

T.C.
GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ

KARAKTER TABANLI DOKÜMAN DİLİ
TANIMA SİSTEMİ TASARIMI

Hidayet TAKCI
DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ
ANABİLİM DALI

171072

TEZ DANIŞMANI
Yrd. Doç. Dr. İbrahim SOĞUKPINAR

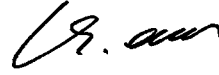
GEBZE
2005

Hidayet TAKCI'nın tez çalışması, G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 3.3.2005 tarih ve 2005/10 sayılı kararıyla oluşturulan jüri tarafından Bilgisayar Mühendisliği Anabilim Dalında DOKTORA Tezi olarak kabul edilmiştir.

JÜRİ

ÜYE

: Prof. Dr. Eşref ADALI



ÜYE

: Doç. Dr. Natali Shvets DİMOĞLU



ÜYE

: Yrd. Doç. Dr. İbrahim SOĞUKPINAR

(Tez Danışmanı)



ÜYE

: Yrd. Doç. Dr. Mehmet GÖKTÜRK



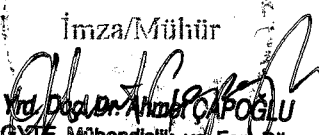
ÜYE

: Yrd. Doç. Dr. F.Erdoğan SEVİLGİN



ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 04.07.2005 tarih ve 2005/24 sayılı kararı.

İmza/Mühür

Yrd. Doç. Dr. Ahmet CAPOĞLU
G.Y.T.E. Mühendislik ve Fen Bil. Enstitüsü Müdür Yardımcısı



ÖZET

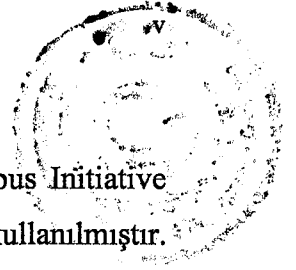
Dil tanıma, başta doğal dil işleme olmak üzere bir çok uygulamada anahtar vazife görmektedir. Dil tanıma, dilin ayırt edici özelliklerine bazı tekniklerin uygulanması ile gerçekleştirilmektedir. Bu kapsamda; terimler (kelime veya kelime öbekleri), harf dizileri veya n-gramlar bugüne kadar dilin ayırt edici özellikleri olarak dil tanımada kullanılmıştır. Dili tanımlayan özelliklere uygulanan en temel iki yöntem ise istatistiksel ve dilbilimsel yöntemdir. Bunlardan istatistiksel yöntemde, dilin istatistiksel özellikleri kullanılır, dilbilimsel yöntemde ise daha çok dilin özel harfleri gibi dilbilimsel özellikleri kullanılmaktadır.

Dil tanımanın geri planında; markov modelleri, n-gram, bayesian modelleri ve vektör uzayı gibi modeller bulunur. Bu modeller dilin özelliklerini sunmada ve özellikler arası ilişkilerin açığa çıkarılmasında kullanılırlar.

Bu çalışmada doküman dili tanıma için karakter özellik kümesini kullanan bir yöntem önerilmektedir. Yöntem; doküman karakter sıklıklarının dokümanın dilini tanımada kullanılabileceği temeline dayanmaktadır. Yöntemin özellik kümesi; Latin kökenli sekiz Avrupa dili ve Türkçenin alfabesinden elde edilmiştir. Karma alfabede toplam 60 karakter bulunmakta olup, bunların 26 tanesi bütün dillerde kullanılan ortak karakterler (genel karakterler) geriye kalan 34 tanesi (özel karakterler) bütün dillerde kullanılmayıp bir veya birden fazla dilde kullanılan karakterlerdir.

Çalışmada dil özelliklerini sunmak için vektör uzayı modeli seçilmiştir. Bu model sayesinde her bir doküman bir vektör ile temsil edilmiş, ayrıca doküman kategorileri de centroid adı verilen bir vektör ile sunulmuştur. Doküman karakter vektörleri her bir dokümanda geçen karakterlerin sıklıklarından elde edilmektedir.

Diller ağırlıklı centroid vektörleri ile, test dokümanları da doküman karakter vektörleri ile sunulur. Bir test dokümanının dilini tespit için diller ile test dokümanı arasındaki benzerlikler hesap edilir. Test dokümanı ile diller arasındaki benzerlik, doküman karakter vektörü ile centroid arasındaki açı benzerliği ile bulunmaktadır. Benzerlik sonrasında, maksimum benzerliği veren centroidin ait olduğu dile test dokümanı atanarak dil tanıma yapılır.



Karakter tabanlı yöntemin eğitimi ve testi için, European Corpus Initiative (ECI) adı verilen uluslar arası kabul görmüş, çok dilli bir külliyat kullanılmıştır. Eğitim için ECI külliyatından, 10 ile 100 KB arasında uzunluklarda 9 adet alt külliyat kullanılmıştır. Külliyat boyutlarındaki farklılığın sebebi, her bir dile ait karakter sıklıklarının farklı uzunlukta kararlı hale gelmesindendir. Karakter sıklıklarının hangi boyutta kararlı hale geldiği dil tanıma doğruluklarının testi ile elde edilmiştir.

Karakter tabanlı yöntemin doğru dil tanıma oranı, 100 karakter uzunluğundaki metinler için 9 dilde ortalama %99 civarındadır. Önerilen yöntemin doğru tanıma oranı diğer yöntemlerden yaklaşık %1 daha düşük ancak yöntemin özellik kümesi diğer yöntemlerin özellik kümelerinden oldukça küçüktür. Karakter tabanlı yöntemin özellik kümesi kısa kelimeler yönteminin 1/16'sı, 3gram yönteminin ise 1/60'ı kadardır. Ayrıca işlem maliyetleri açısından karakter tabanlı yöntem daha az maliyetlidir. Özellik adedinin az olması ve işlem maliyetlerinin düşüklüğü sebebiyle karakter tabanlı yöntem, diğer yöntemlere göre yaklaşık 15 kat daha hızlıdır. Bu durum karakter tabanlı yöntem için önemli bir avantajdır. Hız avantajından dolayı karakter tabanlı yöntem gerçek zamanlı sistemlerde etkin olarak kullanılabilir.

SUMMARY



Language Identification is a key subject in many applications, especially natural language processing. Language identification is performed by applying some techniques to characteristic features of the language. In this scope; terms (words or word groups), sequences of letters or n-grams have been used as characteristic features of the language until today. Statistical and linguistic methods are used for identification of document languages. In statistical method, statistical properties of the language are used and properties like special letters are mostly used in linguistic language identification method.

Models like Markov models, n-gram models, Bayesian models and vector space models lie on the background of the language identification. These models are used for presentation the features of the language and bringing out the relations between these features.

In this work, a method which uses character feature set for language identification is proposed. The method is based on the idea of documents' character frequencies can be used for document language identification. Feature set of the method is obtained from alphabets of eight European languages having Latin origin and Turkish. While there are 60 characters in the mixed alphabet, 26 of them are the common characters which are used in all languages (general characters) and the remaining 34 characters are not used in all languages but are used in one or more languages (special characters).

Vector space model is selected for presentation of the language features in the proposed method. By the help of this model, each document is represented by a vector and in addition, document categories are represented by a vector called weighted centroid. Document character vectors are obtained from the frequencies of characters belonging to each document.

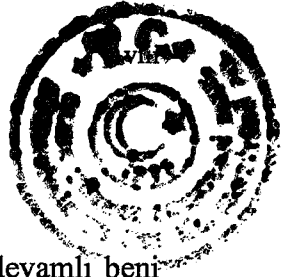
Languages are presented with centroid vectors and test documents are presented with document character vectors. Then, in order to determine the language of a test document similarities between languages and test document are calculated.



The similarity between test document and languages is found by the similarity of the angle between document character vector and centroid vector. After this, language identification is performed by assigning the test document to the category which contains the centroid that gives the maximum similarity.

Multilanguage corpus named as European Corpus Initiative (ECI) have been used for the comparisons between character-based method and other language identification methods. Nine sub-corpus from ECI which have a length between 10 and 100 KB have been used for training of the method. The reason of the differences in corpus lengths is character frequencies belonging to each language becomes steady-state in different lengths. The optimum character frequencies are obtained during the language identification experiments.

The success rate of character based method is about 99% on average in 9 languages for texts whose length are 100 characters. The accuracy rate of this method is approximately 1% less than other methods. However, feature set of the proposed method is 1/16 of short term method and 1/60 of the 3-gram method. For this reason proposed method is approximately 15 times faster than other methods. Thus, character based method can be used effectively in real time systems.



TEŞEKKÜR

Tez çalışmalarım boyunca elinden gelen desteği esirgemeyen, devamlı beni yüreklendiren saygıdeğer hocam Y. Doç. Dr. İbrahim SOĞUKPINAR Beye ve yönlendirmeleri ile faydalı olan Y. Doç. Dr. Mehmet Göktürk ve Y. Doç. Dr. Erdoğan Sevilgen Beylere;

Tez çalışmalarına daha fazla vakit ayırabilmem için gerektiği yerde bana yardım eden arkadaşım Araş. Gör. Rahim KARABAĞ Beye, makale çalışmalarında yardımcı olan Araş. Gör. Türker AKYÜZ ve Araş. Gör. Ali İSKURT Beylere;

Manevi desteklerini her zaman yanımda hissettiğim anneme, babama, kardeşlerime ve tez çalışmam boyunca vakitlerini çaldığım sevgili eşim Nurten Hanım ile sevgili oğullarım Kamil ve Nurettin Beylere;

Teşekkürü bir borç bilirim.

İÇİNDEKİLER DİZİNİ

	<u>Sayfa</u>
ÖZET	iv
SUMMARY	vi
TEŞEKKÜR	viii
İÇİNDEKİLER DİZİNİ	ix
SİMGELER ve KISALTMALAR	xi
ŞEKİLLER DİZİNİ	xii
ÇİZELGELER DİZİNİ	xiii
1.GİRİŞ	1
1.1.Metinsel Dokümanların Dilinin Tanınması	2
1.2.Tezdeki Çalışmalar	6
2.DOKÜMAN DİLİ TANIMA	7
2.1.İlişkili Çalışmalar	7
2.2.Dil Tanımının Özellik Kümeleri	7
2.2.1.Özel Karakterler ve Tekil Harf Kombinezonları	7
2.2.2.Kısa veya Genel Kelimeler	9
2.2.3.Karakter n-gramlar	9
2.2.4.Karakter n-gram Sıklıkları	10
2.2.5.Diğer Özellikler	11
2.3.Dil Tanıma Teknik ve Modelleri	11
2.3.1.İstatistiksel Yöntem ve Modeller	11
2.3.2.Markov Modeller	12
2.3.3.Vektör Uzayı Modelleri (Bilgi Alma Modeli)	13
2.3.4.n-gram Modelleri	14
2.4.Son Zamanlarda Yapılan Çalışmalar	14
2.5.Dil Tanıma için Doküman Sınıflandırma	16
2.5.1.Karakter Tabanlı Dil Tanıma Yöntemi	16
3.KARAKTER TABANLI DİL TANIMA (KTDT)	18
3.1.Karakterlerin Dil Tanımda Kullanılması	18
3.2.Karakter Tabanlı Yöntemin Temelleri	20
3.3.Karakter Tabanlı Dil Tanıma Modeli	21
3.3.1.Modelin Formal Dilde Gösterimi	22

3.4.Karakter Tabanlı Dil Tanıma Sistemi	26
3.4.1.Eğitim Aşaması	29
Metinsel Verinin Sayısal Hale Dönüşümü	30
3.4.2.Test Aşaması	31
3.5.Karakter Tabanlı Dil Tanımanın Özellikleri	33
4.DENEYLER VE SONUÇLAR	36
4.1.Veri Kümesi	36
4.2.Metinsel Verinin Sayısal Hale Dönüşümü	37
4.2.1.Doküman-Karakter Sıklıklarının Bulunması	38
4.2.2.Centroid Değerlerinin Bulunması	39
4.3.Karakter Tabanlı Dil Tanımanın Motivasyonu	41
4.3.1.Karakter Tabanlı Dil Tanıma Karakter Sıklıkları	41
4.3.2.Karakter Tabanlı Dil Tanıma Özel Karakterler	48
4.4.SDM ile Karakter Tabanlı Sınıflandırma	50
4.5.Centroid Tabanlı Dil Tanıma Deneyleri	52
4.5.1.Centroid Değeri	52
4.5.2.Benzerlik/Uzaklık Yöntemlerinin Etkisi	53
4.5.3.Standart Sapmanın Etkisi	55
4.5.4.İlave Yöntem	55
4.6.Karakter Tabanlı Yöntem ve Karşılaştırmalar	57
4.6.1.Doğru Tanıma Açısından Karşılaştırmalar	57
4.6.2.Performans Açısından Karşılaştırma	59
5.SONUÇ ve ÖNERİLER	61
5.1.Öneriler	63
KAYNAKLAR	65
ÖZGEÇMİŞ	69

SİMGELER ve KISALTMALAR

SQL	Structured Query Language
VTBK	Veritabanlarından Bilgi Keşfi
IR	Information Retrieval
NB	Naive Bayesian
CB	Centroid Based
C&RT	Classification and Regression Trees
HMM	Hidden Markov Models
P ve R	Precision ve Recall
NLP	Natural Language Processing
KTDT	Karakter Tabanlı Dil Tanıma
k-NN	k-Nearest Neighbor
ANSI	American National Standards Institute
TF/IDF	Term Frequency Inverse Document Frequency
FR	Sıklık Vektörü
TF	Terim Sıklık Vektörü
N	Doküman veya Örnek Adedi
D	Doküman
$\hat{y}$	Tahmin edilen değer
p	Özellik adedi veya istatistiksel testlerin önem katsayısı
λ	Önem-Önemsizlik derecesi (İstatistiksel Modellemede)
Σ	KTDT modelinde Alfabe
Σ^n	n karakter uzunluklu harf dizilerinin kümesi
Σ^*	Olası bütün kelimelerin kümesi
P(T)	T teriminin olasılığı
P(T D)	D dokümanı içerisinde T teriminin olasılığı
E	Hata miktarı
SDM	Statistica Data Miner
ECI	Dil tanıma deneylerinde kullanılan çok dilli küliyat
$\vec{C}_i$	i. dil için centroid (merkez) değeri

ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
3.1. KTDT Sistem Mimarisi	28
3.2. İsveççe olarak yazılmış örnek metin	29
3.3. Karakter tabanlı ön işlenmiş İsveççe örnek metin	30
3.4. Örnek metinlerin karakter sıklık değerleri	30
3.5. Eğitim setleri için karakter sıklıklarına dayalı doygunluk grafikleri	34



ÇİZELGELER DİZİNİ

Çizelge

Sayfa

2.1.	Dillere göre kısmi harf dizilerinin listesi	8
3.1.	Dillere göre harf sıklıkları	20
3.2.	Farlı uzunluklar için karakter tabanlı dil tanımanın doğruluk oranı	34
4.1.	Eğitim İçin Külliyat Boyutları	36
4.2.	“ADANA” kelimesi için karakter sıklık değerleri	38
4.3.	“ADANA” kelimesi için bağıl sıklık değerleri	38
4.4.	Türkçe için ortalama karakter sıklıkları	39
4.5.	9 Dil için ortalama karakter sıklıkları	39
4.6.	Ortalama karakter sıklıkları ve centroid ilişkisi	40
4.7.	Almanca ile diğer diller arasındaki farklılık testi	44
4.8.	Bir önceki çizelgenin önem bilgisi ile verilmesi	45
4.9.	Hollanda dili için birleştirilmiş Mann Whitney U testi sonuçları	46
4.10.	‘a’ karakteri için sıklık farklarının p değerine göre gösterimi.	47
4.11.	‘b’ için diller arası sıklık farklarının p değerine göre gösterimi.	47
4.12.	‘z’ için diller arası sıklık farklarının p değerine göre gösterimi.	48
4.13.	Özel karakterlerin sıklıklara dayalı gösterimi	49
4.14.	10 karakter için sınıflandırma matrisi	50
4.15.	Farklı metin uzunlukları için doğruluk değerleri (precision)	51
4.16.	Farklı metin uzunlukları için geri çağırma değerleri (recall)	51
4.17.	Logaritma dönüşümünün başarıya etkisi	53
4.18.	Farklı Uzaklık/Benzerlik yöntemlerine göre dil tanımanın başarısı	54
4.19.	3gram yönteminin KTDT’ ye etkisi	56
4.20.	Dillere göre ortalama kelime uzunlukları	57
4.21.	Cümlede geçen kelime adedine göre ortalama başarı oranları	58
4.22.	Cümlede geçen kelime adedine göre başarı (Türkçe/İsveççe)	58
4.23.	Bulunan dokümanların aranan dokümanlar olup olmadığının testi	59
4.24.	Dil Tanıma Yöntemlerinin Dil Tanıma süreleri	59

1. GİRİŞ

Etnolojik arařtırmalara g re řu an d nyada 6800 dil aktif olarak konuřulmaktadır [1]. Ge erli dil tanıma sistemleri ise bu dillerden ancak 250-260 tanesini tanıma yeteneğine sahiptir [2]. Kullanılan dil sayısı ile tanınabilen dil sayılarına bakıldığında yeni dil tanıma sistemleri geliřtirmenin bir zorunluluk olduđu ortadadır. Ayrıca; farklı dillerde yazılmış metinleri anlayabilmek, farklı dillerde m řterilere hizmet sunabilmek, metin  zetleme, soru cevaplama ve  eviri tipinde iřlemler i in de yeni dil tanıma sistemleri geliřtirmek bir ihtiya tır [3].

 eviri sistemleri, dođal dil iřleme ve konuřma tanımada ilk adım dil tanımadır. O y zden, dil tanıma,  eviri sistemleri ve konuřma tanıma ile yakından ilgilidir. Geliřtirilen ilk dil tanıma sistemleri konuřma tanıma teknikleriyle yerine getirilmiřtir.  eviri sistemlerinin dil tanıma konusunda iki temel stratejisi bulunmakta olup, bu stratejilerden birincisi; dil tanımada  zel harflerin kullanımına dayalıdır [4]. Bu tip dil tanımada  zel harflerden bir tanıma listesi oluřturulmakta ve hızlıca dil tanıma iřlemi yerine getirilebilmektedir. İkinci strateji,  eřitli dillerde kullanılan kısa kelimelerin bir listesi ile  alıřma temeline dayalıdır. Dilin tanınabilmesi i in dile ait kısa kelimeler aranır [5]. Kimi zamanda kelimelerin hangi harf dizileriyle son bulunduđu t r nden bilgilere bakılır.

Dil tanıma i in her bir dilde sıklıkla yer alan kısmi harf dizileri ile birlikte d ř k seviyeli n-gramlar da kullanılmaktadır [6, 7, 8, 9].

Bilgi alma etki alanında vekt r uzayı modeli ve olasılıksal model gibi bařarılı modeller bulunmakla birlikte, dil tanıma modelleri daha  ok konuřma tanıma etki alanından alınmıřtır [10, 11]. Dil tanımada kullanılan bir ok istatistiksel model  nce konuřma tanıma konusuna uygulanmıřtır. Shannon [12] tarafından geliřtirilen kelime ve harf dizileri kullanımı bunlara bir  rnektir. Aynı zamanda n-gram modelleri ve Hidden Markov Modeli (HMM) bug n hala sıklıkla dil tanımada kullanılmaktadır. Geliřtirilen dil modelleri bilgi alma etki alanında da kullanılmıřtır. Sorgu sonu larının iyileřtirilmesi bilgi almada dil modellerinin kullanımına bir  rnektir.

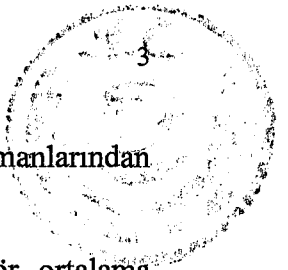
Dil tanıma çalışmalarında n-gram yönteminin kullanımı son zamanlarda artmıştır. N-gram yöntemi metin sınıflandırma için de kullanılan bir yöntemdir. Bu yöntem dil tanıma ile metin sınıflandırma çalışmalarının aynı tekniklerle yapılabilirliğini göstermesi bakımından önemlidir. Cavnar ve Trenkle tarafından geliştirilen, n-gram tabanlı metin sınıflandırma modeli üzerine inşa edilen Texcat isimli dil tanıma uygulaması bugün dil tanımada etkin olarak kullanılmaktadır.

Bu çalışmada, doküman dili tanıma için, bireysel karakterlere dayalı bir metin sınıflandırma yöntemi önerilmektedir. Yöntem unigram da sayılabilecek özellik kümesi ile çalışmaktadır. Özelliklerin sunumu için geleneksel vektör uzayı modeli seçilmiştir, yöntemin ağırlık modeli tfidf benzeri bir yöntemdir. Bu yöntemde dil tanıma işlemi dokümanlar ile doküman kümeleri arasındaki benzerliklerden faydalanılarak yapılmaktadır.

1.1. Metinsel Dokümanların Dilinin Tanınması

Doküman dillerini tanımada takip edilmesi gereken bir yol vardır. Bu yolun şeklini seçilen dil tanıma modeli belirler. Eğer doküman dillerini tanımada geleneksel vektör uzayı modeli seçilmişse, izlenecek adımlar şunlardır:

- Eğitim ve test dokümanları doküman külliyyatından seçilir.
- Dokümanlar üzerinde gerekli ön işlem yapılır.
 - Dokümanlar seçildikten sonra gerekli ön işlemin yapılması ve dokümanların dil tanımaya uygun hale getirilmeleri lazımdır. Bunun için ön işlem aşamasında dokümanlara temizleme ve dönüşüm uygulanır. Temizleme; dokümandan, anlamsız kelime ve işaretlerin kaldırılmasıdır. Dönüşüm ise genellikle özetleme şeklinde yapılır.
- Bütün dokümanlar bir vektör yardımıyla sunulur.
 - Sunum aşamasında özellik seçimi yapılır. Özellik seçimi dokümanı en iyi sunacak özelliklerin seçilmesi işlemidir. Bu aşamada özellik değerleri için eşik değeri kullanılabilir. Örneğin meydana gelme olasılığı 0.01 den daha düşük olan özellik elenir.
 - Dil tanıma sisteminin özellik kümesinin hangi elemanlardan oluştuğu önemli bir konudur. Özellik kümesi için genellikle harf veya kelime dizileri ile n-gramlar kullanılmaktadır.

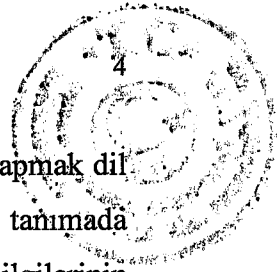
- 
- Eğitim aşamasında her bir dili sunmak için bir vektör eğitim dokümanlarından elde edilir.
 - Eğitim aşamasında elde edilen ve dil kategorilerini ifade eden vektör, ortalama özellik değerleriyle ilgiliyse centroid olarak isimlendirilir. Centroid değerleri her doküman kategorisi için bir tanedir.
 - Test dokümanlarının dilini tespit için, test dokümanı vektörleri ile centroid vektörleri arasındaki benzerlikler hesaplanır.
 - Vektör uzayı modelinde vektörler arasındaki benzerlikler kosinüs benzeri açı benzerliklerinden elde edilebilmektedir.
 - Bir test dokümanının hangi dilde olduğunu bulmak için doküman ile dile ait centroid vektörleri arasındaki maksimum benzerliğe bakılır.
 - Maksimum benzerliği veren centroid vektörünün ait olduğu dil test dokümanının dilidir.

Dil tanımda sıklıkla kullanılan bazı yöntemler vardır. N-gramlar, tekil harf kombinasyonları, kısa veya genel kelimeler en bilinen yöntemlerden birkaç tanesidir. Bu yöntemlerin özellik kümeleri genellikle büyük boyutludur. Özellik kümelerinin büyük olması, dil tanımanın işlem maliyetini artırdığından dolayı bir problemdir.

Bunun yanı sıra dil tanıma sistemleri, yapısal olmayan verilerle çalıştıklarından ön işlem miktarı oldukça fazladır. Ön işlem miktarının fazla olması, dil tanıma sistemleri için bir problem olup ön işlem miktarını azaltıcı bir yöntem diğer yöntemlere göre tercih edilecektir.

