1. Importación y carga de Datos

```
#Importación de librerias
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split, GridSearchCV, cross_val_score
from sklearn import model_selection, naive_bayes, svm
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
from sklearn.metrics import f1_score
from sklearn.metrics import confusion_matrix, plot_confusion_matrix, ConfusionMatrixDisplay
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression
import pandas as pd
import numpy as np

#NLP

from nltk.tokenize import sent_tokenize, word_tokenize, RegexpTokenizer
from nltk.stem import WordNetLemmatizer
from nltk.stem.porter import PorterStemmer
from nltk.corpus import stopwords
from nltk.sentiment.vader import SentimentIntensityAnalyzer

from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
```

```
!pip install inflect
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: inflect in /usr/local/lib/python3.7/dist-packages (2.1.0)
```

```
import nltk
nltk.download('wordnet')
nltk.download('omw-1.4')
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
True
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
np.random.seed(500)
```

```
#Carga de datos
df_original=pd.read_csv('/content/drive/MyDrive/BI/Proyecto 1/Data/SuicidiosProyecto.csv', sep=',', encoding = 'ut
df_suicide = df_original.copy()
```

```
df_suicide.head(50)
```

Unnamed: 0	text	class
173271	i want to destroy myselffor once everything wa...	suicide
336321	I kinda got behind schedule with learning for ...	non-suicide
256637	I'm just not sure anymoreFirst and foremost: I...	suicide
303772	please give me a reason to liveThats too much ...	suicide
293747	27f struggling to find meaning moving forwardI...	suicide
205651	Let’s get this bread 😎 Anyone know any good ba...	non-suicide
97174	Day 126 of posting random "fun" facts everyday...	non-suicide
195945	Little brother is self mutilating. Please help...	suicide
305273	Why do women always go in groups to their wash...	non-suicide
69929	Did you guys know that there's no school for g...	non-suicide
111327	Was about to post something... but forgot it w...	non-suicide
341361	Ah shite I said SUCK MY CLIT instead of SUCK M...	non-suicide
86906	if you hate coffee but need the caffeine try t...	non-suicide
281142	General Kenobi, Hello There First one to comme...	non-suicide
329342	Passively SuicidalI feel suicidal all the time...	suicide
197394	I wanna die but there’s so much I haven’t done...	suicide
31588	Trigger warning ⚠️ So I read a post on r/relat...	non-suicide
121402	I'm just tired and it's not worth itI feel lik...	suicide
67135	So I have covid and I'm stuck in my room for a...	non-suicide
33987	I hate my birthday. My life is looking darker ...	suicide
339803	I'm extremely close to suicide, and I could RE...	suicide
67240	I don't see a futureI've struggled for many ye...	suicide

```
141654          I cut myself and sent it too a group chatI am ...       suicide
124586      My ex is going out with me BOIS, but I have a ...   non-suicide
 67736   Death grips ⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀   non-suicide
109589        But it hurts like hellHello.\nI'd like to star...       suicide
228153      guys i am. too edgy fuck edginess it fucking s...   non-suicide
245384      A question for Americans What the fuck is goin...   non-suicide
255442      Does anyone not want to get better?I have good...       suicide
  4015      Posts about how bad posts that say “unpopular ...   non-suicide
174069      Started watching X-Files and that shit 😩😩😩 X-F...   non-suicide
 66167      Anyone need a video editor? I will edit videos...   non-suicide
 12738        plague5467 appreciation post Please I need this   non-suicide
117336      u/MossIsUsuallyGreen appreciation post u are p...   non-suicide
322958   OH MY GOD IN ABOUT TO DIE JESUS CHRIST AT THE ...      non-suicide
 74013               start a bruh chain ill go sleep now  ...   non-suicide
 99974      Day 64 of posting lines from the Bee Movie eve...   non-suicide
219671      Shit I forgot to give reddit my email, and ive...   non-suicide
297484      I blew up at my abusive parents after they cal...   non-suicide
342310      I wish I had a girlfriend But I don’t have any...   non-suicide
273746      Every since my dad passed I want to join him.L...       suicide
 74693        I feel stuckIdk if I want to feel better. Ther...       suicide
```

```
#Información del DataFrame
df_suicide.describe()
```

	text	class
count	195700	195700
unique	195700	2
top	i want to destroy myselffor once everything wa...	non-suicide

```
df_suicide.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 195700 entries, 173271 to 305170
Data columns (total 2 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   text    195700 non-null  object
 1   class   195700 non-null  object
dtypes: object(2)
memory usage: 4.5+ MB
```

