Mekânsal Verilerin Depolandığı İlişkisel Veritabanı Metin Arama İşleminde Sorgu Optimizasyonu

Murat Taşyürek[1]
[1]**Bilgisayar Mühendisliği Bölümü, Kayseri Üniversitesi, Kayseri, Türkiye**
murattasyurek@kayseri.edu.tr

Özet: Konum temelli veriler olarak bilinen mekânsal veriler fiziksel nesnenin coğrafi koordinat sistemindeki sayısal değerler ile temsil edilmesidir. Mekânsal verileri kaydetmek, düzenlemek, görselleştirmek ve analiz etmek için Coğrafi Bilgi Sistemi (Geographic Information System - GIS) yazılımları kullanılır. Mekânsal veriler GIS sistemleri sayesinde yaygın olarak kullanılmakta ve birçok sistemin alt yapısını oluşturmaktadır. Yaygın olarak kullanılan ve yer yüzünde bulunan bütün nesnelerin mekânsal veri olarak tanımlandığı GIS sistemlerinde arama işlemi ve aranan mekânsal verinin bulunması çok önemlidir. Bu çalışmada mekânsal verilerin depolandığı ilişkisel veritabanı metin arama işleminin daha doğru sonuçlar bulması için sorgu optimizasyonu önerilmiştir. Bu bildiride önerilen yöntem Kayseri Büyükşehir Belediyesine ait gerçek mekânsal verileri ve Oracle ilişkisel veritabanını kullanmıştır. Önerilen sorgu optimizasyonu tabanlı yöntem klasik sorgu yöntemi ile karşılaştırılmıştır. Deneysel sonuçlar, önerilen sorgu optimizasyonu tabanlı yöntemin klasik sorgu yöntemine göre kelime sayısı arttıkça çok daha başarı sonuçlar verdiğini göstermiştir.
Anahtar Kelimeler: Mekânsal veriler, Coğrafi bilgi sistemleri, Sorgu optimizasyonu, Akıllı şehir kayseri, Mobil uygulama

1. Giriş

Konum temelli veriler olarak bilinen mekânsal veriler fiziksel nesnenin coğrafi koordinat sistemindeki sayısal değerler ile temsil edilmesidir (Anselin vd., 2010). Mekânsal veriler, belirli bir konumla ilgili veya bu konum hakkında bilgi içeren herhangi bir nesneyi tanımlamak için kullanılır. Mekânsal veriler bina, okul, cami, eczane, göl, dağ veya kasaba gibi yeryüzündeki nesnenin konumunu, boyutunu ve şeklini temsil eder (Taşyürek, 2021). Mekânsal verileri kaydetmek, düzenlemek, görselleştirmek ve analiz etmek için Coğrafi Bilgi Sistemi (Geographic Information System - GIS) yazılımları kullanılır (Taşyürek, 2021). Mekânsal veriler GIS sistemleri sayesinde yaygın olarak kullanılmakta ve birçok sistemin alt yapısını oluşturmaktadır (Reibel, 2007). Yaygın olarak kullanılan ve yer yüzünde bulunan bütün nesnelerin mekânsal veri olarak tanımlandığı GIS sistemlerinde arama işlemi ve aranan mekânsal verinin bulunması çok önemlidir.

Mekânsal veri türleri vektör ve raster olmak üzere kendi içerisinde iki gruba ayrılır. Raster veri grubu genellikle grid formatında hava fotoğrafı ve uygu görüntülerinden oluşurken vektör veri grubu nokta, çizgi ve çokgen geometri veri türünden oluşur (Karduni vd., 2016). İlişkisel veritabanında SQL işlemlerinin ve indeks yapısının hızlı çalışması için mekânsal veriler veri türüne göre farklı farklı tablolarda tutulmaktadır. Verilerin farklı tablolarda tutulması arama işlemini zorlaştırmaktadır ve Taşyürek (Taşyürek, 2021) bu sorunun üstesinden gelmek için veritabanı denormalizasyon yöntemini kullanarak materyalleştirilmiş bir yapı (view) oluşturmuş ve arama işlemini hızlandırmıştır. Ancak denormalizasyon yöntemi ile sorguların cevap verme süresi azalmış olsa da metin tabanlı arama işlemlerinde genellikle yazım yanlışlarından veya aranan mekânsal nesnenin isminin tam olarak bilinmemesinden kaynaklı arama işlemi istenilen performansı verememektedir. Literatürde metin tabanlı arama işlemini için yaygın olarak Elasticsearch yöntemi kullanılmaktadır (Kononenko vd.,2014). Fakat

