**KARAKTER N-GRAM TABANLI DOKÜMAN SINIFLANDIRMA**

Özdenur UÇAR ve Alperen ÇAKIROĞLU

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

[ozdenuru@gmail](mailto:ozdenuru@gmail), alperenckroglu204@gmail.com

**Özet**

*Hangi konu hakkında yazıldığı bilinmeyen haberleri bilgisayarların gücü ve yazılımların zekası ve matematiği kullanarak konularını bulabiliriz. Yazılan program tam anlamıyla konusunu bulmasa da kategorisini bulabilmektedir. Bunun için 2 ve 3 gramlarına bölünmüş haberler üzerinde naive bayes algoritmalarını çalıştırır. Programın haber kategorilerini tahmin etmeden önce eğitilmesi gerekmektedir. Bu işlem 1150 tane haberin yüzde yetmiş beşi ile yapılır. Geri kalan yüzde yirmi beşlik kısım programı test etmek için kullanılmıştır. Projede kullanılan algoritma naive bayes algoritmasıdır. Tahminler naive bayes algoritmasına göre yapılır.*

# Problem Tanımı

Bu projenin amacı 1150 haberden oluşan veri kümesinin n-gram tabanlı olarak Naive Bayes sınıflandırma yöntemi ile sınıflandırılmasıdır.

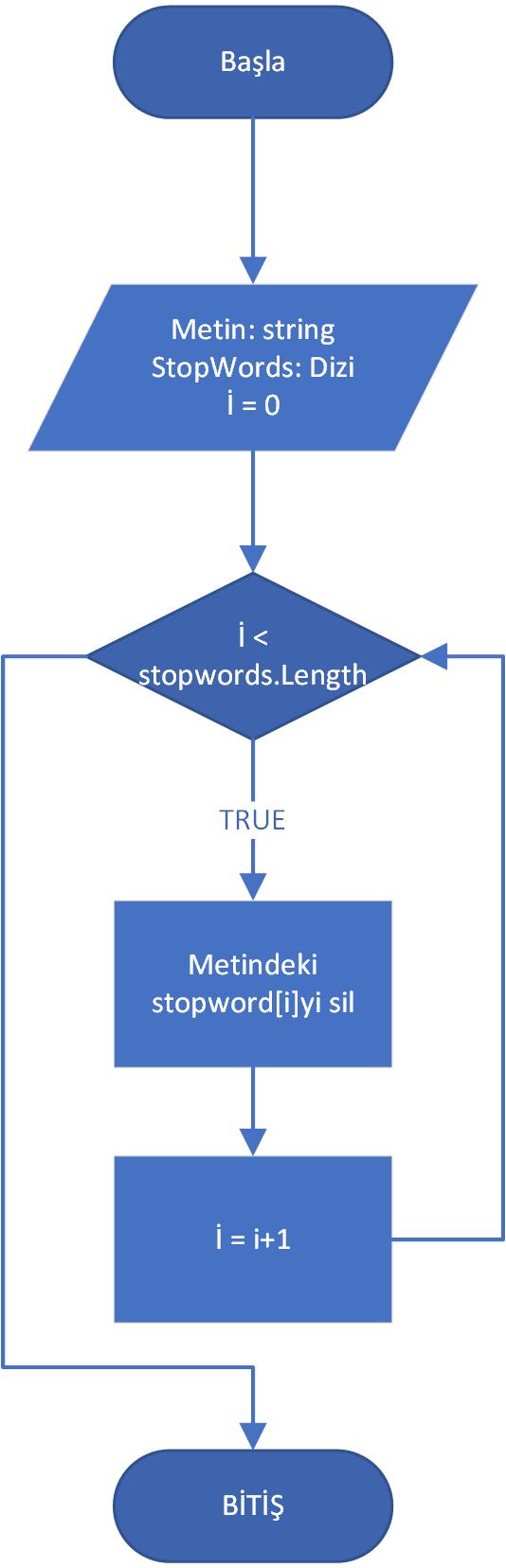
# Yapılan Araştırmalar

Proje C# programlama dili kullanılarak Visual Studio geliştirme ortalamında gerçeklenmiştir.Öncelikle projede n gram tabanlı Naïve Bayes sınıflandırma yöntemi ille sınıflandırma yapabilmek için, temel önişleme adımları uygulanmıştır. Önişleme adımlarından biri olan çekim eklerini ayırma kısmı için uygun bir programlama kütüphanesi araştırılmıştır. C# programlama dilinde kullanılabilen Nüve doğal dil işleme kütüphanesi kullanılmıştır. Naïve Bayes teoremini uygulayabilmek için birçok araştırma yapılmıştır.