Bütün bilgi sistemlerinde olduğu gibi, dil tanıma sisteminde de anlaşılabilirlik ve geliştirilebilirlik tercih sebeplerinden biridir. O yüzden dil tanıma sistemi daha anlaşılır ve daha kolay geliştirilebilir olmalıdır. Bu arada dil tanıma sisteminin başarısı kullandığı sınıflandırma algoritması ile de ilgilidir. Başarılı bir sınıflandırma algoritmasının kullanılması, dil tanıma sisteminin de başarısı manasına gelecektir.

Dil tanıma kimi zaman dilbilimsel yöntemlerle yerine getirilmektedir. Dil bilimsel dil tanıma, dil bilimci olmayı veya dil bilimcilerle çalışmayı gerektirdiği için zor bir yöntemdir. O yüzden dil tanıma için istatistiksel yöntemlerin kullanılması



tercih edilir. Dilin istatistiksel özellikleri yardımıyla dil tanıma çalışması yapmak dil bilimci olmayı gerektirmez. İstatistiksel dil tanımada önemli olan; dili tanımda kullanılabilecek bilgilerin açık olarak ortaya konması ve bunların sayısal bilgilerinin elde edilmesidir. Örneğin; istatistiksel dil tanıma yöntemi terimlere dayalı olarak çalışıyorsa önce dili tanıttıcı terimlerin bir kümesi bulunup daha sonra her bir metinde o terimlerin hangi sıklıklarda yer aldığı ölçülür. Bu bilgi yardımıyla metinlerin dili bulunur.

Kısa veya genel kelimeler, n-gram ve kısmi harf dizileri dil tanıma yöntemlerinden bazılarıdır [3, 5, 6, 7]. Bu yöntemlerden kısa veya genel kelimeler yönteminde, her bir dilde sıklıkla kullanılan kelimelerin bulunması gerekmektedir. Bir A dilindeki kısa kelimeler; $A=(a_1,...,a_p)$ ve X gibi bir dokümandaki kelimeler; $X=(x_1,...,x_n)$ şeklinde verilmişse; $a_i = x_j$ (($i=1,2,...,p$) ($j=1,2,...,n$)) eşleşmesi arttıkça X dokümanının A dilinde olma olasılığı artar. Yöntem bazen kısa metinlerde problem verebilmektedir. Çünkü kısa metinlerde kimi zaman kısa veya genel kelimeler bulunamamaktadır. Bu durumda dil tespitini genel kelimeler yöntemi ile yapmak mümkün olmamaktadır.

N-gram yönteminde ise diller ve dokümanlar profil adı verilen özellikleri yardımıyla sunulur ve dillerin profilleri ile dokümanların profilleri karşılaştırılarak dil tanıma yerine getirir. Profiller arasındaki benzerliklere dayalı olarak test edilen dokümanın dili tespit edilir. Yöntemin en önemli dezavantajı parçalama (tokenizasyon) işlemine ihtiyaç duymasıdır. Ayrıca benzerlik bulmada kullanılan yöntem, klasik benzerlik bulma yöntemlerinden biri değildir [6].

Dil tanıma yöntemlerden bir diğeri kısmi harf dizileri yöntemidir. Bu yöntem bazı harf dizilerinin dilleri ayırt edebileceği fikri üzerine kurulmuştur. Halbuki aynı harf dizileri birkaç dilde birden olabilir. O yüzden harf dizileri doğru seçilmediği takdirde sistem hatalı sonuç vermektedir [6]. Kısmi harf dizileri n-gramlar gibi sabit uzunlukta değildirler. Ayrıca bazı n-gramlar farklı dillerde olabilir ve dilden dile n-gramların meydana gelme sıklıkları farklıdır. Kısmi harf dizilerinde ise sadece bir dilde bulunan harf dizileri seçilmeye çalışılmaktadır.

Bahsedilen dil tanıma yöntemlerine ek olarak bu çalışmada karakter tabanlı bir yöntem önerilmektedir. Karakter tabanlı yöntemin özellik kümesi 60 adet harf karakterinden oluşmakta olup, diğer dil tanıma yöntemlerine oranla özellik kümesi oldukça küçüktür.

Karakter tabanlı dil tanıma, kelime tabanlı ön işleme (heceleme, gövdeleme vs.) ihtiyaç duymaksızın doğrudan ham verilerle çalışabilir durumdadır. Dil tanıma için yapılması gereken tek ön işlem karakterler haricindeki, noktalama işaretleri gibi, işaretlerin dokümanlardan kaldırılması ve dokümanların karakter tabanlı sıklıklarının elde edilmesidir. Karakter tabanlı yöntemin, diğer dil tanıma yöntemlerinde var olan ve maliyet gerektiren kelime tabanlı ön işlemlere ihtiyacı yoktur. Önerilen yöntem ön işlem gereksinimini azaltmıştır.

Yöntemin anlaşılması ve geliştirilmesi kolaydır. Yöntemin eğitimi aşamasında; önce dillere ait ortalama karakter sıklıkları bulunur, ortalama karakter sıklıklarından her bir dil için bir centroid değeri elde edilir. Test aşamasında ise; dili bulunacak dokümanın karakter sıklıkları ile dillerin centroid değerleri arasındaki benzerliklere göre dil tespiti yapılır. Dili bilinmeyen test dokümanı en çok hangi dilin centroid değerine benzer ise o dile atanır.

Önerilen yöntemin diğer diğer dil tanıma yöntemlerine göre bir başka avantajı kullandığı sınıflandırma algoritmasıdır. Yöntem, doküman sınıflandırmada başarılı bir algoritma olan centroid tabanlı sınıflandırma algoritmasını kullanmaktadır. Centroid tabanlı doküman sınıflandırma, n-gram tabanlı metin sınıflandırma gibi bir metin sınıflandırma algoritmasıdır. N-gram tabanlı yöntem gibi centroid tabanlı yöntemde dil tanıma kullanılabilir.

Özet olarak bu çalışmada dil tanıma için yeni bir yöntem önerilmektedir. Yöntem bireysel karakter sıklıklarına dayalıdır. Doküman dili tanıma; dokümanı oluşturan karakterlerin kullanılabileceği, karakter sıklıklarının doküman dili açısından önemli bilgi sağlayabileceği ileri sürülmekte ve bu öneri desteklenmektedir. Bu amaçla istatistiksel testler ve sınıflandırma deneyleri yapılmıştır. Önerilen yöntem bireysel karakterlere centroid tabanlı sınıflandırmanın uygulanması şeklinde çalışmaktadır. Dil tanıma sisteminde diller ve dokümanlar



vektör uzayı modelinde birer vektör ile sunulmakta olup, vektör boyutları karakter sıklık bilgileri tutmaktadır. Dokümanların ağırlıklandırılmasında tfidf benzeri bir ağırlık modeli kullanılmaktadır. Sistem, maksimum benzerliğe göre dil tespiti yerine getirmektedir.

1.2. Tezdeki Çalışmalar

Tezde önerilen yönteme bağlı olarak öncelikle dokümanlar ve diller karakter sıklıklarına dönüştürülmüş daha sonra istatistiksel testler ile doğru dil tanıma deneyleri sıklık verileriyle yerine getirilmiştir. Karakterlerin dilleri sunup sunamayacağı istatistiksel yöntemlerle test edilmiş ve testlerde diller arası farklar ortaya çıkarılmış böylece karakterlerin doküman dili tanımada kullanılabilirliği gösterilmiştir.

Dil tanımadaki başarı oranını artırmak için karakterlerin dil içindeki sıklığına ek olarak diller arasındaki sıklığı da hesaplanmıştır. Bu hesaplamanın amacı dokümanlarda az rastlanan karakterlerin önemini artırmak, sık rastlanan karakterlerin önemini azaltmaktır. Böylesi bir çalışma ile karakterlerin dilleri daha iyi şekilde ayırıştırması sağlanmıştır.

Bir dokümanın dili; doküman ile dillerin karakter sıklıkları arasındaki benzerliğe bağlı olduğundan, en yüksek doğruluğu verecek benzerlik bulma yöntemi araştırılmıştır. Bu araştırma sonrasında kosinüs benzerlik yöntemine benzer bir yöntem olan Metin Puanlama yönteminin en başarılı sonucu verdiği görülmüştür. Metin Puanlama; Takçı ve Soğukpınar tarafından karakter tabanlı dil tanıma için uygun bulunan bir benzerlik yöntemidir [13].



2. DOKÜMAN DİLİ TANIMA

Doküman dili tanıma, dil tanımanın iki alt alanından birisi olup; çeviri sistemleri, konuşma tanıma ve bilgi alma etki alanlarıyla ilişki içerisinde. Aynı zamanda doküman dili tanıma yazılı metinlerin dil tabanlı olarak sınıflandırılmasıdır, o yüzden sınıflandırma algoritmalarıyla da ilişkilidir. Bu bölümde, doküman dillerinin tanınması konusunda yapılmış çalışmalar incelenip karakter tabanlı dil tanımanın konumu bulunmaya çalışılacaktır.

2.1. İlişkili Çalışmalar

Dil tanıma için bugüne kadar çeşitli yöntemler geliştirilmiştir. Bunlardan en bilinen üç tanesi; tekil harf kombinezonları, kısa veya genel kelimeler ve n-gram yöntemidir. Dil tanıma yöntemlerini birbirinden ayıran en önemli iki etken dil tanımada kullanılan özellik kümeleri ile analiz yöntemleridir. Örneğin; tekil harf kombinezonlarının özellik kümesi bireysel veya bileşke harflerden meydana gelirken kısa kelimeler yönteminin özellik kümesi kelimelerden, n-gram yönteminin özellik kümesi n karakter uzunluğundaki harflerden meydana gelmektedir. Aynı şekilde bu yöntemler analiz açısından da farklılıklar göstermektedir. Bu sebeplerden dolayı geçmiş dil tanıma yöntemleri özellik kümeleri ve analiz yöntemlerine göre sınıflandırılarak gözden geçirilecektir.

2.2. Dil Tanımanın Özellik Kümeleri

2.2.1. Özel Karakterler ve Tekil Harf Kombinezonları

Karakter özellik kümesi; Newman [4], Mustonen [14] ve Ziegler [15] tarafından dil tanıma çalışmalarında kullanılmıştır [16]. Newman çalışmasında dil tanıma için her dile ait özel karakterleri kullanmıştır. Özel karakterler bütün dillerde bulunmayıp bir veya birden fazla dilde bulunabilen karakterlerdir. Newman dil tanıma işleminden önce her bir dilin özel karakter listelerini elde etmiş, daha sonra dil tanımayı özel harf listeleri sayesinde hızlı, aynı zamanda kolay şekilde yerine getirmiştir. Özel karakterlerin kullanımı bir nevi filtreleme görevi görmekte ve bu sayede doküman dili tespit edilirken taranan dil sayısı azalmaktadır.

Özel karakterlere ek olarak harflerin bağımsız ve bileşke olasılıkları kullanılmış ve İngilizce ile İspanyolca metinler üzerinde dil tanıma yapılmıştır [17]. Harfler diller hakkında bilgi taşıdığı gibi bileşke harfler de dil hakkında ayırt edici bilgi taşımaktadır.

Karakterlerin dil tanımada kullanımına bir başka örnek tekil harf kombinezonlarıdır. Tekil harf kombinezonları sadece bir dilde yer aldığı tahmin edilen harf dizileridir. Bu harf dizilerinin dokümanda bulunmasıyla dokümanın dilinin tespit edilebileceği düşünülmektedir. Chu tarafından yapılan bir çalışmada bazı dillerde yer alan karakter dizileri Çizelge 2.1’de verilmiştir [9].

Çizelge 2.1. Dillere göre kısmi harf dizilerinin listesi

Dil	Karakter Dizisi
Dutch	“vnd”
English	“ery”
French	“eux”
Gaelic	“mh”
German	“der”
Italian	“cchi”
Portuguese	“seu”
Serbo-croat	“lj”
Spanish	“ir”

Kısmi karakter dizileri güvenilir olarak dili tespit için yeterli değildir. Hiç kimse “derya” gibi bir kelimenin içinde “ery” harf dizisi var diye kelimenin İngilizce olması gerektiğini iddia edemez. Aynı şekilde “killjoy” kelimesi de içinde “lj” yer alıyor diye Sırpça değildir. Yöntemin daha kullanılabilir olması için seçilen tekil karakter dizilerinin, daha zekice ve daha iyi şekilde seçilmeleri gerekmektedir [6].

Karakterler gibi karakterlerin ses karşılıkları da dil tanımada kullanılmaktadır. Bu konuda A. S. House ve E. P. Neuberg [18], fonetik özelliklere dayalı yazılı dil tanıma sistemleriyle ilgili ön çalışmaları sunmuş ve sekiz dile ait fonetik bilgilerin bir dizisinden istatistiksel markov modellerini eğitmişlerdir. Sekiz dille eğitilen markov modelleri dilleri birbirinden harika şekilde ayırtmıştır [19].

2.2.2. Kısa veya Genel Kelimeler

Kelimeler ilk olarak Ingle [20] tarafından dil tanımadaki kullanılmıştır. Ingle'in motivasyonu; her bir dilde sıklıkla geçen kelimelerin, dili karakterize edebileceğidir. Kelime tabanlı dil tanıma konusunda; Henrich [21], Kulikowski [22] ve Batchelder [23] çalışmalar yapmıştır [16]. Kelime tabanlı dil tanıma için en bilinen yöntem genel veya kısa kelimeler yöntemidir. Yönteme kısa kelimeler denilmesinin sebebi, sıklıkla kullanılan kelimelerin genellikle kısa olmasından dolayıdır. Kısa kelimeler; belirteçler, bağlaçlar, edatlar gibi dil bileşenleridir. Bu yöntemin en büyük problemi kısa metinlerde meydana gelmektedir. Kısa metinlerde kimi zaman kısa kelimeler yer alamamakta ve dil tanıma mümkün olamamaktadır.

Kelime tabanlı dil tanıma yöntemi aynı zamanda sözlük tabanlı dil tanıma yöntemidir. Bu tip dil tanımadaki sözlüklerin oluşturulması ve sözlük boyutunun küçültülmesi önemli işlerden birisi olup, hangi kelimelerin önemli hangilerinin önemsiz olduğunu bulmada genellikle kelimelerin sıklıklarına bakılmaktadır.

2.2.3. Karakter n-gramlar

Karakter n-gramlar, Henrich ve Ziegler tarafından kullanılmıştır. Yoshio Ueda [24] çok durumlu HMM ile (tamamen birbirine bağlı), 6 dilin modellenmesi için 2gramlar ve 3gramlar kullanmıştır. Yoshio'nun deneylerine göre 3gramlar, 2gramlardan daha iyi sonuçlar vermiştir fakat, 3gramların işlem maliyetinin oldukça yüksek olması problemidir. Bir 7 durumlu HMM ile 50 karakter uzunluğunda test metni için başarı %99.2'dir. Bu arada, karakter kodlarından dil tanıma yapmanın ses girişlerinden tanıma yapmaya göre daha az sıkı bir problem olduğu ortaya çıkmıştır [19].

Schmitt [25] doküman dili tanıma için trigram tabanlı bir yöntemin patentini almıştır. O, her bir dil için 3gram kümelerinin bir veritabanı ile metinlerden elde edilen 3gramları karşılaştırmış ve bu karşılaştırmalar sonrasında en fazla eşleşmeyi veren dil, kazanan dil yani metnin dili olmuştur [19]. Bu çalışmayla ilgili sonuçlar verilmemiştir.

Ayrıca Kwan ve Hirose [26] konuşma tanıma etki alanında karakter n-gramların kullanımını önermiş ve yapay sinir ağları ile uygulamasını yapmışlardır. Onlar çalışmalarında karakterler yerine fonetik bilgi (ses bilgisi) kullanmışlardır. Benedetto ve Caglioti [27]'de karakter n-gramları dil tanıma ve yazar tanıma konusunda önermişlerdir.

Dil tanımada karakter n-gramlar gibi kelime n-gramlarda kullanılmaktadır [23]. n-gramlar, uzunluğu n olan kelime veya karakter dizileri manasına gelmektedir [21, 7].

2.2.4. Karakter n-gram Sıklıkları

Karakter n-gram sıklıkları Beesley [28], Henrich [21], Cavnar ve Trenkle [7], Dunning [6] ve Damashek [29] tarafından kullanılmıştır. Cavnar ve Trenkle tarafından yapılan çalışma, aslında bir metin sınıflandırma çalışması olarak haberleri otomatik şekilde haber gruplarına dağıtma amaçlı geliştirilmiş olup, daha sonradan dil tanıma için kullanılmıştır. Bu çalışma n-gram sıklıkları ile çalışan en tanınmış yöntemlerden birisidir. Textcat başta olmak üzere bir dizi dil tanıma ürünü, motivasyonunu Cavnar ve Trenkle'in makalesinden almıştır.

Cavnar ve Trenkle [7] kısa harf dizilerinin istatistiksel olarak meydana gelme sıklıklarını hesap etmiş ve bunları sıralamıştır. Onlar ilk olarak test ve eğitim verilerini token adı verilen birimlere ayırmışlar ve parçalama (tokenizasyon) işleminden sonra en genel kısa karakter dizilerini sıklık sırasına göre dizmişlerdir. En sonunda ise farklı dillerdeki külliyat profilleri ile test dokümanının profili arasındaki benzerliklere göre doküman dilini bulmuşlardır. N-gram yönteminin zayıflıklardan birisi giriş verisi olarak işlenmiş veri beklemesidir. Veri işleme ise başta parçalama olmak üzere çeşitli doğal dil işleme görevlerini barındırdığı için zor bir işlemdir.

3gram sıklıkları konusunda Damashek [29]'de çalışma yapmış olup, 3gramları sıklık vektörleriyle sunmuş ve vektörlerin nokta çarpımlarıyla dil tespiti yerine getirmiştir.

2.2.5. Diğer Özellikler

Karakterler, kelimeler, karakter ve kelime n-gramlar ile n-gram istatistiklerine ek olarak dil tanıma için hece karakteristikleri, morfoloji ve yazım kuralları da kullanılmaktadır [14, 15, 19].

2.3. Dil Tanıma Teknik ve Modelleri

Dil tanıma, otomatik olmayan (Ingle [20] ve Newman [4]), yarı otomatik (Kullikowski [22]) ve otomatik yöntemler şeklinde bir dizi analitik teknikle yerine getirilmiştir. Batchelder [23] dilleri ayırt etmek için bir yapay sinir ağı eğitmiş ve yapay sinir ağlarını kullanmıştır. Henrich [21] ve Ziegler [15] dillerin farklılıklarını uzman sistemlere öğretmişlerdir. Mustonen [14] diskriminant analizin formları üzerinde çalışmıştır. Beesley [28], kriptanalizden gelme dil modelleme teknikleri kullanmıştır. Markov modelleri Dunning [6] tarafından kullanılmıştır. Henrich [21] ile Cavnar ve Trenkle [7] ise dil tanıma için istatistiksel modeller inşa etmişlerdir. Damashek [29] vektörlerin nokta çarpımlarını hesap eden bir model ortaya koymuştur.

2.3.1. İstatistiksel Yöntem ve Modeller

İstatistiksel yöntemler, dil bilimci olmadan dil tanımayı mümkün hale getirmektedir fakat sistemin eğitimi söz konusudur. Dil tanımanın bu tipinde dilleri ayırt etmede kullanılacak matematiksel prensiplerin neler olması gerektiği önemlidir [6]. Dil tanıma çalışmalarının önemli bir kısmında istatistiksel yöntemler kullanılmaktadır.

Dil modelleme doğal dildeki örüntüleri yakalayan istatistiksel dağılımların bulunması işidir. İstatistiksel dil modelleri ile ilgili çalışmalar Andrei Markov'a

kadar dayanır. Markov çalışmasında metinsel verilerdeki harf dizilerini modellemiştir [30]. Bir diğer ünlü çalışma Claude Shannon tarafından harf ve kelime dizilerinin modellenmesi çalışmasıdır [12].

İstatistiksel dil modelleri ilk olarak konuşma tanıma konusunda başarıyla uygulanmıştır. Otomatik konuşma tanımanın bir bileşeni dil modelidir ve bu bileşen devam eden konuşmada sıradaki kelimeyi tahmin eder.

Bir istatistiksel dil modeli terimlerin ağırlıkları veya olasılıkları üzerine kurulur. Bir sözlükte yer alan terimlerden bazısı diğer bazılarına göre daha önemlidir. Terimlerin önemi kimi zaman ağırlığı ile kimi zaman da olasılığı ile verilebilir. Önemi yüksek olan kelime ağırlığı veya olasılığı yüksek olan kelimedir. İstatistiksel dil modellemede terimlerin sırası önemli değildir, önemli olan terimlerin değeridir.

Olasılığa dayalı istatistiksel dil modellerinde bazı notasyonlar kullanılmaktadır. Bu notasyonlar kısaca şöyledir. T dil içindeki bir terimi göstermek üzere P(T), T teriminin, P(T|D) ise D dokümanı içindeki T teriminin olasılığıdır. Dillerde hangi terimlerin önemli hangilerinin önemsiz olduğu bilinemez. O yüzden bir terimin önem derecesi λ ile verilebilir, λ için aralık $(0 \leq \lambda \leq 1)$ şeklindedir. Terimleri birbirinden farklı n adet terim; $T_1, T_2, \dots, T_n$ olmak üzere bu diziye ait bir model 2.1. deki gibidir [31].

$$P(T_1, \dots, T_n | D) = \prod_{i=1}^n ((1 - \lambda_i)P(T_i) + \lambda_i P(T_i | D)) \quad (2.1.)$$

2.3.2. Markov Modeller

Bir markov modeli geçerli duruma bağlı olarak bir sonraki durumun olasılık dağılımını hesap eden rasgele prosestir. Biraz formülleştirmek gerekirse; markov model; değerleri X alfabesinden gelen karakter dizileri için; bir S karakter dizisinin olasılığını;

$$P(S) = p(s_1 \dots s_n) = p(s_1) \prod_{i=2}^n p(s_i | s_{i-1}) \quad (2.2.)$$



şeklinde hesap eder. $P(s_1) \rightarrow$ başlangıç olasılığını verirken $P(s_i | s_{i-1}) \rightarrow$ geçiş olasılıklarını vermektedir. En son meydana gelen k adet duruma bağlı olarak bir sonraki durumun olasılığı da bulunabilir.

$$p(s_{i+1} \dots s_{i+k} | s_i \dots s_{i+k-1}) = p(s_{i+k} | s_i \dots s_{i+k-1}) \quad (2.3.)$$

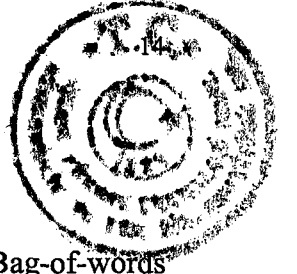
Bu durum sıralı k'nın markov modelleri olarak bilinir. $p(s_1, \dots, s_k)$ başlangıç olasılıkları olmak üzere bütün geçiş olasılıkları $p(s_{i+k} | s_1, \dots, s_{i+k-1})$ dir.

2.3.3. Vektör Uzayı Modelleri (Bilgi Alma Modeli)

Vektör uzayı modeli; sıklıkla, bilgi alma etki alanında kullanılmakta ve daha çok; bir sorgu terimiyle ilişkili dokümanları alıp getirme tipinde işlerde etkin olarak kullanılmaktadır. Bu modelde, dokümanlar ve sorgu kelimeleri birer vektör olarak sunulur. Vektör boyutları doküman veya sorgu kelimelerinin karakteristik özelliklerinden meydana gelir. Örneğin; doküman terim sıklıklarıyla ilgili bir vektör her bir terimin dokümanda meydana gelme sıklıklarını tutar. Doküman terim vektöründe her bir boyut bir terimle ilişkilidir.