2. Preprocesamiento y preparación de datos

```
#Conversión a numérico de la etiqueta

def class_sui(text):
  if text == 'suicide':
    return 1
  else:
    return 0

df_suicide['class'] = df_suicide['class'].apply(class_sui)
df_suicide
```

Unnamed: 0	text	class
173271	i want to destroy myselffor once everything wa...	1
336321	I kinda got behind schedule with learning for ...	0
256637	I'm just not sure anymoreFirst and foremost: I...	1
303772	please give me a reason to liveThats too much ...	1
293747	27f struggling to find meaning moving forwardI...	1
...	...	...
248038	Drop some cool new cereal ideas Like what woul...	0
216516	Unpopular opinion but cats deserve love and re...	0
199341	Hey guys :) How yall doin?	0
145373	uhm I covered my dog in a blanket because the	0

```
#Borramos duplicados
df_suicide.drop_duplicates(inplace = True)
df_suicide
```

Unnamed: 0	text	class
173271	i want to destroy myselffor once everything wa...	1
336321	I kinda got behind schedule with learning for ...	0
256637	I'm just not sure anymoreFirst and foremost: I...	1

```
#Pasamos a minuscula
def minuscula(text):
  texto =[]
  for word in text:
    textos= word.lower()
    texto.append(textos)
  listToStr = ''.join([str(elem) for elem in texto])



  return listToStr

df_suicide['text'] = df_suicide['text'].apply(minuscula)
```

```
#Eliminamos los números
import inflect
def numeros(text):
  x=inflect.engine()
  texto = []
  for word in text :
      word.replace(" ","")
      if word.isdigit():
         textos=x.number_to_words(word)
         texto.append(textos)
      else:
         texto.append(word)
```

```
  return texto
  df_suicide['text'] = df_suicide['text'].apply(numeros)
```

```
a=["126"]
print(numeros(a))
```

```
['one hundred and twenty-six']
```

```
df_suicide.head(10)
```

Unnamed: 0	text	class
173271	i want to destroy myselffor once everything wa...	1
336321	i kinda got behind schedule with learning for ...	0
256637	i'm just not sure anymorefirst and foremost: i...	1
303772	please give me a reason to livethats too much ...	1
293747	27f struggling to find meaning moving forwardi...	1
205651	let’s get this bread 😎 anyone know any good ba...	0
97174	day 126 of posting random "fun" facts everyday...	0
195945	little brother is self mutilating. please help...	1
305273	why do women always go in groups to their wash...	0
69929	did you guys know that there's no school for g...	0

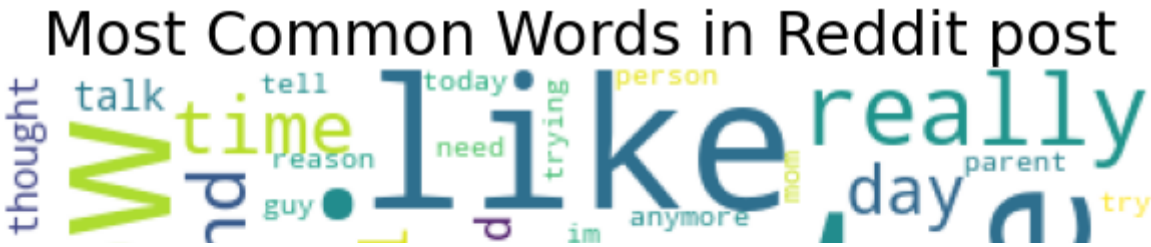
```
#Se generan los token para las palabras
tokenizer = RegexpTokenizer(r'\w+')
lemmatizer = WordNetLemmatizer()
list_token = []
for text in df_suicide['text']:
    # tokenize it
    result = []
```

```
    results = tokenizer.tokenize(text)
    for word in results:
        # lemmatize it
        words = lemmatizer.lemmatize(word)
        result.append(words)
    list_token.append(result)


#Lista de stopwords
count_vec = CountVectorizer(input='content', stop_words='english')
stopw = set(count_vec.get_stop_words())


#Eliminamos artículos, conjunciones, preposiciones, etc

for a in range(0,len(list_token)-1):
  borrables = []
  for b in range(0,len(list_token[a])-1):
    if list_token[a][b] == 'm' or list_token[a][b] == 's' or list_token[a][b] == 've' or list_token[a][b] == 'don'
      borrables.append(list_token[a][b])
  for c in borrables:
    list_token[a].remove(c)


lst = []
for i in list_token:
    listToStr = ' '.join([str(elem) for elem in i])
    lst.append(listToStr)
lst[:2]
```

```
['want destroy myselffor wa starting feel okay came tumbling know used cope reason tearing skin shred
swallowing pill right room wall slowly falling matter time snap finally end all',
 'kinda got schedule learning week testweek 8 test 4 ive studied 2 studied good 2 minimal 4 didnt 3 day
option pull 3 nighters dont tell parent tell freak possible super hard']
```

```
#Mostramos las palabras que más se repiten en el data set
word = ''
for i in lst[0:1000]:
```

```
    # typecaste each val to string
    i = str(i)

    # split the value
    tokenst = i.split()

    # Converts each token into lowercase

    for words in tokenst:
        word = word + words + ' '


wordcloudt = WordCloud(
                background_color ='white',
                stopwords = stopw,
                min_font_size = 10).generate(word)

# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
plt.imshow(wordcloudt)
plt.axis("off")
plt.tight_layout(pad = 0)
plt.title("Most Common Words in Reddit post", fontsize=30)


plt.show()
```




```
#Añadimos los datos limpios
df_suicide['final_text'] = lst
```

```
df_suicide.head(50)
```