# Tasarım

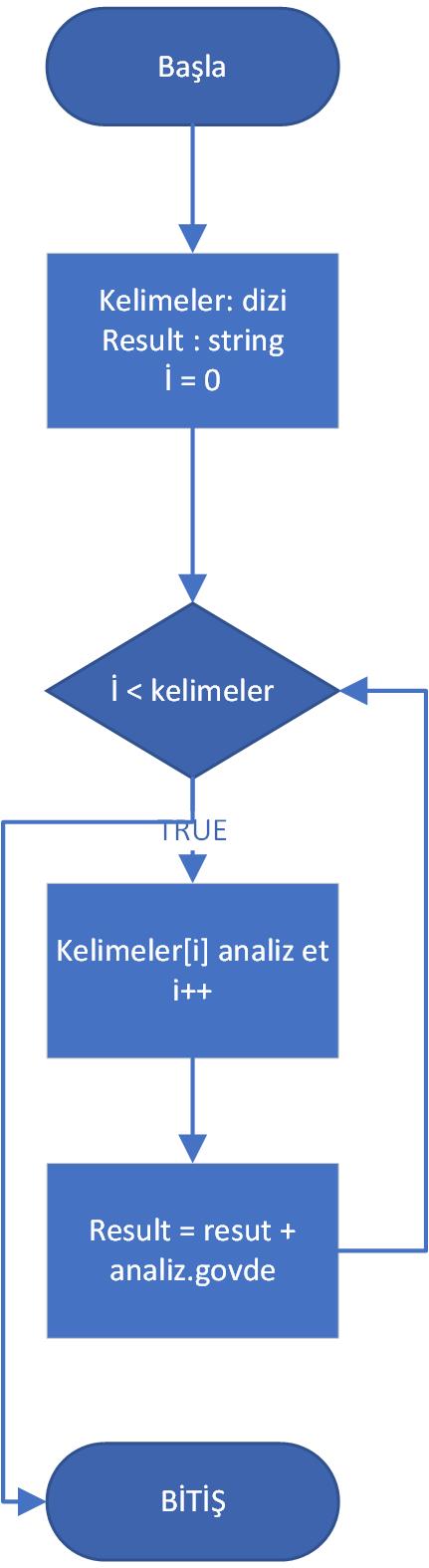
## Akış Şeması

## Kullanılan Fonksiyonlar Ve Veri Yapıları

* *private void btn\_choose\_Click(object sender, EventArgs e):* Program başlatıldıktan sonra kullanıcı arayüzünde yer alan döküman seç butonuna tıklandıktan sonra çalışacak olan ksıımdır. Bu kısımda kullanıcı kategorilerin ve içlerinde text dosyalarının bulunduğu ana klasörü seçer. Klasörün dosya yolu alındıktan sonra, alınan dosya yolu ile DataSet sınıfından bir nesne üretilerek dosya yolu constractor fonksiyonuna verilir. DataSet sınıfında sınıflandırma işlemleri için gerekli olan önişleme adımları, eğitim ve test kümesinin ayrılması gibi işlemler yapılır.Sonrasında ayrılan eğitim dosyaları ile oluşturulan NaïveBayes sınıfından eğitim ve test işlemleri için DataSet sınıfında ayrılan eğitim ve test dosyaları ayrılarak işlem yapılır. Dönen test sonuçlarına göre F-measure, Precision ve Recall değerleri hesaplanır.Hesaplanan değerler ve sonuçlar kullanıcı arayüzüne yazdırılır.
* *class News:* Bu sınıf içerisinde herbir text dosyasına ait, hangi kategoride olduğu, içerisinde yer alan haber ve o habere ait 2 gram ve 3 gram değerleri tutulmaktadır. Bu sınıf sayesinde tüm kategoriler okunduktan sonra okunan haber dosyalarının hangi kategoriden olduğu, o haber dosyasına ait n gramlar bir arada tutulmaktadır.
* *public News(string file, string categori):* News sınfının Constractor fonksiyonudur. New sınıfından nesne türetilirken bu parametrelerin verilmesi gerekmektedir.Parametre olarak dosya ve kategorisi sınıf içerisinde tutulmaktadır.
* *public void Prepare():* New sınıfı içerisinde yer alan bu fonksiyon ile bir haber dosyasına ait temel önişleme adımları yapılmaktadır. Bu adımlar, noktalama işaretlerinin silinmesi, özel unicode karakterlerinin temizlenmesi, stop wordlerin temizlenmesi, kelimelerin gövde haline getirilmesi yani çekim eklerinin temizlenmesi, tüm dosyanın küçük harflere çevrilmesi gibi işlemler yapılmıştır.
* *private string ClearTheStopWords(string text):*Parametre olarak gelen text string ifadesinin içerisinde yer alan, yani herhangi bir haber dosyasının içerisinde yer alan, türkçe stop wordlerin temizlenmesi işlemleri yapılmıştır. Stop wordler txt uzantılı bir dosyada tutulmuş ve dosya yolu program içerisinde belirtilmiştir. Bu fonksiyon içerisinde *Replace()* işlemi ile stop wordler temizlenmiştir.
* **

