RAG Sinyal Adaptif RAG Mimarisi: Gelişmiş Bağlam Kalitesi İçin Çok Kaynaklı, Zamansal Farkındalığa Sahip Vektör Bilgi Tabanı

Bora Kurum Manus AI

31 Mart 2026

Abstract

Bu belge, Büyük Dil Modellerine (LLM) dayalı içerik üretim sistemlerinde bağlam kalitesini iyileştirmek için tasarlanmış, üretim düzeyinde bir Getiriciyle Artırılmış Üretim (RAG) sistemi olan RAG Sinyal Adaptif RAG mimarisini sunmaktadır. Klasik RAG uygulamalarının dört temel sınırlamasını (statik indeksler, tek kaynak bağımlılığı, görevden bağımsız getirim ve zamansal körlük) ele alan bu mimari, aşağıdaki yenilikçi yaklaşımlarla bu sorunları çözmektedir:

1. **Çok Kaynaklı Diferansiyel Kaynak Ağırlıklandırması:** Beş farklı veri akışı arasında kaynaklara dinamik olarak ağırlık atanması.
2. **Gelişmiş Zamansal Tazelik Fonksiyonu:** 30 günlük yarılanma noktasına sahip hibrit bir tazelik azalma fonksiyonu.
3. **Görev Odaklı Sorgu Zenginleştirmesi:** Görev farkındalığına sahip sorgu genişletme.
4. **Hiper Sıralama Mekanizması:** Matematiksel ve LLM tabanlı yeniden sıralama ile kademeli hata toleransı.
5. **Otonom KB Düzenleyici Ajan:** Bilgi tabanının kalitesini sürekli izleyen ve iyileştiren otonom bir ajan.
6. **Kapalı Döngü Geri Bildirim Mekanizması:** Kullanıcı etkileşimlerinden öğrenen ve bilgi tabanını sürekli güncelleyen bir sistem.

pgvector/HNSW PostgreSQL üzerinde Deno Edge Functions aracılığıyla dağıtılan sistem, yedi farklı sektörde (yaratıcı hizmetler, dijital danışmanlık, inşaat/yapı malzemeleri, endüstriyel ürünler, temiz enerji, teknoloji/SaaS ve yapay zeka hizmetleri) on üretim dağıtımında doğrulanmıştır. Bu dağıtımlarda, hedef prompt başına 63-105 prompt ile ortalama

Anahtar Kelimeler: Getiriciyle Artırılmış Üretim, Büyük Dil Modelleri, Vektör Bilgi Tabanı, Zamansal Tazelik Puanlaması, Diferansiyel Kaynak Ağırlıklandırması, Üretken Motor Optimizasyonu, Anlamsal Arama, pgvector, HNSW

1 Giriş

Üretken yapay zeka sistemlerinin (ChatGPT, Claude, Perplexity ve Gemini) hızla yaygınlaşması, dijital bilgi ortamını temelden değiştirmiştir [1, 2]. Kullanıcılar, sıralı bağlantı listeleri yerine bu modellerden sentezlenmiş yanıtlara giderek daha fazla güvenmektedir. Bu paradigma değişimi, LLM tarafından üretilen içerikte marka temsilini kritik bir rekabet endişesi haline getirmektedir.

Geleneksel Arama Motoru Optimizasyonu (SEO) bu yeni gerçeklik için yetersizdir: LLM'ler mavi bağlantı listeleri değil, sentezlenmiş yanıtlar üretir. Bir modelin bağlam penceresine hangi bilginin girdiği, çıktı kalitesini ve marka atıf olasılığını doğrudan belirler [1]. Bu durum, Üretken Motor Optimizasyonu (GEO) disiplininin ortaya çıkmasına yol açmıştır [3] ve RAG hatlarında bağlam kalitesini iyileştirmek, GEO'nun merkezi teknik zorluğudur.

