Metin Benzerliği Algoritmaları ile Veri Tekilleştirme: Oteller Veri Tabanında Bir Uygulama

Data Deduplication with Text Similarity Algorithms: An Application in a Hotels Database

Ünzüle KELEŞ
Yapay Zeka Mühendisi,
Bi.Technology Şirketi, İstanbul, Türkiye
unzule.keles@julienn.com
ORCID: 0000-0001-9539-6981

Nevcihan DURU
Kocaeli Sağlık ve Teknoloji Üniversitesi,
Bilgisayar Mühendisliği Bölümü, Kocaeli, Türkiye
nevcihan.duru@kocaelisaglik.edu.tr
ORCID: 0000-0003-2154-7067

Öz

Hızla ve kontrolsüzce artan veri dünyasında, veri tekilleştirme büyük bir ihtiyaçtır. Veri tekilleştirme, veri tabanında yinelenen verilerin kopyalarını ortadan kaldırmak ya da onları tespit ederek, eşsiz kimlik numarası ile betimlemek için kullanılan bir tekniktir. Oteller veri tabanında; ismi, adresi, acente bilgisi ve fiyat bilgisi bulunan bir otelin, kayıtlarda farklı isim ve kimlik numarası ile bulunması, karşılaştırma olanağı sunmamaktadır. Rakip analizinin tam anlamıyla yapılabilmesi, rakip fiyat durumlarının bilinmesi ve pazar takibi bütün bu otel isimlerinin tekilleştirilmesi ile mümkündür. Bu çalışmanın amacı, otel isimlerini eşleştirerek, eş olan otelleri tek bir kimlik numarası ile tanımlamak ve tekil veriyi elde etmektir.

Veri tekilleştirme için çeşitli metin benzerliği algoritmaları kullanılır. Bu çalışmada, Metin Benzerliği algoritmaları; oteller veri tabanında, otel isimleri üzerinden karşılaştırılmıştır. Mesafe düzenleme bazlı benzerlik, belirteç bazlı benzerlik, diziliş tabanlı benzerlik algoritmaları ile bulanık dize eşleme algoritmaları incelenmiştir. Çalışmanın sonucunda Bulanık dize eşleme algoritmasının ve Jaro Winkler mesafesinin birlikte kullanıldığı hibrit bir model önerilmiştir. Model; test veri seti üz

erinden yapılan karşılaştırmada, kolay veri setinde %94, zor veri setinde %83 Doğruluk (Accuracy) sonucunu vermiştir.

Gönderme ve kabul tarihi: 22.09.2021 - 07.11.2021
Makale türü: Araştırma

Anahtar Sözcükler: *Metin Benzerliği, Veri Tekilleştirme, Bulanık Dize Eşleme, Jaro Winkler Mesafesi*

Abstract

Deduplication is a great need in the world of data that is growing rapidly and uncontrolled. Data deduplication is a technique used to eliminate duplicate copies of data in the database or to identify and describe them with an Id. Finding a hotel with a different name and ID in the records in the database of hotels with name, address, agency, and price information does not allow comparison. Competitor analysis can be done completely, knowing competitor price situations and market follow-up is possible by singularization of all these hotel names. The purpose of this study is to identify identical hotels with a single identification number by matching hotel names and to obtain singular data.

Various text similarity algorithms are used for deduplication. In this study, Text Similarity algorithms; Hotels were compared in the database over hotel names. Distance edit based similarity, token based similarity, sequence based similarity algorithms and fuzzy string matching algorithms were examined. As a result of the study, a hybrid model in which Fuzzy String Matching algorithm and Jaro Winkler distance are used together is proposed. Model; In the comparison made over the test data set, it gave 94% Accuracy results in the easy data set and 83% Accuracy in the difficult data set.

Keywords: *Text Similarity, Data Deduplication, Fuzzy String Matching, Jaro Winkler Distance*

1. Giriş

İnternetin toplum yaşamına girmesi, birçok kullanıcıya ulaşması, büyümesi, gelişmesi yeni teknolojileri de beraberinde getirmiştir. Elektronik Ticaret kısaca E-Ticaret olarak isimlendirilen kavram, toplum hayatına hızlı bir şekilde nüfuz etmiştir [1].

E-Ticaretin yaygınlaşmasıyla birlikte turizm sektörü de bu gelişmelere hızla ayak uydurmuştur. Turizm sektöründe faaliyet gösteren birçok işletme, satış kanallarını güncellemiş, internet üzerinden hizmet sağlamaya başlamıştır. E-Ticarete giren ve gelişmelerden geri kalmayan turizm işletmeleri ciddi kazançlar sağlamış, rekabeti arttırmıştır. Bu noktada pazara, Online Seyahat Acenteleri (Online Travel Agency - OTA) girmiştir [2].

OTA, Turizm ve Otelcilik sektöründe, seyahat acentelerine alternatif olarak geliştirilen ve internet üzerinden satış yapan seyahat acentelerine verilen isimdir. OTA'lar; internet üzerinden satış yapması nedeniyle hem güvenli hem hızlı hem de küresel bir pazar oluşturmuştur. Kullanıcı, istediği bir konumdan, istediği diğer bir konumdaki otele rezervasyon yapabilmektedir.

Ödeme kolaylıkları (taksitle öde, otelde öde, vb.), çeşitli kampanyalar, indirimler, kolay rezervasyon yapabilme gibi özelliklerle kullanıcıyı cezbeden OTA'lar, pazarda büyük pay sahibidir.

Avrupa Otel, Restoran ve Ağırlama Sektörü Birliği (HOTREC), 2020 yılının başında bir çalışma yapmıştır. Bu çalışmaya göre, 2013-2019 yılları dağıtım kanalları için, Avrupa otel sektöründe OTA'nın pazar payı 2013'te 19,7'den 2019'da 29,2'ye istikrarlı bir şekilde artmıştır (Pandemi dönemine ait veriler dahil edilmemiştir) [3].

Aşağıda bazı yerel ve küresel OTA'lara örnek verilmiştir.

- Tatil Sepeti
- Ets Tur
- Booking
- Expedia
- Tatil Budur
- Agoda
- Hotels.com
- Jolly Tur

OTA'larda bulunan oteller incelendiğinde, farklı isim yapıları, farklı konum bilgileri ve farklı fiyat teklifleri bulunduğu görülmüştür. Örneğin Tatil Sepeti'nde (OTA-1) Delta By Marriott İstanbul Haliç ismi ile bulunan otel, otelz.com'da (OTA-2) Delta Hotels By Marriott İstanbul Halic ismi ile bulunur. Tüm pazarda bu kadar büyük paya sahip OTA'larda, bir otelin, farklı isimlerle bulunması, karşılaştırma olanağı sunmaz.

Rekabetin çok büyük olduğu pazarda, Rakip analizinin tam anlamıyla yapılabilmesi, rakip fiyat durumlarının bilinmesi ve pazar takibi bütün bu otel isimlerinin eşleştirilmesi ile mümkündür.

Bu çalışmada veri seti; İstanbul, Muğla ve Antalya illerinin çeşitli ilçelerinde (Taksim, Belek, vs.) bulunan otellerden örneklem alınarak oluşturulmuştur. Toplamda 8746 otelin; isim, adres (il-ilçe-mahalle), konum (enlem-boylam) ve acente (OTA) bilgilerinden oluşur.

Bu çalışmanın amacı, veri setinde farklı acente bilgisi (OTA bilgisi), farklı isim bilgisi ve farklı adres-konum bilgisi bulunan eş otelleri Metin Benzerliği algoritmaları ile eşleştirerek, eşsiz kimlik numarası ile tanımlamak böylece tekil veriyi elde etmektir.