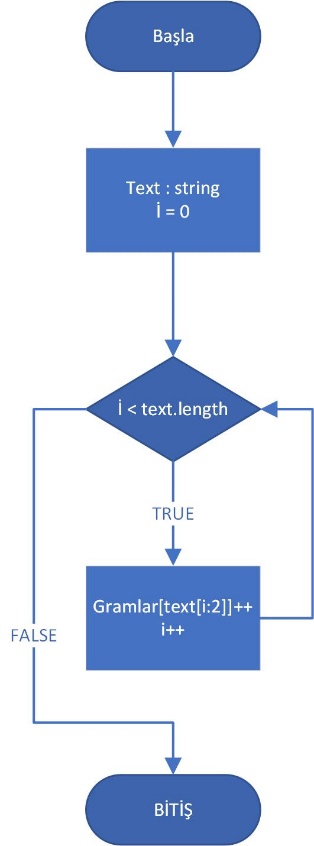
Akış Şeması 1 Clear The StopWords

* *private string DeleteSuffixes(string text):* Parametre olarak verilen text dosyasındaki kelimelerin çekim eklerini temizler, kelimeler gövde haline getirilerek geri döndürülür.

**

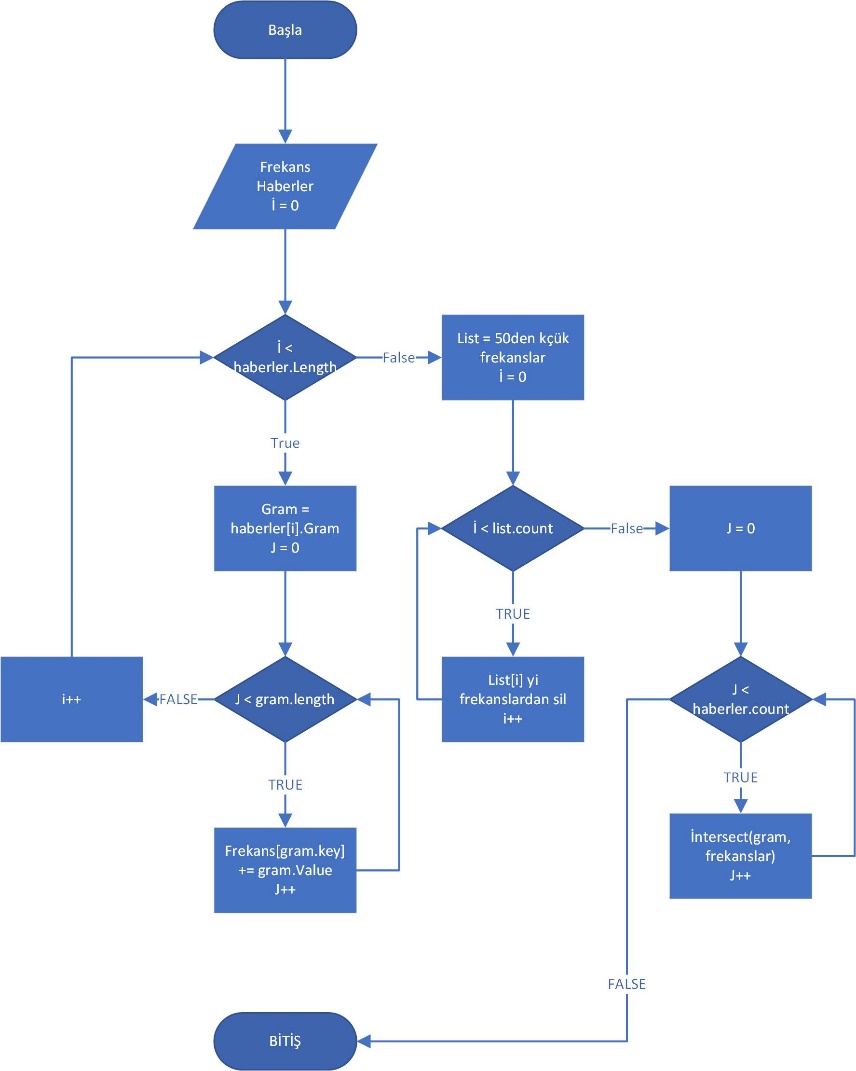
Akış Şeması 2 ÇekimEkleri

* *private Dictionary<string, double> SetGram2()*: News sınıfı içerisinde yer alan bu fonksiyon ile sınıf içerisinde tanımlı olan text üzerinde string ikili altstringlere bölünür. Ve ikili altstringin o text içerisinde ne kadar geçtiği tutulur. Fonksiyon geri dönüş olarak ikili altrstringleri ve geçme miktarlarını döndürmektedir.