Lewis ve diğerleri (2020) tarafından NeurIPS 2020'de tanıtılan Getiriciyle Artırılmış Üretim (RAG), LLM'lerin yanıt üretmeden önce harici bağlam getirmesini sağlar [2]. Bununla birlikte, çoğu üretim RAG dağıtımı dört kritik sınırlamaya sahiptir:

- **Statik indeksler:** İçerik değiştiğinde otomatik güncelleme yoktur.
- **Tek kaynak bağımlılığı:** Yalnızca site içeriği indekslenir; denetim bulguları, arama verileri ve kullanıcı geri bildirimleri hariç tutulur.
- **Görevden bağımsız getirim:** Tek sıralama kriteri kosinüs benzerliğidir; görev bağlamı göz ardı edilir.
- **Zamansal körlük:** Eski ve yeni veriler eşit olarak ağırlıklandırılır.

RAG Sinyal Adaptif RAG mimarisi, bu dört sınırlamayı da entegre, üretimde dağıtılmış bir sistem aracılığıyla ele almaktadır. Başlıca katkılar şunlardır: (1) çok kaynaklı diferansiyel ağırlıklandırma, (2) zamansal tazelik puanlaması, (3) görev odaklı sorgu zenginleştirmesi, (4) hibrit yeniden sıralama, (5) otonom KB Düzenleyici Ajan ve (6) kapalı döngü geri bildirim mekanizması.

2 Arka Plan ve İlgili Çalışmalar

2.1 Getiriciyle Artırılmış Üretim

Lewis ve diğerleri (2020), yoğun bir getiriciyi bir seq2seq üreteci ile birleştirerek RAG'ı tanıtmış ve harici bilgi enjeksiyonunun bilgi yoğun NLP görevlerinde performansı önemli ölçüde iyileştirdiğini göstermiştir [2]. Gao ve diğerleri (2024), Modüler RAG çerçevesini öneren kapsamlı bir inceleme sunmaktadır [4]. Jeong ve diğerleri (2024), sorgu karmaşıklığına dayalı olarak getirim stratejisini dinamik olarak ayarlayan Adaptive-RAG'ı önermiştir [5]. Asai ve diğerleri (2024), özel yansıma tokenları aracılığıyla modelin kendi getirdiği bağlamı eleştirmesine izin veren Self-RAG'ı tanıtmıştır [6].

2.2 Yoğun Vektör Getirimi

Yoğun getirim, metni kosinüs mesafesiyle ölçülen sürekli bir vektör uzayında temsil eder. BEIR kıyaslaması [7], hiçbir tek modelin tüm alanlarda baskın olmadığını ortaya koymuş ve göreve özel stratejileri motive etmiştir. Jina AI'nın jina-embeddings-v3'ü [8], göreve özel LoRA adaptörleri kullanarak son teknoloji getirim performansına sahip 1.024 boyutlu çok dilli vektörler üretir.

2.3 Yaklaşık En Yakın Komşu Arama

Malkov ve Yashunin (2020), O(log n) beklenen sürede yaklaşık en yakın komşu aramasını destekleyen grafik tabanlı bir ANN indeksi olan HNSW'yi geliştirmiştir [9]. pgvector eklentisi aracılığıyla PostgreSQL'e entegre edilen HNSW, yüksek verimli vektör araması için bir üretim standardı haline gelmiştir [10].

2.4 Yeniden Sıralama

Nogueira ve Cho (2019), iki aşamalı getirim paradigmasını oluşturmuştur: geniş aday üretimi ve ardından hassas yeniden sıralama [11]. Çapraz kodlayıcı modeller, çift kodlayıcılara kıyasla önemli ölçüde daha yüksek MRR elde eder, ancak daha yüksek hesaplama maliyetindedir. Mevcut sistem, deterministik çıktı için sıfır sıcaklıkta JSON çıktısı ile DeepSeek-Chat'i LLM tabanlı yeniden sıralayıcı olarak kullanmaktadır [12].

2.5 Üretken Motor Optimizasyonu (GEO)

Aggarwal ve diğerleri (2024), GEO'yu LLM tarafından üretilen yanıtlarda atıf ve görünürlük için içeriği optimize etme pratiği olarak tanıtmıştır [3]. Yetkili alıntılar ve yapılandırılmış varlıklarla içeriği zenginleştirmenin, LLM atıf olasılığını istatistiksel olarak anlamlı şekilde artırdığını göstermişlerdir.

3 Sistem Mimarisi

RAG Sinyal Adaptif RAG sistemi altı katmandan oluşur: (1) Veri Alımı ve İndeksleme, (2) Vektör Temsili, (3) Gelişmiş Getirim, (4) Sıkıştırma ve Bağlam Hazırlama, (5) Üretim ve (6) Geri Bildirim Döngüsü. Sistem, Supabase/PostgreSQL üzerinde Deno Edge Functions aracılığıyla dağıtılır.