Bu çalışma sayesinde metin benzerliği algoritmalarının, otel isimleri veri seti üzerindeki başarısı ölçülmüştür. Bu kapsamda, metin benzerliği ile ilgili literatürde yapılmış çalışmalar özetlenmiştir. Metin benzerliği algoritmaları ele alınmıştır. Ücretsiz ve açık kaynak kodlu python ortamında çalışılmıştır ve model adımları model bölümünde ayrıntıları açıklanmıştır.

2. Literatür İncelemesi

Bu bölümde, metin benzerliği ve metin benzerliği kullanılarak veri tekilleştirme konularının incelendiği çalışmalardan bahsedilmektedir. Mesafe düzenleme bazlı benzerlik algoritmaları, belirteç bazlı benzerlik algoritmaları, diziliş tabanlı benzerlik algoritmaları ve öğrenmeye dayalı metin benzerliği algoritmalarının incelendiği literatürde, çalışmalar analiz edilmiştir.

Bu çalışmada mesafe düzenleme bazlı benzerlik algoritmaları, belirteç bazlı benzerlik algoritmaları ve diziliş tabanlı benzerlik algoritmaları incelenmiştir. Bu algoritmaların otel isimleri ve adresleri üzerindeki başarısı ölçülmüştür. Beş aşamalı bir model oluşturulmuş, iki oteli eşlerken bu model adımları sınanmış ve adımlara puantajlar verilmiştir. Toplanan puanlar kapsamında da bir eşik değeri belirlenmiş ve eşik değerine göre eş-eş değil ataması yapılmıştır.

Pullman ve ark. [4] , bir coğrafi sözlük veya dizin görevi gören, genellikle bir ülkenin, bölgenin, kıtanın coğrafi yapısı, sosyal istatistikleri ve fiziksel özellikleri hakkında bilgi içeren Gazetteer isimli veri seti üzerinde; bölge ve ülke isimlerini, bu isimlerin varyantlarıyla, yer adlarının eşleştirilmesi üzerine çalışmışlardır. Kapsamlı setin performansını ölçmeyi ve eşleştirme tekniklerini karşılaştırmayı amaçlamışlardır. Yirmi bir algoritmanın sınanması sonucunda Levenshtein mesafesi; Tayvan veri kümesi dışındaki tüm veri kümelerinde başarılı performans göstermiştir.

Niwattanakul S. ve ark. [5] Jaccard Benzerlik Katsayısı kullanarak kelimeler arasındaki benzerliği ölçmüştür. Arama motorlarında, arama yapanların gerekli sonuçlara hemen erişmelerini kolaylaştırmak için, anahtar kelimeler ve dizin terimleri arasında bir benzerlik ölçümü yapılır. Sonuçlar, Jaccard katsayısı ile test verisindeki kelimelerin her harfiyle karşılaştırıldığında kelimelerin benzerliğini ölçmede iyi performans gösterebileceğini ortaya koymuştur. Yani, Jaccard benzerlik katsayısı, benzerlik ölçümünde kullanılabilecek kadar uygundur. Test sonuçları ayrıca belirli kelimelerin benzerliğini ölçerken, Jaccard benzerlik katsayısında bazı zayıflıklar olduğunu ortaya koymuştur.

Duarte J. M. ve ark. [6], genetik benzerlikleri ölçmek için 8 farklı benzerlik katsayısı modeli kullanmış ve sonuçları değerlendirmiştir. Katsayılar arasında karşılaştırmalar, bu katsayıların tamamlanmasıyla elde edilen genetik mesafelerin korelasyon analizi ve dendrogram değerlendirmesi yapılmıştır. Sorensen-Dice, iki boyutlu bir alanda daha yüksek verimliliği göstermesi nedeniyle en uygun model olarak kabul edilmiştir.

Jin X. ve ark. [7], ilk olarak Simhash algoritması kullanılmıştır. Bu algoritma bir belgeyi n basamaklı bir imzaya dönüştürür ve imzanın benzerliğini karşılaştırarak belgenin benzerliğini hesaplar. Simhash metni hızlı bir şekilde işler ve bir veri tabanında kolayca saklanabilen parmak izlerini hesaplar; bu nedenle, büyük miktarda metnin benzerlik hesaplaması için çok uygundur. Çalışmada Metin özniteliklerini çıkarmak için TF-IDF (Term Frequency–Inverse Document Frequency) algoritması kullanılmış, aday metinler seçilmiş ve son olarak kosinüs uzaklığı ile en benzer metin seçilmiştir. Deneyler için veri seti, 200.000'den fazla haber metninden rastgele 500 haber öğesi seçilerek oluşturulmuştur. Deneyler, bu yöntemin, değiştirilen metinler için doğruluğu açıkça geliştirdiğini göstermektedir. Bu yöntem, metin benzerliğini karşılaştırmak için uygundur. Bununla birlikte, daha ayrıntılı karşılaştırma için, tam metinden cümle düzeyine benzerlik tespiti elde etmek için gelecekteki çalışma olarak kabul edilen daha farklı bir benzerlik algoritmasına ihtiyaç vardır.

Chen G. ve ark. [8], yazarlar yaygın kullanılan metin benzerliği algoritmalarını ve web sayfası tekilleştirme algoritmalarını analiz etmişlerdir. Çalışmanın sonunda Simhash'e dayalı geliştirilmiş bir web sayfası tekilleştirme algoritması önerilmiştir. Algoritma, metin koleksiyonunu eşleme yoluyla depolamak için Simhash parmak izlerine dönüştürür ve Hamming mesafesi aracılığıyla iki parmak izinin benzerliğini hesaplar ve böylece web sayfasının benzerliğini elde eder. Deneyler, bu yazıda önerilen algoritmanın daha yüksek bir doğruluk oranı ve geri çağırma oranına sahip olduğunu ve kötü niyetli ayna web sitelerinin tanımlanması ve tespit edilmesi için uygulanabileceğini göstermektedir.

Vijaymeena M.K. ve ark. [9], metin benzerliği ile ilgili mevcut çalışmaları, dize tabanlı, bilgi tabanlı ve derlem tabanlı benzerlikler olarak üç önemli yaklaşıma ayırarak tartışmıştır. Algoritmaların hibrit kullanılmasına değinilmiştir. SimMetrics, WordNet Benzerliği ve NLTK (Natural Language Toolkit) gibi faydalı benzerlik paketlerinden bahsedilmiştir. Otomobil sektörü ve Sağlık sektörü gibi ortamlarda kullanılan metin madenciliğinin farklı gerçek zamanlı uygulamaları tartışılmıştır.

Mansoor M ve ark. [10] Uzun Kısa Süreli Bellek Ağı (LSTM) ile Evrişimsel Sinir Ağı'nı (CNN) birleştiren derin öğrenme modelleri kullanılarak yeni bir yaklaşım önermiştir. Çalışmada iki soru arasındaki anlamsal benzerliği ölçmek amaçlanmıştır. Önerilen model, aralarındaki benzerliği ölçmek için cümle çiftlerini girdi olarak alır. Model, herkese açık Quora'nın veri kümesinde test edilmiştir. Mevcut tekniklerle karşılaştırıldığında, %87.,50 doğruluk sonucu ile önceki yaklaşımlardan daha iyi sonuç vermiştir.

Bayrak A.T. ve ark. [11] oteller veri seti üzerinde, otel verilerini tutarlı hale getirmek için aynı oteli temsil eden kayıtları %99.12 doğruluk ile eşleştirmiştir. Levenshtein Benzerliği, kosinüs benzerliği ve Jaro-Winkler benzerliğini otel isimleri ve adresleri üzerinde kullanarak öznitelik oluşturulmuştur. Daha sonra bu öznitelikler; rassal ormanlar, destekçi vektör

makinesi ve yapay sinir ağı modelleri için girdi olarak kullanılmıştır. Otellerin görsel benzerliklerini de modele dahil etmiştir. Sonuç olarak veri tekrarını önleyerek veri boyutunu azaltmışlardır. Bu çalışma, literatürde incelenen çalışmalar arasında, bizim çalışmamıza amaç olarak en yakın çalışmadır fakat bu çalışmada, makine öğrenmesi yöntemleri kullanılarak ikili bir sınıflandırma yapılmıştır. Bizim çalışmamızda yeni bir puanlama sistemi tanıtılmış ve bu sistem üzerinden sınıflandırma yapılmıştır.