Vektör uzayı modelinde terimler önemleriyle ilişkili ağırlıklara sahiptir. Bilgi almada her bir terimin önemi eşit değildir, o yüzden daha önemli terim ile daha önemsiz terim arasındaki farkı belirtmek için ağırlık modelleri kullanılır. Terimlerin önemi genellikle koleksiyon içerisinde yer alma sıklıklarından elde edilir. Vektör uzayı modelinde terimlerin ağırlıklandırılmasında sıklıkla tfidf olarak bilinen ağırlık modeli kullanılmaktadır.

Dokümanların boyut farkından kaynaklanabilecek hataları ortadan kaldırmak için normalleştirme uygulanır. Normalleştirme için genellikle bağıl sıklık kullanılır. Doküman içindeki terim sıklıkları $(\text{bir_terimin_dokümanda_tekrar_adedi} / \text{dokümandaki_bütün_terimlerin_adedi})$ şeklinde bir dönüşümle bağıl sıklıklara dönüştürülür. Bir doküman ile bir sorgu cümlesi arasındaki benzerlik için vektör uzayı modelinde genellikle cosine benzerlik yöntemi kullanılmaktadır.



2.3.4. n-gram Modelleri

n-gram modelleri bag-of-words çözümlemekten daha fazlasıdır. Bag-of-words çözümleme bir metin içerisinde yer alan bütün terimlerin sıklık bilgilerinin kullanılmasını ifade etmekte iken n-gramlar sıklık bilgisine ek olarak terimlerin sırası ile de ilgilenmektedir. Sadece terim sıklıkları ile ilgilenen bir yöntem için “Orta Anadolu” veya “Anadolu Orta” aynı şey iken n-gram için bu ikisi birbirinden farklı şeylerdir.

n-gram modelleme dil tanıma haricinde; yazar tanıma, gen sınıflandırma ve konu başlığı / metin sınıflandırma gibi metinsel görevlerde kullanılmıştır.

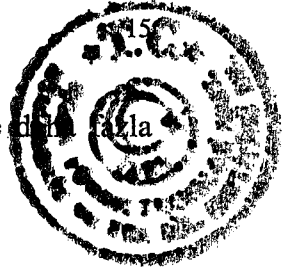
n-gram modelleme geniş bir uygulama alanına sahiptir. Dil tanıma için genellikle 3gram kullanılmakla birlikte digram ve unigramlarda kullanılmıştır. Unigram ile dil tanıma daha çok konuşma tanıma etki alanında kullanılmıştır. Bu yöntemde tek karakter yerine tek ses unigram olarak modellenmiştir [26]. Bu çalışma sesli dil tanıma ile ilişkili olmakla birlikte bir örnek olması açısından faydalıdır. Yöntemin başarısı üç dilden oluşan bir sistem için ortalama %93.5 civarındadır. Digram ve trigram yöntemleri ile ilgili örneklerin sayısı daha fazladır.

2.4. Son Zamanlarda Yapılan Çalışmalar

Kelime tabanlı dil tanıma çalışmaları için bir örnek son zamanlarda gerçekleştirilmiştir. Zhdanova [32] tarafından yapılan çalışmada sözlük oluşturma dilbilimsel yöntemlerle mi yoksa istatistiksel yöntemlerle yerine getirilmesi gerektiği karşılaştırmalı olarak verilmiştir. Dil tanıma için istatistiksel yöntemlerin daha uygun olduğu karşılaştırmalar sonrasında kabul edilmiştir. Geliştirilen dil tanıyıcı ile sekiz dil üzerinde test yapıldığında, sadece bir cümle içeren mesajlar için doğru tanıma oranı ortalama % 99 olarak bulunmuştur.

Dil tanımanın en sık yapıldığı yerlerden birisi web ortamıdır. Web ortamında dil tanıma için geliştirilmiş bir örnek dil tanıyıcı Linguni'dir [33]. Linguni, vektör uzayı tabanlı bir dil tanıma sistemidir. Web dokümanlarının yüzde 5 veya 10'u gibi uzunluklarda Linguni yaklaşık %100 doğru tanıma oranına sahiptir. Sistem, özellikle

konu başlığı tabanlı sınıflandırmalar için uygundur. Test işlemi iki ve dil kategorideki dokümanları sınıflandırma için yapılmıştır.



MacNamara [34] tarafından yapılan bir çalışmada, dil tanıma işlemi, 5 asır boyunca 14 dilde üretilmiş, 1872 adet yayının bibliyografik kayıtlarına uygulanmıştır. Yapay sinir ağları rutin dizi işlemeye izin verdiğinden dolayı dil tanıma için yapay sinir ağları tekniği kullanılmıştır. MacNamara'nın çalışmasında yapay sinir ağları kısa metinler üzerinde dilleri birbirinden ayırt etmek için kullanılmıştır. Yapay sinir ağları bibliyografik bilgiler gibi kısa bilgilerle oldukça başarılı sonuç vermiştir. Sinir ağlarının performansı trigramlarla ve morfolojik analizlerle karşılaştırılmıştır. Bu karşılaştırma sonrasında, aynı veri seti üzerinde; trigramlar %92, yapay sinir ağları %88 ve morfolojik analiz %85 oranında başarılı olmuştur.

Arjen Poutsma [35] dil tanıma çalışmalarında kullanılan özelliklerin adedini düşürmek için dil tanıma monte carlo tekniklerinden faydalanmıştır. Dil tanıma sistemleri için iki önemli adım modelleme ve sınıflandırmadır. Modelleme aşamasında çeşitli modelleme teknikleri kullanılmakta olup bunlardan en bilinenleri genel veya kısa kelimeler ile karakter n-gramlardır. Bilinen tekniklerin en önemli problemi eğitim için sabit ve genellikle büyük veri boyutlarına ihtiyaç duyulmaktadır. Monte Carlo tekniklerinin dil tanıma kullanılması ile birlikte eğitim verilerinin boyutu dinamikleştirilebilmekte ve daha da kısaltılabilmektedir. Bu tekniğin kullanımı ile modellemede daha az sayıda özelliğin kullanımı mümkün hale gelmektedir. Özelliklerin azaltılmasında standart sapma bilgisi kullanılmaktadır. Bu çalışma bir nevi performansla ilgili bir çalışmadır. Çünkü düşük boyutlu bir özellik seti ile dil tanımayı yerine getirmeye çalışmaktadır.

Robertson [36] dil tanıma klasik hale gelen n-gramlar hakkında bir çalışma yapmıştır. Çalışmasında dil tanıma haricinde n-gramların kullanıldığı alanları görmek mümkündür. Makale metinsel bilgi sistemlerinde n-gramların kullanımını anlatmaktadır. N-gramlar ile gerçekleştirilen uygulamalardan bir kısmı şöyledir; heceleme hata tespiti ve düzeltilmesi, sorgu genişletme, serilerle bilgi alma, imza dosyaları, sözlükten arama, metin sıkıştırma ve dil tanıma.

Tsai WH ve Chang WW [37] tarafından yapılan çalışma dillerin farklılaşması için karakteristik ses özelliklerinin modellenmesine odaklanmıştır. Yeni bir stokastik model Gaussian mixture bigram model (GMBM) olarak adlandırılmakta olup bu model akustik özelliklerin bigram istatistikleri üzerine başarılı sonuçlar elde etmeye yaramıştır. Daha fazla etkinlik için sisteme bir minimum classification error (MCE) algoritması uygulanmıştır. Simulasyon sonucu çinceyi kullanan ülkelerden biri olan tayvan için yapılmış ve %94 başarı elde edilmiştir.

2.5. Dil Tanıma için Doküman Sınıflandırma

Doküman dillerinin tanınması veri analizi açısından bir metin madenciliği uygulamasıdır [38]. Dolayısıyla dil tanıma işlemi diğer yöntem ve modellere ek olarak metin madenciliği yöntemleriyle de yerine getirilebilir. Doküman dili tanıma metin madenciliği yöntemlerinden, özellikle doküman sınıflandırma ile sıkı ilişki içerisinde. Hatta kimi sınıflandırmalara göre doküman sınıflandırma ile dil tanıma arasındaki tek fark kategorilerdir. Doküman sınıflandırma; dokümanların, içeriklerine göre daha önceden belirlenmiş kategori veya sınıflara atanmasını ifade ederken dil tanıma dokümanların dil kategorilerine göre sınıflandırılmasını ifade eder.

Doküman sınıflandırma için çeşitli algoritmalar kullanılmakta olup centroid tabanlı metin sınıflandırma bunlardan birisidir [39]. Centroid tabanlı metin sınıflandırma algoritması son zamanlarda kullanılmaya başlanmış, doğrusal zaman karmaşıklığına sahip, kullanımı kolay bir algoritmadır [40]. Bu algoritmanın gerisinde vektör uzayı modeli bulunmaktadır.

2.5.1. Karakter Tabanlı Dil Tanıma Yöntemi

Dil tanıma için temel olarak; özel harfler, kısmi harf dizileri, n-gramlar (harf ve kelime), kısa veya genel kelimeler gibi yöntemler bulunmaktadır. Bu yöntemler incelendiğinde dil tanımada alfabenin en temel birimi olan harflerden kelime öbeklerine kadar bir dizi özellik setinin kullanıldığı görülecektir.



Harfler dil tanımada daha çok ikili ve üçlü diziler şeklinde kullanılmıştır. Unigram yöntemi de dil tanıma ile ilgili literatürde geçmekle birlikte tek karakter manasında değil tek ses manasında kullanılmıştır. Tek ses kimi zaman birden fazla harften meydana gelebilmektedir. Bunlara bir örnek “ch” diğer örnek “sh” tır.

Bu yöntemlere ek olarak dil tanıma için bireysel karakterler kullanılabilir. Karakter tabanlı yöntem Takçı ve Soğukpınar [13, 41] tarafından önerilmiştir. Diğer yöntemlerde karakterler diziler şeklinde kullanılmakta iken bu yöntemde karakterler bireysel olarak ele alınmaktadır. Yani harf dizisi yerine bireysel harfler kullanılmaktadır. Bireysel harflerin dokümanlar için ve diller için sıklıkları elde edilmekte ve karakterlerin sıklık bilgilerine dayalı olarak dil tanıma yerine getirilmektedir. Birçok sınıflandırma uygulamasında olduğu gibi burada da dilin seçiminde maksimum benzerlik kullanılmaktadır.

Karakter tabanlı dil tanıma bireysel karakterlerle (tek karakter) dil tanımayı yerine getirmektedir. Bireysel karakterlerin ikili veya üçlü karakterler kadar bilgi taşımadığı ortadadır. O yüzden ek bazı işlemlerle karakterlerin taşıdığı bilgi miktarını artırmak gerekmektedir. Karakterlere ek bilgi karakterlerin diller arasındaki sıklık bilgilerinden elde edilmiştir. Yönteme yapılan iyileştirmeler sonrasında yaklaşık 3gram yöntemi kadar başarılı sonuçlar elde edilmiştir.

Bireysel karakterlerle dil tanıma diğer dil tanıma yöntemlerine oranla çok daha hızlı olduğu için (60 adet özellik için 2gram yönteminde $60 \times 60 = 3600$, 3gram yönteminde $60 \times 60 \times 60 = 216000$ durum bulunur) gerçek zamanlı sistemlerle birlikte çalışabilir durumdadır. Ayrıca 2gram ve 3gram yöntemlerinde eğitim sistemi için olası bütün n-gram istatistikleri kullanılamayacağı için o yöntemlerde eğitim eksik kalabilmektedir.

3. KARAKTER TABANLI DİL TANIMA (KTDİT)

Karakter tabanlı dil tanıma; bireysel karakter ismi verilebilecek tek karakterlerle yerine getirilen bir dil tanıma yöntemidir. Karakterler diğer dil tanıma yöntemlerinde de geçmiş fakat onlarda daha çok diziler şeklinde kullanılmıştır. Bu çalışmada karakterler bireysel olarak ele alınmış ve bireysel karakterler üzerinde çalışma yapılmıştır. Özellik kümesini oluşturan karakterler daha çok harf karakterleridir. Yönteme harf tabanlı yerine karakter tabanlı denilmesinin sebebi kimi zaman harfler haricindeki karakterlerin de ayırt edici bilgi sağlayabilmesinden dolayıdır. Örneğin İtalyanca ve Fransızca dillerindeki kesme işareti birçok harf özelliğinden daha fazla bilgi sağlayabilmektedir. Sisteme zamanla harfler haricindeki karakterlerin ilavesi düşünülmektedir. Hatta yerine göre boşluk karakterleri bile ayırt edici bilgi sağlayabilmektedir.

3.1. Karakterlerin Dil Tanımda Kullanılması

Dil tanıma çalışmalarının önemli problemlerinden birisi özellik kümesinin boyutu ile ilgilidir. Özellik kümesi boyutu genellikle çok büyüktür. Bu yüzden dil tanımda boyut azaltma bir ihtiyaçtır. Dokümanların daha küçük özellik kümesiyle sunulması dil tanıma sistemleri için önemli bir özellik olan etkinliği sağlayacaktır. Örneğin, kelimeler yerine karakterlerin (özellikle de harfler) dil tanımda kullanımı ile boyut azaltılabilir. Kelimeler bir doküman içinde nasıl rasgele dağılmıyorsa karakterler de rasgele dağılım göstermezler. Farklı dillerde yazılmış dokümanlar incelendiğinde aynı dilde yazılmış dokümanların birbirine daha yakın karakter sıklıklarına sahip oldukları görülmüştür. Karakter sıklıkları, dokümanın dili gibi bilgileri elde etmede kullanılabilir. Çünkü doküman dili tanıma, dilin fiziksel özellikleriyle ilişkili bir çalışma olup karakter sıklıkları da dokümanın fiziksel özelliğini ortaya koyabilmektedir.

Karakterler, karakter n-gramlar şeklinde yazar tanıma tipi işlerde kullanılmıştır. D. Benedetto ve arkadaşlarının [27] makalesinde de karakterlerin dil tanıma ve yazar tanıma tipi işlerde kullanılabileceği yönünde bilgiler mevcuttur. Konuyla ilgili literatür tarandığında karakterlerin genellikle ikili ve üçlü diziler

şeklinde kullanıldığı görülmüştür. İkili ve üçlü karakter dizilerinin özellik kümesi boyutu maalesef büyüktür. 30 karakterden oluşan bir alfabede ikili karakter dizilerinin adedi $30 \times 30 = 900$, üçlü karakter dizilerinin adedi ise $30 \times 30 \times 30 = 27000$ dir.

Dil tanıma için karakter dizileri yerine kelimeler kullanıldığında da özellik kümesinin boyutu sözlükteki kelimeler kadar olacaktır. Orta büyüklükte bir sözlükte 1500 ile 15000 civarında kelime bulunduğu düşünüldüğünde kelime tabanlı dil tanımanın özellik kümesi boyutunun da küçük olmadığı görülecektir [42].

Karakter dizileri ve kelimelerin yerine alternatif olarak bireysel karakterler kullanılabilir. Bireysel karakterler için özellik adedi alfabede yer alan harf adedi kadardır. Eğer alfabe 30 harften oluşuyorsa özellik adedi 30, 60 harften oluşuyorsa 60 dır. Bireysel karakterler ikili veya üçlü karakter dizileri kadar bilgi taşımamakla birlikte etkinlik bakımından onlardan üstündür. O yüzden bireysel karakterler etkinliğin önem kazandığı, gerçek zamanlı uygulamalarda dil tanıma için kullanılabilir. Bu arada ikili ve üçlü harf dizileri daha fazla bilgi taşımakla birlikte aradaki doğru tanıma farkı fazla değildir. Aradaki farkın az olmasının iki sebebi vardır. Bunlardan birincisi; karakter dizileri yönteminde eğitim işlemi genellikle eksik kalmaktadır. Örneğin, alfabesi 30 karakterden oluşan bir dil için ngram adedi binlerce olabilir, bu ngramların tamamı eğitim için kullanılamayacağından bir seçme yapılarak sıklığı en fazla olan 300-500 tanesi seçilir. Böylece bütün ngramların sistemde kullanılamayışından dolayı eğitim eksik kalmaktadır. Diğer sebep ise; bireysel karakterler yönteminde, bireysel karakterlerin diller arasındaki yer alma olasılıklarından her bir karaktere maksimum ayırt edici özellik değeri verilir. Farklı bir ağırlıklandırma yöntemi ile bu işlem yapılmaktadır. Dolayısıyla bireysel karakterler de karakter dizileri ve kelimeler gibi dil tanımada kullanılabilecek özelliklerdir.

Karakter özellik kümesi, özellik adedini diğer yöntemlere oranla önemli derecede düşürmekte ve bu durum da karakter tabanlı dil tanıma için en büyük motivasyonu oluşturmaktadır.

3.2. Karakter Tabanlı Yöntemin Temelleri

Karakter tabanlı dil tanıma iki temele dayanmaktadır. Bunlardan birincisi; *“Karakter sıklık değerleri dil içinde ve diller arasında birbirinden farklı değerler almaktadır. Bir dilde yazılmış dokümanların karakter sıklıkları birbirine daha yakın farklı dillerde yazılmış dokümanların karakter sıklıkları birbirine daha uzaktır”*. Buradan şu sonuca varmak mümkündür; *“Karakterler metinler için ayırt edici bilgi sağlayabilir ve metinler içerdikleri karakterlerin sıklıklarına dayalı olarak sınıflanabilir. Her metin yazıldığı dil ile ilgili gruba atanabilir”*.

Çizelge 3.1. Dillere göre harf sıklıkları

Kar	9 DİL İÇİN ORTALAMA HARF FREKANSLARI								
	İNG	FRA	ALM	HOL	İTAL	POR	TÜRK	İSP	İSVEÇ
A	7,59	8,12	5,65	7,34	10,5	12,7	10,9	12,6	8,49
B	1,09	1,08	2	1,58	0,89	0,89	2,24	1,27	1,57
C	4,4	3,3	2,67	1,57	4,54	3,6	0,92	4,14	1,32
D	3,65	4,22	4,65	5,71	3,54	5,23	3,87	5,28	4,32
E	12,3	12,7	15,9	18,6	11	10,8	11	12,8	9,26
F	2,04	1,02	1,94	0,65	1,23	0,95	0,53	0,72	2,09
G	1,79	1,25	3,04	3,04	2,07	1,23	2,04	1,08	3,12
H	2,97	0,81	3,95	2,42	1,82	0,87	0,96	0,76	1,75
İ	8,19	7,35	7,55	7,21	10,7	6,33	11,8	6,42	5,24
J	0,06	0,47	0,28	1,66	0,06	0,33	0,11	0,45	0,57
K	0,38	0,21	1,55	2,09	0	0,01	4,24	0,48	3,25
L	4,15	5,1	3,73	3,83	5,62	2,63	7,8	5,73	4,97
M	2,71	2,77	2,46	2,45	2,96	4,27	4,15	2,43	3,36
N	8,29	7,32	9,54	9,84	7,1	4,95	7,14	6,99	7,91
O	7,93	5,4	2,89	5,57	8,67	10,6	2,43	8,09	4,02
P	2,53	2,76	1,11	1,43	2,73	2,88	0,64	2,65	1,72
Q	0,07	0,79	0,01	0,01	0,56	1,01	0	0,7	0,01
R	6,89	6,73	7,71	6,27	6,94	6,84	7,25	6,6	8,89
S	6,29	7,21	6,45	4,15	4,95	8,09	3,64	7,27	6,09
T	9,8	6,1	6,49	6,4	5,78	4,8	3,89	4,52	8,08
U	2,6	5,23	3,64	1,94	3,38	3,78	2,48	3,8	1,81
V	1,34	1,19	0,96	2,49	1,81	1,39	1,3	1,08	2,09
W	0,98	0,14	1,31	1,43	0	0,01	0,03	0,04	0,07
X	0,21	0,36	0,07	0,04	0,02	0,2	0	0,19	0,23
Y	1,23	0,21	0,09	0,08	0	0,01	2,52	0,74	0,58
Z	0,11	0,19	1,12	1,23	0,44	0,29	1,25	0,65	0,01

Çizelge 3.1’de Latin kökenli dillerin birçoğunda kullanılan karakterlerin sıklık bilgileri yer almaktadır. Çizelge, dillere göre harflerin kimisinin daha sıklıkla çıktığını kimisinin daha seyrek çıktığını göstermektedir. Çizelge 3.1’de görülen sıklık değerleri arasındaki farklılıkların fazla olması dil tanıma açısından istenen bir durumdur.

Çizelge 3.1. de görüldüğü gibi karakter sıklıkları dil içinde ve diller arasında birbirinden farklı değerler içermektedir. Sıklık değerlerinde bazı yakınlıkların olması dillerin birbirine tarihsel yakınlığından dolayıdır. Fakat bu yakınlık dokümanları farklı sınıflarda gruplamaya engel olmayacak derecede yakınlıklardır.

Karakter tabanlı dil tanıma için ikinci temel dayanak; “*Her dilin kendine has özel karakterlere sahip olmasıdır*”. Örneğin, ‘ş’, ‘Ş’, ‘ı’, ‘İ’ gibi karakterler sadece Türkçe’de bulunan karakterlerdir. Metin içinde bu özel karakterlerden biri veya birkaçına rastlanması o metnin hangi dilde yazıldığını, yani dilini verecektir. Bu işlem dil tespitini hızlandırması bakımından önemlidir. Dolayısıyla metinler öncelikle içinde özel karakterler olup olmamasına göre sınıflanmaya çalışılır, içinde özel karakterlerden biri veya birkaçı olmayan metinler, metin içindeki karakterlerin sıklıkları sayılarak sınıflanır.

Dile ait özel karakterler dil tanıma için en önemli bilgiyi içermektedir. Hemen her sınıflandırma uygulamasında olduğu gibi dil tanımada da amaç az rastlanan özelliklere maksimum değeri vererek ayrışmayı derinleştirmektir. Özel harfler genellikle az rastlanan karakterlerdir. Bunlara uygulanacak bir dönüşüm ile değerleri artırılabilir. Örneğin, özel harf için ham sıklık değeri (fr) ise aynı harf için dönüştürülmüş sıklık değeri (1/fr) olacaktır.

3.3. Karakter Tabanlı Dil Tanıma Modeli

Bir model, gerçek dünya proseslerinin soyut bir sunumudur. Her bir model parametreler ve değişkenlerden meydana gelir ve verilerin model yapısı için uygun parametreler tahmin yoluyla seçilir. Veriye model uydurma için skor fonksiyonları kullanılır ve model için en uygun parametreler seçilir. Bütün bunların öncesinde modele uygun bir fonksiyonel form seçilir [43].

Karakter tabanlı dil tanıma, karakter sıklıklarından doküman dilini tahmin etmeye çalıştığı için bir tahminsel modelleme çalışmasıdır. Bir tahminsel modelde, değişkenlerden biri diğerlerinin bir fonksiyonu olarak sunulur. Cevap değişkeninin değeri diğer değişkenlerin değerlerinden tahmin edilir. Burada tahminde kullanılan

değişkenler tahminci adını alırlar. Karakter tabanlı modelde karakterler tahmincilerdir.

Model tipi seçiminde tahmin tipine uygun bir model tipi belirlenir. Seçilen tahmin tipi regresyon ise model olarak bir yapay sinir ağı kullanılabilir. Eğer seçilen tahmin tipi sınıflandırma ise bunun için karar ağacı gibi bir algoritma kullanılabilir. Modellemede sıklıkla istatistiksel modellerde kullanılmakta olup bunlardan bazıları; logistic regresyon, discriminant analysis veya genel doğrusal modellerdir [44].

Bu çalışmada tahmin tipi için sınıflandırma seçilmiştir. Sınıflandırma; önceden belirlenen sınıflara yeni elemanların dağıtılması işlemidir. Dağıtma işlemi yapılırken elemanların özelliklerinden faydalanılır. Eleman özellikleri X , elemanın atanacağı sınıf Y ile verilecek olursa; sınıflandırma, X 'ten Y 'ye bir eşleme olarak düşünülebilir. Eşlememin sol tarafında bulunan X değişkenleri genellikle ölçüm değerlerini diğer tarafta bulunan Y değişkeni ise tahmin değerini tutar.

3.3.1. Modelin Formal Dilde Gösterimi

Karakter tabanlı dil tanıma modeli eğitim ve test aşamalarından meydana gelir. Model önce eğitim dokümanları tarafından eğitilmekte ardından test dokümanları ile test edilmektedir. Karakter tabanlı model formal dilde aşağıdaki şekilde ifade edilebilir.