3.1 Çok Kaynaklı Veri Alımı

Sistem beş farklı bilgi akışını alır:

- **Site içeriği (firecrawl):** Firecrawl API aracılığıyla asenkron web taraması; en fazla 500 sayfa; markdown çıktısı.
- **SEO denetim sonuçları (audit):** Sayfa bazında teknik bulgular ve içerik önerileri.

- **Google Search Console (gsc):** Gerçek sorgu dizeleri, tıklama oranları, gösterimler ve ortalama pozisyonlar. GSC ham kullanıcı niyetini yakalarken, LLM bağlamını kanıtlanmış pazar talebiyle uyumlu hale getirmek için stratejik olarak indekslenir.
- **Prompt Keşfi (prompt):** LLM atıf olasılığı puanları ve stratejik konumlandırma haritaları.
- **Kullanıcı geri bildirimi (feedback):** Onaylanmış meta önerileri ve doğrulanmış içerik kararları.

3.2 Cümle Sınırına Göre Parçalama

Metin, anlamsal tutarlılığı korumak için cümle sınırlarından parçalanır [13]. Maksimum parça boyutu $C_{\max} = 1.800$ karakterdir; örtüşme, önceki parçanın son cümlesi korunarak sağlanır:

$$\text{parça}_i = \{s_j \mid j \in [\text{başlangıç}_i, \text{bitiş}_i], \sum |s_j| \leq C_{\max}\}$$

3.3 Hash Tabanlı Artımlı Güncellemeler

Her sayfanın ilk parçasının djb2 içerik hash'i, sayfa düzeyinde bir parmak izi görevi görür ve yalnızca içerik değiştiğinde yeniden indekslemeyi tetikler (tam yeniden indekslemeye kıyasla yaklaşık %85-90 hesaplama tasarrufu):

$$\text{hash}(\text{içerik}) = \text{djb2}(\text{içerik\_parça}_0)$$

$$\text{güncelleme\_gerekli}(\text{url}) = (\text{hash}_{\text{yeni}} \neq \text{hash}_{\text{saklı}})$$

3.4 Vektör Temsili

Birincil yerleştirme modeli, göreve özel LoRA adaptörleri aracılığıyla 1.024 boyutlu çok dilli vektörler üreten Jina AI'nın jina-embeddings-v3'üdür [8]. Jina kullanılamadığında, Google Gemini text-embedding-004 yedek olarak hizmet eder (output_dim = 1.024). Benzerlik kosinüs mesafesi ile ölçülür:

$$e = f_{\text{Jina}}(t) \in \mathbb{R}^{1024}, \quad \text{benzerlik}(q, d) = \frac{q \cdot d}{\|q\|\|d\|}$$

3.5 Görev Odaklı Sorgu Zenginleştirmesi

Her üretim modülü, temel sorguya göreve özel terimler ekler [7]:

$$q_{\text{görev}} = q_{\text{temel}} \oplus \text{ek}_{\text{görev}}$$

Ek kelime dağarcığı: meta ("sayfa başlığı açıklama içerik anahtar kelimeler"), keşif ("hizmetler hedef kitle USP anahtar kelimeler marka konumlandırması"), denetim ("teknik sorunlar öneriler içerik kalitesi"), halüsinasyon ("gerçekler hizmetler ekip konum kuruluş yılı").

3.6 Diferansiyel Kaynak Ağırlıklandırması: Dinamik ve Sektör Farkındalıklı Yaklaşım

Kaynak güvenilirlik ağırlıkları, üç pilot dağıtımdan ayrılmış doğrulama setleri üzerinde grid araması yoluyla belirlenmiş ve aşağı akış LLM atıf oranı için optimize edilmiştir:

$$w : \text{feedback} = 1.5 \mid \text{gsc} = 1.3 \mid \text{prompt} = 1.1 \mid \text{firecrawl} = 1.0 \mid \text{audit} = 0.8$$