Elasticsearch yöntemi özellikle büyük veri kümesinde etkili çalışmaktadır ve mekânsal verilerin depolandığı ve güncellendiği ilişkisel veritabanında Elasticsearch yöntemini kullanmak içim mimarinin yeniden yapılandırılması gerekmektedir ve bu işlem birçok sistemi etkileyeceği için zordur.

Bu çalışmada, mekânsal verilerin depolandığı ve güncellendiği ilişkisel veritabanı metin arama işleminin daha doğru çalışması ve aranan kayıtları bulması için sorgu optimizasyonu tabanlı bir yöntem önerilmiştir. Bu bildiride önerilen yöntem Kayseri Büyükşehir Belediyesi'ne (KBB) ait gerçek mekânsal verileri Oracle ilişkisel veritabanı sisteminde kullanmıştır.

2. Problem ve Çözümler

Problem, KBB Akıllı Şehir Kayseri Mobil Uygulaması (Taşyürek vd., 2020) metin arama işleminde ortaya çıkmıştır. Öncelikle, metin arama işlemlerinde kullanılan klasik yöntem olan SQL (Structured Query Language) Like operatörü ile probleme çözüm üretilmeye çalışılmıştır. Fakat, klasik metin arama yöntemi ile mobil uygulamayı kullanan kullanıcıların aramış olduğu konumları (cami, okul, eczane vb.) bulamadığı tespit edilmiştir. Bu sorunun üstesinden gelmek için aranan metin ifadesi kelimelere bölünerek ve metin içerisinde geçen bütün kelimeleri kapsayacak şekilde SQL ifadesi optimize edilerek aranan konum bulunmaya çalışılmıştır.

2.1 Metin Arama Klasik Çözüm

İlişkisel veritabanlarında metin arama işlemleri için Like operatörü kullanılmaktadır. Like operatörü char, nchar, varchar ve varchar2 metin alanlarında arama yapmak için kullanılır. Like operatörü WHERE operatörü ile birlikte kullanılır ve belirtilen özel ifadeye uyan kaydın olup olmadığının sorgulanmasını sağlar. Like operatörü kullanım açısından diğer operatörlerden farklı olarak özel ifade içerisinde bazı karakterler kullanarak arama yapılır (Kim, 1982). Like operatörünün kullanabildiği karakter ve kullanım amacı aşağıda listelenmiştir.

- % ifadesi, % karakterinin bulunduğu yere sıfır veya daha fazla sayıda karakter geleceğini ifade etmektedir.
- _ ifadesi, _ karakterinin bulunduğu yere yalnızca bir karakter geleceğini ifade etmektedir.
- [] ifadesi, yalnızca parantez içerisine yazan karakterlerin geleceğini ifade etmektedir.
- [^] ifadesi, parantez içerisindeki karakterlerin geçmeyeceğini ifade etmektedir.

Tablo 1'de KBB'den alınan ve bütün mekânsal verilerin depolandığı tablonun yapısı gösterilmiştir.

Tablo 1. Veritabanı Tablo Yapısı

Sütun Adı	Veri Tipi	Açıklama
KayitNo	Rakam	Her bir kayıt için oluşturulan ve tekil olan alandır.
Adi	Metin	Mekânsal verinin adıdır.
Geometri	Mekânsal veri	Nokta, çizgi veya çokgen şeklinde olan geometrik veridir.
Turu	Metin	Mekânsal nesnenin türünü göstermektedir.
Ilce	Metin	Mekânsal verinin hangi ilçe sınırları içerisinde bulunduğunu göstermektedir.
Mahalle	Metin	Mekânsal verinin hangi mahalle sınırları içerisinde bulunduğunu göstermektedir.
CSBM	Metin	Mekânsal verinin hangi ilçe sınırları CSBM'ye bağlı olduğunu göstermektedir.
BinaAdi	Metin	Mekânsal verinin bulunduğu binanın ismini göstermektedir.

