

# Makine Öğrenmesi ile Adaptif Otel Öneri Sistemi

Ahmet Sayar<sup>1</sup>, Nursultan Turdaliev<sup>2</sup>

<sup>1</sup>Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Kocaeli Üniversitesi, Kocaeli, Türkiye

<sup>2</sup>Bilgisayar Mühendisliği ABD, Fen Bilimleri Enstitüsü, Kocaeli Üniversitesi, Kocaeli, Türkiye

<sup>1</sup>ahmet.sayar@kocaeli.edu.tr, <sup>2</sup>nursultan.turdaliev@aboutyou.de

**Özet.** Otel rezervasyon yöntemleri gelişen teknoloji ile birlikte sürekli yenilenmektedir. Rezervasyonda en önemli problem kullanıcıların en az para ile en çok memnuniyet verebilecek otellerin tespitidir. Bu tür problemlerin çözümünde son yıllarda gelişen makine öğrenmesi tekniklerinin kullanıldığı görülmektedir. Ancak mevcut sistemler, statik yapıda çalışmakta ve otelleri belirli aralıklarda puanlamaktadırlar. Rezervasyonlarda bu ölçüt anahtar rol oynamaktadır. Ancak, müşteriler oteli seçip rezervasyon yaptıktan sonra, eğer diğer otellerdeki fiyatlar ve diğer şartlar değişirse müşteriler fırsatları kaçırmaktadırlar ve dolayısıyla bu durum müşterilerde memnuniyet kaybına neden olmaktadır.

Bu çalışmada, müşterilerin memnuniyet / fiyat oranını yükseltmeye odaklı dinamik ve makine öğrenmesi tabanlı otel öneri sistemi geliştirilmiştir. Önerilen sistemde, müşteri rezervasyonunu yapmış olsa dahi, tatilin fiilen başlamasına kadar, daha iyi fırsatlar olduğunda müşteriye öneriler sunacak ve dolaylı olarak müşteri memnuniyetini ve otel doluluk oranlarını artırıcı bir etki yaratacaktır. Sistemin işleyişi makinenin eğitilmesine dayalıdır. Bunun için, çeşitli özelliklerine göre birbirine üstünlükleri bilinen oteller ile sistem eğitilmiştir. Daha sonra, sistemde olmayan bir otelin diğer otellerle otomatik olarak karşılaştırılıp daha iyi fırsatın olup oluşmadığı otonom ve dinamik olarak tespit edilmiştir. Karşılaştırma ve sınıflandırma işlemleri Destek Vektör Makinaları (SVM) algoritması ile gerçekleştirilmiştir. Eğitim verileri yaklaşık 680 satırdan oluşmuştur. Bu veriler, 1154983 otel üzerinden rastgele seçilen ikili eşleştirmelerin seyahat acentelerine gönderilip, karşılaştırma ve sonuç bilgilerinin manuel girişi ile elde edilmiştir. Sistemin başarımı, gerçek otel verileri üzerinde test edilmiş ve 0.85 doğruluğunda makine öğrenme modeli geliştirilmiştir.

**Anahtar Kelimeler:** Öneri Sistemleri, Otel Sıralaması, Makine Öğrenmesi Sıralaması, Denetimli Öğrenme, Sınıflandırma

# Adaptive Hotel Recommender System with Machine Learning

Ahmet Sayar<sup>1</sup>, Nursultan Turdaliev<sup>2</sup>

<sup>1</sup>Computer Engineering Department, Engineering Faculty, Kocaeli University, Kocaeli, Turkey

<sup>2</sup>Computer Engineering Department, Graduate School Of Natural and Applied Sciences, Kocaeli University, Kocaeli, Turkey

<sup>1</sup>ahmet.sayar@kocaeli.edu.tr, <sup>2</sup>nursultan.turdaliev@aboutyou.de

**Abstract.** Hotel reservation methods are being upgraded with development of a technology. The most important problem in hotel reservation is proving that customers with least money get the most qualifying hotels. Machine learning methods are being used to solve this problem in recent years. However, the existing systems are working in static structure and scoring hotels in a limited interval. This criterion plays a key role in reservations. However, once customers select and book a hotel, if prices and other conditions change in other hotels, they are missing out on opportunities and this causes customer dissatisfaction.

In this study, a dynamic and machine learning based hotel recommendation system focused on raising the customer satisfaction / price ratio has been developed. In the proposed system, even if the customer makes a reservation, he/she will get new suggestions if there are better opportunities, and this will indirectly increase customer satisfaction and occupancy rates. The operation of the system is dependent on the training dataset of the machine learning model. Then system was autonomously and dynamically tested whether there is a better opportunity or not with an external hotel successfully. Comparison and classification operations was made with. Comparison and classification operations were performed with Support Vector Machines (SVM) algorithm. Training dataset contains approximately 680 rows. This dataset was obtained by sending random two hotels from 1154983 hotels to travel agents to compare them manually. The performance of the system has been tested on real hotel data and a machine learning model has been developed with an accuracy of 0.85.