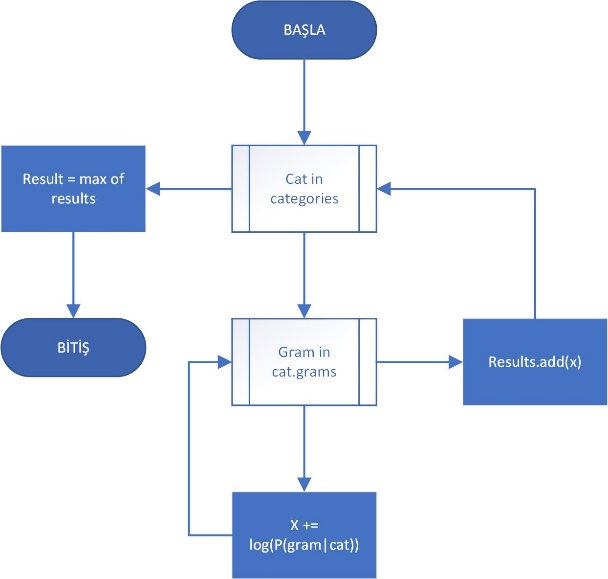
Akış Şeması 3 Gram

* *private Dictionary<string, double> SetGram3():* News sınıfı içerisinde yer alan bu fonksiyon ile sınıf içerisinde tanımlı olan text üzerinde string üçlü altstringlere bölünür. Ve üçlü altstringin o text içerisinde ne kadar geçtiği tutulur. Fonksiyon geri dönüş olarak üçlü altrstringleri ve geçme miktarlarını döndürmektedir.
* *private void SetGrams():* Bu fonksiyon içerisinde, daha önceden hesaplanmış olan 2 ve 3 gram değerleri birleştirilir.
* *class DataSet :* Kullanıcı kategorilerin ve kategorilere ait dökümanların dosya yolunu seçtikten sonra bu sınıf devreye girmektedir. Sınıf içerisinde üzerinde sınıflandırma yapılacak olan beş kategori için bir dizi ve *News*  ismindeki sınıftan eğitim ve test işlemlerine ait dökümanları tutmak için listler tanımlanmıştır.
* *public void ShuffleAndSplit(Dictionary<string, List<News>> dict):* DataSet sınıfı içerisinde yer alan bu fonksiyon içerisinde, parametre olarak gelen *Dictionary* yapısında kategori ve ktegoriye ait haber dosyaları gelmektedir. Projede Eğitim kümesi için tüm dökümanların %75’i, test kümesi için %25’i şeklinde ayrılması gerekmektedir.
* *class TurkishStopWords:* Sınıflandırma işleminde kullanılacak olan dökümanlarda bulunan stop word kelimelerinin yani Türkçede bağlaç, ek, zarf vs şekilde kullanılan kelimelerin temizlenmesi için bu sınıf kullanılmıştır.
* *void PrepareFiles():* Bu fonksiyon içerisinde eğitim ve test kümeleri için ayrılan dosyalar için *Prepare* fonksiyonu çağrılarak eğitim ve test kümeleri hazırlanmıştır. Fonksiyon içerisinde Parallel Foreach yapısı kullanılarak programın daha hızlı çalışması sağlanmıştır.İşlemci bir kerede tüm işi, iş parçacıklarına(threadlere) bölerek tek seferde optimize çalışır ve daha hızlı sonuç alınır.
* *class NaiveBayes:* Bu sınıf içerisinde Naïve Bayes teoremi kullanılarak sınıflandırma yapmak için belirli hesaplamalar yapılmaktadır.
* *public class Train:* Bu sınıf içerisinde News tipinde eğitim kümesine ait veriler bir liste halinde tutulmaktadır.Sınıf içerisinde sınıflandırma işlemi için gerekli olan eğitim işlemi yapılmaktadır.
* *public Train(List<News> trainingSet):* Train sınıfın constractor fonksiyonudur. Bu fonksiyon içerisinde, sınıf içerisinde yer alan eğitim kümesi, parametre olarak gelen eğitim kümesi olur. Sonrasında eğitim kümesinde her kategoriden ne kadar dosya olduğu saydırılır ve eğitim işlemi için eğitim kümesi hazırlanır ve Naïve Bayes işlemi için varyans ve ortalama değer hesaplanır.
* *void PrepareTrainingSet():[5] Train* sınfıında tanımlı olan bu fonksiyon içerisinde, eğitim kümesinde yer alan her bir dosya içerisinde yer alan döküman içerisindeki 2 ve 3 gramlar ve frekansları bulunur. Eğitim kümesinde yer alan herbir döküman için 2 ve 3 gramlar bulunduktan sonra tüm dökümanlarda geçme sayılarına bakılarak frekansı 50 den küçük olan 2 veya 3 gramlar eğitim kümesine dahil edilmemektedir. Bu işlemin kontrolü bu fonksiyon içerisinde yapılmaktadır. Kontrol yapılırken 50 den küçük olan 2 ve 3 gramlar *Frequencies* Dictionary yapısından silindiği için ilk eklenen gramlar ile 50 den büyük olan gramlar kümesinin kesişimi alınarak eğitim kümesinde kullanılacak olan very seti oluşturulmuş olur.

**

Akış Şeması 4 Frekans Ayarı

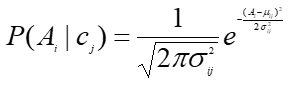
* *void SetVarianceAndMean():* MeanAndVariance içerisinde hangi kategorideki hangi gramların ne kadar geçtiğinin tutulduğu Dictionary yapısının değerleri ayarlanır. Bu fonksiyon içerisinde her bir kategoriye ait 2 ve 3 gramların geçme sayıları(frekansları) tutulmaktadır. Daha sonra herbir gram için hangi kategorideki ortalama ve varyans değeri hesaplanmaktadır.

