submission file generated: /kaggle/working/submission.csv")
```

```
Submission file generated: /kaggle/working/submission.csv
```