```
In [21]: import pandas as pd
         data = pd.read_csv("accident_news_v2.csv", encoding="latin1",delimiter = ";")
         df = data.copy()
In [22]: df.head()
Out[22]:
                                                                                                              Vehicles Province_Code otomobil motosiklet kamyonet ... scooter at Year Month Day Hour Minute
                                                 URL
                                                           Content
                                                                          Date License_Plate Location
                                                                                                                                                                                                                                       Location
                                                          Aydýn'da 2
                                                          otomobilin
             https://www.aydinhedef.com.tr/aydinda-feci-
                                                                      1.02.2019
                                                                                   ('09 DL 861',
                                                                                                                                                                        0 ...
                                                                                                                              ['09', '35']
                                                                                                                                                                                                0 2019
                                                                                                                                                                                                              2
                                                                                                                                                                                                                 1
                                                                                                                                                                                                                        13
                                                                                                                                                                                                                                           Söke
                                                           çarptýðý
                                                                                                   (None,) ['otomobil']
                                                                                    '35 DKY 66')
                                                                          13:08
                                                           yaya aðýr
                                                             yaral...
                                                           Aydýn'ýn
                                                               Söke
                 https://www.aydinhedef.com.tr/aydinda-
                                                                      1.02.2019
                                                                                                            ['kamyonet',
                                                                                                                                                                        1 ...
                                                                                  ('09 SU 913',)
                                                                                                                                  ['09']
                                                           ilçesinde
                                                                                                      Söke
                                                                                                                                                             0
                                                                                                                                                                                      0
                                                                                                                                                                                                0 2019
                                                                                                                                                                                                              2 1 15
                                                                                                                                                                                                                                  27 Germencik
                                            trafik-k...
                                                                          15:27
                                                                                                              'kamyon']
                                                         kamyonetin
                                                        devrilmesi ...
                                                           Aydýn'ýn
                                                                                                             ['otomobil',
                                                          Germencik
                                                                      8.02.2019
                 https://www.aydinhedef.com.tr/aydinda-
                                                                                   ('07 FSZ 99',
                                                                                                                                                                        0 ...
         2
                                                                                                Germencik
                                                                                                              'kamyon',
                                                                                                                              ['07', '09']
                                                                                                                                                             0
                                                                                                                                                                                                0 2019
                                                                                                                                                                                                              2
                                                                                                                                                                                                                   8
                                                                                                                                                                                                                         10
                                                                                                                                                                                                                                          Nazilli
                                                          ilçesindeki
                                                                          10:02
                                                                                    '09 K 6439')
                                                              trafik
                                                                                                                   'týr']
                                                           kazasýn...
                                                       (MEHMET ALÝ
         a https://www.aydinhedef.com.tr/sultanhisarda-
                                                         CÝNTOSUN) 16.02.2019