3. Materyal ve Metot

Bu bölümde, veri tekilleştirme tanımlanmış, otellerin isimleri ve adresleri arasındaki benzerliği ölçmek için kullanılacak metin benzerliği algoritmaları incelenmiş, sonuçlar karşılaştırılmıştır.

Veri tekilleştirme, benzer özelliklere sahip bir veri kümesindeki girdileri tanımlama ve bunlarla ilişki kurma sürecidir.

Veri Tekilleştirme sonucu,

- Doğru ve tutarlı veriye ulaşılır.
- Veri takip edilebilir ve kontrol edilebilir olur.
- Verinin kontrollü azaltılması sayesinde kümeleme, sınıflandırma vs. işlemlerin daha hızlı ve daha doğru yürütülmesi sağlanır.
- Disk alanı ihtiyacını azaltır.
- Gereksiz veya tekrarlayan bilgileri kaldırarak veri kalitesini arttırır [12] .

Bu çalışmada veri tekilleştirme, otel isimlerini tekilleştirme amacı ile kullanılmıştır.

3.1 Metin Benzerliği Algoritmaları

İşlemlerin özelliklerine bağlı olarak, metin benzerlik algoritmaları bir grup alanda sınıflandırılabilir. Her alanda bahsedilen algoritmalar incelenmiştir.

Mesafe Düzenleme Bazlı Benzerlik, bu kategori altında yer alan algoritmalar, bir dizeyi diğer dizeye dönüştürmek için gereken işlem sayısını (ekleme, silme, değiştirme vs.) hesaplamaya çalışır. İşlem sayısı arttıkça, iki dize arasındaki benzerlik daha az olur. Bu kategoriye giren algoritmalar; Levenshtein Mesafesi, Hamming Mesafesi, Jaro-Winkler Mesafesi, Needleman Wunsch Mesafesi ve Smith-Waterman Mesafesidir.

Belirteç Bazlı Benzerlik, bu kategoride beklenen girdi, tam dizelerden ziyade bir dizi belirteçtir. Fikir, her iki sette de benzer belirteçleri (token) bulmaktır. Ortak belirteç sayısı arttıkça, setler arasındaki benzerlik de artar. Bu kategoriye giren algoritmalar; Jaccard Benzerlik Katsayısı, Sorensen Dice Benzerlik Katsayısı, Kosinüs Benzerliği, Tversky İndeksi, Monge Elkan Mesafesi ve Bag Mesafesidir.

Diziliş Tabanlı Benzerlikte, benzerlik, iki dize arasındaki ortak alt dizelerin bir faktörüdür. Algoritmalar, her iki dizede mevcut olan en uzun diziyi bulmaya çalışır, bu dizilerden daha fazlası bulunursa, benzerlik puanı daha yüksektir. Bu kategoriye giren algoritma; Ratcliff/Obershelp Benzerliğidir.

Aşağıdaki bölümlerde, yukarıda bahsedilen her algoritma incelenecek, otel isimleri üzerinden benzerlik karşılaştırması yapılacaktır. Başarı sonuçlarına göre de bu çalışmada kullanılacak algoritma ya da algoritmalar seçilecektir.

3.1.1 Levenshtein Mesafesi

Levenshtein mesafesi iki dize arasındaki benzerliğin bir ölçüsüdür. Algoritma, a dizesini b dizesine dönüştürmek için gereken minimum değişiklik sayısı olarak tanımlanır. Bu, a dizesine bir karakter ekleyerek, silerek veya değiştirilerek yapılır. Levenshtein mesafesi ne kadar küçükse, dizeler de bir o kadar benzerdir [13] . Aşağıda formülize edilmiştir.

$$lev_{a,b}(i,j) = \begin{cases} \max(i,j), & \text{if } \min(i,j) = 0, \\ \min \begin{cases} lev_{a,b}(i-1,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \end{cases} & \text{otherwise} \end{cases} \quad (1)$$

Levenshtein mesafesini hesaplamak için, her kelimeden geçmeli ve en az + 1 kullanmalıyız.

(i-1, j): sol kutu anlamına gelir (silme)

(i, j-1): üst kutu anlamına gelir (ekleme)

(i-1, j-1): sol üst kutu anlamına gelir (yer değiştirme) [14].

3.1.2 Hamming Mesafesi

Bir dizeyi diğerine dönüştürmek için gereken minimum yer değiştirme sayısını veya bir dizeyi diğerine dönüştürmüş olabilecek minimum hata sayısını ölçer. Daha genel bir bağlamda, Hamming mesafesi, iki dizi arasındaki düzenleme mesafesini ölçmek için kullanılan çeşitli dize metriklerinden biridir. İsmini Amerikalı matematikçi Richard

Hamming'den alır. Aşağıda bir örneği verilmiştir.

"karolin" ve "kathrin" için Hamming mesafesi 3'tür [15].

3.1.3 Jaro-Winkler Mesafesi

Jaro algoritması veri bağlantı sistemlerinde, isim eşleşmesi için yaygın olarak kullanılır. Ekleme, silme ve yer değiştirme işlemlerini açıklar. Algoritma, ortak karakterlerin c sayısını (daha uzun dizenin yarısında kalan karakterleri kabul ederek) ve yer değiştirme sayısını t'yi hesaplar. Bir benzerlik ölçümü olarak aşağıdaki gibi hesaplanır:

$$sim_{jaro}(s_1, s_2) = \frac{1}{3}\left(\frac{c}{|s_1|} + \frac{c}{|s_2|} + \frac{c-t}{c}\right) \quad (2)$$

Winkler algoritması, genellikle isimlerin başında daha az hatanın meydana geldiğini bulan ampirik çalışmalara dayanan fikirler uygulayarak, Jaro algoritmasını geliştirir. Winkler algoritması bu nedenle ilk karakterleri kabul etmek için Jaro benzerlik ölçüsünü arttırır.

$$sim_{wink}(s_1, s_2) = sim_{jaro}(s_1, s_2) + \frac{s}{10}\left(1 - sim_{jaro}(s_1, s_2)\right) \quad (3)$$

s, kabul edilen karakter sayısıdır.

iki dizenin başlangıcı için, örneğin; "peter" ve "petra" nın s = 3 değeri vardır [16].

3.1.4 Needleman Wunsch Mesafesi

1970 yılında, Saul B. Needleman ve Christian D. Wunsch, Needleman-Wunsch algoritması olarak da adlandırılan, sekans hizalaması için sezgisel bir homoloji algoritması önerdi. Bu algoritma, hesaplama adımlarını gerektiren (ve hizalanan iki dizinin uzunlukları olan) küresel bir hizalama algoritmasıdır. Küresel hizalamayı göstermek amacıyla bir matrisin yinelemeli hesaplamasını kullanır. Karakter dizisinin sırasını dikkate aldığı için dize karşılaştırması için en iyisidir.

Diziler arasında en uygun hizalama çözümünü bulur. Hizalamayı yapmak daha fazla zaman alır, bu da performansı azaltır [17] .

3.1.5 Smith-Waterman Mesafesi

Needleman-Wunsch algoritması kullanılırken, takip eden on yıl içinde Sankoff, Reichert, Beyer ve diğerleri gen dizilerini analiz etmek için alternatif sezgisel algoritmalar geliştirdi. 1976'da Waterman ve arkadaşları orijinal ölçüm sistemine boşluklar kavramını ekledi. 1981'de Smith ve Waterman, yerel hizalamayı hesaplamak için Smith-Waterman algoritmasını yayınladı. Büyük uzunlukta iki dizeyi hizalamak için: Smith-Waterman algoritması optimize edilir.