Σ sembolü ortak alfabeyi göstermek üzere; alfabe $\Sigma = \{a, b, c, \dots, z\}$ şeklinde verilebilir. Σ alfabesini kullanarak yazılabilecek n karakterli kelimeler; Σ^n olacaktır. Σ^n aynı zamanda n -gram olarak ta bilinen n karakterli harf dizileridir. Σ alfabesi ile üretililecek olası bütün kelimelerin kümesi ise Σ^* sembolü ile verilebilir. O zaman Σ^* için 3.1'deki denklemi yazmak mümkündür.

$$\Sigma^* = \bigcup_{n=1}^{\infty} \Sigma^n \quad (3.1.)$$

Her bir doküman Σ^* uzayının bir alt kümesidir. Çünkü dokümanlar farklı uzunluklardaki harf dizilerinin farklı sayıda ve farklı sırada bir araya getirilmesi ile elde edilir. O yüzden, $D \subseteq \Sigma^*$ ifadesi yazılabilir. Bu arada ortak alfabeyi

kullanarak farklı dillerde dokümanlar üretmek mümkündür. Dokümanlar ile onların dilleri (diller numaralarla verilmekte) arasındaki ilişki 3.2’de sunulmuştur.

$$c : D \rightarrow \{1,2,...,k\}$$

(3.2.)

Dillerin bir kümesi $\{1,2,...,k\}$ olarak verilmiştir. 3.2. ifadesinde dokümanlar k adet dilden birinde olabilir. Dil tanıma sisteminde k adet dil bulunduğu varsayılmaktadır. Dil tanıma çalışmasında tahmin tipi olarak sınıflandırma seçildiği için tahmin edilen değerler kategoriktir.

D külliyyattaki dokümanların bir kümesini ve d_i külliyyat içerisindeki i. dokümanı vermek üzere; D ile d_i arasındaki ilişki 3.3’de verilmiştir.

$$D = \{d_1, d_2, ..., d_i, ..., d_n\} \quad (3.3.)$$

Her bir d_i dokümanının çeşitli şekillerde sunumu mümkündür. Bu sunumlarda genellikle dokümanlar özellikleri cinsinden sunulur. Bu çalışmada doküman özellikleri olarak karakterler, özellik değerleri olarak ta karakterlerin doküman içerisinde kaç kez yer aldıkları yani sıklık bilgileri kullanılmıştır.

Belirlenen özellik değerleri doküman alma etki alanında genellikle kullanılan vektör uzayı modelinde sunulmuştur. Vektör uzayı modelinde; d_{ip} i. dokümandaki p. karakterin sıklığı olmak üzere bir d_i dokümanı aşağıdaki gibi bir doküman harf vektörü ile ifade edilmiştir.

$$\vec{d}_i = (d_{i1}, d_{i2}, ..., d_{im}) \quad (3.3.)$$

fr_{ip} i. dokümandaki p. karakterin sıklığı ve d_{ip} , fr_{ip} değerinin bağıl sıklığı veya normalleştirilmiş değeri olmak üzere fr_{ip} değerinden d_{ip} değerinin hesabı 3.4’deki gibi yapılmaktadır. Bağıl sıklığın hesaplanması doküman uzunluklarından kaynaklanabilecek hataları ortadan kaldırmak için yapılır. fr_{ip} ile d_{ip} arasındaki ilişki şöyledir.

$$d_{ip} = \frac{fr_{ip}}{\sum_{p=1}^m fr_{ip}} \quad (3.4.)$$

Bu arada bağıl karakter sıklıkları toplamı 1 değerine eşittir.



$$\sum_{p=1}^m d_{ip} = 1$$

d_{ip} değeri d_i dokümanına ait p . karakterin özellik değeri, aynı zamanda d_i vektörünün p . boyutudur. Doküman vektörünün her bir boyutu bir karaktere ait karakter sıklık değerini tutmaktadır.

Dili bilinen dokümanlar (d_i, y_i) gibi çiftler şeklinde sunulurlar. Burada y_i , i . dokümanın dilini bir rakam ile vermektedir. Örneğin 123 numaralı dokümanın TÜRKÇE olsun ve TÜRKÇE diller kümesinde 7. sırada yer alsın. O zaman bahsedilen doküman eğitim verileri arasında $(d_{123}, 7)$ şeklinde sunulacaktır. (d_i, y_i) çiftlerindeki i değeri 1 ile n arasında, y_i 1 ile k arasında değer alacaktır ($i=\{1,2,\dots,n\}$, $y_i=\{1,2,\dots,k\}$).

Eğitim aşamasında dokümanın dili bilinmekte ve y_i ile sunulmaktadır. Fakat test aşamasında dokümanın dili bilinmemekte ve tahmin edilmeye çalışılmaktadır. Dili bilinmeyen dokümanlar genellikle X ile ifade edilmekte ve bu dokümanlardan i . sıradaki doküman x_i ile verilmektedir. x_i için tahmin edilecek dil değeri ise $\hat{y}_i$ ile sunulur. Bu değer y_i değerinden farklıdır çünkü $\hat{y}_i$ değeri tahmin edilen bir değeri ifade etmektedir. Dili bilinmeyen test dokümanlarının dilini bulmak için $\hat{y}_i$ ile x_i arasındaki ilişkiyi açığa çıkarmak gerekmektedir. Bu ilişkiyi açığa çıkarmak için 3.6. denklemi kullanılabilir.

$$\hat{y}_i = \arg \max_{j=1..k} (Sim(\vec{x}_i, \vec{C}_j)) \quad (3.6)$$

3.6 ile bulunan $\hat{y}_i$ değeri her zaman doğru olmayabilir. Tahmin sonucu tespit edilen dil hakikaten dokümanın dili ise tahmin işlemi başarı ile sonuçlanmış aksi taktirde başarısız sonuçlanmıştır. Hatalı tahminlerin toplamı sistemin hata miktarını verecektir. Eğer tahmin edilen dil değeri doğru dil değeri ise sistemin o doküman için hata miktarı 0 (hatasız, doğru), aksi taktirde hata miktarı 1 olacaktır. y_i olması gereken dil bilgisini vermek üzere; n elemandan oluşan bir tespit sisteminde, y_i ve $\hat{y}_i$ değerleri arasındaki farklılıklar toplamı veya sistemin hata oranını bulmak için 3.7. yazılabilir. (Sistemin hata oranı E_T sembolü ile gösterilmiştir.)

$$E_T = \frac{\sum_{i=1}^n E(y_i, \hat{y}_i)}{n} \quad (3.7.)$$

$E(y_i, \hat{y}_i)$ değeri doğru tahmin için 0 hatalı tahmin için 1 değerini verir. 3.6'da görüldüğü gibi dil tanıma için C_j değerine ihtiyaç duyulmaktadır. Bu değer j . dil için centroid değeridir. Yani, j . dildeki karakterlerin özellik değerlerini tutan vektördür. Centroid değeri eğitim verilerinden elde edilmekte olup, bir dil için centroid hesabı aşağıdaki şekilde yapılır.

Centroid hesabından önce her bir dilin karakterlere dayalı ortalama sıklıklarının bulunması lazımdır. Bu işlem bir dilden örneklerle ait karakter sıklıklarının bir ortalamasıdır. Örneğin, 1KB uzunluğunda 100 adet örnek doküman için karakter sıklıkları elde edip daha sonra bu sıklıkların bir ortalamasını almak dil için ortalama karakter sıklığını hesap etmede kullanılır. Ortalama karakter sıklıklarının hesabı 3.8. deki gibidir.

$$\vec{A}_j = \frac{1}{n} \sum_{i=1}^n \vec{d}_{ji} \quad (3.8.)$$

A_j , bir dil için ortalama sıklık değerlerini, m , özellik adedini göstermek üzere her bir dilin sıklık değerleri, $A_j = (a_{j1}, a_{j2}, \dots, a_{jm})$ şeklinde ifade edilebilir. Ortalama karakter sıklıklarının birden fazla olduğu düşünüldüğünde o zaman bütün diller için ortalama karakter sıklıklarının bir kümesi; $\{A_1, A_2, \dots, A_k\}$ olacaktır. A_k , k . dile ait ortalama sıklık değerini vermektedir. İndisi j olan dil için özellikler cinsinden ortalama sıklık değerlerinin sunumu aşağıdaki gibi yapılabilir.

$$A_j = (a_{j1}, a_{j2}, \dots, a_{jm}) \quad (3.9.)$$

Ortalama karakter sıklıklarından centroid hesabı için bir dönüşüme daha ihtiyaç duyulmaktadır. O dönüşüm, ortalama karakter sıklıklarına bir ağırlık değeri verilmesi ile ilgilidir. Bahsedilen dönüşüm için her bir karakterin diller içerisindeki meydana gelme olasılığını kullanmak ve daha sonra bu değeri log dönüşümü ile düzgünleştirmek uygun olacaktır. Çünkü log fonksiyonu IR etki alanında sıklıkla kullanılan bir düzgünleştirme fonksiyonudur. $a_{(T)p}$, p . karakterin bütün dillerdeki sıklıkları toplamını ve c_{jp} j . dildeki p . karakter için centroid değerini göstermek üzere aşağıdaki denklemler yazılabilir;

$$a_{(T)p} = \sum_{i=1}^k a_{jp} \quad (3.10.)$$

$$c_{jp} = \log(a_{jp} * 100 / a_{(T)p}) \quad (3.11)$$

3.6. denkleminde benzerlik bulma için Sim isminde bir fonksiyon kullanılmaktadır. Bu fonksiyon cosine benzerlik fonksiyonudur. Fonksiyon açık olarak 3.12. de verilmiştir. 3.12 denkleminde payda kısmının kaldırılması ile de metin puanları elde edilmektedir.

$$Sim(\vec{x}_i, \vec{C}_j) = \frac{\sum_{p=1}^m \vec{x}_{ip} \cdot \vec{C}_{jp}}{\sqrt{\sum_{p=1}^m (\vec{x}_{ip})^2} \cdot \sqrt{\sum_{p=1}^m (\vec{C}_{jp})^2}} \quad (3.12.)$$

Hata oranının minimize edilmesi dil tanıma sisteminin en önemli konularından birisidir. Hata oranını düşürmek için karakter tabanlı dil tanıma modelinde yapılması gereken üç önemli iş vardır.

1. Dokümanlar en uygun özelliklerle sunulmalıdır. Bunun için özellik seçimi tipinde işler yapılmalıdır.
2. Doymuş centroid değerlerinin hesap edilmesi için gerekli eğitim verisi miktarı ayarlanmalıdır.
3. Farklı benzerlik veya uzaklık bulma yöntemleri sisteme entegre edilerek hata oranı düşürülmelidir.

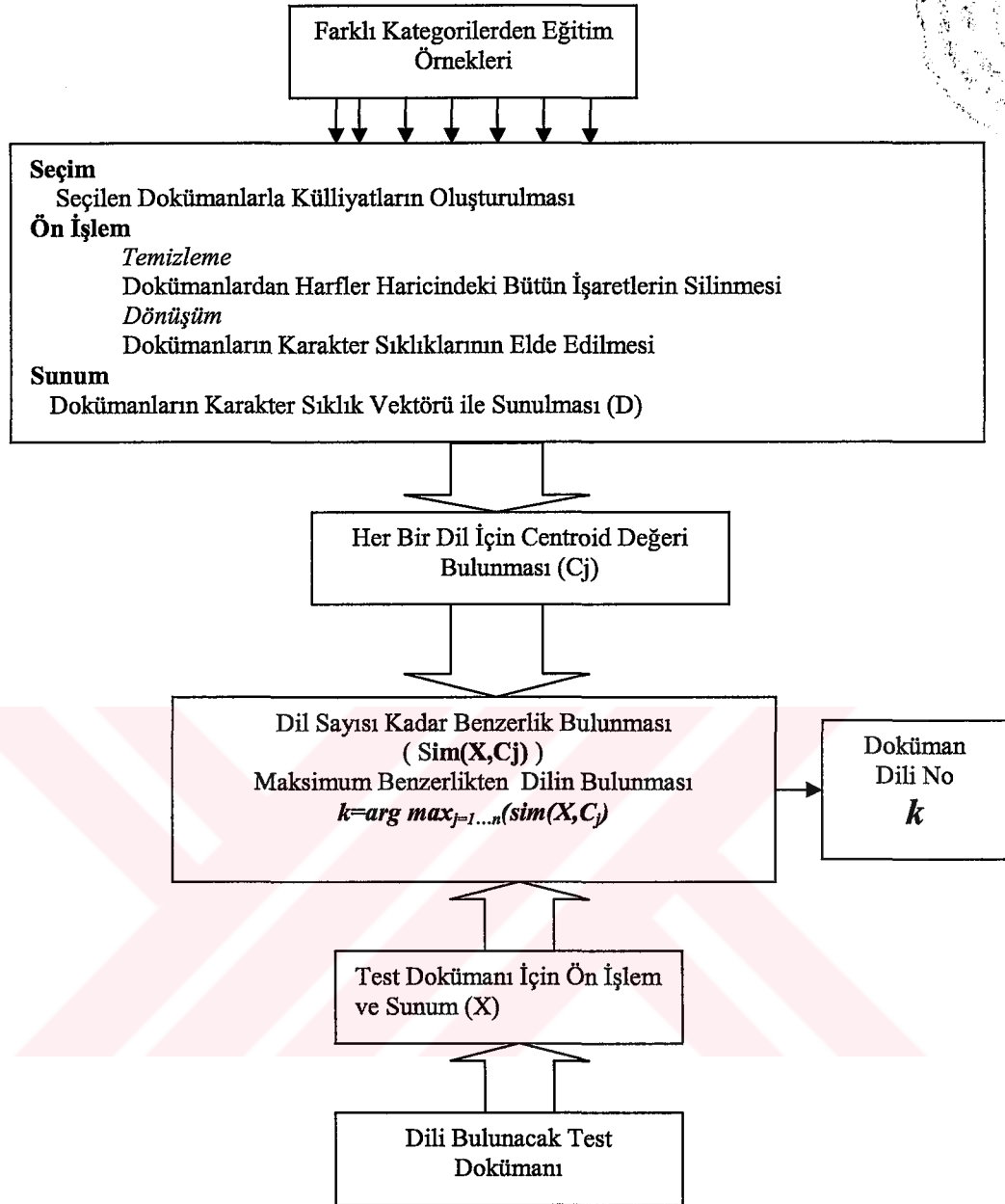
3.4. Karakter Tabanlı Dil Tanıma Sistemi

Dil tanıma işlemini yerine getiren yazılım ve donanımlara dil tanıma sistemi veya dil tanıyıcı adı verilir. Dil tanıma sistemleri sayısallaştırılmış konuşmaların veya metinsel mesajların dillerini tanımaya yarayan sistemlerdir. Bu sistemler; arama motorlarında, çağrı merkezlerinde, iletişim acentelerinde, kısacası çok miktarda elektronik verinin olduğu her yerde kullanılırlar. Dil tanıma sistemlerinin girişinde bir metinsel mesaj veya sayısallaştırılmış konuşma metni çıkışında ise metnin veya sayısallaştırılmış konuşmanın hangi dilde olduğu ile ilgili bilgi bulunmaktadır. İşlem kısmında metne bazı yöntemler uygulanarak metnin dili bulunmaya çalışılır. Bu aşamada yapılan işlemler dil tanıma yöntemlerine göre farklılıklar gösterir. Kimi dil tanıyıcılar metin içindeki harf dizilerini kontrol ederek, kimi dil tanıyıcılar dillere ait

sözlüklerde yer alan kelimelerin var olup olmadığına bakarak, kimileri de tekil harf kombinezonlarını kontrol ederek dili tespiti çalışır.

Dil tanıma sistemlerinin geliştirilmesi için çeşitli sebepler bulunmaktadır. Bu sebeplerin başında İnternet sayesinde uluslar arası iletişimlerin artmış olması vardır. İnternette alışveriş yapan taraflar birbirlerinin dillerini anlamaya ihtiyaç duymaktadırlar. Bir diğer sebep web üzerinde dolaşan verilerin ve elektronik veritabanlarının büyümüş olmasıdır. İnternet'te birbirinden farklı dillerde dokümanlar bulunmakta ve bu dokümanların içeriğini anlayabilmek içinde önce onların dillerini tanımak gerekmektedir. Elektronik mesaj sınıflandırma ve otomatik cevaplandırma sistemleri ile ilgili problemleri çözmenin ilk aşamalarından birisi de dil tanımadır. Sistem önce dili tanıyacak daha sonra tanınan dile göre cevaplama ve sınıflandırma işlemini yerine getirecektir. Dil tanıma sistemleri için istenen en önemli iki özellik ise; kısa mesajlarda en yüksek doğruluk, uzun mesajlarda ise en yüksek hızın yakalanmasıdır [32].

Karakter tabanlı dil tanıma sisteminin en önemli iki özelliği; bireysel karakterler ve centroid tabanlı sınıflandırmadır. Sistemin mimarisi Şekil 3.1'de gösterilmiş olup sistemin iki temel aşaması bulunmaktadır. Birinci aşamada eğitim verileri ile dil tanıma sistemi eğitilmekte ve eğitim verilerinden centroid değerleri öğrenilmekte, ikinci aşamada ise öğrenilen centroid değerleri ile test dokümanlarının dili tespit edilmektedir. Dil tespitinde benzerliklerden faydalanılmaktadır.



Şekil 3.1. KTDT Sistem Mimarisi

Karakter tabanlı dil tanıma sisteminin geri planında vektör uzayı modeli bulunduğu için sistemde her bir doküman ve her bir dil bir vektör ile sunulmaktadır. Bu vektörler bireysel karakter sıklıkları içerir. Sistemi diğer dil tanıma sistemlerinden farklı hale getiren en önemli özellik, kullanılan özellik kümesinin bireysel karakterlerden oluşmasıdır.

Bu çalışmada dil tanıma bir sınıflandırma uygulaması olarak gerçekleştirilmiştir. Doküman sınıflandırma için centroid tabanlı sınıflandırma algoritması seçilmiştir.

Algoritmanın yapısına uygun olarak her bir dil centroid adı verilen bir vektör ile sunulmuş, daha sonra dili bulunacak test dokümanları da bir doküman karakter vektörü ile sunularak işleme hazır hale getirilmiştir. Centroid vektörleri ile doküman karakter vektörleri arasındaki maksimum benzerliklerden test dokümanının dili tespit edilmiştir.

3.4.1. Eğitim Aşaması

Karakter tabanlı dil tanımanın eğitim hedefi; eğitim dokümanlarından centroid değerlerinin elde edilmesidir. Centroidler elde edilmeden önce ham dokümanların ön işleminden geçirilmesi ve karakter sıklıkları cinsinden sunulması bir ihtiyaçtır. Doküman karakter sıklıklarına dokümanın dil etiketi de eklenerek dil tanıma için gerekli eğitim verisi elde edilmektedir.

3.4.1.1. Ön işlem

Karakter tabanlı ön işlemde diğer dil tanıma işlemlerinde yer alan heceleme, gövdeleme gibi ön işlem adımlarına gerek duyulmaz sadece temizleme ve dönüşüm işlemine gerek duyulmaktadır. Temizleme; harfler haricindeki karakterlerin metinden çıkarılması, dönüşüm de, metnin karakterlere dayalı sıklıklarının bulunmasıdır. Dönüşüm sonrasında metinlerin sıklık tabanlı özetleri elde edilir. Şekil 3.4’de İsveç dilinde yazılmış örnek bir metin yer almaktadır.

Tjänsten gick i stället till C. A. Hägglund. Tvisten upptäcker på allvar Edith Södergran. Han träffar Hagar En vansinnig världs klokhet [...] enligt all sannolikhet för att få ro oavsiktligt intagit en för nietzscheanskt färgade vilja att skapa den nya människan.

Şekil 3.2. İsveççe olarak yazılmış örnek metin

Temizleme

Veri setinin karakter tabanlı yöntemle hazır hale getirilebilmesi için, daha doğrusu karakter sıklıklarının elde edilebilmesi için önce külliyatta yer alan metinler temizleme işlemine tabi tutulur. Karakter tabanlı temizlemede karakterler haricindeki bütün işaretler ve boşluk karakterleri çıkarılır. Karakter tabanlı temizleme yapıldıktan sonra metin Şekil 3.3’deki hale gelecektir.

Tjänstengickistället till CA Hägglund Tvistenuptäcker på allvar Edith Södergran Hantärf
far Hagar Envansinnig världskloketenligtallsannolikhet för att få oavsiktligt tagitenf
örnietzscheanskt färgade vilja att skapad en ny människan

Şekil 3.3. Karakter tabanlı ön işlenmiş İsveççe örnek metin

Külliyyat, temizleme işleminin ardından dönüşüme hazır hale gelmiştir. Dönüşüm esnasında dokümanların karakter tabanlı sıklıkları bulunur.

Metinsel Verinin Sayısal Hale Dönüşümü

Metinsel veriye sınıflandırma algoritmaları uygulanabilmesi için, metinsel verinin dönüşümü bir ihtiyaçtır. Dönüşüm; dokümanların bir formdan başka bir forma dönüştürülmesi işlemidir. Dönüşüm için en sık kullanılan yöntem istatistiksel özetleme yöntemidir [45]. Bu aşamada dokümanların istatistiksel özetleri çıkarılır ve bu aşamanın sonunda dokümanların karakter sıklık bilgileri elde edilmiş olur. Dokümanların terim (bu çalışmada karakter) sıklıkları ile özetlenmesi fikri Luhn tarafından önerilmiştir [46]. Bu çalışmada uygulanacak dönüşüm işleminde metinler karakter sıklıklarına dönüştürülecektir.

Metinler, dönüşüm sonrasında karakter sıklıklarına çevrilir. Karakter sıklıkları dizi veya vektör gibi bir veri yapısında tutulabilir. Bu dizide veya vektörde; boyutlar karakterleri ve boyut değerleri karakterin metin içindeki tekrar sayısını verir. Örneğin; ‘a’ karakter sıklık vektörünün 1 numaralı boyutu olsun, karakter sıklık vektörünün 1. boyutunun değeri metinde geçen ‘a’ karakterlerinin adedini verir. Karakter sıklık vektörü için kayıt yapısı ve seçilmiş bazı metinlerin karakter sıklık vektörü değerleri Şekil 3.4’de verilmiştir.

LNG	A	à	á	Â	ã	ä	å	Æ	b	c	ç	...	v	w	X	y	z
SWE	1	0	0	0	0	3	0	0	0	2	0	...	1	0	0	0	0
SWE	4	0	0	0	0	2	1	0	0	1	0	...	1	0	0	0	0
SWE	5	0	0	0	0	1	0	0	0	0	0	...	2	0	0	0	0
SWE	4	0	0	0	0	0	1	0	0	0	0	...	1	0	0	0	0
SWE	6	0	0	0	0	1	0	0	0	1	0	...	1	0	0	0	1
SWE	3	0	0	0	0	2	1	0	0	0	0	...	1	0	0	1	0

Şekil 3.4. Örnek metinlerin karakter sıklık değerleri



3.4.1.2. Özellik Seçimi (Feature Selection)

Ön işlemde geçirilen dokümanlar için bir sonraki aşama özellik seçimidir. Özellik seçiminin amacı dokümanları en iyi ifade edecek özelliklerin bulunmasıdır. Karakter tabanlı dil tanıma işleminde özellikler olarak harf karakterleri seçilmiştir.

Karakter Özellikleri

Özellik setini oluşturan karakterler (a, b, c, d, e, f, g, h, i, k, l, m, n, o, p, r, s, t, u, v, y, z, ...) Latin alfabesinden seçilmiş 60 karakterdir. Bu karakterler deneylerde kullanılacak dillerin alfabelerinden elde edilmiştir. Karakter özellik kümesi özellik adedini önemli ölçüde azaltacaktır. Karakter kullanımı ile sağlanan boyut azaltma işlemi sayesinde doküman sınıflandırmanın hızında önemli bir artış meydana gelecektir.