**Keywords:** Recommender Systems, Hotel Ranking, Machine Learned Ranking, Supervised Learning, Classification

## 1 Giriş

Önceden insanlar otel rezervasyon yapma işini telefon üzerinde veya seyahat acentesi yardımıyla, otelin durumu hakkında güncel bilgisi olmadan, önceden kalan müşterilerin otel hakkında düşüncelerini bilmeden, piyasada olan otellerin fiyata en uygun, kaldığı yerin müşterinin gezmek istediği yerlere göre en uygun yer olup olmadığını bilmeden rezervasyon yaparlardı. Günümüzde var olan online rezervasyon sistemleri bu olanakların hemen hemen hepsini sunmaktadır. Fakat müşteri sadece rezervasyon yapma zamanındaki en iyi ve uygun oteli rezervasyon yapabilmektedir. Rezervasyon yaptıktan sonraki yeni fırsatları kaçırmaktadırlar.

Günümüzde oteller, oda fiyatlarını, doluluk ve kârlılığını sürekli optimize eden modern sistemleri kullanıyorlar. Yapılan analize göre, otelin sekizi konuk gelmeden, son otuz gün içinde, otel fiyatlarını birçok kez değiştiriyor. Sonuç olarak, gezginler için iyi fiyata en iyi otelin rezervasyon yapılması zorlaşıyor.

Otel rezervasyon yapma işi zorlaştığı için, gezginler de davranışını buna göre değiştirmiştir. Gfk'nın Google ile birlikte yaptığı araştırmasına göre, Almanyalı gezginler bir otel teklifini seçmeden önce 24 gün sürekli internette geçirir. Buna rağmen, gezginlerin yüzde sekseni otel odalarına çok fazla öderler.

Trivago gibi portaller bu problemi çözememişlerdir. Aslında, onlar arama zamanındaki en iyi fiyatları gösterir. Bu çalışmada geliştirilen platform diğer taraftan, piyasadaki değişen fiyatları izler. Eğer rezervasyon yaptıktan sonra otel fiyatı düşerse, önceki yapılan rezervasyonu iptal ederek, yeni rezervasyon yapar. Sonuç olarak, geliştirilen platform müşterilere geleneksel fiyat karşılaştırma portallerinden on kez daha tasarruflu harcamayı sunar.

Fiyatlar düştüğünde aynı odayı yeniden rezervasyon yapmak kolaydır. Platform eğer aynı veya düşük fiyata daha iyi otel rezervasyon yapabilme seçeneği varsa, otel değiştirme seçeneğini sunar. Bir otelin başka otelden iyi veya kötü olduğunu nasıl öğreniriz? Otel kalitesi ölçek üzerinde çalışmayan subjektif bir alandır. Bu problemi çözmemiz için otelleri otomatik öneri yapabilecek otel karşılaştırma sistemine ihtiyacımız vardır.

Üstelik, otomatik otel öneri sisteminin doğruluğunu gün geçtikçe arttıran sistemi geliştirmek isterdik. Aşamalı doğruluk iyileştirme yapabilen otomatik otel öneri sistemi geliştirebilmek için, sistemi makine öğrenme teknikleriyle geliştirerek sürekli doğru verilerle eğitmek gerekir.

Bu çalışmada var olan sistemlerden değişik çözüm sunmakla kullanıcıların memnuniyetini arttırmak amacıyla otomatik otel rezervasyon yükseltici sistemi geliştirmeye çalışılmıştır. Genelde iki durumda sistem otomatik otel yükseltme işini yapar. Birinci durum, eğer aynı fiyata daha iyi otel bulunduğu kullanıcıya iletişim kurmadan oteli yükseltir. İkinci durum, ise aynı kategorideki otel daha düşük fiyata bulununca, rezervasyon yapılmış otel iptal edilerek yeni otel rezervasyon yapılır. Ayrıca geri kalan para tamamıyla müşteriye geri ödeme yapılır. Başka bir deyişle otel rezervasyon tecrübesini daha bir yüksek seviyeye çıkaracaktır.

Makine öğrenme alanındaki sıralama, karşılaştırma problemleri üzerinde yapılan çalışmaları üzerinden araştırma yaptıktan sonra, gelişmelerin ve uygulamanın hep 'bilgi alma' alanında oldu öğrenilir. 'Bilgi alma' alanında sıralama veya karşılaştırma

her zaman ‘sorguyla’ benzerlik oranına göre sıralama yapılmıştır. Otelleri karşılaştırma için ‘bilgi alma’ alanındaki gelişmeler uygulanamaz. Hazır veya temel çözümlerin olmaması yeni bulguların, çözüm yöntemlerinin geliştirilmesini gerektirir. Çalışma aşağıdaki bölümlerden oluşmaktadır.

Çalışmanın diğer kısımlarında ilkönce ilgili çalışmalar ve bu alandaki son gelişmeler hakkında bilgi verilecektir. Sonraki kısımda sistem tasarımı ve geliştirilen uygulama iş akışı ve sözde kodlarla birlikte detaylı anlatılacaktır. Bir sonraki kısımda ise elde edilen sonuçlar ve gelecek çalışmalar sunulacaktır.

## 2 İlgili Çalışmalar

Otel öneri sistemleri hemen hemen tamamı otel puanlama veya ‘bilgi alma’ alanı ile ilişkilidir ve dolayısıyla araştırmaların çoğu otel sıralama mantığına dayanmaktadır. Bu odaklı çalışmalar aşağıda özetlenmiştir.