Needleman-Wunsch algoritmasından en büyük fark şunları içerir:

- En iyi yerel hizalamaları vurgulamak için negatif puanlama matris hücreleri sıfıra ayarlanır.
- Traceback prosedürü en yüksek puan alan matris hücresinden başlar ve sıfır puan alan bir hücreye ulaşılana kadar devam eder [18].

3.1.6 Jaccard Benzerlik Katsayısı

Jaccard benzerlik katsayısı, örnek kümelerin benzerliğini ve çeşitliliğini ölçmek için kullanılan bir istatistiktir. Jaccard katsayısı sonlu örnek kümeleri arasındaki benzerliği ölçer ve kesişim boyutunun örnek kümelerin birleşiminin boyutuna bölünmesiyle tanımlanır. Aşağıda gösterilmiştir.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (4)$$

Sonucun 1'e eşit olması iki kümenin eş olduğu anlamına gelir. Jaccard Mesafesi, 1'den Jaccard benzerlik katsayısının çıkarılması ile bulunur [19].

$$d_j(A,B) = 1 - J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|} \quad (5)$$

3.1.7 Sorensen Dice Benzerlik Katsayısı

Sorensen – Dice katsayısı, iki örneğin benzerliğini ölçmek için kullanılan bir istatistiktir. Bağımsız olarak sırasıyla 1948 ve 1945'te yayınlanan botanikçiler Thorvald Sørensen ve Lee Raymond Dice tarafından geliştirilmiştir. Aşağıda formülize edilmiştir [20].

$$DSC = \frac{2\,|X \cap Y|}{|X| + |Y|} \quad (6)$$

3.1.8 Kosinüs Benzerliği

Kosinüs benzerliği, bir iç çözüm uzayının iki vektörü arasındaki benzerliği ölçer. İki vektör arasındaki açının kosinüsü ile ölçülür ve iki vektörün kabaca aynı yönü gösterip göstermediğini belirler. Genellikle metin analizinde belge benzerliğini ölçmek için kullanılır. Bir belge, her biri belirli bir kelimenin (anahtar kelime gibi) veya belgede yer alan cümlenin frekansını kaydeden binlerce özellik ile temsil edilebilir. Dolayısıyla, her belge, terim frekansı

vektörü olarak adlandırılan bir nesne ile temsil edilir. Terim frekans vektörleri tipik olarak çok uzun ve seyrektir (yani, çok sayıda 0 değerine sahiptir). Bu yapıları kullanan uygulamalar arasında bilgi erişimi, metin belgesi kümelenmesi, biyolojik taksonomi ve gen özellik haritalaması bulunur. Kosinüs benzerliği, belgeleri karşılaştırmak veya belirli bir sorgu kelimesi vektörüne göre bir belge sıralaması vermek için kullanılabilecek bir benzerlik ölçüsüdür. Karşılaştırma için x ve y iki vektör olsun. Kosinüs ölçüsünü benzerlik fonksiyonu olarak kullanarak aşağıdaki gibi formülize edilir [21].

$$sim(x,y) = \frac{x \,.\, y}{\| x \| \| y \|'} \quad (7)$$

3.1.9 Tversky İndeksi

Adını Amos Tversky'den alan Tversky indeksi, bir dizeler üzerindeki asimetrik bir benzerlik ölçüsüdür. Tversky indeksi, Sorensen-Dice katsayısının ve Jaccard bir genellemesi olarak görülebilir. Aşağıda formülize edilmiştir.

$$S(X,Y) = \frac{|X \cap Y|}{|X \cap Y| + \alpha|X - Y| + \beta|Y - X|'} \quad (8)$$

X ve Y kümeleri için Tversky indeksi, 0 ile 1 arasında bir sayıdır. Burada α, β ≥ 0 Tversky indeksinin parametreleridir. Formülde α = β = 1 ise Jaccard katsayısı; α = β = 0.5 ise Sorensen-Dice katsayısı elde edilir. X'i prototip ve Y'yi varyant olarak düşünürsek, α prototipin ağırlığına karşılık gelirken, β varyantın ağırlığına karşılık gelir. Tversky indeksi α + β = 1 olarak formülize edilebilir [22] .

3.1.10 Monge Elkan Mesafesi

Monge-Elkan, her bir belirteç için en iyi eşleşmeyi bulan, belirteçlere ve iç benzerlik işlevine dayanan genel bir metin dizesi karşılaştırma yöntemidir. Bu bağlamda belirteçler, çoğu insan dilinde olduğu gibi kelimelere bölünmüş karakter dizileridir. Algoritma, tipik metin alanlarının özyinelemeli yapısını kullanır. İki dize, aynı atomik dize ise veya biri diğerini kısalttıysa, derece 1 ile eşleşir. Aksi takdirde, eşleşme derecesi 0 'dır [23]. Aşağıda formülize edilmiştir.

$$sim_{MongeElkan}(A,B) = \frac{1}{|A|}\sum_{i=1}^{|A|} max\{sim'(a_i,b_j)\}_{j=1}^{|B|} \quad (9)$$

3.1.11 Bag Mesafesi

Bag mesafe algoritması, x dizesindeki y dizesindeki bir karakterle benzersiz bir şekilde eşleştirilemeyen her karakteri numaralandırır ve bunun tersi de geçerlidir; iki değerden maksimum olanı Bag mesafesidir. Bag mesafesi Levenshtein mesafesine bir üst sınır koyar ve bunun hızlı bir yaklaşımı olarak önerilmiştir [24].

3.1.12 Ratcliff/Obershelp Benzerliği

Ratcliff / Obershelp benzerliği algoritması 1983 yılında John W. Ratcliff ve John A. Obershelp tarafından tanıtıldı. Bu algoritmanın eğitim yazılımı endüstrisi üzerinde etkisi oldu. Önceden, eğitim yazılımı genellikle sadece çoktan seçmeli testler sunmuştu, çünkü girilen verilerin işlenmesi ve kontrol edilmesi için kullanıcı tarafından yazılan cevap algoritmaları gerekiyordu. Örneğin, 18. hanedanın Mısır firavunun kim olduğu sorusu için, Tutankhamun, Tutenkhamun, Tutankhamen, Tutankhamon'un cevapları doğru olarak düşünülmelidir. Ayrıca, bir kullanıcı çift "m" girmiş veya başka tür bir yanlış yazmış olabilir. Ratcliff / Obershelp algoritması bu sorunun çözülmesine yardımcı oldu. Jaro-Winkler mesafe algoritması gibi Ratcliff / Obershelp, 0 ile 1 arasında bir değer döndürür; burada 1, verilen iki dize için tam bir eşleşmedir. Aşağıda formülize edilmiştir.

$$D_{RO} = \frac{2 * K_m}{|S_1| + |S_2|} \quad (10)$$

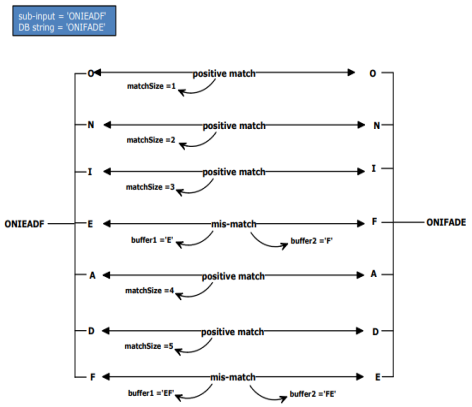
Burada K_m eşleşen harf sayısıdır [25] .