3.4.1.3. Sunum

Seçilen özelliklerin sunumu için uygun bir modele ihtiyaç vardır. Karakter tabanlı dil tanıma uygulamasında, dokümanlar vektör uzayı modeli kullanılarak sunulurlar [10]. Bu modelde, her bir d dokümanı karakter uzayında bir vektör olarak düşünülebilir. En basit formda her bir doküman $\vec{d}_k = (kf_1, kf_2, \dots, kf_n)$ şeklinde bir karakter sıklık vektörü ile sunulabilir, burada kf_i dokümandaki i . karakterin doküman içinde meydana gelme sıklığıdır. Karakter Frekans (KF) vektörü için ağırlık modeli olarak frekans ağırlık modeli seçilmiştir. Dolayısıyla her bir doküman için KF vektörü onun aynı zamanda ağırlık vektörü de olacaktır. Sunumun son aşamasında her bir doküman vektörü birim vektör haline dönüştürülerek normleştirme sağlanacaktır. Bu model sayesinde dokümanlar karakter uzayında noktalar halinde düşünülebilir ve dokümanlar arası benzerlikler geometrik yöntemlerle hesap edilebilir.

3.4.2. Test Aşaması

Eğitim aşamasında olduğu gibi bu aşamada da ön işlem, özellik seçimi ve sunum işlemleri aynı şekilde yerine getirilir. Test aşamasının son işlemi test dokümanları ile diller arasındaki benzerlikler yardımıyla dil tanımanın yapılmasıdır.



3.4.2.1. Benzerliklerin Bulunması

Test dokümanlarının dilini bulmak için gerekli bilgi maksimum benzerliktir. Vektör uzayı modelinde, dokümanlar arasındaki benzerliği veya bir doküman ile bir dil arasındaki benzerliği hesap etmek için genellikle kosinüs fonksiyonu (3.12) kullanılır. Kosinüs fonksiyonunun euclidian gibi benzerlik yöntemlerine oranla daha iyi sonuç verdiği deneylerle de gözlenmiştir.

3.4.2.2. Test Dokümanlarının Sınıflandırılması

Yeni bir x dokümanının dilini bulmak için x dokümanı ile k adet dil arasındaki benzerlikler (3.13) kullanılarak hesaplanır. Centroid değerinin dili temsil ettiği varsayılmaktadır.

$$sim(\vec{x}, \vec{C}_{j,j \in k}) = \frac{\vec{x} \cdot \vec{C}_j}{\|\vec{x}\|_2 * \|\vec{C}_j\|_2} \quad (3.13.)$$

Sonuçta bu benzerliklere dayalı olarak x dokümanı kendine en benzer centroid değerini veren sınıfa atanır. Bu durum şöyle gösterilebilir.

$$\arg \max_{j=1, \dots, k} (sim(\vec{x}, \vec{C}_j)) \quad (3.14.)$$

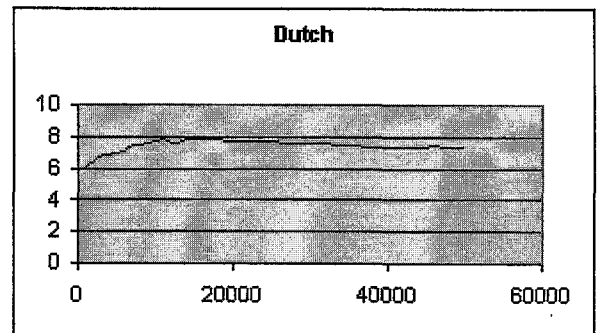
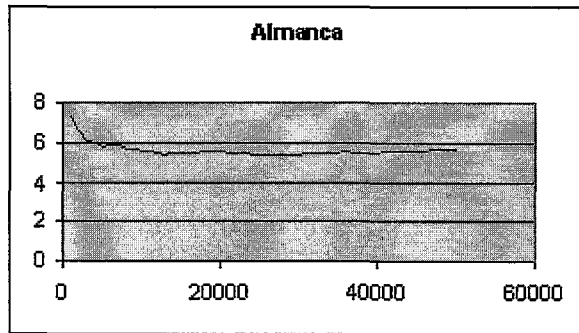
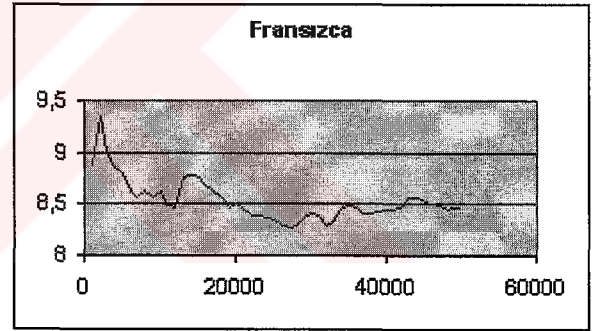
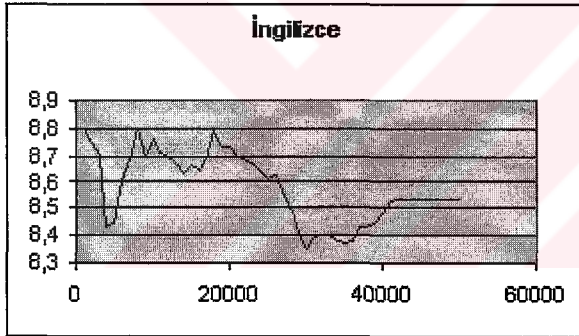
Eğitim ve test aşamalarının sonunda dokümanların dili tespit edilmiş olacaktır. İleriki bir çalışma test alt sisteminin çıktısının eğitim alt sistemine girdi olarak verilmesidir. Bu yöntem daha hassas centroid değerlerinin bulunmasını sağlayacaktır.

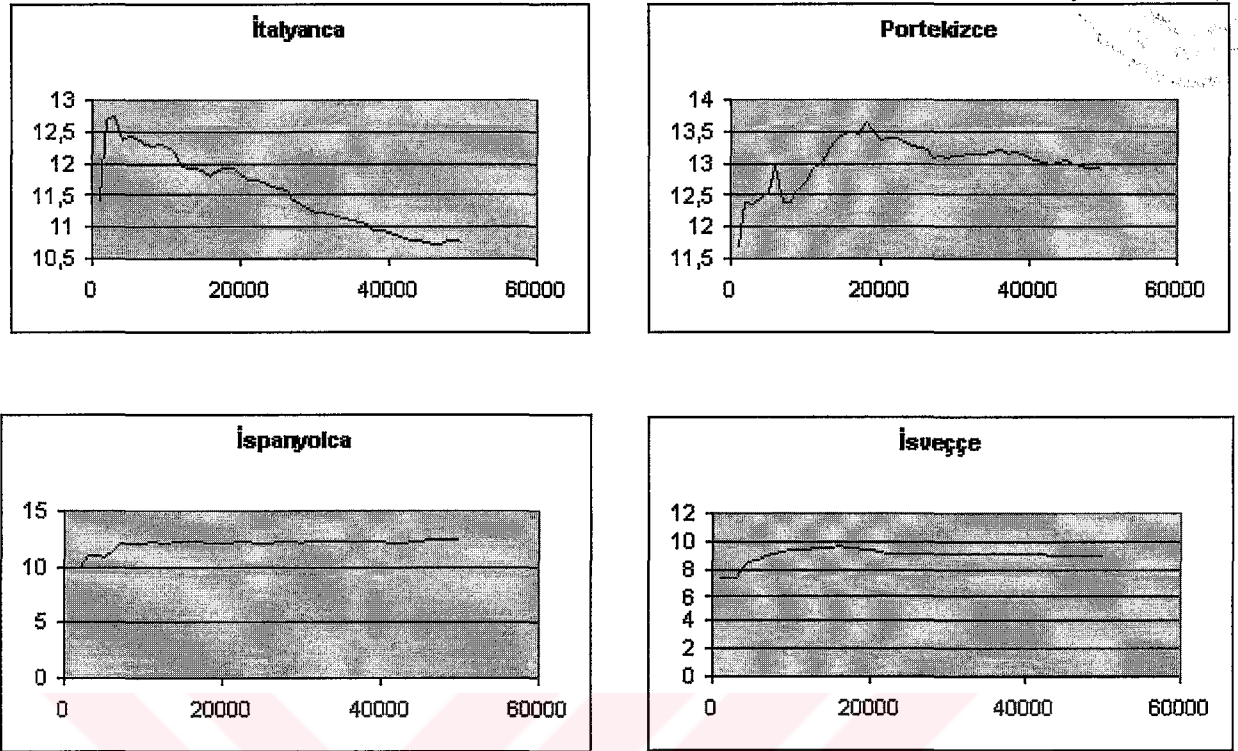
Centroid tabanlı bu sınıflayıcılar için öğrenim aşamasının hesaplamalı karmaşıklığı doküman sayısı ve eğitim setindeki karakter sayıları ile paraleldir. Yani algoritma doğrusal bir karmaşıklığa sahiptir. Hemen hemen bütün k centroid değerleri eğitim setinde tek aşamada hesaplanabilir, her bir centroid ilgili sınıfın dokümanlarının ortalamalarına bağlı olarak hesaplanır. Yeni bir x dokümanını sınıflayabilmek için gerekli zaman karmaşıklığı $O(km)$ dir. Burada m değeri x dokümanı içinde sunulan karakterlerin adedini vermektedir. Böylece algoritmanın

hesaplamalı karmaşıklığı çok düşük ve Naive Bayesian (NB) gibi çok hızlı bir sınıflayıcıdır.

3.5. Karakter Tabanlı Dil Tanımanın Özellikleri

Bir dil tanıma sisteminin başarısını etkileyen bazı faktörler bulunur. Bu faktörlerin başında dil tanıma sisteminin eğitimi için gereken veri miktarı vardır. Karakter tabanlı yöntemin doygun hale gelmesi için gerekli veri boyutu araştırıldığında kimi dillerde bu değerin 10 KB kimilerinde ise 100 KB olduğu görülmüştür. Karakter sıklıklarının doygunlaşması ile ilgili deneyler için örnek olarak 'A' karakteri alınmış ve farklı boyutlarda sıklığın nasıl değiştiği aşağıdaki şekilde sunulmuştur 'A' karakteri deney yapılan dillerin hepsinde kullanıldığı için tercih edilmiştir. Şekil 3.5. sadece bir karakter için doygunluğu vermekte olup bütün eğitim kümesinin doygunluğu için bileşke bir değer alınmalıdır.





Şekil 3.5. Eğitim setleri için karakter sıklıklarına dayalı doygunluk grafikleri

Bir diğer faktör dili tanınan metinlerin uzunluğudur. Karakter tabanlı dil tanıma sistemi 30 karakter uzunluğundaki metinler için ortalama %75, 60 karakter uzunluğunda ortalama %90 başarı oranına sahiptir. 90 ve 100 karakterden sonra tanıma oranı birçok dil için %100 civarındadır.

Çizelge 3.2. Farlı uzunluklar için karakter tabanlı dil tanımanın doğruluk oranı

Dil	Metinlerin karakter olarak uzunlukları				
	30	60	90	100	120
İngilizce	0,540	0,845	0,951	0,970	0,980
Fransızca	0,728	0,865	0,970	0,980	0,980
Almanca	0,792	0,940	0,952	0,980	0,990
Hollanda Dili	0,774	0,929	0,969	0,990	0,980
İtalyanca	0,707	0,917	0,970	0,980	0,970
Portekizce	0,844	0,959	0,990	1,000	0,970
Türkçe	0,970	0,990	1,000	1,000	1,000
İspanyolca	0,693	0,871	0,960	0,961	0,971
İsveççe	0,734	0,990	1,000	1,000	1,000

Dil tanıma sistemi tarafından desteklenen dillerin adedi bir başka faktördür. Başta 4 dil için geliştirilen dil tanıma sistemi daha sonra 7 dili ardından 9 dili

destekler hale getirilmiştir. Latin alfabesini kullanmak koşulu ile daha fazla dilin desteklenmesi sadece dilin centroid değerinin elde edilmesine bağlıdır. Sistemin genişletilmesi oldukça kolay şekilde yerine getirilebilmektedir.

Bir başka faktör özellik kümesidir. Karakter tabanlı dil tanıma yönteminin en önemli artısı özellik kümesinin karakterlerden oluşmasıdır. Dil tanıma bir sınıflandırma uygulaması olarak yerine getirildiğinden özellik kümesinin boyutu doğrudan dil tanımanın performansını etkilemektedir. Sınıflandırma için seçilen algoritmanın hesaplamalı karmaşıklığı k ve m değerlerine bağlıdır. k değeri centroid (aynı zamanda sınıf adedi) adedini verirken m değeri özellik setinin boyutunu vermektedir. Özellik setinin boyutu kelime tabanlı yöntemlerde 1500 ile 15000 arasındadır karakter tabanlı yöntemde ise özellik setinin boyutu sadece 60'dır. Dolayısıyla karakter özellik kümesinin kullanımı m değerini düşürmekte ve performansı ciddi oranda artırmaktadır.

4. DENEYLER VE SONUÇLAR

4.1. Veri Kümesi

Karakter tabanlı dil tanıma yöntemini test etmek için ECI [47] isimli, çok dilli bir külliyat kullanılmıştır. Külliyat dilbilimsel çalışmalarda sıklıkla tercih edildiğinden ve Avrupa dilleri ile Türkçe için örnek verilere sahip olduğundan tercih edilmiştir. Külliyatın kaynağını ağırlıklı olarak Almanca (34 milyon kelime), Fransızca (4.1 milyon kelime) ve Hollanda dilinde (5.5 milyon kelime) yayın yapan basın yayın organları oluşturmaktadır. Ayrıca; İngilizce, Fransızca ve İspanyolca dillerinde paralel metinlerin yer aldığı bir organizasyondan da 5 milyon civarında kelime içeren bir külliyat ECI içinde bulunmaktadır. Diğer kaynaklar ise bütün Avrupa dilleri, Türkçe, Rusça ve Japonca'dır.

Tezde 9 dil üzerinde çalışma öngörülmüş olup bu diller; İngilizce, Fransızca, Almanca, Hollanda Dili, İtalyanca, Portekizce, İspanyolca, İsveççe ve Türkçe'dir. ECI verilerini kullanmadan önce alt külliyatlar oluşturulmuştur. Alt külliyatlar oluşturulurken külliyatların boyutu önemli bir konudur. 9 alt külliyatının hangi boyutta olması gerektiğini tespit için dilin doygunluğuna bakılmıştır. Külliyat boyutu artırıldığı halde dil tanıma doğruluğunun değişmediği noktada külliyat boyutu artırma sona erdirilmiş ve ilgili külliyatın doygunluk boyutuna ulaştığına karar verilmiştir. Diller için en uygun külliyat boyutları doygunluk yöntemiyle ölçüldüğünde Çizelge 5.1. deki değerler elde edilmiştir.

Çizelge 4.1. Eğitim İçin Külliyat Boyutları

Dil	Külliyat Boyutu
İngilizce	20 KB
Fransızca	20 KB
Almanca	20 KB
Hollanda Dili	10 KB
İtalyanca	10 KB
Portekizce	100 KB
Türkçe	10 KB
İspanyolca	100 KB
İsveççe	30 KB

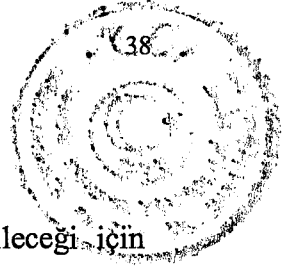
Külliyyat boyutuyla ilgili deneyler sonrasında dillerin külliyyat boyutlarının farklı olması gerektiği bulunmuştur. Dillerden bazıları 20 KB uzunluğunda en yüksek tanıma doğruluğunu verirken diğer bazıları 100 KB uzunluğunda en yüksek doğruluğu vermektedir. Deneyler sırasında bütün dillerin aynı boyutta olma durumu da test edilmiş fakat en yüksek doğruluğu vermemiştir. Külliyyat boyutları centroid değerlerinin hangi boyutta doygunluğa eriştiği ile ilgili bilgi vermektedir. Örneğin, Çizelgede Almanca için 20 KB uzunluğu verilmiştir, bu değer Almanca için karakter sıklık bilgilerinin doygunluğa eriştiği değerdir. O uzunluğun altında da üstünde de dil tanıma oranları düşmektedir.

4.2. Metinsel Verinin Sayısal Hale Dönüşümü

ECI'den elde edilen veriler ham halde bulunmaktadır. Ham metinsel verilere sınıflandırma tipinde sayısal algoritmalar uygulamak mümkün olmadığından metinsel dokümanların karakter tabanlı ön işlemden geçirilmeleri gereklidir.

Karakter tabanlı ön işlemin en temel iki aşaması temizleme ve dönüşümdür. Veri temizleme aşamasında veriler filtre edilir. Filtreleme doküman bazında ve karakter bazında olabilir. Doküman bazında filtreleme yapılırken çok fazla bilgi taşımayan, içinde sık sık tekrarlar bulunan dokümanlar elenir. Çünkü, tekrarların fazla olması dokümandaki kelime ve karakter sıklıklarını olumsuz yönde etkilemektedir. Karakter bazında temizlemede ise harf karakterleri haricindeki karakterler dokümanlardan temizlenir. Karakter tabanlı dil tanıma daha çok harf karakterleri ile çalıştığından bu işlem yapılmaktadır.

Dokümanlar temizleme işleminden geçirildikten sonra ön işlemin ikinci aşaması olan dönüşüme hazır hale gelirler. Dönüşüm işlemi, ham karakter bilgilerini karakter sıklık bilgilerine dönüştürme işlevi görmektedir.



4.2.1. Doküman-Karakter Sıklıklarının Bulunması

Karakter tabanlı dil tanıma sınıflandırma yöntemiyle icra edileceği için metinsel veriden sayısal verinin elde edilmesi bir ihtiyaçtır. Ayrıca, karakter sıklıklarına dayalı istatistiksel testler için de karakter sıklıklarının elde edilmesi gerekmektedir.

Sıklığı bulunacak karakterler harf karakterleridir. Karakter sıklıkları bulunurken doğrudan karakterler sayılır. Örneğin, “ADANA” kelimesinde toplam 5 karakter bulunmakta ve bu kelime “3 adet A, 1 adet D ve 1 adet N” karakterinden meydana gelmektedir. “ADANA” kelimesinin sıklığa dayalı özeti bulunmuştur. Bu özet bilgi bir vektör üzerinde aşağıdaki çizelgede olduğu gibi sunulabilir.

Çizelge 4.2. “ADANA” kelimesi için karakter sıklık değerleri

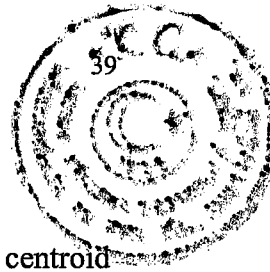
Dok_No	A	B	C	D	...	M	N	...	Z
1	3	0	0	1	...	0	1	...	0

Aynı kelime bağıl sıklık cinsinden sunulmak istenirse o zaman kelime içerisinde geçen karakterlerin bütün içindeki oranlarına bakılır. “ADANA” kelimesindeki ‘A’ karakterinin bağıl sıklığı ($3/5=0,6$), ‘D’ karakteri ile ‘N’ karakterinin bağıl sıklıkları 0,2’dir. Örnek kelime bağıl sıklıklar cinsinden aşağıdaki gibi verilebilir.

Çizelge 4.3. “ADANA” kelimesi için bağıl sıklık değerleri

Dok_No	A	B	C	D	...	M	N	...	Z
1	0,6	0	0	0,2	...	0	0,2	...	0

Doküman karakter sıklıkları bulunduktan sonra her bir doküman karakter uzayında bir doküman vektörü yardımıyla sunulur. Bu vektörün 60 adet boyutu bulunmaktadır. Her bir boyut bir karakterin sıklık değerini tutmakta olup değerler bağıl değerler şeklindedir. Bu sayede bütün dokümanların aynı boyutta dokümanlar gibi ele alınması sağlanmıştır. Doküman karakter vektöründe sıra önemli değildir. Yani karakterlerin sıklık sırasına göre dizilme ihtiyacı yoktur. Bu ihtiyacın olmaması da performans için istenen bir durumdur.



4.2.2. Centroid Değerlerinin Bulunması

Karakter tabanlı dil tanımanın en önemli işlerinden birisi dillere ait centroid değerlerinin bulunmasıdır. Centroid, her bir dili diğerinden ayırt etmeye yarayan, dilin karakteristiklerini sunan bir vektördür. Doküman karakter vektörlerinin dokümanları temsil ettiği gibi centroid vektörleri de dilleri temsil ederler. Centroid değerlerinin doküman karakter vektörlerinden bir farkı vardır. Doküman karakter vektörleri sadece karakterlerin doküman içerisindeki sıklık bilgilerinden elde edilirken centroid vektörleri karakterlerin diller arasındaki sıklıklarından elde edilir.

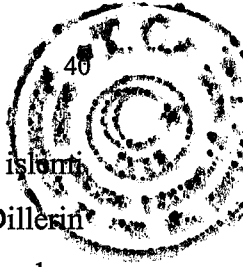
Her bir dil için elde edilecek centroid vektörlerinden önce her bir dil için ortalama karakter sıklıkları elde edilir. Ortalama karakter sıklıkları, doküman karakter vektörlerinin bir ortalamasından elde edilir. Çizelge 4.4'te 4 KB uzunluğunda 10 adet eğitim dokümanından ortalama karakter sıklıklarının elde edilmesini göstermektedir.