Expedia online otel rezervasyon yapma platformu da otel işinde ayrıcalık olmak, kullanıcıların memnuniyetini arttırmak amacıyla birkaç araştırmada bulunmuştur [1]. Online seyahat ajanslarının tek amacı vardır: siteye gelen her misafir kullanıcının otel rezervasyon yapma olasılığını maksimuma çıkarmak. Bu problemi kullanıcının aradığı kriterleri, tarihsel verileri kullanarak var olan otelleri satın alma olasılığına göre sıralamayla çözmek mümkündür. Nishith veya başkaları bu problemi Naive Bayes, Lojistik regresyon, Random Forest, Ranking SVM algoritmalarını kullanarak en iyi doğruluğu elde etmeye çalışmışlardır [1].

Walek ve arkadaşları [9] “Otel Rezervasyon Sistemine Uzman Sistemi Önerisi” çalışmasında otel rezervasyon sistemlerinin otel servislerinden başka otel etrafındaki veya şehirdeki etkinlik, aktivite veya ilgi noktalarıyla bağ kurmadığını söyleyerek, bu boşluğu dolduracak çözüm üzerinden çalışmışlardır. Aslında bu ilginç ve yaygın kullanılabilir özelliklerdir. Çünkü müşterilerin çoğunluğu otel rezervasyon yaparken etrafında ilgi noktalarının, tarihi noktaların, gezi yerlerinin oldukça çok olmasını ister. Maalesef mevcut olan otel veri setinde otel etrafındaki etkinlik, aktivite noktalarının olmadığı için ve bunları toplamak daha çok iş gücünün talep edeceği düşünüldüğü için, çalışma bu fırsatı gelecekte kullanarak modelin daha iyileştirebilecek nokta olarak bırakmıştır.

Otel sıralamasını makine öğrenmesinin geleneksel algoritmalarıyla çözmeye çalışmak biraz yanlış olacağını, geleneksel algoritmalar sonucu verinin belirli bir parçasıyla elde edebileceğini, sıralama probleminde ise verinin tamamının sonuca etkisi olduğunu Tiziano Papini “SortNet: Sıralamayı sinirsel tercih fonksiyonu ile öğrenmeye çalışmak” adlı çalışmasında iyice kanıtlamaktadır [10]. Bu çalışma kullanılan yöntemin daha karmaşık modelini kullanmaktadır. Çalışmada ise karşılaştırma verilerini geleneksel makine öğrenme yöntemi olan sınıflandırma probleminde dönüştürerek, standart sınıflandırma algoritmalarıyla çözmeye çalışılmıştır.

Nishith Khantal ve arkadaşları [3] Expedia.com platformu için satın almayı yükseltmek amacıyla makine öğrenmesi yoluyla otel sıralama sistemini geliştirmiştir. Ryosuke Saga ve arkadaşları [4] ise Tokyodaki müşterilerin geçmişteki rezervasyonlarını kullanarak, model tabanlı yöntemle müşterilerin otel geçiş grafiğini

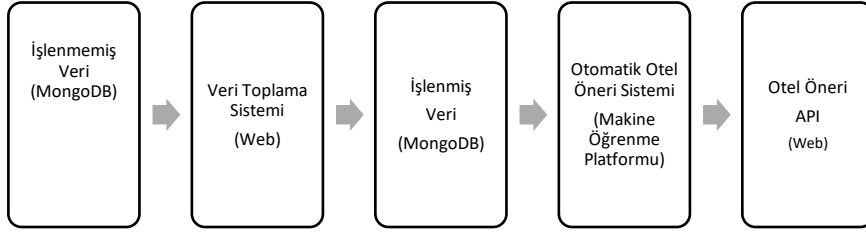
kurarak otel öneri problemini çözmeye çalışmıştır. Michael Arruza ve arkadaşları [6] Expedia şirketinin Kaggle websitesinde yaptığı otelleri kullanıcıların sorgularına göre en uygun sıralama yarışmasının sonucunda elde ettiği ortalama doğruluğu 0.02260 olan sonucu 0.30 oranında yükseltmişlerdir. Anindya ve arkadaşları [7][18] otel sıralama sonucunu arama sorgusuna göre en uygun değil, müşteriye göre en uygun şekilde sıralama yöntemini geliştirmişlerdir. Otel sıralama sonucu bir iş seyahatçısı için başka, turistler için başka olması daha etkili olduğunu kanıtlamışlardır. İş seyahatçısı fiyata en az duyarlı, turistler ise en çok duyarlı geldiği öğrenilmiştir. İş seyahatçısı için otelin toplu taşımaya kolay erişimli, otoyola yakın olması çok önemlidir. Tersine, turistler için sahile yakın olması, hizmet seviyesinin en yüksek olması önemlidir. Aynı otel sıralama sonucunu değişik ülkeler için değişik sırada göstermemiz daha etkili olabileceğini Elena Marchiori ve arkadaşları [8] kanıtlamışlardır. Araştırmanın amacı otel sıralamasını daha iyi yapmak değil, aynı ülkeden gelen turistlerin otel değerlendirme yapmasında uyumlu yorumlar, değerlendirmeler yaptığını öğrenmektir. Bu sonuçları kullanarak potansiyel müşterinin geldiği ülkeye göre kişiselleştirmeyi daha bir adım iyileştirmek mümkündür. Otel sıralama, puanlama, karşılaştırma işleri yapmak için en önemli gereksinimlerden biri de otelin hangi özellikleri daha değerli olduğunu bilmemiz gerekir. Bu bilgi veri üzerinde ön işleme, dönüştürme ve seçim yapma işlerinde çok faydası vardır. Bazı özelliklerin veya olanakların hiç faydası olmadığını önceden bilmemiz veri seçiminde daha doğru karar vermemizi sağlar. Bu alanda araştırma işleri hangi aşamada, nasıl gelişmeler olduğu hakkındaki bilgileri Sara Dolnicar ve T.Otter [9] “Which Hotel attributes Matter? A review of previous and a framework for future research” çalışmasında detaylı bilgiler sunmuştur. Sonuç olarak otel özellikleri önemi geniş ve son derece heterojen araştırma alanı olduğu öğrenilmiştir. AltexSoft’un [10] veri bilimi blogunda yayımlandığı makalede ise makine öğrenme tekniklerinin otel ve seyahat servisinde nasıl kullanıldığı ve geliştirirken ne tür zorluklarla karşılaştığı, değişik şirketlerin otel rezervasyon işlerini kişiselleştirmeyi nasıl uyguladığı hakkında detaylı bilgi verilir. Sonucu kişiselleştirmede özel verilerin etkisi çok olduğu için ilk önce müşterilerin kişisel verileri paylaşma isteği olması ve izin vermesi gerektiğinden, bazen yasal problemler üzerinden çalışılması gerektiğinden bahseder. Demek, müşteri verisi üzerinden kişiselleştirilmiş önerilerde bulunmak, kişisel sıralama uygulamak için yasal sorunları da göz önünde bulundurmamız lazım.