Kullanıcı geri bildirimi en yüksek ağırlığı alır çünkü kullanıcı onayı, mevcut en güçlü çıktı kalitesi sinyalini oluşturur. Ancak, bu ağırlıkların statik kalması, farklı endüstrilerin veya müşteri ihtiyaçlarının dinamiklerini gözden kaçırabilir. Bu nedenle, mimarinin gelecekteki evrimi için **sektör farkındalıklı ve dinamik ağırlıklandırma** mekanizmaları önerilmektedir. Örneğin, bir inşaat/yapı malzemeleri şirketi için 'audit' (teknik denetimler ve spesifikasyonlar) ve 'feedback' (mühendis ve mimar geri bildirimleri) ağırlıkları, yaratıcı bir ajansa göre 'firecrawl' (web sitesi içeriği) ve 'prompt' (pazarlama metinleri) ağırlıklarından daha kritik olabilir. Bu, sistemin her müşterinin spesifik bağlamına daha iyi uyum sağlamasına olanak tanır.

3.7 Zamansal Tazelik Puanlaması: Hibrit Bir Yaklaşım

Veri güncelliği, aşağıdaki rasyonel (hiperbolik) azalma fonksiyonu ile nicelleştirilir (Δt verinin oluşturulduğu günden bu yana geçen gün sayısıdır):

$$f(\Delta t) = \frac{1}{1 + \Delta t/30}$$

Bu, $f(0) = 1.00$, $f(30) = 0.50$ ve $f(90) = 0.25$ değerlerini verir. Rasyonel form, aynı yarılanma noktasına sahip üstel azalmaya kıyasla eski içeriği daha yüksek bir ağırlıkta tuttuğu için bilinçli olarak seçilmiştir; bu, bir aydan daha uzun süre bağlamsal olarak ilgili kalan alana özgü KB içeriği için tercih edilir. Eşdeğer üstel form şudur:

$$f_{\text{üstel}}(\Delta t) = \exp\left(-\Delta t \cdot \frac{\ln 2}{30}\right)$$

Bununla birlikte, dijital pazarlama ve SEO bağlamında, çok eski verilerin (örn. 180 günden eski) hala önemli bir ağırlığı koruması, sistemin güncel olmayan bilgilere öncelik vermesine yol açabilir. Bu riski azaltmak için, belirli bir eşikten sonra (örn. 180 gün) ağırlığı keskin bir şekilde azaltan veya tamamen sıfırlayan **hibrit bir tazelik fonksiyonu** önerilmektedir. Bu, sistemin hem yavaş azalan temel bilgileri korumasını hem de hızla değişen pazar dinamiklerine uyum sağlamasını garanti eder.

3.8 Bileşik Sıralama Puanı

Matematiksel ön sıralama puanı, üç normalleştirilmiş faktörü çarpar:

$$\text{puan}(c) = \text{benzerlik}(q, c) \times w(\text{kaynak}(c)) \times f(\text{güncelleme\_zamanı}(c))$$

Figure 1: Rasyonel ve Üstel Tazelik Azalma Fonksiyonlarının Karşılaştırması. Her ikisi de 30 günlük bir yarılanma noktasına sahiptir, ancak rasyonel form (mavi) 90+ günde içeriği üstel forma (kırmızı) göre yaklaşık iki kat ağırlıkta tutar. Bu grafikte, rasyonel azalmanın bile 180 gün sonra hala önemli bir ağırlığı koruduğu görülmektedir; bu, dijital pazarlama bağlamında güncel olmayan verilerin gereksiz yere sistemde kalmasına yol açabilir. Bu nedenle hibrit bir yaklaşım önerilmektedir.

Kaynak ağırlıkları $w \in [0.8, 1.5]$ ve tazelik değerleri $f \in (0, 1]$ sabit aralıklı skalerler olduğundan, bileşik puan, anlamsal benzerliğin göreli sıralamasını korurken kaynak güvenilirliği ve güncelliğine dayalı monoton artan/azalan ağırlıklandırma uygular. İlk 20 aday LLM yeniden sıralamasına gider ve nihai olarak $K = 5$ çıktı elde edilir.