Akıllı Şehir Kayseri Mobil Uygulaması için Metin Arama Klasik yöntemde Tablo 1'de bulunan Adi sütunu için Like operatörü ile arama yapılmıştır. Oracle ilişkisel veritabanı için klasik metin arama yönteminin sorgusu Şekil 1'de gösterilmiştir. Şekil 1'de bulunan MV_KBB_ALL Tablo 1'de detaylıca anlatılan veritabanı tablosunu ve P_ARANAN_CUMLE ise aranan metin ifadesini göstermektedir.

```
SELECT * FROM MV_KBB_ALL WHERE ADI LIK '%||P_ARANAN_CUMLE||%'
```

Şekil 3. Klasik Metin Arama Sorgusu

Ancak kullanıcılar aramış oldukları mekânsal noktaların isimlerini yanlış yazabilmekte veya kelime dizilişi olarak veritabanında kayıtlı olan dizilişten farklı olarak arama yapmakta bu nedenle aramış olduğu sonuca ulaşamamaktadır. Kullanıcıların aramış oldukları verileri erişebilmesi için sorgu optimizasyonu tabanlı metin arama yöntemi önerilmiştir.

2.2 Metin Arama Sorgu Optimizasyonu Tabanlı Çözüm

İlişkisel veritabanında klasik metin arama işlemi ile bir cümle arama yapıldığında arama yapılan cümlenin veritabanında depolanan cümle ile aynı dizilişte olması gerekmektedir. Aranan cümle veritabanında depolanan dizilişinden farklı olur ise sorgu cümlesi kayıt bulamayacaktır. Bu sorunun üstesinden gelmek ve Akıllı Şehir Kayseri Mobil Uygulamasında kullanıcıların aramış oldukları mekânsal nesneleri bulması için

sorgu optimizasyon yöntemi önerilmiştir. Oracle ilişkisel veritabanında 5 kelime içeren cümle için optimize edilmiş metin arama yönteminin sorgusu Şekil 2'de gösterilmiştir.

```
SELECT * FROM (
SELECT M.*, UTL_MATCH.EDIT_DISTANCE_SIMILARITY(ADI,P_ARANAN_CUMLE) BENZERLIK_ORANI
FROM MV_KBB_ALL M WHERE  ( ADI LIKE '%||P_ARANAN_KELIME1||%'
OR ADI LIKE '%||P_ARANAN_KELIME2||%'
OR ADI LIKE '%||P_ARANAN_KELIME3||%'
OR ADI LIKE '%||P_ARANAN_KELIME4||%'
OR ADI LIKE '%||P_ARANAN_KELIME5||%') )
ORDER BY BENZERLIK_ORANI ASC
```

Şekil 4 Optimize Edilmiş Metin Arama Sorgusu

Şekil 2'de kullanıcıların aramış oldukları mekânsal noktaları bulması ve aranan cümlede bulunan kelime dizilişlerinin farklı olmasından kaynaklanan sorunları ortadan kaldırmak için cümle kelimelere bölünerek kelime bazlı arama işlemi yapılmıştır. Ancak, kelimelere bölerek arama işleminde sorgu istenilen sonucu verse aranan nokta kayıtlarda en sonda gelmektedir. Bu sorunun üstesinden gelmek için UTL_MATCH. EDIT_DISTANCE_SIMILARITY fonksiyonu kullanılmıştır. UTL_MATCH kütüphanesinde bulunan EDIT_DISTANCE_SIMILARITY fonksiyonu iki kelime arasındaki değişen karakter sayısını göstermektedir (Urbanoski vd., 2018). Değişim sayısı az olan kayıt kullanıcının aramış olduğu kayıt olma ihtimali yüksektir ve order by operatörü ile bu kayıtların ilk sırada gelmesi sağlanmıştır.