Yukarıda bahsedilen otel puanlama ve sıralama tabanlı otel öneri sistemleri sürekli ve dinamik olmadığından, önerilen çözümler genelde rezervasyona kadar kullanıldığı için müşterilerin memnuniyetini yeterince arttıramamaktadır. Geliştirilen sistemler çoğu zaman statik olarak kalmaktadır. Çalışmanın sonucunda geliştirilen sistem bu çalışmalardan dinamik ve adaptif olmasıyla, rezervasyondan sonraki fırsatları müşterilere sunma özelliğiyle müşteri memnuniyetini, otel rezervasyon yapma işini yeni seviyeye yükseltmektedir.

### 3 Makine Öğrenme ile Adaptif Otel Öneri Sistemi

Sistem iki temel alt sistemden oluşmaktadır. Birincisi, sistem makine öğrenmesi için temel gereksinimlerden olan veriyi toplamak için tasarlanmıştır. Veri toplama işini otel rezervasyon işinde uzman seyahat acenteleri manuel olarak yapmaktadır. Veri toplama işi sadece iki oteli karşılaştırma ve sonucu veritabanına saklama işinden ibarettir. Ayrıca birinci sistem toplanmış veriyi ikinci sisteme aktarmada veri ön işleme, dönüştürme işlerinin yüzde seksenini uygulamaktadır. İkinci uygulama ise birinci uygulamanın yardımıyla toplanmış verileri makine öğrenme yöntemini kullanarak otomatik otel öneri sistemini geliştirmiştir. Veri toplama işini bir web platformu üzerinden, makine öğrenme problemi ise web platforma bağımsız olduğu için iki ayrı uygulama olarak geliştirilmiştir.

Sistemin akış diyagramı **Şekil 1**'de gösterilmiştir.



**Şekil 1.** Sistem akış diyagramı

Görüldüğü gibi sistem verinin üç ayrı sistem arasında akıştan ibarettir. İlk önce otel hakkında işlenmemiş veri MongoDB veritabanında saklanmaktadır. Sonra veri toplama sistemi işlenmemiş veriyi seyahat acentelerinin yardımıyla işlenmiş veriye dönüştürmektedir ve geri MongoDB veritabanına ayrı koleksiyona saklamaktadır. Otomatik otel puanlama sistemi ise işlenmiş veri üzerinden makine öğrenme yoluyla elde edilmiş eğitilmiş modeli hazırlamaktadır. Sonunda ise eğitilmiş model otel karşılaştırma ve öneri işini REST API olarak başka sistemlere veya halka sunmaktadır.

#### 3.1 Verinin Ön Hazırlığı

Makine öğrenme algoritmaları veriden öğrenir. Çözülmesi gereken problemi çözme için doğru veri ile algoritmayı besleme çok önemlidir. İyi veri olsa bile, yararlı bir ölçekte, formatta ve anlamlı özelliklerin dahil olması gerekir. İyi ve anlamlı veri formatını elde etmek için aşağıdaki adımlar toplanan veri üzerine uygulanmıştır:

1. Veri seçmesi
2. Veri ön işleme
3. Veri dönüştürümü

**Veri Seçmesi.** Bu adımda var olan tüm veri kümesinden belirli bir alt kümeyi seçme işi yapılır. Genelde “çok daha iyi” felsefesiyle var olan tüm veri setini seçme isteği vardır. Bu doğru olmayabilir. Hangi verinin gerçekten çözülmesi gereken probleme etkisi olduğunu bilmek lazım. Önemli olan veriler hakkında varsayımlar üzerinde ve sonradan varsayımı doğrulayabilecek şekilde veri seçimi yapılabilir.

