# DOĞAL DİL İŞLEME DERSİ PROJESİ

# **Spam mail tespiti**



Muhammed Ali SOYLU

# 16011131

04.02.2021

**Spam mail NEDİR?**

Kelime anlamı itibariyle “istenmeyen” anlamına gelen **spam mail**; ticari reklam amaçlı veya belli bir konuda kamuoyu oluşturmak amacıyla gönderilen toplu maillerdir. Spam mailler, farklı amaçlarla gönderilebilir. Bunların bazıları aşağıdadır.

• Ticari reklamlar ve pazarlama faaliyetleri

• Kampanya ve duyurular

• Yanıltıcı ve gerçek olmayan vaatler (Hızlı zengin olma yöntemleri vb.)

• Yasal olmayan veya yarı yasal faaliyet/servis duyuruları

• UBE (Unsolicited Bulk E-Mail: Talep Edilmemiş Kitlesel E-Posta)

• Politik veya ideolojik bir görüşün propagandası

• Belli bir konuda kamuoyu oluşturma

• Kimlik avı saldırısı

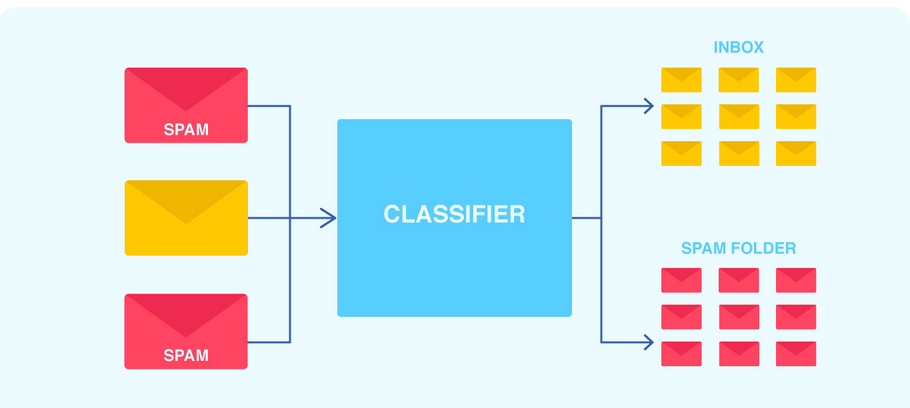
**Doğal Dil İşleme NEDİR?**

**Doğal Dil İşleme** insan dillerinin bilgisayarlar tarafından işlenmesidir. Bilinen en eski uygulama olarak spam mail tespit eden algoritmaları gösterebiliriz. Mailin konusu ve içeriğine bakarak gereksiz olup olmadığına karar verilir. Genelde Makine Öğrenmesi ilkeleri uygulanır.

**Doğal Dil İşleme ile Spam mailleri nasıl tespit edebiliriz?**

İnsan dillerinin bilgisayarlar tarafından işlenmesini kullanarak verilen bir Mail’in spam mı yoksa değil mi olduğuna karar vermenin yolu aslında temel olarak basittir.

Öncelikle makinemizi eğitmek için bir veri seti kullanmak zorundayız. Bu veri seti büyüklüğü ve doğru model kullanımı sayesinde, yeni maillerimizi test edebiliriz.

****

**KAYNAK KOD**

from google.colab import drive

import pandas as pd

# veri setinin yüklenmesi

drive.mount('/content/drive')

file\_url = '/content/drive/My Drive/Colab Notebooks/SMSSpamCollection.txt'

data = pd.read\_csv(file\_url, sep = '\t', header=None, names=["label", "sms"])

data

import string

import nltk

nltk.download('stopwords')

nltk.download('punkt')

stopwords = nltk.corpus.stopwords.words('english')

punctuation = string.punctuation

print(stopwords[:5])

print(punctuation)

def pre\_process(sms):

    remove\_punct = "".join([word.lower() for word in sms if word not in punctuation])

    tokenize = nltk.tokenize.word\_tokenize(remove\_punct)

    remove\_stopwords = [word for word in tokenize if word not in stopwords]

    return remove\_stopwords

#İslenen mesaj verileri ekleme

data['processed'] = data['sms'].apply(lambda x: pre\_process(x))

print(data['processed'].head())

def categorize\_words():

    spam\_words = []

    ham\_words = []

    #Spam olan mesajlari isleme

    for sms in data['processed'][data['label'] == 'spam']:

        for word in sms:

            spam\_words.append(word)

    #Ham olan mesajlari isleme

    for sms in data['processed'][data['label'] == 'ham']:

        for word in sms:

            ham\_words.append(word)

    return spam\_words, ham\_words

spam\_words, ham\_words = categorize\_words()

print(spam\_words[:5])

print(ham\_words[:5])

def predict(sms):

    spam\_counter = 0

    ham\_counter = 0

    #Veri setindeki spam ve ham mesajları sayarak, spam tespiti yapma

    for word in sms:

        spam\_counter += spam\_words.count(word)

        ham\_counter += ham\_words.count(word)

    print('\*\*\*RESULTS\*\*\*')

    #Eğer mesaj ham ise

    if ham\_counter > spam\_counter:

        accuracy = round((ham\_counter / (ham\_counter + spam\_counter) \* 100))

        print('messege is not spam, with {}% certainty'.format(accuracy))

    #Eğer spam ve ham sayisi yakin cikarsa

    elif ham\_counter == spam\_counter:

        print('message could be spam')

    #Eğer mesaj spam ise

    else:

        accuracy = round((spam\_counter / (ham\_counter + spam\_counter)\* 100))

        print('message is spam, with {}% certainty'.format(accuracy))

user\_input = input("Spam tespitinde kullanılacak test cümleyi giriniz\n")

#Spam tespitinden önce inputu ön isleme yapma

processed\_input = pre\_process(user\_input)

predict(processed\_input)

**KAYNAKÇA**

[**http://www.dt.fee.unicamp.br/~tiago/smsspamcollection/**](http://www.dt.fee.unicamp.br/~tiago/smsspamcollection/)

[**https://archive.ics.uci.edu/ml/datasets/sms+spam+collection**](https://archive.ics.uci.edu/ml/datasets/sms+spam+collection)