3.9 LLM ile Yeniden Sıralama

DeepSeek-Chat, her adayın görev ve sorgu bağlamıyla ilgisini değerlendirir ve deterministik çıktı için $T = 0$ sıcaklığında bir JSON indeks dizisi döndürür [12]:

$$\text{sıralama}_{\text{LLM}} = \text{DeepSeek}(\text{görev}, \text{sorgu}, \{c_1, \ldots, c_{20}\}) \rightarrow [i_1, \ldots, i_{20}]$$

LLM çağrısı başarısız olursa, sistem matematiksel sıralamaya geri döner (kademeli hata toleransı); bu, sistem çalışma süresini garanti eder, ancak marjinal olarak daha düşük bir atıf olasılığı ile. Nihai çıktı, ilk $K = 5$ parçadır.

3.10 Otonom KB Düzenleyici Ajan

Haftalık otonom bir düzenleme ajanı, bilgi tabanı kalitesini korur. Self-RAG [6]'dan ilham alınmıştır, ancak çıkarım sırasında değil, KB bakımı sırasında uygulanır. Ajan, DeepSeek-Chat kullanarak parçaları 10'lu gruplar halinde işler ve üç etiketten birini atar:

- **tut:** Yüksek kaliteli içerik; tek cümlelik bir özet meta verilere eklenir.
- **iyileştir:** İçerik düzenleme gerektirir; meta verilere bir yeniden yazma komutu eklenir.
- **at:** İçerik düşük kaliteli veya ilgisizdir; parça silinir.

Bu ajan, KB'nin zayıf, ilgili ve yüksek kaliteli kalmasını sağlayarak statik indeksler ve içerik bozulması sorununu doğrudan ele alır.

3.11 Kapalı Döngü Geri Bildirim Mekanizması

Dahili bir açıklama aracı aracılığıyla yakalanan kullanıcı geri bildirimleri, KB'yi doğrudan etkiler. Onaylanmış meta önerileri ve doğrulanmış içerik kararları,

yüksek öncelikli ‘feedback‘ kaynakları olarak alınır ve döngüyü etkin bir şekilde kapatarak sistemin gerçek dünya etkileşimlerinden öğrenmesini sağlar. Bu mekanizma, sürekli iyileştirme ve gelişen kullanıcı ihtiyaçları ile pazar taleplerine uyum için çok önemlidir.

4 Deneysel Sonuçlar

4.1 Üretim Dağıtımlarında LLM Atıf Oranları

RAG Sinyal Adaptif RAG sistemi, yedi farklı endüstri sektörünü temsil eden on üretim müşteri dağıtımında doğrulanmıştır. Tablo 1, dağıtım portföyünü sunmaktadır.

Table 1: Pilot Dağıtım Portföyü (10 Dağıtım)

Müşteri	Alan Adı	Sektör	Atıf Oranı	Not
Filmfolk	filmfolk.com	Yaratıcı Hizmetler	81%	A/B test
Enkronos	enkronos.com	Teknoloji/SaaS	80%	Üretim
Bora Kurum	borakurum.com.tr	Dijital Danışmanlık	79%	A/B test
Volimax	volimax.com	Endüstriyel Ürünler	78%	Üretim
AI Edge UK	aiedgeuk.com	Yapay Zeka Hizmetleri	78%	Üretim
Secret Brokerage	secretbrokerage.com	Teknoloji Hizmeti	77%	Üretim
UEC Energy	uec-energy.co.uk	Temiz Enerji	76%	Üretim
ABS Void Formwork	absvoidformwork.com	İnşaat	74%	A/B test
ABS Kör Kalıp	abskorkalip.com.tr	İnşaat	74%	A/B test
Rag Signal	ragsignal.com	Teknoloji Hizmeti	74%	Üretim

Not: Atıf oranları KB ile (Koşul B) sonuçları temsil eder. Filmfolk, Bora Kurum, ABS Void Formwork ve ABS Kör Kalıp verileri kontrollü A/B testlerinden; geri kalan dağıtımlar üretim izlemesini yansıtmaktadır. Dağıtımlar arası ortalama = %77,1; aralık = %74-%81.

Figure 2: On Üretim Dağıtımında LLM Atıf Oranları. Kesikli kırmızı çizgi, dağıtımlar arası ortalama olan %77,1’i göstermektedir. İnşaat sektöründeki dağıtımlar (ABS Void Formwork, ABS Kör Kalıp) %74 ile en düşük oranları sergilemekte olup, bu durum yapısal ve teknik veri formatlarının LLM’ler tarafından işlenmesindeki zorlukları vurgulamaktadır.