Adaptif otel öneri problemi için otel adresi, adı önemli olmadığı varsayılabilir. Çünkü bir otelin adının Motel One veya İbis otel olması hiç önemli değildir. Fiyat, olanakların (wifi, ütüleme servisi, garaj) belirli derecede etkisi olduğu varsayılabilir.

Otelin aşağıdaki özellikleri seçilmiştir:

1. Yıldız sayısı
2. Açıklama metni
3. TripAdvisor derecelendirme sayısı
4. TripAdvisor derecelendirmesi
5. Oda olanakları (klima, televizyon, saç kurutma makinesi)
6. Genel olanaklar (ücretsiz WIFI, otopark, kapalı havuz)
7. Resimler

**Veri ön işleme.** Üç yaygın veri ön işleme adımı vardır: biçimlendirme, temizleme ve örnekleme. Tüm ön işleme adımları otomatik otel öneri verisi üzerinde uygulamaya çalışılmıştır.

Bildiğimiz gibi veri üzerinde temizleme ve örnekleme adımlarını yapmadan önce biçimlendirme işi yapılması gerekir. Otomatik otel puanlama sisteminin verileri MongoDB veri tabanında saklandığı için, biçimlendirme adımı kaçınılmazdır. Uyguladığımız makine öğrenme algoritması belge tabanlı veritabanı üzerinde uygulanamaz. Veri biçimlendirme işi PHP programlama dilinde yapılmıştır. Biçimlenen veri ise bildiğimiz CSV dosya formatında dışa aktarılmıştır. Veri görünüşü kısaca aşağıdaki gibidir.

**Tablo 1.** Biçimlendirilmiş veri görünüşü.

| Yıldız Sayısı | Açıklama                     | Resimler sayısı | TripAdvisor Değerlendirmesi | TripAdvisor Değerlendirme Sayısı | Teras | Özel Plaj | Veya başka olanaklar listesi |
|---------------|------------------------------|-----------------|-----------------------------|----------------------------------|-------|-----------|------------------------------|
| 4             | Make yourself at home<br>... | 43              | 8.8                         | 1230                             | 1     | 0         | 0 veya 1                     |
| 5             | Make yourself at home<br>... | 8               | 9.0                         | 5000                             | 1     | 1         | 1 veya 0                     |

Yukarıdaki Tablo 1’de gördüğümüz veri biçimlendirmesi makine öğrenmesi algoritmalarının uygulanmasına uygundur. Tabloya verinin tüm nitelikler sığmadığı için burada gösterilmemiştir. Gösterilmeyen nitelikler son kolonda var olup

olmadıklarına göre 0 veya 1 ile gösterildikleri ifade edilmiştir. Niteliklerin tüm sayısı 162 dir. Gösterilmeyen bu niteliklerden bazıları, tüm otel ve oda olanakları (wifi, kapalı havuz, yıkama servisi, garaj vb.) olarak açıklanabilir.

Veri ön işleminin ikinci adımı ise temizleme işidir. Otomatik otel öneri sisteminde veri seçme işini yaparken gereksiz diye düşünülen nitelikler atıldığı için ve verinin tutarlılık oranı, ticari veri seti olduğu için çok yüksektir. Hemen hemen yüzde yüz tutarlı olduğu için temizleme işine çok da ihtiyaç duyulmamıştır.

Son adım olarak da örnekleme işidir. Ticari veriler genelde çok miktarda olur. Genelde tüm veri seti üzerinde makine öğrenme algoritmasını uygulamak biraz yanlış olur. Çünkü algoritma çalışma zamanı, bellek ve hesaplama gereksinimi de çok yüksek olması gerekir. MongoDB veritabanında bulunan veri setimizdeki otel sayısı 1154983 tür. Toplam veri boyutu ise 4716MB'tır. Bir önceki bölümde veri toplama işinde, seyahat acentelerinin otel karşılaştırma yaptığına milyon otelin tamamından iki rastgele oteli karşılaştırmayı sormak yerine, belirli sayıdaki şehir içindeki oteller manuel olarak seçilmiştir. Otel türünün, sunduğu olanakların da çok çeşitli ve tüm veri setini temsil edebilecek şekilde hazırlanmıştır. Örnekleme işinin uygulanmasıyla, prototipleme ve uygun çözüm üzerinde çalışma işinde kolaylık sağlanmıştır.

**Veri Dönüştürümü.** Son adım ise veri dönüştürüm işidir. Kullanılan algoritma, problem alanı bilgisi bu adımı etkiler. Genelde birden çok veri dönüştürüm işleri yapılması gerekebilir. Üç genel veri dönüştürüm yöntemleri vardır: ölçekleme, öznelik ayrıştırma, birleştirme. Bu adım özellik mühendisliği olarak da bilinir.

İlk olarak ölçekleme işinin nasıl uygulandığını anlatalım. Orijinal veride resimler URL'lerin dizisi formatında, açıklama ise uzun metin, TripAdvisor değerlendirmesi 0 ve 10 arasında, değerlendirme sayısı ise tamsayı formatında, olanaklar ise olanakların dizisi formatındadır.