3.1.13 Bulanık Dize Eşleme

Bulanık dize eşleme işlemi, birbirine benzeyen iki dizeyle karşılaştığında son derece önemlidir. Algoritma mantığı, iki dize arasında doğrudan bir ilişki olmadığında, dizelerin hala bazı ortak noktalara sahip olabileceği gerçeğinden doğar. Bulanık dize eşlemesi, yanlış yazılmış dizeler, tutarsız girişler, eksik bağlam, farklı sıralama ve belirsiz veriler içeren bazı kirli veri sınıflarından kaynaklanan riske karşı çözüm sunar.

"Onifade" ve "onitade" dizelerini düşünün. Tipik bir eşleme algoritması bu girişle karşılaştığında farklılık ihmal edilir. Çünkü incelenirse dizeler aynıdır, 't 'karakteri hariç. Bununla birlikte, belirsiz bir şekilde bakıldığında, iki dizenin birçok ortak noktası vardır. İlk olarak, iki dize aynı anda analiz edildiğinde "oni" ve "ade" alt dizelerinin aynı konumda olduğunu belirleyebiliriz. Diğer bir nokta, her ikisinin de aynı sayıda karaktere sahip olmasıdır ve bu nedenle asıl sorun ya yazım hatası ya da aktarımdır.

Kullanıcının dizesini eşzamanlı olarak karşılaştırmak için, işlemin başlangıcında iki dinamik ara bellek oluşturulur. Biri "buffer1"; kullanıcı alt girdisinin eşleşmeyen karakterlerini, diğeri "buffer2", veri tabanı alt dizesinin eşleşmeyen karakterlerini tutar. Algoritma daha sonra iki dizinin karakter içeriğini aynı anda tarar. Karakterler benzer olduğunda, kaç karakterin eşleştiğini gösteren değişken artırılır. Karakterler farklıysa, iki karakter sırasıyla buffer1 ve buffer2'de saklanır. Tüm karakterler karşılaştırıldıktan sonra, dizelerden birinin sonuna gelir (iki dizenin boyutunun aynı olmaması durumunda), bulanık eşleşme değeri, kapsama veya aidiyet düzeyine göre hesaplanır.



Şekil-1: Bulanık dize eşleme algoritması geçiş örneği

Şekil 1'de örnek bir sorgu dizesinin bulanık dize eşleme algoritmasından geçişi gösterilmiştir [26].

3.2 Metin Benzerliği Algoritmalarının Karşılaştırılması

Yukarıda işlenen Metin Benzerliği algoritmaları, python ortamında incelenmiş ve otel isimleri üzerinden karşılaştırılmıştır. Eş ve kayıtlarda iki farklı isimde bulunan otellerden örneklem seçilerek isimlerin benzerliği 0 ile 1 arasında derecelendirilmiştir. Bu derecelendirme python programlama dili içinde bulunan textdistance kütüphanesinin normalized_similarity özelliği ile normalize edilmiştir. İki ismin eş olma durumu 1, eş olmama durumu ise 0'dır. Aradaki değerler ise eş olmaya ya da eş olmamaya yakınlığı gösterir. Böylece tüm algoritmalar karşılaştırılmıştır. Aşağıda bir otel üzerinden örnek gösterilmiştir.

Çizelge-1: Bir Otel Örneği Üzerinden Metin Benzerliği Algoritmalarının Sonuçları

Otel İsimleri	Levenshtein M.	Hamming M.	Jaro-W. M.	Needleman W.M.	Smith-W. M.	Jaccard K.	Sorensen Dice K.	Kosinüs B.	Tversky	Monge E. M.	Bag M.	R/O B.	Bulanık Dize E.
Corinne Art Boutique Hotel	0.23	0.04	0.60	0.37	0.08	0.50	0.33	0.71	0.50	0.02	0.5	0.36	0.86
Hotel Corinne													

Veri setindeki eş otel çiftlerinden (tek otel iki farklı isim) 50 çift otel seçilerek, benzerlik algoritmalarının başarıları ölçülmüştür. Aşağıda sonuçlar verilmiştir.

Çizelge-2: Metin Benzerliği Algoritmalarının Sonucu

Algoritma	0.85 ve üzeri benzerlik sonucu veren çift sayısı	0.85 altı benzerlik sonucu veren çift sayısı
Levenshtein M.	32	18
Hamming M.	12	38
Jaro-Winkler M.	40	10
Needleman W.M.	23	27
Smith-W. M.	12	38
Jaccard K.	29	21
Sorensen Dice K.	13	37
Kosinüs B.	35	15
Tversky	23	27
Monge E. M.	10	40
Bag M.	22	28
Ratcliff/Obershelp B.	16	34
Bulanık Dize Eşleme A.	48	2

Çizelge-2'de verilen 50 sonuçtan, Bulanık dize eşleme algoritması ve Jaro Winkler algoritması, eş otellerde en çok 0.85 ve üzeri benzerlik sonuçları verdiği için, modelde kullanılmaya uygun bulunmuştur.

4. Veri Seti

Veri toplayıcı servislerinden çeşitli OTA'lara ait (tatil sepeti, tatilbudur, ets tur vs.) veriler toplanmıştır. İstanbul, Muğla ve Antalya illeri için, otel ismi, OTA bilgisi, adres bilgisi ve konum bilgisi veri tabanına kaydedilmiştir. Toplamda 8746 veri, veri tabanına kaydedilmiştir.

Veri setinde konum bilgisi değerlerinde yanlışlıklar ve eksikler olması sebebi ile, bu bilgiler doğrulama gereksinimi oluşturmuştur. Model için konum bilgisi çok önemli bir girdidir. Doğrulama için Google Places uygulama programlama arayüzü (Google Places Api), kullanılmıştır. Google Places Api, HTTP (Hyper Text Transfer Protocol) isteklerini kullanarak yerler hakkında bilgi döndüren bir hizmettir [27].

Otel ismini ve bölgesini girdi olarak kullanarak Api'den dönen enlem- boylam bilgisi, isim bilgisi ve adres bilgisi kaydedilmiştir. Api'den gelen lokasyon bilgisi ile veride bulunan lokasyon bilgisi karşılaştırılmıştır. Aralarındaki mesafe metre ve kilometre bazında python'da bulunan geopy.distance kütüphanesi ile ölçülmüştür. Geopy, python geliştiricilerinin, coğrafi kodlayıcıları ve diğer veri kaynaklarını kullanarak dünya genelindeki adreslerin, şehirlerin, ülkelerin ve yer işaretlerinin lokasyonlarını bulmasını kolaylaştıran bir kütüphanedir [28].

Veri setinde Konum Doğru isimli bir öznitelik oluşturulmuştur ve Google Places Api'den dönen konum ile veri setindeki konum değeri arası fark 10 kilometreden az ise verilerin Konum Doğru bilgisine 1 değeri atanmıştır. İşlemler sonucunda veri setindeki konumların %90'ı doğrulanmıştır. Aşağıda veri seti örneği verilmiştir.

Çizelge-3: Veri Seti Örneği

Otel No	Otel İsmi	OTA	Bölge	Alt Bölge	Bölge Detay	Enlem	Boylam	Konum Doğru	Google İsmi
ets-58536	Korsan Ada Hotel	ets	Antalya	Kaş		36.194301	29.594155	1	Korsanada hotel
tatilsepeti-5013	Skalion Hotel Spa	tatilsepeti	İstanbul	Fatih		41.0025228	28.964836	1	Skalion Hotel & Spa
ets-20193	Dorman Suites Hotel	ets	Muğla	Bodrum	Bitez	37.027195	27.386739	1	Dorman Suites Hotel

Model adımına geçilmeden önce, modelin başarısını ölçmek için test verileri oluşturulmuştur. Yani eş otellere, uzman bilgisi ile eşsiz bir otel numarası atanmıştır. Böylelikle modeldeki puanlamada yapılan değişikliklerin etkisi ölçülmüştür. Otellere, Global Otel No özniteliği, kontrollü olarak atanmıştır.