Unnamed: 0	text	class	final_text
173271	i want to destroy myselffor once everything wa...	1	want destroy myselffor wa starting feel okay c...
336321	i kinda got behind schedule with learning for ...	0	kinda got schedule learning week testweek 8 te...
256637	i'm just not sure anymorefirst and foremost: i...	1	just sure anymorefirst foremost brazil judge s...
303772	please give me a reason to livethats too much ...	1	reason livethats dont reason live like anymore...
293747	27f struggling to find meaning moving forwardi...	1	27f struggling meaning moving forwardi admit b...
205651	let’s get this bread 😎 anyone know any good ba...	0	let bread know good bakery store
97174	day 126 of posting random "fun" facts everyday...	0	day 126 posting random fun fact everyday forge...
195945	little brother is self mutilating. please help...	1	little brother self mutilating help brother 15...
305273	why do women always go in groups to their wash...	0	woman group washroom hey guy wa watching coupl...
69929	did you guys know that there's no school for g...	0	did guy know school gay pride month ahhhh ahhh...
111327	was about to post something... but forgot it w...	0	wa post forgot wa weekend stupid crap wa damn ...
341361	ah shite i said suck my clit instead of suck m...	0	ah shite said suck clit instead suck cock frie...
86906	if you hate coffee but need the caffeine try t...	0	hate coffee need caffeine try starbucks grande...
281142	general kenobi, hello there first one to comme...	0	general kenobi hello comment doe award gotten ...
329342	passively suicidali feel suicidal all the time...	1	passively suicidali feel suicidal time know so...
197394	i wanna die but there’s so much i haven’t done...	1	wanna die haven yeti seriously wanna end readi...
31588	trigger warning ⚠️ so i read a post on r/relat...	0	trigger warning read post r relationship_advic...
121402	i'm just tired and it's not worth iti feel lik...	1	just tired worth iti feel like sleeping time t...
67135	so i have covid and i'm stuck in my room for a...	0	covid stuck room week bit aaaaaaaaaaaaaaaaaaaa...
33987	i hate my birthday. my life is looking darker ...	1	hate birthday life looking darker year bythis ...
339803	i'm extremely close to suicide, and i could re...	1	extremely close suicide really use advice help...
67240	i don't see a futurei've struggled for many ye...	1	futurei struggled year happiness life ha happe...