                                                                                   ('07 HOP 69',
                                                                                                             ['otomobil',
                                                                                                                                                             0
                                                                                                                                                                        0 ...
                                                                                                                                                                                                                                  57
                                                                                                     Nazilli
                                                                                                                              ['07', '09']
                                                                                                                                                                                      0
                                                                                                                                                                                                0 2019
                                                                                                                                                                                                              2 16
                                                                                                                                                                                                                          9
                                                                                                                                                                                                                                          Nazilli
                                                                                   '09 CED 35')
                                                  fe... Alýnan bilgiye
                                                                          09:57
                                                                                                                   'týr']
                                                         göre, Cen...
                                                               Kaza
                                                       Yukarýörencik
               https://www.aydinhedef.com.tr/nazillide-is-
                                                                     16.02.2019
                                                       Mahallesi' nde
                                                                                                                                      []
                                                                                                                                                                        0 ...
                                                                                                                                                             0
                                                                                                                                                                                                0 2019
                                                                                                     Nazilli
                                                                                                                                                                                                              2
                                                                                                                                                                                                                 16
                                                                                                                                                                                                                                            Çine
                                                                          17:04
                                                              11:00
                                                            sýralar...
         5 \text{ rows} \times 25 \text{ columns}
In [23]: df.drop(columns = ["URL" , "Date","License_Plate","Province_Code","Hour","Minute","Vehicles"],inplace = True)
In [24]: def extract_province_codes(text):
             codes = []
             plate = re.findall(r'\b\d{2}\s[A-Z]{1,3}\s\d{3}\b', text)
             plate2 = re.findall(r'\b\d{2}\s?[A-Z]+\s?\d{4}\b', text)
             plate3 = re.findall(r'\b\d{2}\s?[A-Z]+\s?\d{2}\b', text)
             for p in plate + plate2 + plate3:
                 code = p.split()[0][:2]
                 codes.append(code)
             return codes
         df['Province_Codes'] = df['Content'].apply(extract_province_codes)
         for code in set(df['Province_Codes'].explode()):
             df[code] = df['Province_Codes'].apply(lambda x: 1 if code in x else 0)
         df_concatenated = df.drop(columns=['Province_Codes'])
In [25]: df.columns
Out[25]: Index([
                        'Content',
                                        'Location',
                                                          'otomobil',
                                                                         'motosiklet',
                       'kamyonet'
                                         'minibüs',
                                                           'kamyon',
                                                                                'týr'
                        'traktör'
                                        'bisiklet',
                                                           'scooter',
                                                                         'at arabasý',
                          'Year',
                                          'Month',
                                                              'Day', 'Next Location',
                         'Death'
                                         'Injured', 'Province_Codes',
                            '15',
                                              '19',
                                                                '08',
                                                                                 nan,
                             '05',
                                              '51',
                                                                '06',
                                                                                 '41',
                            '27',
                                              '54',
                                                                                 '32',
                                                               '63',
                            '49'
                                              '60',
                                                               '77',
                                                                                 '46',
                            '81',
                                              '70',
                                                               '18',
                                                                                 '59',
                             '01',
                                              '80',
                                                                '07',
                                                                                 '68',
                             '64',
                                              '20',
                                                               '53',
                                                                                 '67',
                            '43',
                                              '17',
                                                               '25',
                                                                                 '66',
                            '42',