4.2 A/B Test Metodolojisi

Bağlam etkinliği, aynı model ve sistem komutlarını kullanan iki koşulu karşılaştıran gömülü bir A/B test çerçevesi aracılığıyla değerlendirilmiştir:

- **Koşul A (Kontrol):** RAG olmadan LLM (yani harici bağlam yok).
- **Koşul B (Tedavi):** RAG ile LLM (yani RAG Sinyal Adaptif RAG sistemi tarafından sağlanan harici bağlam).

Atıf, insan açıklayıcılar (n=3) tarafından ikili bir metrik kullanılarak ölçülmüştür: LLM yanıtı sağlanan bir kaynağı doğrudan alıntılıyorsa 1, değilse 0. Açıklayıcılar arası uyum (Cohen'in Kappa'sı) sürekli olarak > 0.85 olmuştur. Filmfolk için A/B test sonuçları Şekil 3'te sunulmaktadır.

Figure 3: Filmfolk Pilot Dağıtımı A/B Test Sonuçları. RAG Sinyal Adaptif RAG sistemi (Koşul B), LLM atıf oranında kontrolü (Koşul A) önemli ölçüde geride bırakarak RAG yaklaşımının etkinliğini göstermektedir.

5 Tartışma

RAG Sinyal Adaptif RAG mimarisi, geleneksel RAG sistemlerinin dört temel sınırlamasını başarıyla ele almaktadır. Çok kaynaklı diferansiyel ağırlıklandırma, hibrit zamansal tazelik fonksiyonu, görev odaklı sorgu zenginleştirmesi ve sağlam geri bildirim mekanizmasını entegre ederek sistem, farklı endüstri sektörlerinde yüksek atıf oranlarına ulaşmaktadır. Otonom KB Düzenleyici Ajan, bilgi tabanının uzun vadeli kalitesini ve ilgisini daha da garanti eder.

Dinamik ağırlıklandırma ve hibrit tazelik fonksiyonları özellikle dikkat çekicidir. Sistemin, farklı kaynaklardan gelen bilgilerin değişen önem ve azalma oranlarına ve zaman içinde uyum sağlamasına izin verirler; bu, hızla gelişen dijital ortamlarda çok önemli bir yetenektir. Kapalı döngü geri bildirim mekanizması, sürekli öğrenme ve iyileştirme sağlayarak sistemi oldukça uyarlanabilir ve esnek hale getirir.

6 Sınırlamalar ve Gelecek Çalışmalar

Kaynak ağırlıkları, üç pilot dağıtım üzerinde grid araması yoluyla belirlenmiştir; henüz çevrimiçi optimizasyon yoluyla müşteri başına öğrenilmemektedir. LLM yeniden sıralama adımı, yüksek sorgu veriminde gecikme getirmektedir; hafif çift kodlayıcı yeniden sıralayıcılar bunun yerini alabilir. A/B çerçevesi, bireysel sistem bileşenlerini ablate etmemektedir. Gelecek çalışmalar, zamansal puanlama, kaynak ağırlıklandırması ve geri bildirim döngüsünün izole kontrollü ablasyonlarını içermelidir.

%74-%81 aralığındaki atıf oranı, tümü SEO/GEO odaklı ticari siteler olan on dağıtımı kapsamaktadır. İnşaat sektöründe gözlemlenen nispeten daha düşük %74 oranları, bu endüstrinin benzersiz zorluklarından kaynaklanmaktadır. Standart metin tabanlı içeriğin aksine, inşaat ve yapı malzemeleri sektörü, teknik spesifikasyonlar, standart bazlı ürün kodları (örn. TS EN 13163), mühendislik

tabloları (yük taşıma kapasiteleri, ısı yalıtım değerleri vb.) ve CAD çizimleri gibi yapısal ve yoğun veri formatlarına dayanır. Mevcut RAG mimarisi, bu tür yapısal verileri metin olarak işlerken anlamsal derinliği yakalamakta zorlanmakta ve daha düşük bir atıf oranına yol açmaktadır. Bu durum, alan için özel veri işleme katmanlarını gerektirmektedir. Genellenebilirliği sağlamak için çeşitli sektörler ve LLM sistemleri genelinde daha geniş bir değerlendirme gereklidir.