Çizelge 4.4. Türkçe için ortalama karakter sıklığının bulunması

DİL	BOYUT	SIRA	A	B	C	D	E	F	...	X	Y	Z
TRK	4000	1	13,2	3,62	0,89	5,07	11,4	0,22	...	0	3,87	2,23
TRK	4000	2	12,5	3,52	1,11	5,25	12,1	0,8	...	0	3,66	1,28
TRK	4000	3	13,7	3,69	0,76	5,32	11,6	0,53	...	0	3,36	1,93
TRK	4000	4	13	3,32	0,92	4,93	11,6	0,45	...	0	3,51	2,56
TRK	4000	5	13,2	3,08	0,88	4,4	11,5	0,34	...	0	3,75	2,29
TRK	4000	6	12,1	2,39	0,84	4,17	10	0,24	...	0	3,06	1
TRK	4000	7	10,6	1,55	1,42	3,61	12,4	0,57	...	0	2,58	0,98
TRK	4000	8	10,9	1,54	0,95	3,7	12,5	0,72	...	0	1,88	0,95
TRK	4000	9	8,13	1,11	0,8	4,58	14,3	0,82	...	0	1,67	0,9
TRK	4000	10	11,8	1,62	1,03	4,16	12,2	0,54	...	0	2,23	0,69
ORTALAMA			11,9	2,54	0,96	4,52	12	0,52	...	0	2,96	1,48

Çizelge 4.5. Bütün diller için ortalama karakter sıklıkları

DİL	A	B	C	D	E	F	...	X	Y	Z
ENG	8,446	1,754	3,208	5,188	12,66	2,506	...	0,05	1,429	0,05
FRE	9,161	0,852	4,129	4,8	12,95	1,471	...	0,619	0,439	0,052
GER	6,186	2,377	3,528	4,882	16,31	2,071	...	0,026	0,026	1,176
DUT	6,96	1,753	0,926	5,834	18,9	0,551	...	0,1	0,025	1,602
ITA	12,46	1,133	4,48	3,272	9,791	1,661	...	0	0	0,352
POR	12,86	0,957	4,062	5,537	11,36	0,88	...	0,103	0	0,259
TRK	11,92	2,543	0,959	4,52	11,97	0,523	...	0	2,959	1,482
SPA	11,28	1,706	3,896	5,067	13,7	0,815	...	0,051	0,942	0,28
SWE	8,807	1,548	1,094	4,937	9,421	2,402	...	0,053	0,48	0



Çizelge 4.4. de Türkçe için örneği gösterilen ortalama sıklık bulma işlemi diğer diller için de yerine getirildiğinde Çizelge 4.5. elde edilmiştir. Dillerin ortalama karakter sıklıklarından her bir karakterin sadece bir dil içerisinde yer alma olasılığı değil aynı zamanda bütün koleksiyon içerisinde meydana gelme olasılığı elde edilecektir. Örneğin; 'A' karakteri 9 dilde de yer almaktadır. Dil tanıma sistemine bir 'A' karakteri geldiğinde sistem bu karakterin % de kaç olasılıkla hangi dilde yer alacağını hesap ederken her bir karakterin dillere göre olasılıklarına ihtiyaç duyacaktır. Bu bilginin hesabı Çizelge 4.5'deki değerlere göre şöyledir.

$$\text{TOPLAM('A')}=(8.446+9.161+6.186+6.96+12.46+12.86+11.92+11.28+8.807)=(88.08)$$
$$P('A'|ENG)=(8.446/88.08)=0.095$$

Çizelge 4.6. Ortalama karakter sıklıkları ve centroid ilişkisi

9 DİL İÇİN ORTALAMA HARF FREKANSLARI

9 DİL İÇİN BAĞIL HARF FREKANSLARI

Karakter	İngilizce	Fransızca	Almanca	Hollandaca	İtalyanca	Portekizce	Türkçe	İspanyolca	İsviçre	Toplam	İngilizce	Fransızca	Almanca	Hollandaca	İtalyanca	Portekizce	Türkçe	İspanyolca	İsviçre
A	7,6	8,1	5,6	7,3	10	13	11	13	8,5	84	3	3	2,8	2,9	3,1	3,2	3,1	3,2	3
B	1,1	1,1	2	1,6	0,9	0,9	2,2	1,3	1,6	13	2,9	2,9	3,2	3,1	2,9	2,8	3,2	3	3,1
C	4,4	3,3	2,7	1,6	4,5	3,6	0,9	4,1	1,3	26	3,2	3,1	3	2,8	3,2	3,1	2,5	3,2	2,7
D	3,6	4,2	4,7	5,7	3,5	5,2	3,9	5,3	4,3	40	3	3	3,1	3,1	2,9	3,1	3	3,1	3
E	12	13	16	19	11	11	11	13	9,3	111	3	3	3,1	3,2	3	3	3	3	2,9
F	0	2,8	0	0	0,2	0,5	0	0,2	0	3,8	0	3,9	1,4	2	2,7	3,1	0	2,8	1,4
G	2	1	1,9	0,6	1,2	1	0,5	0,7	2,1	11	3,3	3	3,2	2,8	3	2,9	2,7	2,8	3,3
H	1,8	1,2	3	3	2,1	1,2	2	1,1	3,1	19	3	2,8	3,2	3,2	3	2,8	3	2,8	3,2
I	3	0,8	3,9	2,4	1,8	0,9	1	0,8	1,8	16	3,3	2,7	3,4	3,2	3	2,7	2,8	2,7	3
J	8,2	7,4	7,5	7,2	11	6,3	12	6,4	5,2	71	3,1	3	3	3	3,2	3	3,2	3	2,9
K	0,4	0,2	1,6	2,1	0	0	4,2	0,5	3,2	12	2,5	2,2	3,1	3,2	0	0,9	3,5	2,6	3,4
L	4,2	5,1	3,7	3,8	5,6	2,6	7,8	5,7	5	44	3	3,1	2,9	2,9	3,1	2,8	3,3	3,1	3,1
M	2,7	2,8	2,5	2,5	3	4,3	4,1	2,4	3,4	28	3	3	3	2,9	3	3,2	3,2	2,9	3,1
N	8,3	7,3	9,5	9,8	7,1	5	7,1	7	7,9	69	3,1	3	3,1	3,2	3	2,9	3	3	3,1
O	7,9	5,4	2,9	5,6	8,7	11	2,4	8,1	4	56	3,2	3	2,7	3	3,2	3,3	2,6	3,2	2,9
P	2,5	2,8	1,1	1,4	2,7	2,9	0,6	2,7	1,7	18	3,1	3,2	2,8	2,9	3,2	3,2	2,5	3,2	3
Q	0,1	0,8	0	0	0,6	1	0	0,7	0	3,2	2,3	3,4	1,5	1,6	3,3	3,5	0	3,3	1,2
R	6,9	6,7	7,7	6,3	6,9	6,8	7,2	6,6	8,9	64	3	3	3,1	3	3	3	3,1	3	3,1
S	6,3	7,2	6,4	4,2	5	8,1	3,6	7,3	6,1	54	3,1	3,1	3,1	2,9	3	3,2	2,8	3,1	3,1
T	9,8	6,1	6,5	6,4	5,8	4,8	3,9	4,5	8,1	56	3,2	3	3,1	3,1	3	2,9	2,8	2,9	3,2
U	2,6	5,2	3,6	1,9	3,4	3,8	2,5	3,8	1,8	29	3	3,3	3,1	2,8	3,1	3,1	2,9	3,1	2,8
V	1,3	1,2	1	2,5	1,8	1,4	1,3	1,1	2,1	14	3	2,9	2,8	3,3	3,1	3	3	2,9	3,2
W	0,2	0,4	0,1	0	0	0,2	0	0,2	0,2	1,3	3,2	3,4	2,7	2,5	2,1	3,2	1,1	3,2	3,2
X	1,2	0,2	0,1	0,1	0	0	2,5	0,7	0,6	5,5	3,4	2,6	2,2	2,2	0,9	1,2	3,7	3,1	3
Y	0	0,1	0	0	0	0	0,1	0	0	0,2	2,8	3,6	2,7	2,2	0	0	3,5	3,2	2,4
Z	0,1	0,2	1,1	1,2	0,4	0,3	1,3	0,7	0	5,3	2,3	2,6	3,3	3,4	2,9	2,7	3,4	3,1	1,4



Karakterlerin dillere göre ortaya çıkma olasılıklarının bir logaritma dönüşümünden geçirilmesi sonrasında dilleri karakterize edecek olan centroid değeri elde edilir. Logaritma dönüşümü düzgünleştirme için kullanılmaktadır. Ortalama karakter sıklıkları ile centroid arasındaki ilişki Çizelge 4.6'da verilmiştir. Bu işlemin ardından karakter tabanlı dil tanıma işlemi test dokümanı ile centroid değerleri arasındaki benzerlik veya uzaklığa göre yerine getirilmektedir. Benzerlik için metin puanlama yöntemi uygulandığında doküman vektörü ile centroid vektörlerinin nokta çarpımlarından test dokümanının dil puanları elde edilir ve bu puanlardan maksimum olanın test dokümanının dilini bulmada kullanılır.

4.3. Karakter Tabanlı Dil Tanımanın Motivasyonu

Karakter tabanlı dil tanımanın en önemli motivasyonu karakterlerin iki önemli özelliğidir. Öncelikle, karakterler dilden dile farklı sıklıklara sahiptir ayrıca her dilin kendine has özel karakterleri bulunmaktadır. Karakter sıklıklarının doküman dilini ayırt edecek derecede farklı olması karakter tabanlı yöntemin ispatı manasına gelmektedir. Karakter tabanlı dil tespiti deneylerine başlamadan önce bu motivasyonun test edilmesi gereklidir.

4.3.1. Karakter Tabanlı Dil Tanıma Karakter Sıklıkları

Karakterlerin rasgele dağılmadığı daha önceki çalışmalarda belirtilmiş olmakla birlikte karakterlerin sahip olduğu bilgilerin yeterliliği burada bir kez daha test edilecektir. Karakter sıklıkları için bu çalışmada parametrik olmayan testlerden Mann Whitney U ile Kruskal Wallis testi uygulanacaktır. Testin yerine getirilmesinde Statsoft firmasına ait Statistica sürüm 6.0 dan faydalanılmıştır [48].

4.3.1.1. İstatistiksel Testler: Mann Whitney U ve Kruskal Wallis

Karakter sıklıkları dilden dile değişim göstermektedir. Sıklık bilgilerindeki değişim miktarının istatistiksel olarak değerli bilgi taşıyıp taşımadığı test edilerek sıklık bilgilerindeki sapmanın dil tespitinde kullanılıp kullanılamayacağı anlaşılabilir. Bir gruplama değişkeninin bir başka değişken veya değişkenler üzerindeki etkisini test etmek için istatistiksel bazı testler vardır. Bu testlerden birisi

Mann Whitney U testidir. Bu testte bir grupta değişkeninin diğer değişkenleri ne oranda etkilediği bulunmaya çalışılır. Ayrıca bir değişkenin grup değişkenine bağlı olarak değişip değişmediği ile değişkenlerin yeterli bilgi taşıyıp taşımadıkları bu test ile bulunabilir.

Gruplama değişkeninin diğer değişkenleri ne oranda etkilediği veya grupta değişkeni değişince diğer değişkenlerin de hissedilir derecede değişip değişmediğini anlamak için grupta değişkenine göre farklılıklar izlenir. Karakter tabanlı dil tanıma açısından testlere bakmak gerekirse; eğer karakterlerin sıklıkları iki veya daha fazla dilde paralel (benzer) dağılım gösteriyorsa, yani dil değiştiği halde karakter sıklıklarında istatistiksel bakımdan önemli bir değişim meydana gelmiyorsa bu karakter tabanlı dil tanımanın motivasyonuna terstir. Aksi taktirde; eğer dile bağlı olarak karakter sıklıkları istatistiksel bakımdan önemli derecede farklı değerler alıyorsa bu karakter tabanlı yöntem açısından istenen bir durumdur.

İstatistiksel testlerde benzerliği bulmak için genellikle p katsayısı kullanılmaktadır. Belirlenen p değerinin altında kalan benzerlikler önemsiz, belirlenen p değerinin üstünde kalan benzerlikler önemli benzerliklerdir. Karakter tabanlı yöntem için; benzerliğin istatistiksel olarak önemsiz olması, dolayısıyla benzemezliğin fazla olması karakterin yöntem açısından önemli bilgi taşıdığı manasına gelirken, benzerliğin istatistiksel olarak önemli olması karakterin yöntem için önemli bilgi taşımadığı manasına gelir. Bu tip durumlar daha çok sıklık bilgisi birbirine çok yakın olan karakterlerde meydana gelir.

İstatistiksel önem derecesi olan p için bu çalışmada 0.05 ($p=0.05$) değeri uygun görülmüştür. Özellik kümesinde yer alan karakterler tek tek incelenerek dilden dile karakter sıklıklarının birbirine benzerliğine bakılmıştır. Testlerin çıktısı ikili veya çoklu grupta değişkenine göre karakter sıklıklarındaki benzerliklerdir.

4.3.1.2. Hipotez ve Hipotezin Test Edilmesi

Karakterlerin dil tanıma kullanımı önerilirken karakter sıklıklarının dilden dile farklılıklar gösterdiği, dolayısı ile karakter sıklık bilgileri kullanılarak dil tanıma yapılabileceğinden bahsedilmişti. Karakter tabanlı dil tanımanın temel

öğeleri karakterler olup karakter sıklıkları birbirinden farklı davrandıkça bu durum karakter tabanlı dil tanımının başarısını artıracaktır. Karakter tabanlı dil tanımının sıklıklara dayalı temel motivasyonunu test etmek amacıyla aşağıdaki hipotezler kurulabilir.

H_0 : Karakterler dilden dile birbirine benzer sıklıklara sahiptir (benzer dağılıma sahiptir), o yüzden karakterler dil tanımada kullanılamaz.

H_1 : Karakter sıklıkları dilden dile farklı sıklık değerlerine sahiptir, o yüzden karakterler dil tanımada kullanılabilir.

Hipotez testi için; ikili gruplar arasındaki farklılıkları test etmeye yarayan t veya Mann Whitney U testi ile çoklu gruplar arasındaki farklılıkları test etmeye yarayan F veya Kruskal Wallis testi kullanılabilir. Testlerde, örnek adedinin azlığından kaynaklanabilecek hataları ortadan kaldırmak için parametrik olmayan testler; Mann-Whitney U ve Kruskal Wallis seçilmiştir. Parametrik testler ile parametrik olmayan testler arasındaki en önemli fark parametrik olmayan testlerin daha az eleman ile de daha doğru sonuçlar verebilmesidir.

Seçilen istatistiksel testler bazı çıktılar vermektedir. Bu çıktılardan birisi de istatistiksel önem değerini veren p katsayısıdır. Bu katsayı karşılaştırılan iki değer arasında istatistiksel önemde bir benzerlik bulunup bulunmadığını tespiti, yani farklılığı veya benzerliği tespiti yarar. İstatistiksel anlamda iki olay arasında bir ilişki veya benzerlik olması p katsayısının 0.05 gibi bir değerden büyük ($p > 0.05$) olması ile ölçülür. Eğer p katsayısı 0.05 den daha büyük bir değere sahipse bunun anlamı; test edilen iki olay arasında bir benzerlik vardır, bu iki olay benzer özelliktedir. Eğer iki olay arasındaki p katsayısı 0.05 den daha küçük ise; iki olay birbirinden farklıdır veya bu iki olay arasında bir ilişki bulunmamaktadır. Dolayısı ile karakter sıklıkları arasındaki ilişkide p değerinin 0.05 den büyük olduğu durumlar H_0 hipotezini destekleyen durumlar p değerinin 0.05 den küçük olduğu durumlar H_1 hipotezini destekleyen durumlardır.

İstatistiksel testlerden ilk olarak Mann Whitney U testi eğitim verilerine uygulanmıştır. Test ikişerli gruplara izin verdiğinden, diller ikişerli gruplanmış ve test uygulanmıştır. Sistemde k adet dil bulunduğundan ikişerli grupların adedi;

$kombinasyon(k,2)=k*(k-1)/2$ 'dir. İkişerli gruplar Çizelge 4.7. de olduğu gibi gruplanarak sonuçların daha rahat izlenmesi amaçlanmıştır. Çizelge 4.7. (Almanca, Hollandaca), (Almanca, Fransızca), (Almanca, İngilizce), (Almanca, İtalyanca), (Almanca, Portekizce), (Almanca, Türkçe) gruplarının tek bir Çizelgede toplanması sonucu elde edilmiştir. Testler bütün karakterler üzerinde değil bütün dillerde ortak kullanılan karakterler üzerinde yapılmıştır.

Çizelge 4.7. Almanca ile diğer diller arasındaki farklılık testi

Karakter	Hol.	Fra.	İng.	İta.	Por.	Türk
A	0,000336	0,000000	0,000000	0,000000	0,000000	0,000000
B	0,000344	0,001490	0,000000	0,000000	0,000000	0,268944
C	0,000000	0,178156	0,402643	0,000000	0,320357	0,000000
D	0,396929	0,107046	0,043995	0,000000	0,601274	0,139969
E	0,000001	0,000000	0,000000	0,000000	0,000000	0,000000
F	0,000000	0,891984	0,000000	0,000003	0,000000	0,000000
G	0,390485	0,000000	0,000000	0,000134	0,000000	0,000000
H	0,000374	0,024487	0,000000	0,000000	0,000000	0,000000
I	0,184336	0,085282	0,796369	0,000000	0,000059	0,372757
J	0,000000	0,410667	0,001255	0,322741	0,045876	0,058301
K	0,000631	0,000000	0,000000	0,000000	0,000000	0,000000
L	0,001809	0,728922	0,000000	0,000441	0,000017	0,000000
M	0,004204	0,101681	0,025030	0,014706	0,000000	0,000052
N	0,563186	0,000000	0,002084	0,000005	0,000000	0,017862
O	0,000000	0,000000	0,000000	0,000000	0,000000	0,152588
P	0,088186	0,000002	0,000000	0,000000	0,000000	0,831180
R	0,000000	0,000028	0,000029	0,016204	0,005039	0,254535
S	0,000000	0,898922	0,000001	0,000031	0,031501	0,000000
T	0,929027	0,000000	0,098555	0,595031	0,000018	0,000000
U	0,000000	0,001310	0,000004	0,011374	0,062241	0,000100
V	0,000000	0,069588	0,676870	0,000000	0,000101	0,149188
Y	0,000004	0,000000	0,000004	0,156256	0,156256	0,000000
Z	0,223746	0,000000	0,000000	0,000000	0,000000	0,227268

Çizelge 4.7'deki değerlerin daha anlaşılır bir hale gelmesi için benzerlik miktarı 0.05 değerinden daha küçük olanlar ÖNEMLİ etiketiyle etiketlenmiştir. Çizelge 4.8. deki ÖNEMLİ etiketinin manası; ÖNEMLİ hücrenin bulunduğu satır başındaki karakterin Almanca ile ÖNEMLİ hücrenin bulunduğu sütun başlığındaki dil arasında ayırt edici bilgi taşımasıdır. Örneğin, Çizelge 4.8. de I karakteriyle ilgili satırda sadece iki adet ÖNEMLİ etiketi bulunmaktadır. Bu etiketlerden birisi İtalyanca sütununda diğeri ise Portekizce sütunundadır. Bu örneğe göre I karakteri Almanca ile İtalyanca ve Almanca ile Portekizce arasında ayırt edici bilgi taşımaktadır. Yani, I karakterinin sıklığı Almanca ve İtalyanca



arasında, ayrıca Almanca ile Portekizce arasında farklı dağılmakta, benzer değer almamaktadır. Diğerleri de bu şekilde yorumlanabilir.

Çizelge 4.8. Bir önceki çizelgenin önem bilgisi ile verilmesi

Karakter	Hollanda dili	Fransızca	İngilizce	İtalyanca	Portekizce	Türkçe
A	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
B	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	
C	ÖNEMLİ			ÖNEMLİ		ÖNEMLİ
D			ÖNEMLİ	ÖNEMLİ		
E	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
F	ÖNEMLİ		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
G		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
H	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
I				ÖNEMLİ	ÖNEMLİ	
J	ÖNEMLİ		ÖNEMLİ		ÖNEMLİ	
K	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
L	ÖNEMLİ		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
M	ÖNEMLİ		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
N		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
O	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	
P		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	
R	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	
S	ÖNEMLİ		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ
T		ÖNEMLİ			ÖNEMLİ	ÖNEMLİ
U	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ		ÖNEMLİ
V	ÖNEMLİ			ÖNEMLİ	ÖNEMLİ	
Y	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ			ÖNEMLİ
Z		ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	ÖNEMLİ	

ÖNEMLİ şeklinde etiketlenmiş hücrelerin sayısının daha fazla olması, H_0 hipotezinin reddi H_1 hipotezinin kabulü manasına gelmektedir. Çizelge 4.8’de görüldüğü gibi ÖNEMLİ etiketi ile etiketlenmiş hücrelerin adedi diğerlerinden daha fazladır, bu yüzden H_0 hipotezi reddedilmeli ve H_1 hipotezi kabul edilmelidir. İkinci bir örnek olarak Hollanda dili ile diğer dillerin karakter sıklıkları arasındaki benzerlikler test edilebilir.

Çizelge 4.8’deki değerler gibi Çizelge 4.9’da da istatistiksel manada ÖNEMLİ olan hücrelerin sayısı fazladır. Bahsedilen hücreler diğerlerinden ayırt edilebilir diye renklendirilmiştir. Çizelge 4.9’da yer alan değerler H_0 hipotezinin reddi ve H_1 hipotezinin kabulünü gerektirmektedir. Çizelge 4.8. ve Çizelge 4.9’daki değerler gibi diğer olası bütün değerler bir araya getirildiğinde de çoğunlukla H_1 hipotezinin kabul edilmesini gerektirecek sonuçlar çıkmıştır. Hipotezlerin kabulü veya reddi için bu çalışmada çoğunluğa bakılmaktadır.



Örneğin; iki dilin karakter sıklıkları kontrol edilirken 26 karakterden 20 tanesinin birbirinden farklı sıklıklara sahip ise o zaman bu iki dilin birbirinden farklı sıklıklara sahip olduğu kabul edilmektedir.

Çizelge 4.9. Hollanda dili için birleştirilmiş Mann Whitney U testi sonuçları

Karakter	Fransızca	İngilizce	İtalyanca	Portekizce	Türkçe
A	0,001254	0,000019	0,000000	0,000000	0,000000
B	0,000002	0,914929	0,000009	0,000000	0,034216
C	0,000000	0,000000	0,000000	0,000000	0,012244
D	0,397253	0,507328	0,000001	0,180770	0,748694
E	0,000000	0,000000	0,000000	0,000000	0,000000
F	0,572825	0,000000	0,025911	0,342184	0,005195
G	0,000000	0,000000	0,003949	0,000000	0,000000
H	0,000000	0,000000	0,000004	0,000000	0,000000
İ	0,121816	0,812001	0,000000	0,011688	0,696549
J	0,000000	0,000000	0,000000	0,000000	0,000000
K	0,000000	0,000000	0,000000	0,000000	0,000000
L	0,002992	0,006515	0,581266	0,000000	0,000000
M	0,538553	0,207402	0,718437	0,000000	0,262884
N	0,000036	0,000000	0,000000	0,000000	0,000967
O	0,862764	0,000001	0,000000	0,000000	0,000000
P	0,000000	0,001178	0,000000	0,000000	0,130386
R	0,037179	0,157101	0,002293	0,004962	0,000019
S	0,000000	0,000000	0,000004	0,000000	0,076486
T	0,142463	0,000000	0,650048	0,000065	0,000000
U	0,000000	0,000001	0,000000	0,000000	0,006512
V	0,000000	0,000000	0,000163	0,000000	0,000000
Y	0,984294	0,000000	0,000000	0,000000	0,000000
Z	0,000000	0,000000	0,000000	0,000000	0,019759

Mann Whitney U testi ile diller arasındaki farklılık veya benzerlikler ikili gruplar halinde bulunmuştur. Diller arasındaki benzerlikleri bulmada kullanılabilecek diğer test Kruskal-Wallis ANOVA testidir. Bu test ikişerli gruplar yerine çoklu gruplara izin vermektedir. Kruskal Wallis testi uygulandığında, her bir karakter için dilden dile farklılık veya benzerlikler elde edilmektedir. Mann Whitney U testinde iki dilin bütün karakterlere dayalı benzerliği bulunmaktadır. Kruskal Wallis testinde ise tek bir karakterin bütün diller için ayırt ediciliği ölçülür. Referans noktası Mann Whitney U testinde karakterler, Kruskal Wallis testinde ise dillerdir. Örnek olarak 'a', 'b' ve 'z' karakterleri seçilmiştir. 'a' karakteri için Çizelge 4.10'daki değerler elde edilmiştir.

Çizelge 4.10. 'a' karakteri için sıklık farklarının p değerine göre gösterimi.

		Multiple Comparisons p values (2-tailed); a (FR100HARF_26HarfOzellik)						
		Independent (grouping) variable: DIL						
		Kruskal-Wallis test: H (6, N= 700) =322,1881 p =0,000						
Depend.:		GER	DUT	FRE	ENG	ITA	POR	TRK
a		R:140,87	R:228,38	R:297,58	R:328,65	R:390,88	R:533,78	R:533,35
GER			0,046449	0,000001	0,000000	0,000000	0,000000	0,000000
DUT	0,046449			0,326296	0,009545	0,000000	0,000000	0,000000
FRE	0,000001	0,326296			1,000000	0,023206	0,000000	0,000000
ENG	0,000000	0,009545	1,000000			0,621143	0,000000	0,000000
ITA	0,000000	0,000000	0,023206	0,621143			0,000012	0,000013
POR	0,000000	0,000000	0,000000	0,000000	0,000012			1,000000
TRK	0,000000	0,000000	0,000000	0,000000	0,000013	1,000000		

Kruskal Wallis testinde birden fazla grubun aynı anda farklılık testi yapılır. p katsayısı için 0.05 eşik değeri burada da geçerlidir. Çizelge üzerinde 0.05 değerinden daha düşük çıkan benzerlikler sayma yöntemiyle bulunduğunda; toplam 42 hücrenin 34 tanesinde p değeri 0.05 ten daha küçük, 8 tanesinde ise p değeri 0.05 ten daha büyüktür. Çizelge 4.10'da elde edilen değerler 'a' karakterinin karakter tabanlı dil tanıma için faydalı bilgi taşıdığını göstermektedir.

Çizelge 4.11. 'b' için diller arası sıklık farklarının p değerine göre gösterimi.