**

Akış Şeması 5 Varyans ve Ortalama hesabı

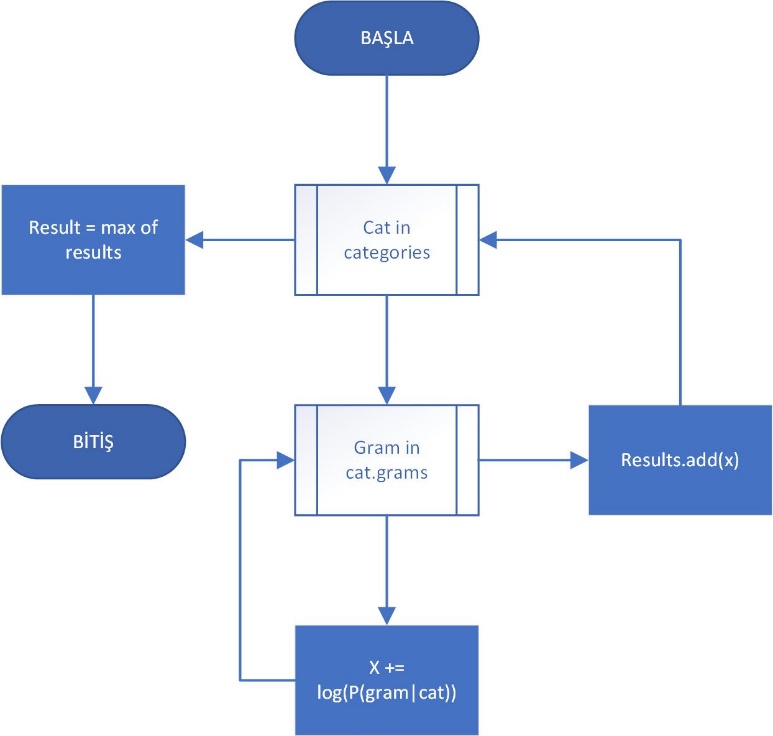
* *public class Test:* Eğitim ve Test işlemlerinden oluşan Naïve Bayes için Test işlemini yapmak için oluşturulmuştur. Sınıf içerisinde üzerinde test işlemi yapılacak olan eğitim kümesi ve kategoriler tutulmaktadır.
* *public Test(Train train, string[] categories):* Test Sınıfının constractor fonksiyonudur. Test sınfıından bir nesne çağrıldığında bu fonksiyon çalışmaktadır. Bu fonksiyon içerisinde paratmetre olarak gelen Tran sınıfına ait tran nesnesi ve kategoriler gelmektedir. Gelen bu değerleri sınıf içerisinde tutulmakta olan eğitim kümesine ve kategorilere verir.

* *public string Predict(News file):* Test sınıfı içerisind eyer alan bu sınıf içerisinde parametre olarak dosya üzerinde tahmini sınıflandırma işlemi yapılmaktadır. Bunun için öncelikle olasılık değerleri hesaplanmaktadır. Herbir kategori dolaşılarak, kategorilere ait eğitim kümesindeki herbir gramın ortalama ve varyans değerleri ile olasılık hesabı yapılmaktadır bunun için aşağıdaki denklen kullanılmıştır.

