

# LangChain

LangChain kütüphanesi, yazılım geliştiricilerin yapay zeka (AI) ve makine öğrenimi (ML) alanında büyük dil modellerini (LLM) diğer harici bileşenlerle birleştirerek LLM tabanlı uygulamalar geliştirmelerine olanak tanıyan açık kaynaklı bir çerçevedir. 2022 yılında Harrison Chase ve Ankush Gola tarafından başlatılan bu proje, güçlü LLM'leri (GPT-3.5, GPT-4, Llama gibi) çeşitli harici veri kaynaklarına bağlayarak doğal dil işleme uygulamalarının avantajlarından yararlanmayı amaçlamaktadır. LangChain, Python ve JavaScript dillerinde kullanılabilir.

## LangChain'in Temel Bileşenleri

**Model I/O:** Dil modelleriyle veri alışverişini yönetir, model giriş ve çıkışlarını formatlar.

**Prompts:** Dil modeline verilen girdilerin formatlanmasını sağlar, doğru ve etkili yanıtlar üretilmesine yardımcı olur.

**Chat Modeller ve LLM'ler:** Sohbet tabanlı modeller ve metin tabanlı giriş ve çıkışlarla çalışan dil modelleri için arayüz sağlar. LangChain, OpenAI, Cohere, Hugging Face gibi birçok model sağlayıcıyla entegrasyona sahiptir ve bu modellerle etkileşim kurmak için standart bir arayüz sunar.

**Retrieval ve Document Loaders:** Belirli uygulamalara özgü verileri almak ve işlemek için kullanılan bileşenlerdir. Belge yükleyicileri, verileri kaynaktan alıp daha sonraki işlemler için hazır hale getirir.

**Text Splitters ve Embedding Models:** Metinleri daha yönetilebilir parçalara böler ve metinlerin anlamsal temsillerini oluşturur. Doğal dil aramaları ve diğer uygulamalar için önemlidir.

**Vectorstores ve Retrievers:** Vektör depoları, yapılandırılmamış veriler üzerinde doğal dil araması yapabilen özel veri tabanlarıdır. Retrieverlar, yapılandırılmamış sorgulara uygun belgeler döndüren genel arayüzlerdir.

**Composition ve Tools:** Farklı sistemleri ve LangChain bileşenlerini birleştirir. Dil modelinin harici sistemlerle etkileşime girmesini sağlar.

**Agents ve Chain:** Araçları kullanarak yüksek düzeyde direktifler verir ve bu direktiflere uygun eylemleri belirler. Zincirler çeşitli bileşenlerin bir araya getirilmesiyle oluşturulan işlem akışlarıdır.

**Memory ve Callbacks:** Bellek bir zincirin çalışmaları arasında uygulama durumunu korur. Geri çağrılar ise zincirin ara adımlarını günlüğe kaydeder ve yayınlar.

## LangChain'in Faydaları

### 1. Dil Modellerini Yeniden Kullanma

LangChain, mevcut dil modellerini yeniden kullanmayı kolaylaştırarak geliştirme süreçlerini hızlandırır. Bu sayede yazılım ekipleri önceden eğitilmiş modelleri entegre ederek daha hızlı ve verimli bir şekilde projelerini hayata geçirebilir.

### 2. Yapay Zekâ Geliştirmeyi Basitleştirme

LangChain, veri kaynağı entegrasyonlarının karmaşıklığını soyutlayarak yapay zekâ (AI) geliştirmeyi basitleştirir. LangChain'in sağladığı şablonlar ve kütüphaneler, yazılım ekiplerinin iş mantığını programlamadan geliştirme süresini azaltmalarını sağlar.

### 3. Geliştirici Desteği

Açık kaynaklı ve aktif bir topluluk tarafından desteklenmektedir. Kuruluşlar, LangChain'i ücretsiz kullanabilir ve bu konuda yetkin diğer geliştiricilerden destek alabilirler.

## Agents Kullanımı ve Proje Fikirleri

LangChain'deki agents, karmaşık uygulamalar için güçlü ve esnek bir yapı sunar. Bir agent, belirli bir amaca ulaşmak için hangi araçların ve adımların kullanılacağını belirler.

### 1. Akıllı Müşteri Destek Sistemi

Bir müşteri destek sistemi oluşturmak için agents kullanılabilir. Sistem gelen müşteri sorularını anlamak ve en uygun yanıtları vermek için çeşitli araçları ve veri kaynaklarını kullanır. Örneğin:

Soruları Anlama: Girdi metnini analiz eder ve anlamını çıkarır.

Bilgi Alma: Sorunun yanıtını dahili belgelerden veya dış kaynaklardan alır.

Yanıt Üretme: Uygun yanıt üretir ve kullanıcıya sunar.

### 2. Çok Dilli Çeviri ve Bilgi Sağlama

Bir çeviri ve bilgi sağlama sistemi oluşturmak için agents kullanılabilir. Bu sistem kullanıcıların sorgularını birden çok dile çevirir ve gerekli bilgileri sağlar. Örneğin:

Dil Algılama: Kullanıcının girdiği dil belirlenir.

Çeviri: Metin hedef dile çevrilir.