**Tablo 2.** Veri özellikleri üzerinde ölçekleme işleri.

| Özellik Adı                      | Orijinal Format      | Yeni Format   | Yeni Format Açıklama   |
|----------------------------------|----------------------|---------------|--|
| Yıldız Sayısı                    | Tamsayı [0-5]        | Rasyonel Sayı | 0 ve 1 arasındaki rasyonel sayı                                      |
| TripAdvisor Değerlendirmesi      | Tamsayı [0-10]       | Rasyonel Sayı | 0 ve 1 arasındaki rasyonel sayı                                      |
| TripAdvisor Değerlendirme Sayısı | Tamsayı [0-10000]    | Rasyonel Sayı | 0 ve 1 arasındaki rasyonel sayı                                      |
| Açıklama                         | Uzun metin           | Rasyonel Sayı | Metin uzunluğu 0 ve 1 arasındaki rasyonel sayı                       |
| Resimler                         | URL dizisi           | Rasyonel Sayı | Resimler sayısı 0 ve 1 arasındaki rasyonel sayısına dönüştürülmüştür |
| Oda olanakları                   | Olanaklar adı dizisi | Boole         | Her olanak ayrı bir özelliktir. Olanak var ise 1, yoksa 0            |
| Otel olanakları                  | Olanaklar adı dizisi | Boole         | Her olanak ayrı bir özelliktir. Olanak var ise 1 yoksa 0.            |



Tablo 1'e bakıldığında ise bu problemlerin çoğuna birleştirme, ayrıştırma veya ölçekleme işi ön işleme adımında birlikte yapıldığı anlaşılır. Örneğin, URL'lerin dizisi formatındaki resimler üzerinde birleştirme yapılarak, resim sayısı adında yeni özellik eklenmiştir. Yine de veriler tamamen aynı formatta değildir. Aynı formata dönüştürmek için tüm sayısal özellikler 0 ve 1 arasındaki değere ölçeklendirilmiştir. Tablo 2'de verilerin orijinal formatı ve ölçeklendirme yapıldıktan sonraki formatı sunulmuştur.

İkinci adım ise ayrıştırma işidir. Otel verilerinde hiçbir özellik üzerinde ayrıştırma işi yapılamadığı için, bu adım uygulanmamıştır.

Birçok özellikler üzerinde birleştirme işi yapılmıştır. Örneğin resimler resimlerin URL'lerinin dizi formatında olduğu için, tüm resmi algoritmaya sunmak veri miktarını çok büyüteceği bilindiği için, URL metni olarak bir anlam taşımadığı için, resim sayısı olarak yeni özellik temsil edilmiştir. Açıklama özelliğinde de benzer birleştirme işi yapılmıştır. Açıklama uzun metin olarak temsil edildiğinde bir anlamı taşımayabileceğini, metin üzerinde metin analiz yapma işi daha karmaşık olabileceği düşünüldüğü için, analiz yapıldığı anda bile ana problemi çözmede etkisi çok küçük olabileceği düşünüldüğü için, metin uzunluğunda yeni özellik eklenmiştir. Metin uzunluğu tamsayı olarak kullanıldığında sonuca etkisi çok büyük olabileceği bilindiği için, 0 ve 1 arasındaki rasyonel sayıya ölçekleme işi yapılmıştır.

Otomatik otel öneri veri seti bir denetimli makine öğrenme kategorisine daha uygundur. Özellikle sınıflandırma yöntemine uygundur. Bu nedenle her bir karşılaştırma belgesi ayrıca bir sınıflandırma verisinin dizisi olarak temsil edilebilir. MongoDB belge formatını makine öğrenme algoritmasına uygun formata dönüştürülmüş hali aşağıdaki gibidir.

**Tablo 3.** Karşılaştırma sonuçlarının makine öğrenmesi algoritmasına uygunlaştırılmış hali.

| Birinci Otelin Özellikleri |                                  |  |     | İkinci Otelin Özellikleri |                                  |  |     | Sınıf |
|----------------------------|----------------------------------|--|-----|---------------------------|----------------------------------|--|-----|-------|
| Yıldız Sayısı              | TripAdvisor Değerlendirme Değeri | Açıklama                               | ... | Yıldız Sayısı             | TripAdvisor Değerlendirme Değeri | Açıklama                               | ... |       |
| 3                          | 6.6                              | Be our guest at this historic hotel... | ... | 4                         | 8.8                              | Make yourself at home in one of the... | ... | 0     |
| 5                          | 9.2                              | Located 400 metres ...                 | ... | 5                         | 9.0                              | Only 50 metres from Mitre...           | ... | 1     |

Tablo 3'te görüldüğü gibi karşılaştırma sonucu basit bir sınıflandırma problemine dönüştürülmüştür. Verinin bu gibi formatta olması veri setinin özellik sayısını çok yüksek yapmıştır. Bir otelin biçimlendirilmiş hali 162 özelliği içerirdi. Şimdi 162'den iki, yani 324 özellikli bir veriye dönüşmüş oldu.

Genelde özelliklerin çok sayıda olması, eğitim veri setinin miktarı büyük olmasını gerektirir. Değilse makine öğrenme algoritma performansı düşük olma olasılığını yükseltir.