		Multiple Comparisons p values (2-tailed); b (FR100HARF_26HarfOzellik)						
		Independent (grouping) variable: DIL						
		Kruskal-Wallis test: H (6, N= 700) =133,7615 p =0,000						
Depend.: b		GER	DUT	FRE	ENG	ITA	POR	TRK
		R:477,21	R:385,33	R:262,10	R:378,63	R:263,89	R:248,43	R:437,91
GER			0,027584	0,000000	0,011891	0,000000	0,000000	1,000000
DUT	0,027584			0,000344	1,000000	0,000456	0,000036	1,000000
FRE	0,000000	0,000344			0,000966	1,000000	1,000000	0,000000
ENG	0,011891	1,000000	0,000966			0,001264	0,000111	0,801837
ITA	0,000000	0,000456	1,000000	0,001264			1,000000	0,000000
POR	0,000000	0,000036	1,000000	0,000111	1,000000			0,000000
TRK	1,000000	1,000000	0,000000	0,801837	0,000000	0,000000		

'b' karakteri için p değerlerine bakıldığında görülecektir ki 42 değerden sadece 10 tanesi H_0 hipotezinin kabulünü gerektirirken 32 tanesi H_1 hipotezinin kabulünü gerektirmektedir. Çoğunluğa dayalı karar verildiğinden dolayı H_1 hipotezi kabul edilmesi gereken hipotezdir.

Çizelge 4.12. ‘z’ için diller arası sıklık farklarının p değerine göre gösterimi.

Multiple Comparisons p values (2-tailed); z (FR100HARF_26HarfOzellik) Independent (grouping) variable: DIL Kruskal-Wallis test: H (6, N= 700) =276,3488 p =0,000							
Depend.: z	GER	DUT	FRE	ENG	ITA	POR	TRK
GER	R:477,48	R:502,04	R:213,20	R:227,40	R:281,38	R:300,44	R:451,56
DUT	1,000000		0,000000	0,000000	0,000000	0,000000	1,000000
FRE	0,000000	0,000000		1,000000	0,359539	0,047963	0,000000
ENG	0,000000	0,000000	1,000000		1,000000	0,223608	0,000000
ITA	0,000000	0,000000	0,359539	1,000000		1,000000	0,000000
POR	0,000000	0,000000	0,047963	0,223608	1,000000		0,000003
TRK	1,000000	1,000000	0,000000	0,000000	0,000000	0,000003	

Diğer karakterler için test yapıldığında da çoğunlukla H_1 hipotezini doğrulayıcı sonuçlar çıkmaktadır. Dolayısı ile Mann-Whitney U testi ve Kruskal Wallis testi ile hipotezler test edilmiş olup bu testlerin sonrasında H_0 hipotezinin reddedilmesi H_1 hipotezinin ise kabul edilmesi uygun bulunmuştur. H_1 hipotezinin kabulü ise karakter tabanlı dil tanımının motivasyonunun desteklenmesi manasına gelmektedir.

4.3.2. Karakter Tabanlı Dil Tanımda Özel Karakterler

Karakter sıklıkları ile ilgili söylenenler daha çok genel karakterler (bütün dillerde ortak kullanılan karakterler) için söylenmiş olup karakter tabanlı dil tanıma da genel karakterlere ek olarak özel karakterlerde yer almaktadır. Özel karakterler bütün dillerde bulunmayıp sadece bazı dillerde bulunan karakterlerdir. Türkçe’de bulunan ‘Ş’ ve ‘Ğ’ karakteri özel karakterlere bir örnektir.

Özel karakterlerin dilleri birbirinden ayırt etmede kullanılabilir olduğunu göstermek için özel karakterlerin sıklıklarını vermek yeterli olacaktır. Özel karakterler genel karakterlerden daha fazla ayırt edici bilgi taşırlar. Çünkü özel karakterlerin sıklıklarındaki sapma miktarı daha fazladır ve özel karakterler daha ayırt edici bilgi sağlamaktadır. Dolayısıyla özel karakterlerin varlığı dil tespiti için istenen bir durumdur.

Çizelge 4.13’de görüldüğü gibi özel karakterlerin bir çoğunda sıklık bilgisi 0 değerine sahiptir. Bu şunu göstermektedir özel karakterlerin sıklık değerleri kimi

$$\text{Yataydaki } \text{\text{Ş}} \text{ değeri} = ((0.46)/(0+0+0+0+0+0+0.46)) * 100 = (\text{fr}_{\text{\text{Ş}}} / (\text{fr}_{\text{TOPLAM}})) * 100$$
[illegible]

4.4. SDM ile Karakter Tabanlı Sınıflandırma

Karakterlerin dil tanımada kullanılabilirliği test edildikten sonra karakter sıklıklarına dayalı dil tanıma deneyleri yapılmıştır. Dil tanıma bir çeşit sınıflandırma uygulaması olduğundan dil tanıma deneyinde karar ağacı algoritmalarından Sınıflandırma ve Regresyon Ağaçları (C&RT) algoritması kullanılmıştır. Algoritmanın gerçekleştirimi için Statistica Data Miner (SDM) isimli üründen faydalanılmıştır. SDM, Statsoft firmasının veri madenciliği uygulamaları için geliştirdiği bir paket programdır [48].

C&RT için hazırlanan deney verisi; her satırında bir dokümana ait dil kodu ve karakter sıklık değerleri bulunan doküman karakter vektörlerinden meydana gelmektedir. Sınıflandırma algoritması için dil kodu bağımlı değişken, karakterler ise bağımsız değişkenler olarak belirlenir. Algoritma çalıştırıldıktan sonra yazılım raporlar kısmında çeşitli çıktılar üretir. Bu çıktılardan birisi de sınıflandırma matrisidir. Sınıflandırma matrisinde gözlemlenen (observed) DİL değerleri (y_i) ile tahmin edilen (expected / predicted) DİL değerleri ($\hat{y}_i$) yer almaktadır. Çizelge 4.14'de 10 karakter uzunluklu test dokümanları için sınıflandırma sonuçları sınıflandırma matrisi yardımıyla verilmiştir.

Çizelge 4.14. 10 karakterlik dokümanlar için sınıflandırma matrisi

Classification matrix 1 (FR10HARF)							
Dependent variable: DIL							
Options: Categorical response, Tree number 1, Analysis sample							
	Class GER	Class DUT	Class FRE	Class ENG	Class ITA	Class POR	Class TRK
GER	89,00000	6,00000	3,00000	7,00000	5,00000	4,00000	3,00000
DUT	6,00000	86,00000	4,00000	3,00000	3,00000	1,00000	2,00000
FRE	1,00000	3,00000	89,00000	4,00000	3,00000	4,00000	2,00000
ENG	1,00000	1,00000	1,00000	82,00000	2,00000	4,00000	1,00000
ITA	2,00000	1,00000	2,00000	1,00000	86,00000	1,00000	
POR		3,00000	1,00000	2,00000	1,00000	86,00000	
TRK	1,00000			1,00000			92,00000

Çizelgenin her bir sütunundaki değerler toplamı 100 dür, çünkü sisteme her dilden 100 adet örnek verilmiştir. Satır değerleri ise verilen dildeki dokümanın hangi dilde tahmin edildiği ile ilgilidir. Çizelge üzerindeki birinci satır ve birinci sütun

incelendiğinde 89,0 değeri görülür. Bunun anlamı Almanca olarak etiketlenmiş 100 adet dokümanın 89 tanesi yine Almanca olarak tahmin edilmiştir. Bilgi alma etki alanında doküman almanın başarısı doğruluk (precision) ve geri çağırma (recall) ile ölçülür. Yukarıdaki çizelgeye sadık kalmak üzere diller için doğruluk ve geri çağırma değerleri aşağıdaki gibi bulunabilir. Bu arada doğruluk ve geri çağırma için Almanca örnek olarak alınmıştır.

$$\text{Doğruluk} = 89 / (89 + 6 + 3 + 7 + 5 + 4 + 3) = 0,76 \quad \text{Geri Çağırma} = 89 / (89 + 6 + 1 + 1 + 2 + 1) = 0,89 \text{ olarak bulunur.}$$

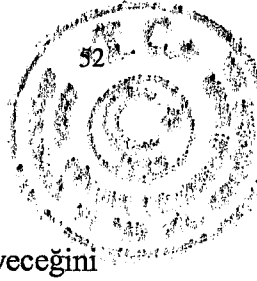
Çizelge 4.15’de her bir dil için 30, 50, 60, 90 ve 100 karakterlik metin parçaları ile yapılmış sınıflandırma sonuçları verilmiştir. Metin boyutunun doğruluk üzerindeki etkisini ölçmek ve farklı diller için doğruluk değerlerini görmek için Çizelge 4.15 açıklayıcı niteliktedir.

Çizelge 4.15. Farklı metin uzunlukları için doğruluk değerleri (precision)

Dil	Metinlerin karakter olarak uzunlukları				
	30	50	60	90	100
Almanca	0,87	0,93	0,97	0,96	1
Hollanda Dili	0,89	0,92	0,99	0,96	0,99
Fransızca	0,94	0,84	0,96	0,97	1
İngilizce	0,95	0,98	0,95	0,98	0,99
İtalyanca	0,92	0,95	0,98	0,98	0,97
Portekizce	0,95	1	1	0,99	0,99
Türkçe	1	1	0,99	1	1

Çizelge 4.16. Farklı metin uzunlukları için geri çağırma değerleri (recall)

Dil	Metinlerin karakter olarak uzunlukları				
	30	50	60	90	100
Almanca	0,93	0,98	0,99	0,98	1
Hollanda Dili	0,90	0,98	0,96	0,96	0,99
Fransızca	0,96	0,96	0,96	0,97	0,96
İngilizce	0,89	0,98	0,97	0,99	0,99
İtalyanca	0,92	0,93	0,98	0,97	1
Portekizce	0,93	0,96	0,98	0,99	0,99
Türkçe	0,99	0,98	1	1	1



4.5. Centroid Tabanlı Dil Tanıma Deneyleri

Karakter tabanlı dil tanımanın daha yüksek doğruluğu verip vermeyeceğini anlamak için bazı deneyler yapılmıştır. Bu deneylerde dillerin hangi sayısal değerlerle sunulması gerektiğinden, hangi benzerlik yönteminin tercih edilmesi gerektiğine kadar bazı testler yerine getirilmiş ve sonuçlar yorumlanmıştır.

Bu kapsamda yöntemin dil tanıma doğruluğunu etkileyebileceği düşünülen faktörlerden bazıları şunlardır;

- Centroid değeri
- Benzerlik/uzaklık yöntemleri
- Standart sapma
- İlave yöntem

Dil tanımanın başarısına etki edebilecek faktörler sırayla test edilerek dil tanıma için en uygun form bulunmaya çalışılmıştır. Deneyler sonrasında; benzerlik için kosinüs türevi bir yöntemin kullanılması gerektiği anlaşılmıştır. Ayrıca deneyler sırasında sisteme yapılabilecek ekler test edilmiştir. Geliştirilen yöntem doğru tanıma bakımından n-gram ve short term yöntemlerine yaklaşıırken performans bakımından diğer yöntemlerden daha iyi sonuçlar vermektedir.

4.5.1. Centroid Değeri

Dilleri karakterize etmede kullanılabilecek iki seçenekten birisi dillerin ham ortalama sıklıkları diğeri ise dillerin centroid değerleridir. Centroid değeri ile ortalama karakter sıklıkları arasındaki önemli farklardan birisi logaritma dönüşümüdür. Dolayısıyla centroid kullanımıyla elde edilen başarı aynı zamanda logaritma faktörünün başarısıdır. Dillerin ortalama karakter sıklıkları da dilleri bir miktar karakterize etmektedir fakat ortalama sıklıklar ham verilerden oluştuğu için başarı oranı logaritma dönüşümünden geçirilmiş centroid yöntemine göre daha düşüktür. Bu durum Çizelge 4.17’de gösterilmiştir. Çizelgede yer alan değerler ortalama karakter sıklıklarıyla centroid değerleri arasındaki farkın doğruluğa etkisini göstermektedir.



Çizelge 4.17. Logaritma dönüşümünün başarıya etkisi

	Yöntemler		
	Ortalama	Centroid	İyileşme
İngilizce	98,00	100,00	2,00
Fransızca	100,00	100,00	0,00
Almanca	100,00	99,00	-1,00
Hollanda Dili	82,00	98,00	16,00
İtalyanca	87,00	97,00	10,00
Portekizce	97,00	97,00	0,00
Türkçe	100,00	100,00	0,00
İspanyolca	90,00	96,00	6,00
İsveççe	100,00	100,00	0,00
Başarı Ort. (%)	94,89	98,56	3,67

(Değerler 100 karakter uzunluğundaki metinler içindir)

4.5.2. Benzerlik/Uzaklık Yöntemlerinin Etkisi

Karakter tabanlı dil tanıma, dil tanımayı benzerlik tabanlı bir sınıflandırma uygulaması olarak ele almakta ve bu yüzden benzerlikler dil tanımada önem kazanmaktadır. Bilgi alma etki alanında benzerlik veya uzaklık bulma için genellikle kosinüs tipi açı benzerlikleri ve öklidyen tipi uzaklık bulma yöntemleri kullanılmaktadır [49]. Karakter tabanlı dil tanımada hangi benzerlik/uzaklık bulma yönteminin kullanılması gerektiği farklı benzerlik/uzaklık bulma yöntemlerinin testi sonucu elde edilmiştir.

Bu çalışmada kosinüs benzerlik yöntemi yerine onun bir türevi olan metin puanlama yöntemi kullanılacaktır. Metin puanlama yöntemi centroid vektörü ile doküman karakter vektörünün nokta çarpımından bir puan elde etmeye dayalıdır. Metin puanlamanın kosinüs yönteminden farkı paydasında bir değer olmamasıdır. Aynı veri kümesi üzerinde metin puanlama benzerlik yöntemi, öklidyen ve boyutların maksimumu yöntemleri ile karşılaştırıldığında Çizelge 4.18'deki değerler elde edilmiştir.

Çizelgedeki değerler aynı veri kümesi üzerinde farklı yöntemlerin başarı oranlarını vermektedir. Çizelgede; TS, metin puanlama (Text Scoring - TS)



yöntemini, EUC, öklidyen yöntemini ve MAX, boyutların maksimum uzaklık bulma yöntemini ifade etmektedir.

Çizelge 4.18. Farklı Uzaklık/Benzerlik yöntemlerine göre dil tanınmanın başarısı

	İngilizce	Fransızca	Almanca	Hollandaca	İtalyanca	Portekizce	Türkçe	İspanyolca	İsviçre	METOD
İng.	97	2	1	0	0	0	0	0	0	TS
	75	3	3	1	5	2	1	6	4	EUC
	83	0	6	1	3	2	0	5	0	MAX
Fra.	0	99	0	0	0	0	0	1	0	TS
	3	84	2	3	3	0	0	5	0	EUC
	6	82	2	0	6	0	0	4	0	MAX
Alm.	0	0	100	0	0	0	0	0	0	TS
	4	1	89	4	0	0	1	0	1	EUC
	3	0	87	9	0	0	0	0	1	MAX
Hol.	2	0	1	97	0	0	0	0	0	TS
	4	0	6	83	0	0	0	6	1	EUC
	3	0	6	88	1	0	0	2	0	MAX
İta.	2	0	0	0	95	0	0	3	0	TS
	4	2	0	3	70	5	0	16	0	EUC
	5	0	0	0	81	6	0	8	0	MAX
Por.	0	0	0	0	1	97	0	2	0	TS
	0	2	0	0	4	90	0	4	0	EUC
	0	0	0	0	2	93	0	5	0	MAX
Türk.	0	0	0	0	0	0	100	0	0	TS
	0	2	1	2	0	0	94	1	0	EUC
	0	0	1	1	0	0	97	1	0	MAX
İsp.	1	0	0	0	1	0	0	98	0	TS
	4	6	0	3	7	10	0	70	0	EUC
	3	5	0	2	4	6	0	80	0	MAX
İsv.	0	0	0	0	0	0	0	0	100	TS
	1	0	3	0	0	0	0	0	96	EUC
	1	0	3	0	0	1	0	0	95	MAX

Deneyler metin puanlama yönteminin en başarılı sonucu verdiğini göstermektedir. Bu yöntemin başarılı sonuç vermesinin bazı sebepleri vardır. Başta metin puanlama yöntemi vektörler arasındaki açıya dayanarak benzerliği verdiğinden vektör boyutlarındaki sapmalardan etkilenmez, öklidyen yöntemleri ise vektör

değerlerindeki sapmalardan olumsuz yönde etkilenir. Ayrıca; karakter tabanlı yöntem metin puanlama için daha uygun bir yöntemdir. Çünkü metin puanlamada centroid değerlerinin kullanımı bir sakınca oluşturmazken öklidyen yöntemlerinde centroid değerleri sakınca oluşturur. Öklidyen yöntemlerinde vektörler birbiriyle aynı uzunlukta olmak zorunda iken metin puanlama yönteminde böyle bir zorunluluk yoktur. Öklidyen yöntemlerinin aynı boyutta vektörlerle çalışma zorunluluğundan dolayı diller ortalama karakter sıklıklarıyla sunulurlar.

4.5.3. Standart Sapmanın Etkisi

Karakter tabanlı dil tanıma yönteminde genellikle orta değerler üzerinde deneyler yapılmakta ve dili orta değerlerin daha iyi sunduğu temeli üzerinde çalışma yapılmaktadır. Orta değerler yerine uç değerlerin kullanılması durumunda sonucun ne oranda değişeceği, artıp artmayacağı incelenebilir. Bu testi yapmak için öncelikle her bir dil için ortalama karakter sıklıklarına ek olarak standart sapmalar elde edilir. Her bir dildeki her bir karakterin standart sapması birbirinden genellikle farklı olduğu için her bir dilin 60 elemandan oluşan standart sapma vektörü elde edilir. Dil tanıma sonuçları önce ortalama değerlere göre, ardından artı standart sapma ve eksi standart sapmaya göre elde edilir ve sonuçlar izlenir. Böylece standart sapmanın dil tanıma başarısına etkisi ölçülerek standart sapma faktörünün dil tanıma sisteminde kullanılıp kullanılmayacağına karar verilir.

Bu test ile karakter sıklıklarındaki dalgalanma veya standart sapma etkisi centroid değerine katılmaktadır. Centroid değerine standart sapmanın etki ettirilmesi ile doğru tanıma oranında kimi zaman düşmeler kimi zaman da yükselmeler meydana gelmiştir. Standart sapmanın etkisine toplamda bakıldığında doğru tanıma oranını düşürdüğü görülecektir. Elde edilen değerler dili centroid (orta) değerinin daha başarılı şekilde sunduğunu göstermektedir.

4.5.4. İlave Yöntem

Karakter tabanlı dil tanımanın diğer yöntemlere göre bir miktar daha başarısız olduğu metin uzunluklarında (Örn. 30 karakter) diğer yöntemlerin özellik setinden faydalanmak bir çözüm olabilir. Tek karakter yerine iki veya üç karakter kullanmak

doğru tanıma oranı artırıyorsa bireysel karakter tabanlı dil tanımanın başarısız olduğu yerlerde iki ve üç karaktere göre dil tanıma bir çözüm olabilir. Böylece kombine bir çözüm ile performans yönünden diğer yöntemlerden daha performanslı, doğru tanıma bakımından da karakter tabanlı dil tanımadan daha başarılı sonuçlar elde edilebilir.

30 karakter uzunluğundaki örnek metinler için karakter tabanlı dil tanıma sonuçları ve ikinci yöntemin başarıya etkisi Çizelge 4.19’da sunulmuştur.

Çizelge 4.19. 3gram yönteminin KTDT’ ye etkisi

Dil	KTDT	3gram	Başarı Artımı	Son Durum
İngilizce	84.7	61.0	9.33	94.03
Fransızca	76.0	66.0	15.84	91.84
Almanca	75.0	57.0	14.25	89.25
Hollanda Dili	77.6	71.0	15.904	93.50
İtalyanca	74.4	45.0	11.52	85.92
Portekizce	73.7	41.0	10.824	84.52
Türkçe	92.9	60.0	4.26	97.16
İspanyolca	69.2	72.0	25.872	95.07
İsveççe	92.5	84.0	6.3	98.80

Çizelge 4.19’daki değerlerden KTDT sütununda yer alan değerler 30 karakter için karakter tabanlı yöntemin ne kadar başarılı olduğunu verir. 3gram sütunundaki değerler karakter tabanlı yöntemin başarısız olduğu metinlerde 3gram yöntemi tarafından doğru tanınan metinlerin oranını verir. Başarı artımı ise ilave yöntem kullanımının doğru tanımayı ne oranda artırdığını vermektedir.

Çizelgede dikkate değer bir konu 3gram yönteminin bazı dillerde daha fazla başarıyı artırdığı bazılarında ise daha az başarıyı artırdığıdır. Bunun en önemli sebebi dillerdeki 3gram sıklıklarıdır. Örneğin İsveççe, İspanyolca ve Hollandaca dillerinde yer alan 3gramlar dili ayırt edici özelliğe daha fazla sahip olduklarından bunlarda 3gram yöntemi daha başarılı sonuç vermektedir. Türkçe’de ise 3gram yöntemi en düşük başarıyı vermektedir. Çünkü Türkçe’de dil hakkında bilgi verici ngramlar daha çok 2gramlardır.

4.6. Karakter Tabanlı Yöntem ve Karşılaştırmalar

Karakter tabanlı dil tanıma ile n-gram ve short term yöntemleri arasındaki karşılaştırmalar doğruluk (doğru tanıma oranı) ve performans açısından yapılabilir. Bu karşılaştırmalar sonrasında karakter tabanlı dil tanıma için en uygun kullanım alanı tespit edilebilir.

4.6.1. Doğru Tanıma Açısından Karşılaştırmalar

Doğru tanıma; dil tanıma sistemine sunulan bir dokümanın dilinin doğru olarak tespit edilme oranıdır. Dil tanıma sistemi seçiminde önemli parametrelerden bir tanesidir. Bu çalışmada karakter tabanlı yöntem aynı veri seti üzerinde daha önceden deneyleri yapılan yöntemlerle karşılaştırılacaktır. Bu karşılaştırma öncesinde; karşılaştırma yapılan doğru tanıma değerleri kelime uzunluklarına dayalı olarak sunulduklarından ortalama kelime uzunluğunun kaç karaktere denk geldiği belirlenmelidir. Her bir dile ait külliyatlardan ve internetten ortalama kelime uzunlukları için Çizelge 4.20'deki değerler elde edilmiştir.

Çizelge 4.20. Dillere göre ortalama kelime uzunlukları

Dil	Ort. Kelime Uz.
İngilizce	5,89
Fransızca	5,45
Almanca	6,07
Hollanda dili	4,74
İtalyanca	5,31
Portekizce	4,99
Türkçe	6,95
İspanyolca	5,26
İsveççe	6,00

Testlerde her bir dilin kelime uzunluklarının farklı değerler aldığı görülmüştür. O yüzden dilden dile kelimelerin karakter karşılıkları değişecektir. Örneğin, İngilizce için 2 kelimedeki tanıma oranını bulurken karakter tabanlı yöntemde yaklaşık 12 karakter kullanılabilir. Karşılaştırma sonuçları 4.21'de verilmiştir.

