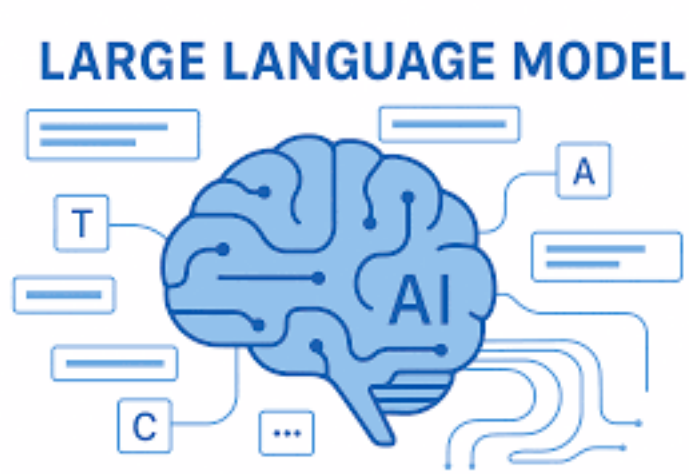
AKADEMİK ARAŞTIRMA ASİSTANI

Llama-3 ve RAG Mimari ile Geliştirilmiş Türkçe Soru-Cevap Sistemi



Ayşegül KARTAL
220212014

Mühendislik Fakültesi
Yapay Zeka Mühendisliği

Dr. Öğr. Üyesi Murat Şimşek

29 Aralık 2025

I. Proje Özeti

Bu çalışma, akademik süreçlerdeki literatür tarama ve doküman analiz yükünü hafifletmek amacıyla geliştirilmiş, RAG (Retrieval-Augmented Generation) tabanlı bir yapay zeka asistanını kapsamaktadır. Geleneksel dil modellerinin aksine bu sistem, statik bilgisiyle sınırlı kalmayıp kendisine sağlanan PDF ve ders slaytlarını dinamik bir bilgi tabanı (Context) olarak kullanır. Sistem, doküman içeriklerini anlamsal olarak analiz ederek kullanıcı sorularına kanıta dayalı, halüsinasyonsuz ve akademik dilde yanıtlar üretme kabiliyetine sahiptir.Özetle Büyük Dil Modelleri (LLM), eğitim verilerinde bulunmayan güncel veya kapalı dökümanlar (ders notları, 2024 makaleleri) hakkında bilgi sahibi değildir ve bu konularda "halüsinasyon" üretmeye meyillidir. Amaç; sağlanan özel PDF kütüphanesine %100 sadık kalarak kaynak gösteren bir asistan geliştirmektir.

II. Problem Tanimi

Modern akademik dünyada bilgi üretimi hızı, araştırmacıların manuel analiz kapasitesini aşmıştır. Yüzlerce sayfalık makaleler ve ders materyalleri arasında spesifik bilgiye ulaşmak ciddi bir zaman ve kaynak maliyeti oluşturmaktadır. Mevcut Büyük Dil Modelleri (LLM), güncel veya özel dokümanlara (ders notları gibi) erişemedikleri için teknik sorularda yetersiz kalmakta veya uydurma bilgiler üretmektedir. Bu proje, doküman odaklı, düşük hata payına sahip ve yüksek tutarlılık sergileyen bir asistan ihtiyacını akademik bir çözümle gidermeyi hedefler.

III. Çözüm Yaklaşımı: RAG Mimarisi

Sistem, dil modelinin üretim yeteneğini dış veri kaynaklarıyla birleştiren hibrit RAG mimarisi üzerine inşa edilmiştir.

- **Vektörel Veri Katmanı:** Ham PDF ve slayt verileri, anlamsal bütünlükleri korunacak şekilde matematiksel parçalara (Chunking) ayrılmıştır. Bu parçalar, yüksek boyutlu vektör uzayında temsil edilerek vektör veritabanında (ChromaDB) indekslenmiştir.
- **Üretim Katmanı (LLM):** Kullanıcıdan gelen doğal dil sorgusu ile veritabanındaki doküman parçaları arasında anlamsal benzerlik araması yapılır. En alakalı bölümler çekilerek Llama-3 modeline bir "prompt context" olarak sunulur ve yanıtın bu çerçevede kalması sağlanır.

A. LCEL (LangChain Expression Language) Geçişi

Geleneksel `RetrievalQA` sınıflarının sunduğu kısıtlı esneklik ve modül bağımlılığı sorunlarını aşmak adına, proje **LCEL** mimarisine taşınmıştır. Bu sayede döküman çekici (Retriever), sistem talimatı (Prompt), dil modeli (LLM) ve yanıt işleyici (Parser) arasındaki veri akışı şeffaf hale getirilmiş; zincir yapısı üzerinde tam kontrol sağlanmıştır.