### 3.2 Sınıflandırma İçin Modelin Oluşturulması

Sınıflandırma algoritmasını kullanmadan önce veri üzerinden veri ön seçme, ön işleme ve dönüştürüm işleri dikkatlice uygulanmıştır. En önemli adım olarak da karşılaştırma sonucunun bir sınıflandırma problemine dönüştürme adımı olmuştur. Sınıflandırma problemine dönüştürme karşılaştırılan otellerin özellikleri bir dizi olarak biçimlendirilmiştir. Sınıf olarak karşılaştırma sonucu sıfır ve bir ile etiketlenmiştir. Sonuç olarak ikili sınıflandırmaya uygun biçim hazırlanmıştır. Bu yöntemle hazırlanmış olan karşılaştırma sonuçları eğitim verisi olarak kullanılmıştır. Daha sonra dışarıdan bir otelin müşterinin rezerve ettiği otelden daha iyi mi veya daha kötü olduğu test edilmiştir.

SVC algoritmasından en önemli parametrelerden olan “kernel” adında parametre vardır. Kernel “linear”, “poly”, “rbf”, “sigmoid” “precomputed” değerlerinden birini alabilir. SVC algoritması kernel fonksiyonunu kullanarak giriş verisini çıkış verisine değişik yollarla değiştirme yapar.

Algoritmanın en basit kullanımı aşağıdaki gibidir.

```
clf = SVC(C=5.0, kernel='rbf')
clf.fit(x_train, y_train)
predicted = clf.predict(x_test)

print "Normal RBF", accuracy_score(y_test, predicted)
```

Şekil 2. Sınıflandırma algoritması

SVC algoritması C adında parametreyi alır. C hata teriminin penaltı parametresidir. Algoritma kurulduktan sonra, fit adındaki fonksiyonu eğitim verisiyle çağrılır. Şimdi model eğitilmiştir. Eğitim zamanında kullanılmayan veri yardımıyla modelin doğruluk oranını öğrenebiliriz. Doğruluk puanı ne kadar yüksek ise o kadar algoritma doğru çalışmaktadır demektir. SVC algoritmasını var olan tüm kernel fonksiyonlarıyla çalıştırdıktan sonra aşağıdaki tabloda gösterilmiş sonucu elde etmişizdir.

Table 4. SVC kernel fonksiyonları sonuçları.

| Kernel Foksiyon | Doğruluk Puanı |
|-----------------|----------------|
| RBF             | 0.89552238806  |
| Poly            | 0.865671641791 |
| Linear          | 0.865671641791 |
| Sigmoid         | 0.7910447761   |

Tablo 4’te görüldüğü gibi RBF kernel fonksiyonunun doğruluk oranı en büyüktür. Ama gerçekten de RBF kernel fonksiyonu bizim problem için en uygun mudur? Yoksa veriyi eğitim ve test parçalarına bölerken RBF için en uygun biçimde bölünmüş müdür? Bu soruya cevap almak için veri setini 20 ve 80 oranına birkaç sefer rastgele ayırma

yaparak, ayırma yapılmış veri üzerinde tüm kernel fonksiyonlarını çalıştırarak genel ortalama doğruluk oranını almamız lazım.

Ortalama doğruluk oranı ise aşağıdaki gibidir.

**Tablo 5.** Doğruluk oranı ortalaması

| Kernel  | Puan   |
|---------|--------|
| RBF     | 0.843  |
| Poly    | 0.8425 |
| Linear  | 0.831  |
| Sigmoid | 0.842  |

Yine doğruluk puanını yükseltmek mümkün mü? Bu soruya cevap bulmak için sadece kernel parametresini değil C ve gamma parametreleri için de en uygun değer ne olduğu bulunmaya çalışılmıştır. C = 10.0 ve gamma= 0.0017 olduğunda en iyi sonuç elde edilmiştir (Tablo 6).

**Tablo 6.** C=10.0 ve 0.0017 olduğu zaman ortalama doğruluk puanı.

| Kernel  | Ortalama Puan |
|---------|---------------|
| RBF     | 0.85          |
| Poly    | 0.83          |
| Linear  | 0.84          |
| Sigmoid | 0.85          |

Yukarıdaki en yüksek doğruluk puanı sunan algoritma parametre değerlerini arama sonuçları üzerinden ilginç analizler yapılabilir. İlk olarak, 0.843 en yüksek puan olmadığını görürüz. Yeni en yüksek puan ise 0.85'tir.

RBF yine de en yüksek puanı gösteren kernel fonksiyonun olduğunu ikinci defa kanıtlamıştır. Gamma değeri sıfıra yaklaşınca sigmoid kernel fonksiyonu da en yüksek doğruluğu vermektedir. Demek parametrelere doğru değerleri vererek doğruluk oranını yükseltmek mümkün.

Sonuç olarak, kurduğumuz model ile en yüksek 0.85 doğruluk puanına erişile bilinmiştir. Yüzde seksen beş makine öğrenme alanında iyi bir performans olarak bilinmektedir.

## 4 Sonuçlar

Bu çalışmanın amacı; otomatik otel yükseltici, otomatik daha iyi otele değişim yapma işini otomatikleştirmektir.