141654	i cut myself and sent it too a group chati am ...	1	cut sent group chati depressed cut took pictur...
124586	my ex is going out with me bois, but i have a ...	0	ex going bois mouth leave friend zone prove go...
67736	death grips ⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀	0	death grip
109589	but it hurts like hellhello.\ni'd like to star...	1	hurt like hellhello like start saying think ac...
228153	guys i am. too edgy fuck edginess it fucking s...	0	guy edgy fuck edginess fucking suck man cringe...
245384	a question for americans what the fuck is goin...	0	question american fuck going meme protest shit...
255442	does anyone not want to get better?i have good...	1	doe want better good day wish didn wanting mis...
4015	posts about how bad posts that say “unpopular ...	0	post bad post say unpopular opinion sexism bad...
174069	started watching x-files and that shit 😩 😩 😩 x-f...	0	started watching x file shit x file awesome ki...
66167	anyone need a video editor? i will edit videos...	0	need video editor edit video free need really ...
12738	plague5467 appreciation post please i need this	0	plague5467 appreciation post need this
117336	u/mossisusuallygreen appreciation post u are p...	0	u mossisusuallygreen appreciation post u pog
322958	oh my god in about to die jesus christ at the ...	0	oh god die jesus christ end day going school m...

3. Modelamiento

219671	shit i forgot to give reddit my email, and ive...	0	shit forgot reddit email ive lost account sob ...

Inspirado en: https://github.com/tw1270/Web-APIs-and-Predicting-Subreddit/blob/main/Reddit.ipynb (también parte del pre-procesamiento)

273746	every since my dad passed i want to join him l...	1	dad passed want join year december dad suddenl...

Modelamiento con Regresión Logistica con CountVectorizer

[illegible]	short advice from a boy, about boys boys are p...	0	short advice boy boy boy pretty dense simple i...

```
#Creamos una copia de los datos
df_suicide2 = df_suicide.copy()
X = df_suicide2['final_text']
```

```
y = df_suicide2['class']
#Sacamos los datos de entrenamiento y test
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify = y, random_state = 42)
```

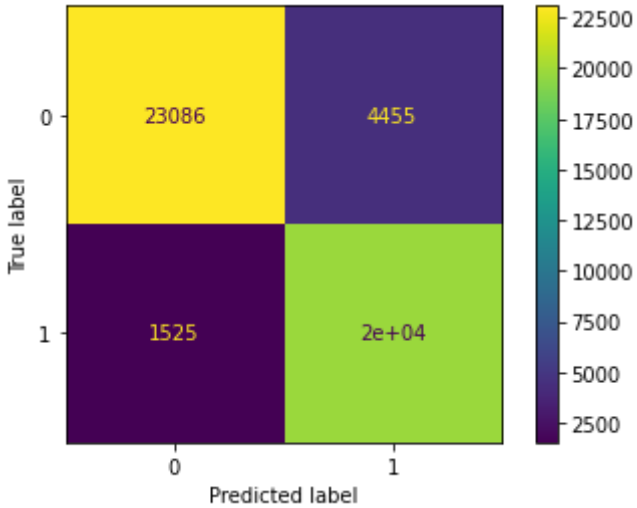
```
from nltk.tokenize import TweetTokenizer

#Usamos un TweetTokenizer para la tokenización del texto.
def tokeni(text):
  tt = TweetTokenizer()
  return tt.tokenize(text)


#Hacemos la matriz de apariciones
cv = CountVectorizer(tokenizer = tokeni, stop_words = stopw)
X_train = cv.fit_transform(X_train)
X_test = cv.transform(X_test)


#Ejecutamos el modelo de regresión logistica
lr = LogisticRegression(penalty='l2',
                        tol=0.00001,
                        C=1.0,
                        fit_intercept=True,
                        intercept_scaling=1,
                        class_weight='balanced',
                        random_state=1,
                        solver='saga',
                        max_iter=5000,
                        n_jobs=-1)
lr.fit(X_train, y_train)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_sag.py:354: ConvergenceWarning: The max_iter was
  ConvergenceWarning,
LogisticRegression(class_weight='balanced', max_iter=5000, n_jobs=-1,
                   random_state=1, solver='saga', tol=1e-05)
```

```
#Hacemos la predicción con los conjuntos de entrenamiento y test
y_train_lr_predict = lr.predict(X_train)
```

```
y_test_lr_predict = lr.predict(X_test)


#Imprimimos las métricas
print(f"LR Accuracy: {accuracy_score(y_test, y_test_lr_predict):.4%}")
print(f'LR Precision: {precision_score(y_test,y_test_lr_predict):.4%}')
print(f'LR Recall: {recall_score(y_test,y_test_lr_predict):.4%}')
print(f'LR F1 Score: {f1_score(y_test, y_test_lr_predict):.4%}')
ConfusionMatrixDisplay.from_predictions(y_test, y_test_lr_predict)
```

```
LR Accuracy: 87.7772%
LR Precision: 81.6772%
LR Recall: 92.8685%
LR F1 Score: 86.9141%
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7fa83eff2450>
```