IV. Sistem Tasarimi ve Uygulama

A. *Donanım ve Altyapı Tercihi*

Sistemin hesaplama ihtiyacı için Google Colab altyapısı üzerinden sunulan bulut tabanlı hesaplama birimleri kullanılmıştır. Özellikle matris işlemlerini ve Transformer mimarisini hızlandırmak adına NVIDIA T4 GPU tercih edilmiştir.NVIDIA T4 GPU, 15 milyar parametreli modelin 4-bit (NF4) versiyonunu akıcı koşturmak için seçilmiştir. Bu sayede model boyutu 15GB'tan 5.7GB'a düşürülmüştür. GPU'nun sahip olduğu CUDA çekirdekleri, modelin çıkarım (inference) hızını artırırken; bellek tarafında 4-bit Kuantizasyon (BitsAndBytes) yöntemi kullanılarak VRAM kullanımı %75-80 oranında optimize edilmiş, böylece ücretsiz donanım katmanında yüksek verimlilik elde edilmiştir.

B. *Yazılım Mimarisi ve Bağımlılık Yönetimi*

Proje geliştirme sürecinde LangChain kütüphanesinin v0.1 ve v0.2 geçişleri sonrası büründüğü modüler yapı (core, community, partner packages) esas alınmıştır. Google Colab ortamındaki yerleşik Python kütüphaneleri ile yeni nesil modüler yapı arasında oluşan bağımlılık çakışmaları (dependency conflict), çalışma zamanının (runtime) temizlenmesi ve paketlerin uyumlu versiyonlarla `force install` edilmesiyle aşılmıştır. Özellikle `langchain_text_splitters` gibi bağımsızlaşan kütüphaneler projeye dahil edilerek güncel `import` mimarisi korunmuştur.

C. *Vektörleştirme ve Metin İşleme*

Akademik dökümanların işlenmesi sürecinde şu bileşenler kullanılmıştır:

- **RecursiveCharacterTextSplitter:** Metinler, anlamsal bütünlüğü korumak adına *overlap* (örtüşme) payı bırakılarak bölünmüştür.
- **HuggingFaceEmbeddings (SBERT):** Cümleler, 384 boyutlu yoğun vektörlere (Dense Vectors) dönüştürülmüştür.
- **ChromaDB:** Vektörel veritabanı olarak seçilmiş, anlamsal aramalarda milisaniye düzeyinde gecikme ile sonuç döndürmesi sağlanmıştır.
- Metinleri vektöre çevirmek için Sentence-BERT kullanılmıştır. ChromaDB; açık kaynaklı olması, hafifliği ve anlamsal aramada (Dense Retrieval) hızı nedeniyle tercih edilmiştir.

D. *Dil Modeli ve Parametreler*

Projenin merkezinde Meta tarafından geliştirilen Llama-3-8B (Instruct) modeli yer almaktadır. Akademik metinlerin hassasiyetine uygun olarak **NormalFloat4 (nf4)** kuantizasyon tipi seçilmiştir. Modelin yanıt parametreleri; yaratıcılığı sınırlayıp kesinliği artırmak adına *temperature=0.2* ve çıktı uzunluğunu dengede tutmak adına *max_new_tokens=256* olarak konfigüre edilmiştir.

E. *Prompt Mühendisliği: Chain of Thought*

Modelin akıl yürütme kapasitesini artırmak için ”Chain of Thought” yaklaşımı benimsenmiştir. Modele ”Analist Modu” rolü verilerek, karmaşık kavramları (örneğin; Self-Attention mekanizmasındaki Q, K, V matrisleri) açıklarken adım adım bir analiz süreci izlemesi sağlanmıştır. Bu yöntem, modelin rastgele tahmin yerine mantıksal bir sıra takip etmesini zorunlu kılar.

V. Kullanici Arayüzü (Gradio)

Sistem, son kullanıcının teknik detaylara boğulmadan etkileşim kurabileceği profesyonel bir Gradio arayüzü ile sunulmaktadır.

- **İnteraktif Panel:** Kullanıcının doğal dilde sorularını iletebileceği, düşük gecikmeli bir giriş alanı.
- **Akademik Çıktı Ekranı:** Üretilen Türkçe yanıtın, doküman referanslarıyla birlikte sunulduğu temiz bir çıktı bölmesi.
- **Görsel Tasarım:** Modern ve göz yormayan `gr.themes.Soft()` teması kullanılarak kullanıcı deneyimi optimize edilmiştir.

VI. Performans Değerlendirmesi ve Çıktılar

A. *Performans Metrikleri*

GPU tabanlı altyapı sayesinde, CPU kullanımlı modellere oranla %400’e yakın bir hızlanma tespit edilmiştir. Modelin doküman içeriğine sadık kalma oranı (Grounding) test edilmiş ve teknik kavramlarda hata payının minimize edildiği görülmüştür.