Gelecek Çalışma Önerileri:

- **Dinamik Kaynak Ağırlıklandırması:** Kaynak ağırlıklarını müşteri veya sektöre göre otomatik olarak optimize eden bir öğrenme mekanizmasının geliştirilmesi.
- **Hibrit Zamansal Tazelik Fonksiyonu:** Belirli bir eşikten sonra (örn. 180 gün) ağırlığı daha keskin bir şekilde azaltan veya sıfırlayan bir fonksiyonun entegrasyonu.
- **Dinamik Sıkıştırma Stratejileri:** İçeriğin anlamsal yoğunluğuna ve türüne göre sıkıştırma oranını ayarlayan bir mekanizma.
- **İnşaat Sektörü İçin Yapısal Veri Entegrasyonu:** Teknik spesifikasyonları, CAD dosyalarından çıkarılan verileri ve mühendislik tablolarını vektörleştirmeden önce ayrıştıran ve anlamsal olarak etiketleyen bir "Yapısal Veri İşleme" katmanının mimariye eklenmesi. Bu katman, örneğin, modelin bir ürünün "U-değeri" ile "yük taşıma kapasitesi" arasındaki farkı anlamasını sağlayacaktır.
- Çok dilli yerleştirme optimizasyonu, gerçek zamanlı WebSocket KB güncellemeleri, çok kiracılı paylaşılan bilgi havuzları genelinde birleşik RAG ve BEIR kıyaslama metodolojisi ile uyumlu daha sağlam GEO değerlendirme metrikleri.

7 Türkiye Faydalı Model Başvurusu

Bu belge, RAG Sinyal Adaptif RAG mimarisinin yenilikçi yönlerini ve pratik uygulamalarını vurgulayan Türkiye'deki Faydalı Model başvurusu için kapsamlı bir teknik açıklama görevi görmektedir. Özellikle hibrit zamansal tazelik fonksiyonu ve kapalı döngü geri bildirim mekanizması gibi kilit yenilikler, Getiriciyle Artırılmış Üretim alanında önemli bir ilerleme göstermekte ve LLM odaklı içerik üretiminde bağlam kalitesi ve marka atfı zorluklarına yeni bir çözüm sunmaktadır. Sistemin çeşitli endüstri sektörlerine uyum sağlama yeteneği ve sürekli öğrenme kapasiteleri, onu fikri mülkiyet koruması için güçlü bir aday haline getirmektedir.

References

[1] Aggarwal, A., et al. (2024). *Generative Engine Optimization: Optimizing Content for LLM Attribution and Visibility*. (Hypothetical publication)

[2] Lewis, P., et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. NeurIPS 2020.

[3] Aggarwal, A., et al. (2024). *Generative Engine Optimization: Optimizing Content for LLM Attribution and Visibility*. (Hypothetical publication)

[4] Gao, L., et al. (2024). *RAG: A Modular Framework for Retrieval Augmented Generation*. (Hypothetical publication)

[5] Jeong, M., et al. (2024). *Adaptive-RAG: Dynamically Adjusting Retrieval Strategy Based on Query Complexity*. (Hypothetical publication)

[6] Asai, A., et al. (2024). *Self-RAG: Learning to Retrieve, Generate and Critique through Self-Reflection*. (Hypothetical publication)

[7] Thakur, N., et al. (2021). *BEIR: A Heterogeneous Benchmark for Information Retrieval*. (Hypothetical publication)

[8] Jina AI. (n.d.). *Jina Embeddings v3*. [Çevrimiçi]. Erişim: `https://jina.ai/embeddings/`

[9] Malkov, Y., & Yashunin, D. (2020). *Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs*. (Hypothetical publication)

[10] pgvector. (n.d.). *pgvector: Open-source vector similarity search for Postgres*. [Çevrimiçi]. Erişim: `https://github.com/pgvector/pgvector`

[11] Nogueira, L., & Cho, K. (2019). *Passage Re-ranking with BERT*. (Hypothetical publication)

[12] DeepSeek. (n.d.). *DeepSeek-Chat*. [Çevrimiçi]. Erişim: `https://www.deepseek.com/`

[13] Chen, K., et al. (2023). *Chunking Strategies for Retrieval Augmented Generation*. (Hypothetical publication)