Bilgi Sağlama: Çevrilen metin üzerinden bilgi alınır ve yanıt hazırlanır.

### 3. Otomatik İçerik Oluşturma

Bir içerik oluşturma sistemi kurmak için agents kullanılabilir. Belirli konular hakkında makaleler, blog yazıları veya raporlar oluşturabilir. Örneğin:

Konu Belirleme: Kullanıcıdan veya bir kaynaktan alınan konuyu belirler.

İçerik Araştırma: Konuyla ilgili bilgileri toplar.

İçerik Yazma: Toplanan bilgileri düzenleyerek özgün bir metin oluşturur.

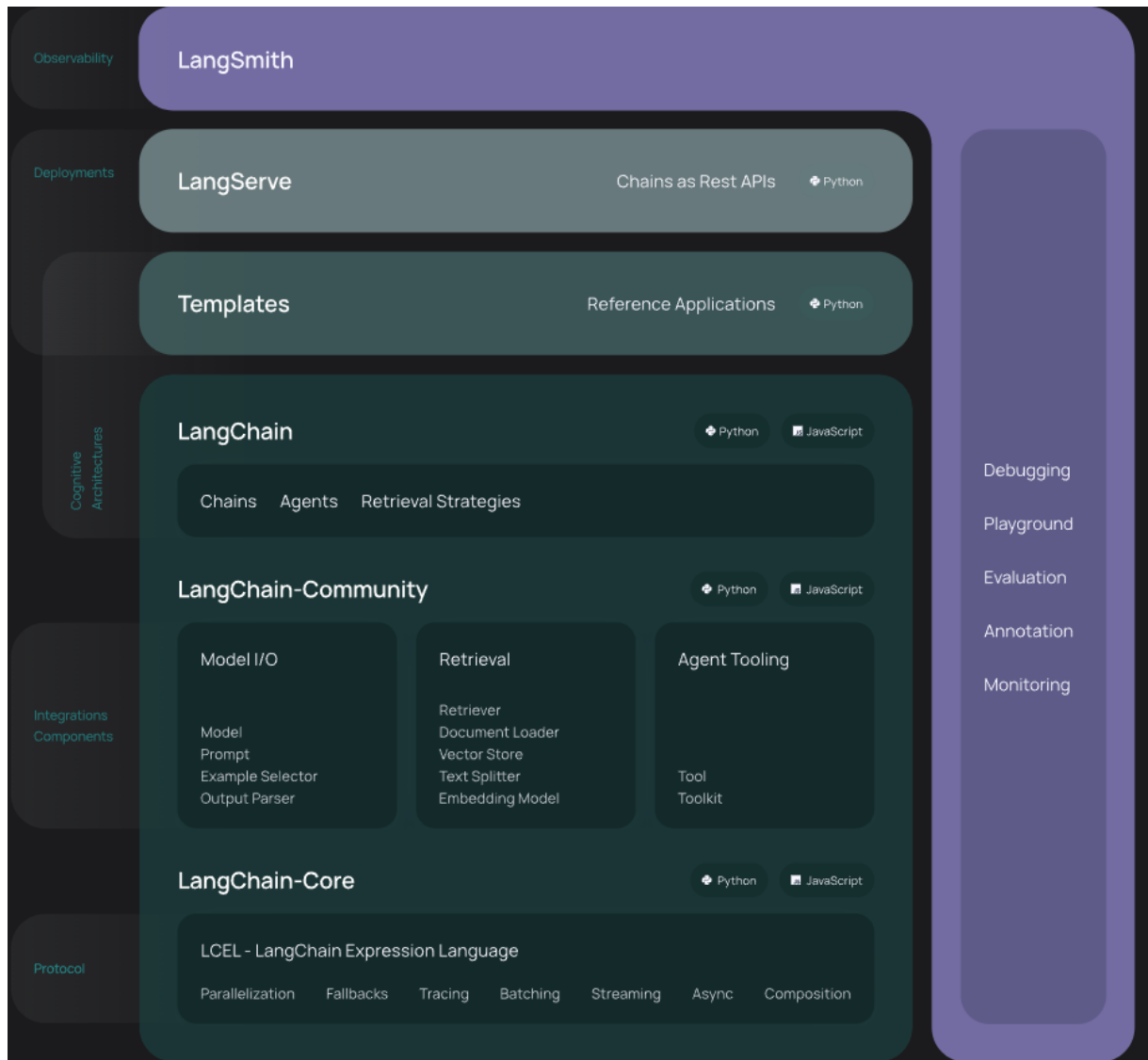
#### 4. Akıllı Veri Analizi ve Raporlama

Veri analizi ve raporlama süreçlerini otomatikleştirmek için agents kullanılabilir. Bu sistem verileri analiz eder ve belirli formatlarda raporlar oluşturur. Örneğin:

Veri Toplama: Farklı kaynaklardan verileri toplar.

Veri Analizi: Verileri analiz eder ve anlamlı sonuçlar çıkarır.

Raporlama: Analiz sonuçlarını rapor formatında sunar.



Şekil 1. LangChain mimarisi ve temel bileşenleri.

<https://github.com/langchain-ai/langchain>

<https://python.langchain.com/v0.2/docs/introduction/>