3. Araştırma Sonuçları

Bu çalışmada KBB'den alınan Oracle ilişkisel veritabanında depolanan mekânsal veriler kullanılmıştır. Tablo 1'de detaylıca anlatılan veritabanı tablosunda 574.274 adet kayıt bulunmaktadır. Deneysel çalışmalar 574.274 adet gerçek veri üzerinde yapılmıştır. Bu çalışmada yapılan deneyler Intel Xenon Gold 5118 CPU 2.30 GHz (4 Çekirdek) 16 GB Ram 600 GB SSD disk özellikleri ve Windows Server 2012 işletim sistemine sahip sunucu üzerinde gerçekleştirilmiştir.

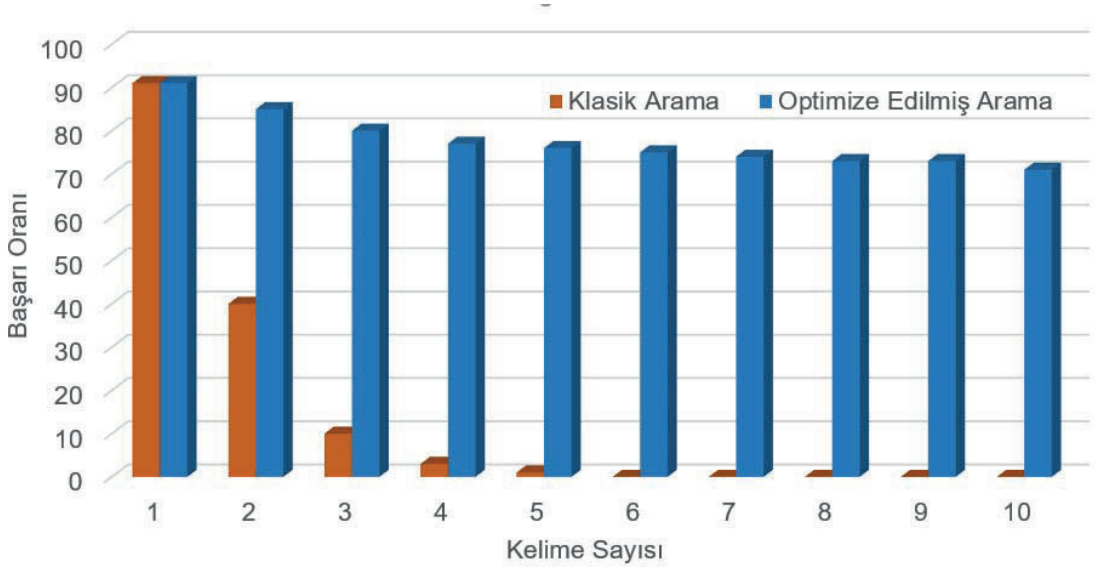
3.1. Deneysel Sonuçlar

Klasik metin arama ve önerilen optimize edilmiş metin arama yöntemlerinin sonuçları karşılaştırmak için arama sonuçlarının başarı oranları incelenmiştir. Aranan kelime sayısı ve toplam arama sayısına göre yöntemlerin başarıları irdelenmiştir. Tablo 2'de aranan kelime sayısı, toplam arama sayısı, klasik arama yönteminin başarı oranı ve önerilen optimize edilmiş arama yönteminin başarı oranı gösterilmiştir. Şekil 3'te ise yöntemlerin aranan kelime sayısına göre başarı oranı grafik olarak sunulmuştur.