Çizelge 4.21 Cümlede geçen kelime adedine göre ortalama başarı oranları

	Test Metni İçinde Geçen Kelime Sayısı							
	1 – 2	3-5	6-10	11-15	16-20	21-30	31-50	>50
Hollanda Dili								
Trigram	70.6	91.3	98.9	99.7	100.0	99.9	100.0	100.0
Short	47.1	84.2	98.5	99.2	99.5	99.6	99.9	100.0
KTDT	47.0	68.0	86.0	91.0	97.0	98.0	100.0	99.0
İngilizce								
Trigram	78.9	97.2	99.5	99.9	100.0	99.9	100.0	100.0
Short	52.6	87.7	97.3	99.8	99.9	100.0	99.9	100.0
KTDT	53.0	82.0	99.0	100.0	100.0	100.0	100.0	100.0
Fransızca								
Trigram	69.2	93.0	94.5	93.6	99.8	100.0	99.9	100.0
Short	30.8	81.8	96.0	97.2	99.8	100.0	100.0	100.0
KTDT	54.0	82.0	94.0	99.0	100.0	99.0	100.0	100.0
Almanca								
Trigram	90.3	97.2	99.3	99.8	99.9	100.0	100.0	100.0
Short	23.1	71.6	89.6	98.2	99.8	100.0	100.0	100.0
KTDT	57.0	85.0	96.0	96.0	96.0	98.0	99.0	100.0
İtalyanca								
Trigram	56.8	92.9	99.6	100.0	100.0	100.0	100.0	100.0
Short	16.7	65.0	96.9	99.8	100.0	100.0	99.9	100.0
KTDT	48.0	72.0	83.0	92.0	98.0	97.0	100.0	100.0
Portekizce								
Trigram	83.5	96.6	99.4	99.9	100.0	99.9	100.0	100.0
Short	51.1	88.9	98.2	99.7	99.9	99.9	100.0	100.0
KTDT	52.0	71.0	87.0	95.0	96.0	99.0	99.0	100.0
İspanyolca								
Trigram	73.8	86.9	97.3	99.0	99.8	99.9	99.9	100.0
Short	8.1	81.5	98.8	99.7	100.0	100.0	100.0	100.0
KTDT	44.0	65.0	87.0	95.0	98.0	99.0	100.0	100.0

Çizelge 4.21’de Trigram ve Short term yöntemleri ile karakter tabanlı dil tanımanın sonuçları karşılaştırılmıştır. Short term ve trigram yöntemi için deney sonuçları Grefenstette’in [42] çalışmasından alınmıştır. Ayrıca Türkçe ve İsveççe için de deneysel çalışma yapılmış ve sonuçlar Çizelge 4.22’teki gibi elde edilmiştir.

Çizelge 4.22 Cümlede geçen kelime adedine göre başarı oranları

	Test Metni İçinde Geçen Kelime Sayısı							
	1 – 2	3-5	6-10	11-15	16-20	21-30	31-50	>50
Türkçe								
Trigram	62.0	87.0	95.0	98.0	100.0	100.0	100.0	100.0
Short	16.0	45.0	74.0	83.0	87.0	92.0	90.0	93.0
KTDT	70.0	90.0	98.0	100.0	100.0	100.0	100.0	100.0
İsveççe								
Trigram	85.0	94.0	99.0	100.0	100.0	100.0	100.0	100.0
Short	30.0	63.0	94.0	99.0	99.0	100.0	100.0	100.0
KTDT	69.0	98.0	99.0	100.0	100.0	100.0	100.0	100.0

Çok dilli dokümanların bulunduğu bir ortamda, bir dokümanın dilini tespit ederken dokümanın dilini kesin olarak söylemek yerine yüzde ihtimalle söylemek daha doğrudur. Örneğin Türkçe olarak tespit edilen bir dokümanın yüzde kaç ihtimalle gerçekten Türkçe olduğu merak edilir. Konuyla ilgili deneyler 30, 60, 90, 100 karakter uzunluğunda dokümanlarla yapılmış ve Çizelge 4.23'deki değerler elde edilmiştir.

Çizelge 4.23. Bulunan dokümanların aranan doküman olup olmadığının testi

Uzunluk	İngilizce	Fransızca	Almanca	Hollandaca	İtalyanca	Portekizce	Türkçe	İspanyolca	İsveççe
30	83	80	81	72	56	80	97	72	96
60	93	92	92	93	85	94	100	81	98
90	97	100	94	96	91	98	100	92	100
100	98	99	97	97	98	99	100	96	100

4.6.2. Performans Açısından Karşılaştırma

Karakter tabanlı yöntem ile diğer yöntemler arasındaki performans farkını görebilmek için hazırlanan deney düzeneğinde işlem süreleri kaydedilmiş ve değerler Çizelge 4.24'te verilmiştir. Deneyler Intel P4 1.8 GHz işlemci ve 256 MB ana belleğe sahip bir bilgisayarda yapılmıştır. Her bir dilden 50 KB uzunluğunda dokümanlar teste tabi tutulmuş ayrıca, 3gram özellik seti ve shortterm özellik seti için 300'er özellik alınmıştır.

Çizelge 4.24. Performans bakımından karşılaştırmalar

	KTDT	3gram	Shortterm
İngilizce	0,5 sn	8 sn	7 sn
Fransızca	0,5 sn	8 sn	8 sn
Almanca	0,5 sn	8 sn	7 sn
Hollandaca	0,5 sn	8 sn	8 sn
İtalyanca	0,5 sn	8 sn	7 sn
Portekizce	0,5 sn	7 sn	7 sn
Türkçe	0,5 sn	8 sn	7 sn
İspanyolca	0,5 sn	8 sn	8 sn
İsveççe	0,5 sn	8 sn	7 sn

Çizelge 4.24'de görüleceği gibi KTDT yöntemi diğer yöntemlere oranla performans bakımından daha üstün bir yöntemdir. Diğer yöntemlerden yaklaşık 15,

16 kat daha hızlıdır. Bu hız farkından dolayı etkin dil tanımada karakter tabanlı yöntem diğerlerinden daha uygundur. Bu yüzden özellikle hıza ihtiyaç duyulan yerlerde kullanılabilir. Karakter tabanlı yöntem ile diğer yöntemler arasındaki hız farkı karakter tabanlı dil tanımının önemli avantajlarından biridir. Bu avantaj özellikle canlı sohbet yapan kişilerin dillerinin bulunması gibi gerçek zamanlı uygulamalarda kullanılabilir.

İşlem sürelerinin neden o şekilde çıktığını anlayabilmek için yöntemleri algoritmik olarak analiz etmek uygun olacaktır. Aşağıda üç yöntemin algoritmik açıklamaları yer almaktadır. Metin uzunluğu için 100 değeri, 3gram ve shortterm yöntemlerinin özellik adetleri için 300 değeri alınmıştır.

KARAKTER TABANLI YÖNTEM

frequency(METİN, fr); // O(n)

score(fr,puanlar); // O(k)

maxindex(puanlar); // O(k)

n metnin uzunluğu, k dil sayısı olmak üzere komplekslik $O(n+k+k)=O(n)$ dir.

N-GRAM YÖNTEMİ

N-gram yöntemi; metinlerden n-gramların çıkarılması ve çıkarılan n-gramlara dayalı olarak dil tanımının yerine getirilmesi şeklinde iki adımda yerine getirilir.

ngram(METİN,tri); // O(n)

count(ngrams); // Geleneksel yöntemde $O(n^2)$, Hızlı n-gram çıkarma yönteminde $O(n)$ [51]

compare(Metin_Ngrams, Dil_Ngrams) // m ngramların adedi olmak üzere $O(nm)$

choice(Benzerlikler) // O(k)

n=100 ve m>100 olmak üzere; $O(n+(n \text{ veya } n^2)+nm)=O(nm)$

SHORT TERMS YÖNTEMİ

shortterms(metin); // O(n)

shorttermpuan(metin); // O(nm)

choice(Benzerlikler) // O(k)

m her bir dile ait sözlükteki kelimelerin adedini vermek üzere shortterm yöntemi içinde karmaşıklık derecesi $O(n+nm+k)=O(nm)$ dir.



5. SONUÇ ve ÖNERİLER

Bu çalışmada dil tanıma için karakter tabanlı bir yöntem önerilmiştir. Bu yöntemde; her bir dil, karakterlerin ortalama sıklıklarıyla ilişkili, ağırlıklı centroid vektörü ile sunulur. Sistemde dil adedi kadar centroid değeri bulunmaktadır. Yöntem dil tespitini doküman ile centroid değerleri arasındaki benzerliklere göre yerine getirir. Doküman karakter vektörü hangi dilin centroid vektörüne daha yakınsa doküman o dildedir.

Yöntem, farklılıkları büyütmek için centroid değerinin hesabında bazı dönüşümler yerine getirmektedir. Bu dönüşümler sayesinde dokümanlarda seyrek rastlanan karakterlerin değeri artırılmakta, sık rastlanan karakterlerin değeri azaltılmaktadır. Bu işlem sayesinde özel karakterlerin daha fazla bilgi taşıması sağlanmıştır.

Karakter tabanlı yöntemde dillerin temsil edildiği, centroid adı verilen vektör değerleri oldukça önemlidir. O yüzden centroid vektörlerinin elde edilme yöntemi karakter tabanlı yöntemin başarısını etkilemektedir. Centroid değerleri elde edilmeden önce; başta, her bir karakterin külliyat içindeki ortalama sıklığından külliyatların ortalama karakter sıklıkları elde edilir. Ortalama karakter sıklıkları bütün diller için elde edildikten sonra bu değerler normalleştirilerek külliyat boyutlarının farklılıklarından kaynaklanabilecek problemler ortadan kaldırılır. Daha sonra ortalama karakter sıklıklarından her bir karakterin koleksiyon içinde yer alma sıklığı elde edilir. En sonunda ortalama karakter sıklıkları bir log dönüşümünden geçirilerek centroid değerleri elde edilir. Log dönüşümü IR etki alanında genellikle kullanılan bir düzgünleştirme fonksiyonudur. Bu işlem TFIDF ağırlık yöntemindeki IDF etkisine benzer [50].

Karakter tabanlı yöntemde centroid değerlerinden sonraki en önemli ikinci konu doküman vektörleri ile centroid vektörleri arasındaki benzerliğin hesap edilmesidir. Benzerlik bulma için öklidyen tipi yöntemler ile kosinüs tipi yöntemler test edildiğinde kosinüs tipi yöntemlerin tercih edilmesi uygun görülmüş ve kosinüs benzerlik yöntemi dil tanıma sisteminde kullanılmıştır.



Geliştirilen yöntemle ilgili sonuçları maddeler halinde yazmak gerekirse;

- Karakter tabanlı yöntemin özellik seti karakterlerden oluşmaktadır.
 - Karakter özelliklerinin dilleri ayırt etmede kullanılıp kullanılamayacağı Mann Whitney U ve Kruskal Wallis testleriyle test edilmiş ve belirlenen p değeri ($p=0.05$) referans alındığında karakterlerin dilleri ayırt etmede kullanılabilir olduğu sonucuna varılmıştır.
- Karakter sıklık değerlerinin sunumu için vektör uzayı modeli kullanılmıştır. Dokümanlar ve diller birer vektör olarak sunulmuş ve doküman vektörü doküman-karakter vektörü olarak, dil vektörü ise centroid olarak isimlendirilmiştir.
- Yöntem dil tanımayı sınıflandırma uygulaması olarak yerine getirdiği için dil tanımada sınıflandırma modeli kullanılmaktadır.
- Seçilen sınıflandırma algoritması benzerliklere dayalı bir algoritma olan centroid tabanlı sınıflandırma algoritmasıdır.
 - Centroid vektörleri dillerin ortalama karakter sıklıklarından elde edilmekte ve her bir centroid vektörü bir dili temsil etmektedir.
- Dokümanın dili; doküman-karakter vektörü ile centroid vektörü arasındaki benzerliklere göre bulunmaktadır.
 - Doküman, maksimum benzerliği veren centroid değerinin dilindedir.
- Deneyler sırasında doküman ile dil arasındaki benzerlik bulmada kosinüs türevi bir benzerlik yöntemi olan metin puanlama yönteminin kullanımının daha uygun olduğu gözlemlenmiştir.
- Karakter tabanlı dil tanımada ortalama karakter sıklıkları yerine centroid değerlerinin kullanımı test edilmiş ve dili temsil için centroid değerlerinin ortalama karakter sıklıklarından % 3.67 daha başarılı sonuç verdiği gözlemlenmiştir.
- Karakter sıklıklarındaki sapmalardan elde edilen standart sapma etkisi test edilmiş ve standart sapma etkisinin sonuçları iyileştirmediği görülmüştür.
- Karakter tabanlı yöntem diğer yöntemlerle doğru tanıma ve performans açısından karşılaştırılmıştır;
 - Doğru tanıma bakımından yapılan karşılaştırmanın sonucu, karakter tabanlı yöntemin kabul edilebilir olduğunu göstermiştir. Karakter tabanlı yöntemin başarı oranı 100 karakter uzunluğundaki metinlerde 9 dil için

ortalama %98.56 civarındadır. Diğer yöntemler ile doğruluk açısından karşılaştırma yapıldığında geliştirilen yöntem %1 oranında daha düşük tanıma oranına sahiptir.

- n-gram yöntemi kısa metinlerde, kısa kelimeler (short terms) yöntemi ise daha uzun metinlerde başarılı sonuçlar vermektedir. Karakter tabanlı dil tanıma ise kısa metinlerde kısa kelimeler yönteminden genellikle daha başarılı sonuçlar vermekte, 15-20 kelimeden daha uzun metinlerde ise diğer yöntemlerde olduğu gibi %100 başarı ile dil tanıma işlemini yerine getirmektedir.
- Yöntemler performans açısından karşılaştırıldığında; n-gram ve short terms yöntemi için dil tanıma süreleri ortalama olarak karakter tabanlı dil tanımanın 15 mislidir. Aradaki hız farkı karakter tabanlı dil tanımanın en önemli avantajıdır. Bu avantaj özellikle gerçek zamanlı dil tespiti için önemlidir.
- Performans; özellik adedi ve işlem miktarına bağlı bir konudur. Yöntemler için özellik adetlerine bakıldığında; n-gram (3gram, $n=3$) yönteminin 2550 ile 3560 arasında özelliğe sahip olduğu, short terms yöntemi için ise 980 ile 2750 arasında özellik değerine sahip olduğu görülecektir. Karakter tabanlı dil tanıma için özellik adedi sadece 60 dır. Dolayısı ile karakter tabanlı dil tanıma bütün metin uzunluklarında diğer yöntemlerden daha etkin olarak dil tanıma işlemi yapacaktır.
- Bireysel karakterler 30 karakter gibi metin uzunluklarında doğru tanıma bakımından diğer yöntemlerden geridedir. Aradaki farkı kapatmak için üç harf kullanımı test edilmiş ve bazı diller için doğru dil tanıma başarısı diğer yöntemleri geçmiştir.
- Doğru tanıma ve performans parametrelerine göre minimum uzunluğu 15-20 kelime olan metinler için karakter tabanlı dil tanıma idealdir.

5.1. Öneriler

Karakter tabanlı dil tanıma gerçek zamanlı sistemler gibi, performansın önem kazandığı durumlarda idealdir. Çünkü bilinen dil tanıma yöntemlerine oranla oldukça hızlı şekilde dil tespitini yerine getirmektedir. Karakter tabanlı dil tanımanın doğru

tanıma oranını benzerlik bulma ve centroid değerlerinin hesabı etkilemekte olup bunlardan benzerlik bulma için uygun olan yöntem kosinüs türevi bir yöntem olan metin puanlamadır. Başarıyı artırmak için centroid hesabında güncellemeler yapılabilir. Her bir karakterin bütün koleksiyon içerisindeki önemini bulma konusunda çalışmalar yapmak doğru tanıma oranını diğer yöntemler seviyesine getirebilir.

Karakter tabanlı dil tanıma yazar tanıma, spam engelleme ve atak tespiti gibi konularda kullanılabilir. Çünkü bu işlemler de metinlerin fiziksel özelliklerine dayalı olarak yerine getirilebilir. Daha geniş bir bakış açısıyla anormallik tespiti tarzı işlemlerde karakter tabanlı yöntem kullanılabilir.



KAYNAKLAR

1. SIL International (2005), Ethnologue data from Ethnologue: Languages of the World, 14th Edition, URL: <http://www.ethnologue.com/>, Bibliography database last modified: November-2003, Publications catalog database last modified: December-2004
2. Xerox Language Identification System (2005), <http://www.languageidentifier.com/>
3. Djoerd Hiemstra and Arjen P. de Vries (2000), Relating the new language models of information retrieval to the traditional retrieval models, Published as CTIT technical report TR-CTIT-00-09, May, <http://www.ctit.utwente.nl>
4. Patricia Newman. (1987). Foreign language identification: First step in the translation process. In Proceedings of the 28th Annual Conference of the American Translators Association, pages 509–516
5. Stephen Johnson (1993), Solving the problem of language recognition Technical report, School of Computer Studies, University of Leeds
6. Dunning, T. (1994). Statistical identification of language. Technical Report CRL Technical Memo MCCS-94-273, University of New Mexico
7. Cavnar, W and Trenkle, J. (1994) "N-gram-based text categorization" in Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp.161—175
8. Judith Hayes (1993), Language Recognition using two and three letter clusters. Technical report, School of Computer Studies, University of Leeds
9. Gavin Churcher (1994), Distinctive character sequences, personal communication
10. Salton, G. and McGill (1983), M. Introduction to Modern Information Retrieval. McGraw Hill
11. S.E. Robertson and K. Sparck Jones (1976). Relevance weighting of search terms. Journal of the American Society for Information Science, 27:129–146.
12. Shannon, C.E. (1948). A mathematical theory of communication. Bell System Technical Journal 27, 379–423, 623–656.
13. H.Takçı, İ.Soğukpınar (2004), "Letter Based Text Scoring Method for Language Identification ", Lecture Notes in Computer Science, Springer-Verlag, Vol. 3261/2004, pages 283-290, October

14. S. Mustonen (1965). Multiple discriminant analysis in linguistic problems. In Statistical Methods in Linguistics. Skriptor Fack, Stockholm. Number 4.
15. D. V. Ziegler (1991). The automatic identification of languages using linguistic recognition signals. PhD thesis, SUNY Buffalo
16. Sibun, P., & Reynar, J. (1996). Language identification: Examining the issues. In Symposium on Document Analysis and Information Retrieval, pp. 125--135, Las Vegas. <http://citeseer.nj.nec.com/sibun96language.html>
17. M. D. Rau (1974). Language identification by statistical analysis. Master's thesis, Naval Postgraduate School
18. A. S. House and E. P. Neuberg (1977). Toward automatic identification of the language of an utterance. I. Preliminary methodological considerations. *Journal of the Acoustical Society of America*, 62(3):708--713
19. R. A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, and V. Zue, editors (1997). Survey of the State of the Art in Human Language Technology. Cambridge University Press, Cambridge, UK.
20. N. C. Ingle (1991). A language identification table. *The Incorporated Linguist*, 15(4):98--101
21. P. Heinrich (1989). Language identification for automatic grapheme-to-phoneme conversion of foreign words in a german text-to-speech system. In *Speech-89*, pages 220--223
22. S. Kulikowski (1991). Using short words: a language identification algorithm. Unpublished technical report
23. E. O. Batchelder (1992). A learning experience: Training an artificial neural network to discriminate languages. Technical Report
24. Yoshio Ueda and Seiichi Nakagawa (1990). Prediction for phoneme/syllable/word-category and identification of language using HMM. In *Proceedings of the 1990 International Conference on Spoken Language Processing*, volume 2, pages 1209--1212, Kobe, Japan, November.
25. J. C. Schmitt (1991). Trigram-based method of language identification, October. U.S. Patent number: 5062143.
26. Kwan, H., and Hirose, K., (1995) "Recognized Phoneme-based N-gram Modeling in Automatic Language Identification" *Proc. EUROSPEECH '95*, pp.1367-1370

27. D. Benedetto, E. Caglioti, and V. Loreto, (2002), Language trees and zipping. Physical Review Letters, 88:4
28. K. R. Beesley (1988). Language identifier: A computer program for automatic natural-language identification on on-line text. In Proceedings of the 29th Annual Conference of the American Translators Association, pages 47--54
29. Damashek, Marc (1995). "Gauging Similarity with n-grams: Language-Independent Categorization of Text", Science, Vol. 267, 10 February
30. Manning, C. and H. Schütze (1999). Foundations of Statistical Natural Language Processing. MIT Press, Boston.
31. Djoerd Hiemstra and Franciska de Jong (2001), "Statistical Language Models and Information Retrieval: natural language processing really meets retrieval", This article was published in Glot International 5(8), pages 288-294, <http://www.glotinternational.com>
32. Zhdanova AV (2002), Automatic identification of European languages, Natural Language Processing and Information Systems Lecture Notes in Computer Science 2553: 76-84
33. Prager JM (1999) Linguini: Language Identification for Multilingual Documents Journal of Management Information Systems 16 (3): 71-101 Win
34. MacNamara S, Cunningham P, Byrne J (1998), Neural Networks for Language Identification: A Comparative Study, Information Processing & Management 34 (4): 395-403 JUL
35. Arjen Poutsma (2001), Applying Monte Carlo Techniques to Language Identification, CLIN 2001 (Twelfth Meeting of Computational Linguistics in the Netherlands)
36. Robertson AM, Willett P (1998), Applications of N-grams in Textual Information Systems, Journal of Documentation, 54 (1): 48-69 JAN
37. Tsai WH, Chang WW (2002), Discriminative training of Gaussian mixture bigram models with application to Chinese dialect identification, SPEECH COMMUNICATION 36 (3-4): 317-326 MAR
38. Ari Visa (2001), Technology of Text Mining, P. Perner (Ed.): MLDM 2001, LNAI 2123, pp. 1-11
39. S. Dumas, J. Plat, D. Heckerman, and M. Sahami (1998). Inductive learning algorithms and representation for text categorization. In Proceedings of CIKM-



- 98, 7th ACM International Conference on Information and Knowledge Management, pages 148-155
40. E. Hong (Sam) Han and G. Karypis (2000), Centroid-Based Document Classification: Analysis & Experimental Results, University of Minnesota, Department of Computer Science, Army HPC Research Center Minneapolis, MN 55455 Technical Report: #00-017
 41. H.Takçı, İ.Soğukpınar, (2004) "Centroid-Based Language Identification Using Letter Feature Set", Lecture Notes in Computer Science, Springer-Verlag, Vol. 2945/2004, pages 635-645, February
 42. Grefenstette, G. (1995), Comparing two Language Identification Schemes, in Proceedings of the 3rd International Conference on the Statistical Analysis of Textual Data (JADT'95), Rome, Italy, December
 43. Principles of data mining, David hand, heikki mannila, padhraic smyth, ISBN 0-262-08290-x, sayfa (165-169)
 44. Two Crows Corp (1998). Introduction to Data Mining and Knowledge Discovery.
 45. T. Lahtinen (2000). Automatic indexing: an approach using an index term corpus and combining linguistic and statistical methods. PhD thesis, Department of General Linguistics, University of Helsinki, Finland
 46. H. P. Luhn (1958). The automatic creation of literature abstracts. IBM Journal of Research and Development, 2:159–165
 47. European Corpus Initiative Multilingual Corpus I (ECI/MCI) (2005), <http://www.elsnet.org/resources/eciCorpus.html>, Page last modified 29-03-2005
 48. StatSoft, Inc. (2003). STATISTICA (data analysis software system), version 6. <http://www.statsoft.com>
 49. R.Ullman (2000), Data Mining Lecture Notes, available at <http://www-db.stanford.edu/~ullman/mining/mining.html>
 50. S. Chakrabarti (2000), "Data mining for hypertext: a tutorial survey," SIGKDD: SIGKDD Explorations: Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining, ACM, vol. 1, pp. 1—11
 51. Xueqiang Lu lxq@pku.edu.cn Le Zhang Institute of Computer Software Theory... In Proceedings of The First International Joint Conference on Natural Language Processing (IJCNLP-04)



ÖZGEÇMİŞ

1974 yılında Sivas/Gürün'de doğdu. 1991 yılında Gürün Lisesi'nden mezun oldu. 1991-1993 yılları arasında 9 Eylül Üniversitesi Mühendislik Mimarlık Fakültesi Jeoloji Mühendisliği bölümünde öğrenim gördü. 1993 yılında Trakya Üniversitesi Mühendislik Mimarlık Fakültesi'nde eğitime başlayıp 1997 yılında Bilgisayar Mühendisi olarak mezun oldu. 1997 yılında Gebze Yüksek Teknoloji Enstitüsü (GYTE) Bilgisayar Mühendisliği Bölümünde yüksek lisansa başlayıp 1999 yılında Bilgisayar Yüksek Mühendisi olarak mezun oldu. 1999 yılında GYTE'de başladığı Doktora eğitimi hala devam etmektedir. Aynı zamanda GYTE'de 1997-2002 yılları arasında Araştırma Görevlisi olarak, 2002 yılından bu yana da Öğretim Görevlisi olarak çalışmalarına devam etmektedir. Evli ve iki çocuk babasıdır.