Adaptif otel öneri sisteminin doğruluk oranı 85% erişilmiştir. Sistemin gelecek versiyonunda kararın ne kadar olasılıkla verildiği hakkında bilgi verecek olanağını ekleme planlanmıştır. Çünkü birinci otelin ikinci otelden daha iyi olma olasılığı yüzde 51-55% ise, genelde otel değiştirmeye veya öneride bulunmaya gerek yoktur. Müşterilerin seçimi ile oynamak, istediğimiz kadarıyla otomatik değiştirmek çok

hassas bir konudur. Bu nedenle, otomatik deęiřtirme yapmandan önce sistem müşterinin yüzde kaç oranda yeni deęiřmelerle mutlu kalacağını hesaplamak çok önemlidir. Otomatik deęiřtirme yapmaktansa sadece öneride bulunmak çoęu zaman müşteri memnuniyetini artırır. Çünkü son karar yine de müşterinin kendi isteęine baęlı kalmaktadır.

Sonunda, çalışmanın anan konusu olan otomatik otel öneri ve puanlama sistemi başarıyla tamamlanmıştır. Model kurulmadan önce doğruluk oranı en azında 96% olması temel bir gereksinimlendendi. Makine öğrenmesinde bazı problemler ne kadar algoritmayı ayarlama yapsan bile, veri üzerinden en doğru ön işleme, veri seçme, dönüřtürme işleri yapılsa bile bazen doğruluk puanını yükseltmek tek bir ekstra adıma baęlıdır: veri miktarını çoęaltmak. Maalesef, bu iş genelde çok zaman gerektirir. Seyahat acenteleri manuel otel karşılaştırma işini yine de devam ederek, makine öğrenme verisini her gün çoęaltmaktadır. Bundan dolayı makine öğrenme modeli her gün gelişmeyi devam etmekte ve doğruluk oranını yükseltmektedir.

## Kaynaklar

1. Michael Arruza, John Pericich, Michael Straka. The Automated Travel Agent: Hotel Recommendations Using Machine Learning. Stanford University (2016)
2. Anindya Ghose, Panagiotis G.Ipeiritis, Beibei Li: Designing Ranking Systems for Hotels on Travel Search Engines to Enhance User Experience, ISIC (2010).
3. Elena Marchiori, Davide Eynard, Alessandro Inversini, Lorenzo Cantoni, Francesco Cerretti : Harvesting Online Contents: an Analysis of Hotel Reviews Websites, DOI: 10.1007/978-3-7091-0503-0\_9
4. Sara Dolnicar, T. Otter. Sydney: Which Hotel attributes Matter? A review of previous and a framework for future research. Proceedings of the 9th Annual Conference of the Asia Pacific Tourism Association (2017).
5. Personalization in Travel: Machine Learning Approaches to Behavior Analytics. <https://www.altexsoft.com/blog/datascience/customer-experience-personalization-in-travel-and-hospitality-using-behavioral-analytics-and-machine-learning/>, ziyaret tarihi 2018/05/05.
6. Xavier Amatriain, Justin Basilico: Netflix Recommendations: Beyond the 5 stars (Part 1). Netflix. <https://medium.com/netflix-techblog/netflix-recommendations-beyond-the-5-stars-part-1-55838468f429>, ziyaret tarihi 2018/05/20.
7. Medium. Netflix. <https://medium.com/netflix-techblog/netflix-recommendations-beyond-the-5-stars-part-2-d9b96aa399f5>, ziyaret tarihi 2018/03/26.
8. Papini, Tiziano. Siena, Italy: SortNet: Learning to Rank by a Neural Preference Function. IEEE, IEEE Transactions on Neural Networks . Vol. 22 (2011)
9. Bogdan Walek, Radim Farana, Oldrich Hosek. Tatranska Lomnica: Proposal of Expert System for Hotel Booking System. IEEE, Carpathian Control Conference (ICCC) (2016).
10. <https://medium.com/deep-learning-turkiye/makine-%C3%B6%C4%9Frenmesi-e%C4%9Flencelidir-b9d50aad3a62>, ziyaret tarihi 2018/03/25.
11. Faiza Dammak, Hager Kammoun, Abdelmajid Ben Hamadou: Improving pairwise learning to rank algorithms for document retrieval. Honolulu, HI, USA : IEEE, IEEE 2017 Symposium Series (2018).
12. Scikit. Support Vector Machines. <http://scikit-learn.org/stable/modules/svm.html>, ziyaret tarihi 2018/03/27.

13. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze: Introduction to Information Retrieval. Cambridge : Cambridge University Press (2009).
14. Eleftherios Tiakas, Apostolos N. Papadopoulos, Yannis Manolopoulos: Skyline Queries: an Introduction. Corfu, Greece : IEEE, 2015. Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference (2015).
15. Kernel Functions-Introduction to SVM Kernel & Examples. <https://data-flair.training/blogs/svm-kernel-functions/>, ziyaret tarihi 2018/05/01.
16. Anindya Ghose, Panagiotis G. Ipeirotis, Beibei Li: Designing Ranking Systems for Hotels on Travel Search Engines by Mining User-Generated and Crowd-Sourced Content, *Journal Marketing Science*, 493-520 (2012).
17. Li, Hang: A Short Introduction to Learning to Rank, IEICE Transactions (2011).