Test verisi, 8746 otel içerisinden 1000 otel kontrollü şekilde seçilerek oluşturulmuştur. Kolay, Zor ve ikisinin toplamını kapsayan Karışık isimli 3 veri seti oluşturulmuştur Kolay veri seti, 500 otel için, kolay bulunabilecek, isim benzerlikleri yüksek ve konumları yakın olan otellerden seçilmiştir. Zor veri seti, 500 otel için, isim benzerlikleri düşük ama aynı olan oteller, ya da isim benzerliği yüksek ama farklı olan oteller arasından seçilmiştir. Karışık, Kolay ve Zor veri setlerinin toplamından 1000 otel ile oluşturulmuştur.

Aşağıda kolay ve zor test verisinden örnekler verilmiştir.

Çizelge-4: Kolay Test Veri Seti Örneği

Otel No	Otel İsmi	Global Otel No
trivago-2030761	7 Art Fesleğen	455f43a2-0fba-11eb
ets-94272	7 Art Fesleğen Hotel	455f43a2-0fba-11eb
tatilSepeti-7315	7 Art Fesleğen Hotel	455f43a2-0fba-11eb

Çizelge-5: Zor Test Veri Seti Örneği

Otel No	Otel İsmi	Global Otel No
trivago-1360186	Yüksel	5ce76356-0fba-11eb
tatilSepeti-9004	Yüksel İstanbul Yenikapı Hotel	5ce76356-0fba-11eb

Global Otel Numaraları test verisine atandıktan sonra, model adımına geçilmiştir.

5. Model

Model 5 aşamadan oluşturulmuştur. Birinci aşamada, otel isimleri arasındaki benzerlik, bulanık dize eşleme algoritması ile ölçülmüştür. Bu ölçümün sonucunda model aşaması, benzerlik yüksek ise yüksek, düşük ise düşük bir puan almıştır.

İkinci aşamada, otel isimleri arasındaki benzerlik Jaro Winkler mesafe algoritması ile ölçülmüştür. Ölçüm sonucunda, bu aşamaya da puan verilmiştir. Üçüncü aşamada, Google Place Api'den dönen otel isimleri karşılaştırılmıştır. Karşılaştırılan isimler eşse puan eklemesi yapılmıştır. Dördüncü aşamada, veri setinde bulunan, Bölge, Alt Bölge ve Bölge Detay bilgileri birleştirilmiştir. Adres ismi verilen birleştirme sonucunda, iki otelin adres benzerliği bulanık dize eşleme algoritması ile ölçülmüştür. Benzerlik sonucuna göre bu aşamaya da puan verilmiştir. Son aşamada, veri setinde bulunan, enlem ve boylam bilgileri kullanılarak, karşılaştırılan iki otel arasındaki uzaklık ölçülmüştür (metre bazında). İki otel arasındaki mesafe az ise aynı otel olmaları, çok ise farklı otel olmaları olasılığı yüksektir. Bu mantık üzerine puan verilmiştir.

Modelin bu kadar aşama ile sınanmasının sebebi; iki otel ismi için sadece isimleri incelenerek "bu otel isimleri tek bir oteli tanımlar, eştir" çıkarımının yapılamamasıdır. Veri setinde rastlanmıştır ki; çok benzer isimlerde (benzerlik algoritması sonucu yüksek) farklı oteller, çok farklı isimlerde (benzerlik algoritması sonucu düşük) eş oteller mevcuttur. Aşağıda puanlama sistemi ve tüm bu aşamalar açıklanmıştır.

5.1 Puanlama Sistemi

Puanlama sistemi 0 ile 100 arası sayılardan oluşturulmuştur. Bu aralıklarla, model aşamalarına verilen puanlar test veri seti üzerinde sınanmıştır. Her aşamaya, puan verilecek özellik sayısına yani sınamaya ve o özelliğin önem derecesine yani puan sınıfına göre, rasgele puanlar verilmiştir. Aşağıda örnek verilmiştir.

Sınama: Otel isimleri arası benzerlik 0.95 (0 -1 aralığında) ve üzeridir.

Puan Sınıfı: Yüksek puan verilmelidir. Aşağıda puan sınıflarının aralığı verilmiştir.

- Yüksek : 90 ile 100 arası
- Orta 1 : 71 ile 89 arası
- Orta 2 : 60 ile 70 arası
- Orta 3 : 50 ile 59 arası
- Orta 4 : 40 ile 49 arası
- Orta 5 : 30 ile 39 arası
- Düşük 1 : 20 ile 29 arası
- Düşük 2 : 16 ile 19 arası
- Düşük 3 : 13 ile 15 arası
- Düşük 4 : 10 ile 12 arası
- Düşük 5 : 5 ile 9 arası
- Düşük 6 : 1 ile 4 arası
- Düşük 7 : -15 ile 0 arası

Yukarıdaki puan aralıklarında her aşama rasgele (puan sınıfı içindeki değerler arasında) puan almıştır. Rasgele Puan kümeleri model aşamaları ve puan sınıflarına göre aşağıdaki gibi oluşturulmuştur.

Çizelge-6: Puanlama Sistemi

Rasgele Puanlar	Model Aşaması	Puan Sınıfı	Puanlar
1	1	Yüksek, Orta 2, Orta 3, Orta 4, Orta 5	90,60,55,40,30
	2	Düşük 3, Düşük 4, Düşük 5	15,10,5
	3	Düşük 3	15
	4	Düşük 3, Düşük 4, Düşük 5, Düşük 7	15,10,5,-10
	5	Orta 5, Düşük 3, Düşük 4,	30,15,10
		Düşük 1, Düşük 3, Düşük 4	25,15,10
2	1	Yüksek, Orta 2, Orta 3, Orta 4, Orta 5	95,63,57,41,30
	2	Düşük 3, Düşük 4, Düşük 5	14,10,6
	3	Düşük 3	14
	4	Düşük 3, Düşük 4, Düşük 5, Düşük 7	13,10,5,-15
	5	Orta 5, Düşük 3, Düşük 4,	35,15,12
		Düşük 1, Düşük 3, Düşük 4	22,15,10
3	1	Yüksek, Orta 2, Orta 3, Orta 4, Orta 5	95,63,58,41,30
	2	Düşük 3, Düşük 4, Düşük 5	14,10,6
	3	Düşük 3	13
	4	Düşük 3, Düşük 4, Düşük 5, Düşük 7	15,10,5,-14
	5	Orta 5, Düşük 3, Düşük 4,	35,15,12
		Düşük 1, Düşük 3, Düşük 4	22,15,10
4	1	Yüksek, Orta 2, Orta 3, Orta 4, Orta 5	95,63,57,41,30
	2	Düşük 3, Düşük 4, Düşük 5	14,10,6
	3	Düşük 3	14
	4	Düşük 3, Düşük 4, Düşük 5, Düşük 7	14,10,5,-12
	5	Orta 5, Düşük 3, Düşük 4,	35,14,11
		Düşük 1, Düşük 3, Düşük 4	22,15,11

Model aşamalarında gösterilen puanlar, test veri setinde, eşik değeri de değiştirilerek yapılan denemeler sonucunda doğruluk metriğine göre en başarılı sonucu vermiştir. Sonuçlar, çalışmanın sonunda verilmiştir.

5.2 Otel İsimleri Arasında Bulanık Dize Eşleme Algoritması ile Metin Benzerliği

Karşılaştırılan iki otelin isimlerinin benzerliği Bulanık dize eşleme algoritması ile ölçülmüştür. Model için bu aşamada sonuçlar aşağıdaki gibi puanlanmıştır.