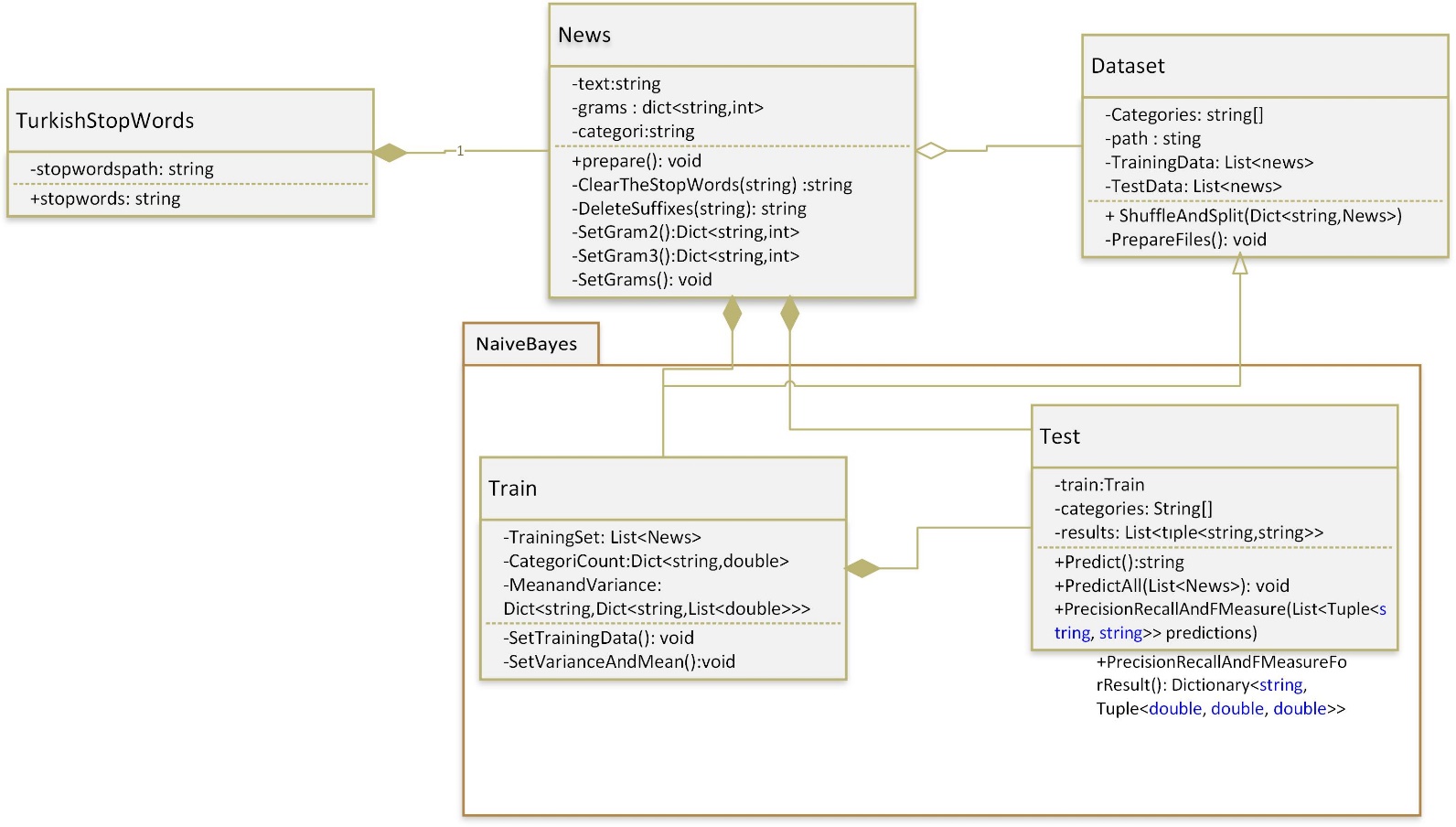


Denklem 1: Normal Dağıtıma göre olasılık hesabı(Normal Distribution)

Herbir gram için olasılık değeri hesaplandıktan sonra, olasılıkların ve hang*i* kategoriye ait olduğunu tutmak için oluşturulan probs Dictionary yapısında eklenir. Bu fonksiyon geri dönüş olarak olasılık değeri en yüksek olan kategori bilgisini geri döndürür.



* *public void PredictAll(List<News> files):*  Test sınıfı içerisinde yer alan bu fonksiyon içerisinde, test kümesinde yer alan herbir doküman için Predict fonksiyonu çağrılıp, dökümanın hangi kategoriye ait olduğu tahmini gelir.Gelen tahmin sonucu ve aslında ait olduğu kategori bilgisi *results* dictionary yapısına eklenir.

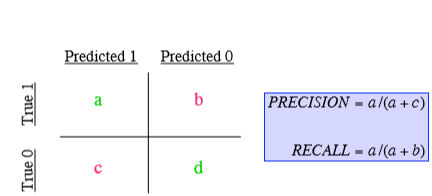
**

* *public Dictionary<string, Tuple<double, double, double>> PrecisionRecallAndFMeasure(List<Tuple<string, string>> predictions):[2]*  Test sınıfı içerisinde yer alan bu fonksiyon içerisinde F-measure, Recall ve Precision değerleri hessaplanmaktadır. Bu değerler aşağıda beliritilen matris yöntemine göre hesaplanmıştır.

F-measure değeri ise

(2 \* precision \* recall) / (precision + recall);

Formülü kullanılarak hesaplanmıştır.

**

# Genel Yapı

**4.1 UML diyagramı**

# 5.Referanslar

1. <http://bilgisayarkavramlari.sadievrenseker.com/2008/12/21/bayes-aglari-bayesian-network/>
2. <https://towardsdatascience.com/model-evaluation-i-precision-and-recall-166ddb257c7b>
3. <https://medium.com/@theflyingmantis/text-classification-in-nlp-naive-bayes-a606bf419f8c>
4. <https://machinelearningmastery.com/naive-bayes-for-machine-learning/>