                                              '58',
                                                               '26',
                                                                                 '00',
                            '50',
                                              '28',
                                                               '38',
                                                                                 '34',
                            '35',
                                              '33',
                                                               '22',
                                                                                 '40',
                            '45'
                                              '44',
                                                               '03',
                                                                                 '31',
                            '09'
                                              '55',
                                                               '21',
                                                                                 '23',
                            '39'
                                              '16',
                                                               '10',
                                                                                 '47',
                            '48'
                                              '65'],
               dtype='object')
In [27]: # df.to_csv("deneme_accident.csv" , index = False)
In [46]: from wordcloud import WordCloud
         import matplotlib.pyplot as plt
         lyrics_text = ' '.join(df['Lyrics'])
         wordcloud = WordCloud(width=800, height=400, background_color='white').generate(lyrics_text)
         plt.figure(figsize=(10, 5))
         plt.imshow(wordcloud, interpolation='bilinear')
         plt.axis('off')
         plt.show()
            Kuþadasý Germencik
                                                                                         Didim
                                        Bozdoðan
                        <u>Ýncirli</u>ova
               Karpuzlu
          Kuyucak
 In [2]: import pandas as pd
         df = pd.read_csv("deneme_accident.csv")
 In [5]: from sklearn.ensemble import GradientBoostingClassifier, RandomForestClassifier
         from sklearn.datasets import make_classification
         from sklearn.model_selection import train_test_split
         from sklearn.metrics import accuracy_score
         from sklearn.tree import DecisionTreeClassifier
         from catboost import CatBoostClassifier
         X, y = df.drop(columns = ["Next Location" , "Death", "Injured", "Province_Codes", "Content", "Location"]), df["Injured"]
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=47)
         gb_model = GradientBoostingClassifier()
         gb_model.fit(X_train, y_train)
         rf_model = RandomForestClassifier()
         rf_model.fit(X_train, y_train)
         y_pred_gb = gb_model.predict(X_test)
         accuracy_gb = accuracy_score(y_test, y_pred_gb)
         print("Gradient Boosting Test Accuracy:", accuracy_gb)
         y_pred_gb_train = gb_model.predict(X_train)
         accuracy_gb_train = accuracy_score(y_train, y_pred_gb_train)
         print("Gradient Boosting Train Accuracy:", accuracy_gb_train)
         print("----")
         y_pred_rf = rf_model.predict(X_test)
         accuracy_rf = accuracy_score(y_test, y_pred_rf)
         print("Random Forest Test Accuracy:", accuracy_rf)
         y_pred_rf_train = rf_model.predict(X_train)
         accuracy_rf_train = accuracy_score(y_train, y_pred_rf_train)
         print("Random Forest Train Accuracy:", accuracy_rf_train)
         tree_model = DecisionTreeClassifier()
         tree_model.fit(X_train, y_train)
         print("----")
         y_pred = tree_model.predict(X_test)
         accuracy = accuracy_score(y_test, y_pred)
         print("Decision Tree Test Accuracy:", accuracy)
         tree_model.fit(X_train, y_train)
         y_pred_train = tree_model.predict(X_train)
         accuracy_train = accuracy_score(y_train, y_pred_train)
         print("Decision Tree Train Accuracy:", accuracy_train)
        Gradient Boosting Test Accuracy: 0.6529968454258676
        Gradient Boosting Train Accuracy: 0.7033227848101266
        Random Forest Test Accuracy: 0.6214511041009464
        Random Forest Train Accuracy: 0.9944620253164557
        -----
        Decision Tree Test Accuracy: 0.5425867507886435
        Decision Tree Train Accuracy: 0.9944620253164557
 In [6]: X, y = df.drop(columns = ["Next Location" , "Death", "Injured", "Province_Codes", "Content", "Location"]), df["Death"]
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=47)
         b_model = GradientBoostingClassifier()
         gb_model.fit(X_train, y_train)
         rf_model = RandomForestClassifier()
         rf_model.fit(X_train, y_train)
         y_pred_gb = gb_model.predict(X_test)
         accuracy_gb = accuracy_score(y_test, y_pred_gb)
         print("Gradient Boosting Test Accuracy:", accuracy_gb)
         y_pred_gb_train = gb_model.predict(X_train)
         accuracy_gb_train = accuracy_score(y_train, y_pred_gb_train)
         print("Gradient Boosting Train Accuracy:", accuracy_gb_train)
         print("----")
         y_pred_rf = rf_model.predict(X_test)
         accuracy_rf = accuracy_score(y_test, y_pred_rf)
         print("Random Forest Test Accuracy:", accuracy_rf)
         y_pred_rf_train = rf_model.predict(X_train)
         accuracy_rf_train = accuracy_score(y_train, y_pred_rf_train)
         print("Random Forest Train Accuracy:", accuracy_rf_train)
         tree_model = DecisionTreeClassifier()
         tree_model.fit(X_train, y_train)
         print("----")
         y_pred = tree_model.predict(X_test)
         accuracy = accuracy_score(y_test, y_pred)
         print("Decision Tree Test Accuracy:", accuracy)
         tree_model.fit(X_train, y_train)
         y_pred_train = tree_model.predict(X_train)
         accuracy_train = accuracy_score(y_train, y_pred_train)
         print("Decision Tree Train Accuracy:", accuracy_train)
        Gradient Boosting Test Accuracy: 0.7539432176656151
        Gradient Boosting Train Accuracy: 0.8283227848101266
        Random Forest Test Accuracy: 0.7413249211356467
        Random Forest Train Accuracy: 0.997626582278481
        Decision Tree Test Accuracy: 0.6466876971608833
        Decision Tree Train Accuracy: 0.9984177215189873
 In [8]: df = df.dropna()
         X, y = df.drop(columns = ["Next Location" , "Death", "Injured", "Province_Codes", "Content", "Location"]), df["Next Location"]
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=47)
         b_model = GradientBoostingClassifier()
         gb_model.fit(X_train, y_train)
         rf_model = RandomForestClassifier()
         rf_model.fit(X_train, y_train)
         y_pred_gb = gb_model.predict(X_test)
         accuracy_gb = accuracy_score(y_test, y_pred_gb)
         print("Gradient Boosting Test Accuracy:", accuracy_gb)
         y_pred_gb_train = gb_model.predict(X_train)
         accuracy_gb_train = accuracy_score(y_train, y_pred_gb_train)
         print("Gradient Boosting Train Accuracy:", accuracy_gb_train)
         print("----")
         y_pred_rf = rf_model.predict(X_test)
         accuracy_rf = accuracy_score(y_test, y_pred_rf)
         print("Random Forest Test Accuracy:", accuracy_rf)
         y_pred_rf_train = rf_model.predict(X_train)
```

accuracy\_rf\_train = accuracy\_score(y\_train, y\_pred\_rf\_train)
print("Random Forest Train Accuracy:", accuracy\_rf\_train)

tree\_model = DecisionTreeClassifier()
tree\_model.fit(X\_train, y\_train)
print("-----")
y\_pred = tree\_model.predict(X\_test)

tree\_model.fit(X\_train, y\_train)

-----

accuracy = accuracy\_score(y\_test, y\_pred)

y\_pred\_train = tree\_model.predict(X\_train)

print("Decision Tree Test Accuracy:", accuracy)

accuracy\_train = accuracy\_score(y\_train, y\_pred\_train)
print("Decision Tree Train Accuracy:", accuracy\_train)

Gradient Boosting Test Accuracy: 0.20253164556962025 Gradient Boosting Train Accuracy: 0.5886075949367089

In [1]: import re

Random Forest Test Accuracy: 0.20886075949367<mark>0</mark>9
Random Forest Train Accuracy: 0.9889240506329114

Decision Tree Test Accuracy: 0.1550632911392405
Decision Tree Train Accuracy: 0.9889240506329114