- Eğer Benzerlik 1 ise, +90
- Eğer Benzerlik 0.95 ve üzeri ise, +60
- Eğer Benzerlik 0.90 ve üzeri ise, +55
- Eğer Benzerlik 0.85 ve üzeri ise, +40
- Eğer Benzerlik 0.75 ve üzeri ise, +30

Benzerlik oranının en düşük 0.75 belirlenmesinin sebebi, benzerlik oranı düşük isimlerin de diğer sınama aşamalarına girmelerini sağlamaktır. Böylelikle, isim benzerliği düşük ama eş olan otelleri belirlemek amaçlanmıştır.

5.3 Otel İsimleri Arasında Jaro Winkler Mesafesi ile Metin Benzerliği

Bu aşamada Jaro Winkler sonuçlarının aldığı puanlar verilmiştir. Modeli, birinci aşamada bulanık dize eşleme algoritması ile sınadıktan sonra tekrar bir algoritma ile daha sınamaktaki amaç; bulanık dize eşleme algoritmasının bazı farklı oteller için yüksek benzerlik oranı vermesidir. Bu aşamada, bu durumun önüne geçmek amaçlanmıştır. Puanlar aşağıdaki gibi atanmıştır.

- Eğer Benzerlik 0.95 ve üzeri ise, +15
- Eğer Benzerlik 0.75 ve üzeri ise, +10
- Eğer Benzerlik 0.55 ve üzeri ise, +5

5.4 Google İsmi Eşitliği

Google Places Api'den dönen Google İsimleri karşılaştırılmıştır.

Aşağıda veri setinden bir örnek verilmiştir.

Çizelge-7: Google İsmi Örneği

Otel No	Otel İsmi	Google İsmi
tatilSepeti-6234	Wise Hotel Spa - Adult Onl	Wise Hotel & Spa
ets-91234	Wise Hotel & SPA - Adult Only	Wise Hotel & Spa

Çizelge-7'te görülen Google isimleri karşılaştırılmış ve iki ismin eş olup olmamasına göre bir puan verilmiştir.

- Eğer Google isimleri eş ise, +15

5.5 Adres Benzerliği

Veri setinde bulunan Alt Bölge ve Bölge Detay bilgileri birleştirilmiştir. Bu birleştirme sonucu, Adres ismi verilen öznitelik oluşturulmuştur. Aşağıda örnek verilmiştir.

- Alt Bölge : fatih
- Bölge Detay : laleli
- Adres : fatih laleli

Adresler arasında Bulanık dize eşleme algoritması ile benzerlik ölçülmüştür. Aşağıda puantajlar gösterilmiştir.

- Eğer Benzerlik 1 ise, +15
- Eğer Benzerlik 0.85 ve üzeri ise, +10
- Eğer Benzerlik 0.55 ve üzeri ise, +5
- Eğer Benzerlik 0.55 aşağısı ise, -10

Eğer Adres kolonları boş ise, model bu benzerlik aşamasına girmez, dolu ise benzerlik puanlarına göre yukarıdaki puanları alır. Benzerlik 0.55 ve üzeri ise puan alır, yani ekleme işleme yapılır. Eğer 0.55'ten küçükse adreslerin farklı olacağı, bu sebeple otellerin de farklı oteller olduğu öngörülerek 10 puan çıkarılır.

5.6 Konum Mesafesi

Veri setindeki konumların doğrulanması sonucunda bu aşama iki kısımda incelenmiştir. Konum doğrulanmış ise ve iki otel arasında 100 metreden az mesafe varsa 30; 1 kilometreden az mesafe varsa 15; 4 kilometreden az mesafe varsa 10 puan eklenmiştir.

Eğer konum doğrulanmamış ise ve iki otel arasında 500 metreden az mesafe varsa 25; 10 kilometreden az mesafe varsa 15; 20 kilometreden az mesafe varsa 10 puan eklenmiştir.

Yukarıdaki puanlar sonucunda bir eşik değeri belirlenmiştir. Toplanan puanların, eşik değerini aşıp aşmamasına göre, eş otel ya da farklı otel etiketi (Global Otel No) atanmıştır. Yani eşik değerini aşan karşılaştırmaların eş oteller olduğu saptanmıştır ve aynı global otel numarası atanmıştır. Eğer eşik değeri aşılmadıysa farklı otel etiketi atanmıştır.

Modelin çalışma mantığı şu şekildedir; Arama havuzu ve Atama havuzu olarak iki veri seti vardır. Atama havuzu, global otel numarası boş olan yani henüz atanmamış olan veri setidir. Arama havuzu ise global otel numarası atanmış veri setidir. Atama havuzundaki her otel arama havuzundaki, ilgili bölgedeki (örneğin Antalya) otelleri arar. Yukarıdaki model aşamalarını ve eşik değerini en yüksek puanla geçen otelin global otel numarası, aranan otel numarasına verilerek arama havuzuna kaydolur. Eğer otel, yukarıdaki puanları alarak eşik değerini aşamadıysa, eşi olmayan bir oteldir ve benzersiz bir numara alarak arama havuzuna dahil edilir. Bu işlemler Atama havuzu boş kalana kadar devam eder.

6. Sonuç ve Gelecekteki Çalışmalar

Modelin, tüm veri setindeki oteller için çalıştırılması sonucu Global Otel No değerleri atanmıştır. Aşağıdaki şekilde gösterilmiştir.

Çizelge-8: Global Otel No Ataması Sonuçları

Otel No	Otel İsmi	OTA	Bölge	Alt Bölge	Bölge Detay	Enlem	Boylam	Konum Doğru	Google İsmi	Global Otel No
ets-58536	Korsan Ada Hotel	ets	Antalya	Kaş		36.194301	29.594155	1	Korsanada hotel	4dm96356-0fba-11eb
tatilsepeti-5013	Skalion Hotel Spa	tatilsepeti	İstanbul	Fatih		41.0025228	28.964836	1	Skalion Hotel & Spa	316ukdj7-0fba-11eb
tatilbudur-3136573	Skalion & Spa	tatilbudur	İstanbul	Fatih		41.0025228	28.964836	1	Skalion Hotel & Spa	316ukdj7-0fba-11eb

Çizelge-8'de görüldüğü üzere, farklı OTA'larda bulunan otel isimleri tek bir global numara ile tekilleştirilmiştir.

Beş aşamadan oluşan modelde, puanlar ve eşik değerleri değiştirilerek çeşitli denemeler yapılmıştır. Test veri setindeki Doğruluk hata metriği değerine göre, aşağıda denemelerin sonuçları verilmiştir.

Çizelge-9: Puan- Eşik Değeri Denemeleri

Puanlama Sistemi	Eşik Değeri	Doğruluk Metriği (Accuracy)		
		Kolay Test Veri Seti	Zor Test Veri Seti	Karışık Test Veri Seti
Rasgele Puanlar 1	80	0.66	0.60	0.67
Rasgele Puanlar 2	80	0.61	0.55	0.64
Rasgele Puanlar 3	80	0.64	0.62	0.63
Rasgele Puanlar 4	80	0.64	0.60	0.61
Rasgele Puanlar 1	75	0.67	0.62	0.69
Rasgele Puanlar 2	75	0.65	0.60	0.64
Rasgele Puanlar 3	75	0.64	0.59	0.61
Rasgele Puanlar 4	75	0.64	0.59	0.63
Rasgele Puanlar 1	100	0.87	0.82	0.88
Rasgele Puanlar 2	100	0.85	0.81	0.84
Rasgele Puanlar 3	100	0.88	0.82	0.85
Rasgele Puanlar 4	100	0.86	0.79	0.83
Rasgele Puanlar 1	95	0.89	0.82	0.90
Rasgele Puanlar 2	95	0.91	0.82	0.85
Rasgele Puanlar 3	95	0.84	0.82	0.82
Rasgele Puanlar 4	95	0.88	0.81	0.83
Rasgele Puanlar 1	**90**	**0.94**	**0.83**	**0.87**
Rasgele Puanlar 2	90	0.91	0.82	0.85
Rasgele Puanlar 3	90	0.89	0.83	0.86
Rasgele Puanlar 4	90	0.94	0.82	0.85

En iyi puanlar model aşamalarında verilen (Rasgele Puanlar 1) puanlardır. Çizelge-9'da yeşil renk ile belirtilen eşik değeri ve puanlar kombinasyonu en iyi sonucu vermiştir.