Modelamiento con SMV y Naibe Bayes

Inspirado en: https://medium.com/@bedigunjit/simple-guide-to-text-classification-nlp-using-svm-and-naive-bayes-with-python-421db3a72d34

```
#Divifimos los datos en entrenamiento y prueba
Train_X, Test_X, Train_Y, Test_Y = model_selection.train_test_split(df_suicide['final_text'],df_suicide['class'],t
```

```
#Se vectorizan las palabras
Tfidf_vect = TfidfVectorizer(max_features=5000)
Tfidf_vect.fit(df_suicide['final_text'])
Train_X_Tfidf = Tfidf_vect.transform(Train_X)
Test_X_Tfidf = Tfidf_vect.transform(Test_X)
```

```
#Vemos el vocabulario
print(Tfidf_vect.vocabulary_)
```

```
{'want': 4810, 'destroy': 1253, 'wa': 4796, 'starting': 4204, 'feel': 1715, 'okay': 3062, 'came': 725, 'know'
```

```
#Vemos la información vectorizada
print(Train_X_Tfidf)
```

```
  (0, 4971)     0.15789190393087366
  (0, 4968)     0.12108781558886988
  (0, 4932)     0.05512505140743203
  (0, 4921)     0.08116153706346857
  (0, 4811)     0.05452334555488876
  (0, 4796)     0.3393057205079025
  (0, 4720)     0.10419022668234496
  (0, 4715)     0.06557159707644827
  (0, 4620)     0.07440055310003953
  (0, 4505)     0.03652410665941144
  (0, 4477)     0.04519778453867124
  (0, 4471)     0.05556894916367853
  (0, 4469)     0.03939649173353486
  (0, 4446)     0.09706943705694931
  (0, 4444)     0.09377269420696342
  (0, 4420)     0.10018404541092622
  (0, 4392)     0.05967736593701985
  (0, 4391)     0.07196657939048232
  (0, 4390)     0.09517490518650859
  (0, 4372)     0.09768439762557464
  (0, 4305)     0.09670475741145555
  (0, 4255)     0.09750321786603043
  (0, 4202)     0.05626633300942276
```

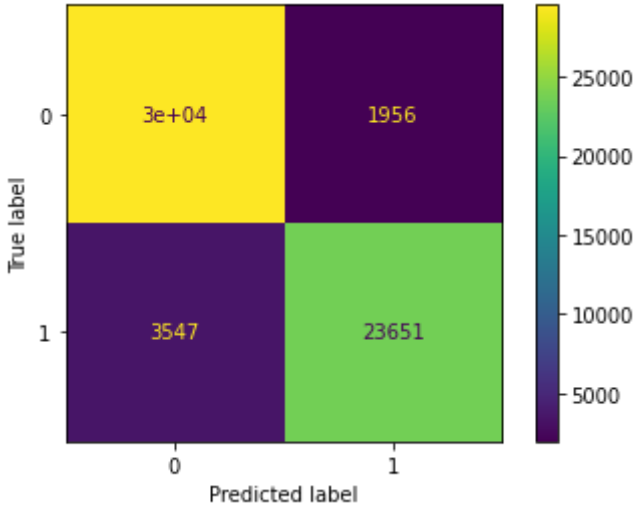
```
(0, 4201)     0.11742188126874643
(0, 4154)     0.07368519769601616
:     :
(136989, 2430)        0.08939135643092938
(136989, 2253)        0.09452846976384015
(136989, 2199)        0.12513607668396168
(136989, 2130)        0.1548519461033701
(136989, 2127)        0.12939593004688751
(136989, 2069)        0.08180994736293477
(136989, 1996)        0.15590953038402053
(136989, 1939)        0.2884961769992133
(136989, 1938)        0.07929397840046744
(136989, 1857)        0.12577086558928122
(136989, 1720)        0.0871634202811577
(136989, 1716)        0.07471038681271407
(136989, 1462)        0.1124947926371009
(136989, 1420)        0.15062924442501247
(136989, 1417)        0.12201862605441316
(136989, 1279)        0.07642444145283464
(136989, 1166)        0.059459499765885195
(136989, 926) 0.11392219793820954
(136989, 711) 0.117638085650697
(136989, 662) 0.10495953690377209
(136989, 642) 0.10292466670138688
(136989, 436) 0.07935984642708853
(136989, 405) 0.12081455708543169
(136989, 310) 0.12926105272376456
(136989, 230) 0.16318362867616393
```

```
#Aplicamos Naive Bayes
# Se hace fit del entrenamiento en el clasificador de NB
Naive = naive_bayes.MultinomialNB()
Naive.fit(Train_X_Tfidf,Train_Y)
# Predicer las etiquetas
predictions_NB = Naive.predict(Test_X_Tfidf)
# Se obtiene la precisión
print("Naive Bayes Accuracy Score -> ",accuracy_score(predictions_NB, Test_Y)*100)
```

```
Naive Bayes Accuracy Score ->  90.62680974280362
```

```
from sklearn.metrics._plot.confusion_matrix import ConfusionMatrixDisplay
#Imprimimos las métricas
print("NB Accuracy Score -> ",accuracy_score(predictions_NB, Test_Y)*100)
print("NB Precision Score -> ", precision_score(predictions_NB, Test_Y)*100)
print("NB Recall Score -> ", recall_score(predictions_NB, Test_Y)*100)
print("NB f1 Score -> ",f1_score(predictions_NB, Test_Y)*100)
ConfusionMatrixDisplay.from_predictions(predictions_NB, Test_Y)
```

```
NB Accuracy Score ->  90.62680974280362
NB Precision Score ->  92.36146366227985
NB Recall Score ->  86.95859989705126
NB f1 Score ->  89.57863838651642
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7f5f32ff12d0>
```