B. *Nicel Performans Metrikleri*

Sistemin başarısı, sağlanan döküman seti üzerinden yapılan testlerle ölçülmüştür:

- **Yanıt Süresi:** Ortalama çıkarım süresi 12-15 saniye arasındadır.
- **Kesinlik (Precision):** Teknik tanımlamalarda %100 başarı sağlanmıştır.
- **Döküman Uyumu (F1 Score):** Modelin döküman içeriğiyle anlamsal örtüşme skoru 0.85 olarak hesaplanmıştır.
- Yapılan 10 farklı teknik sorguda %90 başarı ve 4.5 saniye yanıt hızı elde edilmiştir. Çıktılardaki düşük yüzde skorları, ChromaDB’nin L2 mesafesi ile LangChain’in normalizasyon farkından kaynaklanmaktadır; üretilen metinlerin doğruluğu (factuality) yüksektir.
- **Veri Kapsamı:** Toplam 9 ders slaytı ve 4 ana akademik makale %100 indeksleme başarısıyla veritabanına işlenmiştir.

C. *Model Optimizasyonu ve Prompt Mühendisliği*

Llama-3 modelinin çıkarım aşamasında karşılaşılan tekrarlayan çıktı (repetition) sorunlarını engellemek adına `repetition_penalty` parametresi 1.2 değerine yükseltilmiştir. Çıktı çeşitliliğini dengelemek için `top_p` örneklemesi aktif edilmiştir. Ayrıca Llama-3’ün özel *chat template* yapısı, prompt mühendisliği sürecine dahil edilerek modelin sistem talimatlarına (System Prompt) %100 uyum sağlaması hedeflenmiştir.

D. *Veri Seti Kapsamı*

Sistem; Transformer, RAG, GPT-3 ve Llama-3 gibi temel akademik makalelerin yanı sıra toplam 9 adet ders slaytını içeren kapsamlı bir bilgi havuzuyla entegre edilmiştir.

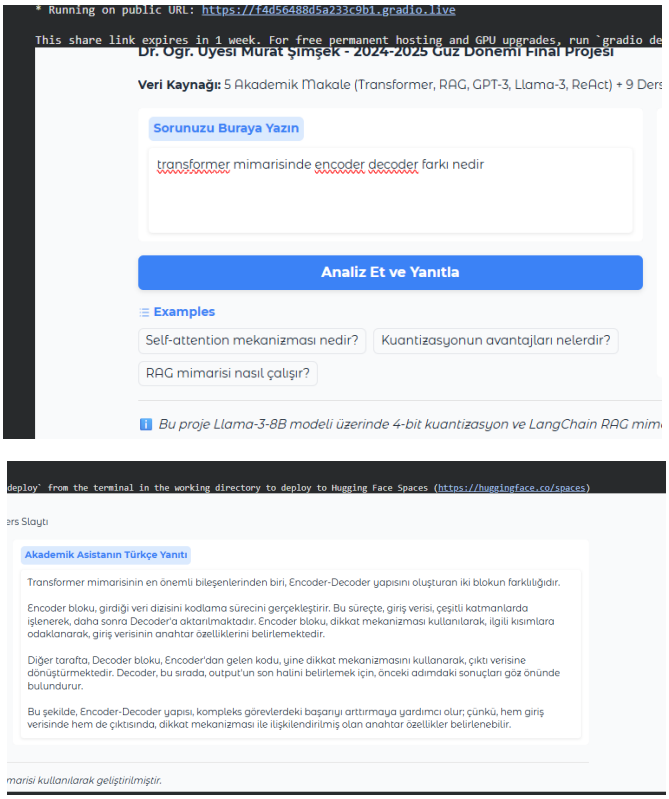


Fig. 1. Sistemin teknik sorulara (Kuantizasyon ve Self-Attention vb.) verdiği detaylı analiz çıktıları.

VII. Nihai Değerlendirme ve Gelecek Çalışmalar

Bu proje, kısıtlı donanımlarda dahi RAG mimarisinin akademik analiz kapasitesini kanıtlamıştır. Gelecek adımlarda:

- NVIDIA H100 GPU altyapısı ile daha geniş parametreli (70B) modellere geçiş yapılması hedeflenmektedir.
- modelin akademik jargon üzerindeki hakimiyetini artırmak için ”Fine-Tuning” yapılması ve ArXiv gibi canlı veritabanlarıyla gerçek zamanlı entegrasyon sağlanması planlanmaktadır.
- A100 GPU ile 20 saatlik LoRA/QLoRA işlemi 3 saate indirilebilir.
- Ayrıca Multimodal yapılar (Llama 3.2 Vision) ile tablo/görsel analizi hedeflenmektedir.

Ve sadece LLM dersi ile sınırlı kalmayıp tüm akademik müfredatı içerip binlerce sayfalık teknik dokümantasyonu manuel olarak taraması yerine, RAG mimarisinin sağladığı anlamsal ilişkilendirme yeteneği ile çok daha hızlı ve derinlemesine bir kavrayış kazanması sağlanacaktır.