Otellerin; isim, adres ve konum bilgilerini kullanarak oluşturulan tekilleştirme modelinin test verisi üzerindeki sonuçları başarılı bulunmuştur. Bu çalışmanın sonucunda tekilleştirilen otel isimleri; OTA bazlı fiyat karşılaştırma, rakip analizi, dinamik fiyatlandırma ve benzeri çalışmalar için bir başlangıç niteliği taşır.

Çeşitli OTA'lardan alınan otel bilgilerinin içerisinde, bu modelde kullanmadığımız otelin fotoğrafı, oda bilgisi ve fiyat bilgisi de vardır. İlerleyen çalışmalarda oda isimleri de tekilleştirilecek ve bir oda fiyatının tüm OTA'lardaki fiyatı karşılaştırılabilecektir.

Bu modeli daha da iyileştirmek için, bir aşama daha eklenebilir. Otellerin son olarak da fotoğrafları arasında görüntü benzerliği algoritmaları kullanılarak görüntülere bir benzerlik oranı atanabilir. Bu orana bir puan verilir ve eşik değeri güncellenebilir.

7. Teşekkür

Bu çalışma, TÜBİTAK tarafından desteklenen TEYDEB 3192318 numaralı, Yapay Zekâ ile Çoklu Tedarikçi Yapılarında, Otellerin ve Oda Tiplerinin Eşleştirilmesi projesinden oluşturulmuştur. Desteği için TÜBİTAK'a teşekkür ederiz.

Kaynakça

[1] S. Kemp, «we are social - Dıgıtal 2021: The Latest Insıghts Into The 'State Of Dıgıtal,» January, New York, 2021.

[2] S. Karacan ve S. Çiftçioğlu, «TURİZM İŞLETMELERİNDE ELEKTRONİK TİCARETİN ETKİSİ,» *Uluslararası Turizm, İşletme, Ekonomi Dergisi*, pp. 245-252, 2018.

[3] Türkiye Otelciler Birliği, «Türkiye Otelciler Birliği,» 2 March 2021. [Çevrimiçi]. Available: http://www.turob.com/tr/istatistikler/hotrec-2020-online-dagitim-kanallari-calismasi.

[4] M. Pullman ve S. Rodgers, «Capacity management for hospitality and tourism: A review of current approaches,» *International Journal of Hospitality Management* , cilt 29(1), p. 177–187, 2010.

[5] S. Niwattanakul, J. Singthongchai, E. Naenudorn ve S. Wanapu, «Using of Jaccard Coefficient for Keywords,» *Proceedings of the International MultiConference of Engineers and Computer Scientists*, cilt 1, 2013.

[6] J. M. Duarte, J. B. d. Santos ve L. C. Melo, «Comparıson Of Sımılarıty Coefficıents Based On Rapd,» *Genetics and Molecular Biology*, cilt 22, no. 3, pp. 427-432, 1999.

[7] X. Jin, S. Zhang, J. Liu ve H. Guan, «Research on Similarity Detection of Massive Text based on Semantic Fingerprint,» *Proceedings of Science*, 2017.

[8] G. Chen, G. Chen, D. Wu, Q. Liu, L. Zhang ve X. Fan, «An improved Simhash algorithm based malicious mirror website detection method,» *Journal of Physics: Conference Series*, 2021.

[9] M.K.Vijaymeena ve K.Kavitha, «A Survey On Sımılarıty Measures In Text Mınıng,» *Machine Learning and Applications: An International Journal (MLAIJ)*, cilt 3, no. 1, pp. 19-28, 2016.

[10] M. Mansoor, M. Shaheen ve Z. U. R. , «Deep Learning Based Semantic Similarity Detection Using Text Data,» *Information Technology and Control*, cilt 4, no. 49, pp. 495-510, 2020.

[11] A. T. Bayrak, E. E. Özbek ve S. Kestepe, «Aynı Oteli Temsil Eden Farklı Kayıtlar için Akıllı Eşleştirme,» *researchgate.*

[12] R. Editör, «Regna,» 29 March 2017. [Çevrimiçi]. Available: https://www.regna.com.tr/veri-tekillestirme-nedir. [Erişildi: 18 September 2021].

[13] R. T. Ionescu ve M. Popescu, Knowledge Transfer between Computer Vision and Text Mining, Switzerland: Springer, 2016.

[14] «wikipedia.org,» [Çevrimiçi]. Available: https://en.wikipedia.org/wiki/Levenshtein_distance. [Erişildi: 16 09 2021].

[15] H. Khudeer ve H. Erbay, «Hibrit Karga-Genetik Algoritmasını Kullanarak 3 Boyutlu Kutu Paketleme Problemi Çözme,» *Veri Bilimi Dergisi*, cilt 4, no. 1, pp. 8-22, 2021.

[16] P. Christen, «A Comparison of Personal Name Matching: Techniques and Practical Issues,» 2006.

[17] K. M. M. Aung ve A. N. Htwe, «Comparison of Levenshtein Distance Algorithm and Needleman-Wunsch Distance Algorithm for String Matching,» *National Journal of Parallel and Soft Computing*, cilt 1, no. 1, pp. 209-213, 2019.

[18] A. Ansari, «bioinfoguide.com,» January 2018. [Çevrimiçi]. Available: https://bioinfoguide.com/index.php/algorithms-and-methods/11-smith-waterman-algorithm. [Erişildi: 10 September 2021].

[19] F. Öztemiz ve A. Karcı, «Akademik Yazarların Yayınları Arasındaki İlişkinin Sosyal Ağ Benzerlik Yöntemleri İle Tespit Edilmesi,» *Uludağ Üniversitesi Mühendislik Fakültesi Dergisi*, cilt 25, no. 1, pp. 591-608, 2020.

[20] B. K. Ülkü, «Panoramik Radyografi Görüntülerinde Maksiller Ve Mandibular Yapıların Sınırlarının Belirlenmesi için Yarı Gözetimli bir Metot,» *TOBB Ekonomi ve Teknoloji Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi*, 2019.

[21] J. Han, M. Kamber ve J. Pei, Data Mining Concepts and Techniques, USA: Morgan Kaufmann, 2012.

[22] S. R. Hashemi, S. S. M. Salehi ve D. Erdogmus, «Tversky as a Loss Function for Highly Unbalanced Image Segmentation using 3D Fully Convolutional Deep Networks,» 2018.

[23] I. Zarembo, A. Teilans, A. Rausis ve J. Buls, «Assessment of Name Based Algorithms for Land Administration Ontology Matching,» *Procedia Computer Science*, cilt 43, pp. 53-61, 2015.

[24] G. Recchia ve M. Louwerse, «A Comparison of String Similarity Measures for Toponym Matching,» *ResearchGate*, 2013.

[25] I. Ilyankou, «Comparison of Jaro-Winkler and Ratcliff/Obershelp algorithms in spell check,» *IB Extended Essay Computer Science*, 2014.

[26] Olufade, F. W. Onifade, M. IAENG ve O. Thiéry, «Dynamic Fuzzy String-Matching Model for Information Retrieval Based on Incongruous User Queries,» *Proceedings of the World Congress on Engineering*, cilt 1, 2010.

[27] «developers.google.com,» google, [Çevrimiçi]. Available: https://developers.google.com/maps/documentation/places/web-service/overview. [Erişildi: 12 September 2021].

[28] «geopy,» pypi.org, [Çevrimiçi]. Available: https://pypi.org/project/geopy/. [Erişildi: 21 July 2021].