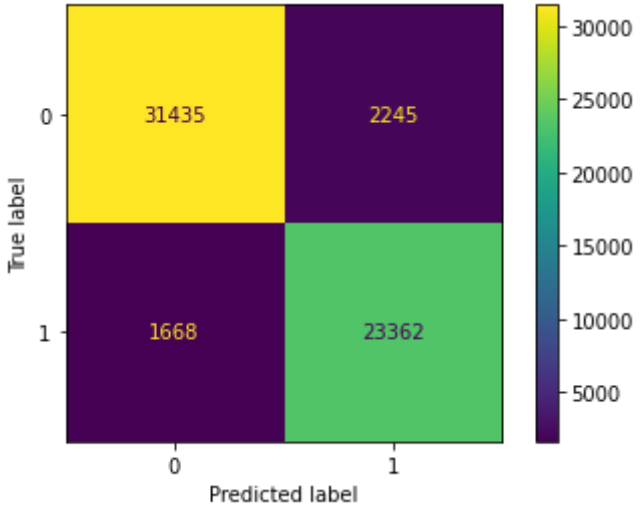



```
# Aplicamos SVM
# Se hace fit del entrenamiento en el clasificador de SMV
SVM = svm.SVC(C=1.0, kernel='linear', degree=3, gamma='auto')
SVM.fit(Train_X_Tfidf,Train_Y)
# Predicer las etiquetas
predictions_SVM = SVM.predict(Test_X_Tfidf)
# Se obtiene la precisión
print("SVM Accuracy Score -> ",accuracy_score(predictions_SVM, Test_Y)*100)
```

```
SVM Accuracy Score ->  93.33503662067791
```

```
#Imprimimos las métricas
print("SVM Accuracy Score -> ",accuracy_score(predictions_SVM, Test_Y)*100)
print("SVM Precision Score -> ", precision_score(predictions_SVM, Test_Y)*100)
print("SVM Recall Score -> ", recall_score(predictions_SVM, Test_Y)*100)
print("SVM f1 Score -> ",f1_score(predictions_SVM, Test_Y)*100)
ConfusionMatrixDisplay.from_predictions(predictions_SVM, Test_Y)
```

```
SVM Accuracy Score ->  93.33503662067791
SVM Precision Score ->  91.23286601319953
SVM Recall Score ->  93.3359968038354
SVM f1 Score ->  92.27244899974328
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7f5f3306a290>
```



✓ 0 s se ejecutó 08:14

No fue posible conectarse al servicio de reCAPTCHA. Comprueba tu conexión a Internet y vuelve a cargar la página para obtener un desafío de reCAPTCHA.