Tablo 2. Arama Genel Bilgisi

Kelime Sayısı	Arama Sayısı	Klasik Yöntem	Önerilen Yöntem
1	50.230	91	91
2	40.021	40	85
3	18.037	10	80
4	10.615	3	77
5	6.123	0	76
6	2.687	0	75
7	1.678	0	74
8	1.378	0	73
9	1.203	0	73
10	785	0	71

Tablo 2 ve Şekil 3'te görüldüğü üzere sadece bir kelime arandığı durumda her iki yöntemde aynı sonucu vermektedir. Çünkü bu durumda her iki yönteminde sorgusu aynı olmaktadır. Kullanıcılar genellikle 1 veya 2 kelime ile arama yapmayı tercih etmektedir. Kelime sayısı artıkça klasik arama yönteminin doğru sonucu bulması için kullanıcının arama işlemini veritabanında kayıtlı olduğu formatta araması gerekmektedir. Bu nedenle kelime sayısı artıkça klasik arama yönteminin performansı düşmektedir. 5 veya üzerinde kelime arandığı durumda kullanıcıların harf hatasından veya veritabanında yazılmış olan ile birebir aynısını arayamadığından dolayı aranan hiçbir kayda ulaşmamıştır. Önerilen optimize edilmiş metin arama yöntemi ise aranan metin ifadeyi kelimelere bölüm her kelime için aslında arama yaptığı için ve en çok benzeyen kaydı ön planda gösterdiği için başarı oranı daha yüksek çıkmıştır. Klasik arama yöntemi 5 veya daha fazla kelime hiçbir kayıt bulamazken önerilen yöntem 10 kelime arandığı durumda bile %71 başarı oranı ile ilgili kayıtları bulmaktadır.

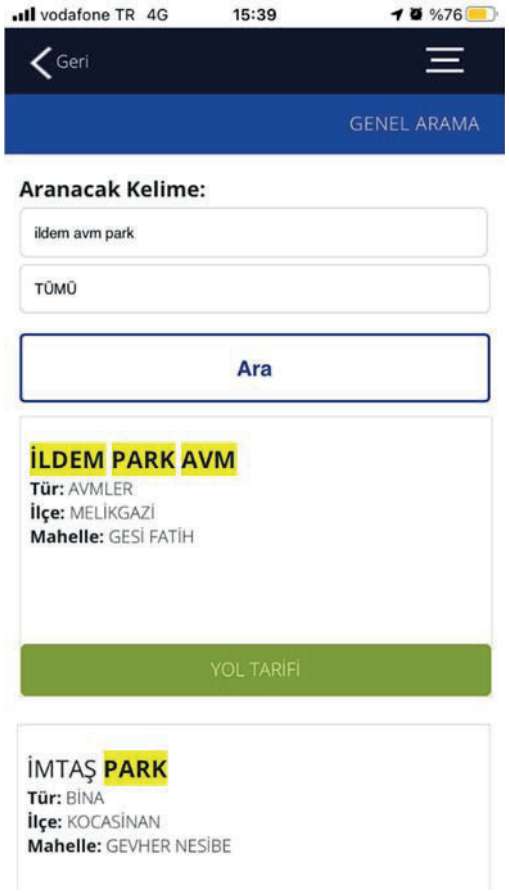


Şekil 5 Yöntemlerin Kelime Sayısına Göre Başarı Oranı

Metin arama işleminde klasik yöntem sadece aranan cümle tek kelimeden oluştuğunda doğru sonuçlar üretmektedir. Aranan kelimede birden fazla kelime olduğunda klasik yöntem sonuç üretememiştir. Önerilen sorgu optimizasyon tabanlı yöntem ise aranan cümleleri kelime bölerek aradığı için doğru sonuçlar üretmektedir.

4.2. Akıllı Şehir Kayseri Mobil Uygulaması Arama Sonucu

Önerilen optimize edilmiş metin arama yöntemi Kayseri Büyükşehir Belediyesi kurumsal mobil uygulaması olan Akıllı Şehir Kayseri Mobil uygulamasında aktif olarak kullanılmaktadır. KBB mobil uygulaması Google Play Store ve App Store'dan ücretsiz olarak indirilebilmektedir. Mobil uygulamada açılan genel arama ekranında arka planda önerilen optimize edilmiş metin arama yöntemi çalışmaktadır. Kullanıcılar kelimeler arasında genellikle boşluk kullanmaktadır ve bu nedenle kelime bölüm işlemi boşluk karakterine göre yapılmıştır. Önerilen yöntem aranan mekânsal konumun isminin kelimeleri yanlış yazılsa bile aranan kayıtları ilk sırada göstermektedir. Şekil 4'te İldem avm park cümlesi aranmıştır. Aranan cümlede bulunan kelime dizilimi ile Oracle veritabanında bulunan kelime dizilimi aynı olmamasına rağmen önerilen yöntem kullanıcının aramış olduğu mekânsal noktayı bulmakta ve sıralamada ilk sırada göstermektedir.



Şekil 6 Metin Arama Sorgu Optimizasyonu Sonucu

4. Sonuçlar

Mekânsal veriler GIS sistemleri sayesinde yaygın olarak kullanılmakta ve birçok sistemin alt yapısını oluşturmaktadır. Yaygın olarak kullanılan ve yer yüzünde bulunan bütün nesnelerin mekânsal veri olarak tanımlandığı GIS sistemlerine arama işlemi ve aranan mekânsal verinin bulunması çok önemlidir. Bu çalışmada, mekânsal verilerin depolandığı ilişkisel veritabanında metin arama işleminin başarısını artırmak için sorgu optimizasyonu tabanlı yöntem önerilmiştir. Önerilen yöntemin sonuçları ile klasik metin arama yöntemin sonuçları KBB'den alınan gerçek veriler kullanılarak ile karşılaştırılmıştır. Deneysel değerlendirmelerde önerilen yöntemin aranan konumun kelime sırasını yanlış yazılsa bile ilgili kaydı ilk sırada getirdiği görülmüştür. Deneysel değerlendirmeler, klasik arama yönteminin 5 ve üzeri kelime arandığında hiçbir sonuç bulamadığını, önerilen yöntemin en kötü durum olan 10 kelime arandığında bile %71 başarı oranı ile ilgili kayıtları bulduğunu göstermiştir.

5. Teşekkür

Kayseri Büyükşehir Belediyesi'ne ilişkisel veritabanında depolanan ve güncellenen mekânsal verileri paylaştığı için teşekkür ederiz.

Kaynaklar

- Anselin, L., Syabri, I. and Kho, Y. (2010). GeoDa: an introduction to spatial data analysis. In Handbook of applied spatial analysis. Springer, pp. 73-89
- Karduni, A., Kermanshah, A. and Derrible, S. (2016). A protocol to convert spatial polyline data to network formats and applications to world urban road networks. Scientific data, 3(1), pp. 1-7.
- Kim, W. (1982). On optimizing an SQL-like nested query. ACM Transactions on Database Systems (TODS), 7(3), pp. 443-469.
- Kononenko, O., Baysal, O., Holmes, R. and Godfrey, M. W. (2014). Mining modern repositories with elasticsearch. In Proceedings of the 11th working conference on mining software repositories, pp. 328-331.
- Reibel, M. (2007). Geographic information systems and spatial data processing in demography: a review. Population Research and Policy Review, 26(5), pp 601-618.
- Taşyürek, M. (2021). Mekânsal Verilerin Sıklıkla Güncellendiği Coğrafi Bilgi Sistemleri Arama İşleminde Denormalizasyon Yöntemi. Avrupa Bilim ve Teknoloji Dergisi, (24), pp. 18-23.
- Taşyürek, M. (2021). Mekân-Zamansal Veri Madenciliği Yöntemi ile Otobüs Durak İhlallerinin Tespiti. Avrupa Bilim ve Teknoloji Dergisi, (24), pp. 449-454.
- Taşyürek, M. and Çelik, M. (2020). Akıllı Durak Sistemindeki Araç Seyahat Sürelerinin Birleşik Yapay Sinir Ağları Kullanarak Tahmini. Avrupa Bilim ve Teknoloji Dergisi, Ejosat Special Issue 2020 (ARACONF), pp. 72-79.
- Urbanoski, K., Kenaszchuk, C., Inglis, D., Rotondi, N. K., and Rush, B. (2018). A system-level study of initiation, engagement, and equity in outpatient substance use treatment. Journal of substance abuse treatment, 90, pp. 19-28.