Büyük Dil Modelleri (LLMs - **Large Language Models**), yapay zeka (AI) ve doğal dil işleme (NLP) alanlarında kullanılan devasa ölçekte eğitilmiş yapay sinir ağı modelleridir. Bu modeller, insan dilini anlamak, üretmek ve simüle etmek amacıyla geniş metin veri setlerinde eğitilir. LLM'ler, özellikle dilsel görevlerde etkileyici sonuçlar elde etmek için milyonlarca veya milyarlarca parametreye sahip olabilir.

**Transformers Mimarisi**

LLM'lerin temelinde, 2017'de Google araştırmacıları tarafından önerilen **Transformer** adı verilen bir mimari yatmaktadır. Transformer, doğal dil işlemede büyük bir devrim yarattı ve günümüzdeki büyük dil modellerinin omurgasını oluşturdu.

**Encoder**: Girdi cümleleri veya paragrafları alarak, bunları daha düşük boyutlu bir vektör temsiline çevirir.

**Decoder**: Metin üretimi gibi görevlerde kullanılır ve önceki girdilere dayalı olarak yeni kelimeler tahmin eder.

LLM'ler genellikle bir dil modelleme görevi üzerine eğitilirler. Bu süreçte, model bir metnin devamını tahmin etmeyi öğrenir. Yani, modelin bir dizi kelime verildiğinde, sıradaki kelimeyi veya kelime dizisini tahmin etmesi gerekir.

**Transfer Öğrenme**: LLM'ler, geniş bir dil modeli olarak eğitildikten sonra, daha spesifik görevler için ince ayar yapılabilir.

Popüler LLM'ler

**GPT Serisi (Generative Pretrained Transformer)**

* OpenAI tarafından geliştirilen GPT serisi, LLM'ler arasında en bilinen modellerdendir. GPT-3 ve GPT-4 gibi sürümler, milyarlarca parametreye sahip olup, geniş çapta dil üretiminde başarılıdır.
* GPT modelleri, önceden eğitilmiş olup, çeşitli görevlerde etkili sonuçlar verecek şekilde ince ayar yapılabilir.

LLM'ler, birçok farklı alanda kullanılabilir:

**Metin Üretimi**- Sohbet Robotları (Chatbots)- **Dil Çevirisi**-Özetleme- Programlama desteği- Soru cevaplama

**LLama (Large Language Model Meta AI)**

**LLama**, Meta (eski adıyla Facebook) tarafından geliştirilen büyük bir dil modelidir. LLama, daha düşük maliyetli bir alternatif olarak, akademik araştırmalar ve dil işleme toplulukları için geliştirilmiş açık kaynaklı bir dil modelidir. LLama, düşük kaynaklarda bile güçlü performans sergileyebilmesi için optimize edilmiştir.

hem daha küçük cihazlarda hem de büyük sunucularda çalışabilecek esneklik sağlar.  **Açık Kaynak**: Meta, LLama'yı akademik kullanıma açmıştır, bu da araştırmacıların ve geliştiricilerin modeli inceleyip kendi ihtiyaçlarına göre özelleştirmelerine olanak tanır.

 **Verimlilik**: LLama, özellikle küçük ve orta ölçekli modellerinde diğer büyük dil modellerine kıyasla daha az donanım kaynağı kullanırken güçlü sonuçlar elde edebilir.

**PaLM (Pathways Language Model)**

**PaLM** (Google tarafından geliştirilen Pathways Language Model), Google’ın Pathways altyapısına dayanan büyük dil modelidir. Google, PaLM ile dil işleme ve diğer genel yapay zeka görevlerinde daha geniş çapta başarı elde etmeyi amaçlamaktadır. PaLM, çok büyük veri setleri üzerinde eğitilen bir model olup, karmaşık dil görevlerinde son derece güçlü performans gösterir.

PaLM, Google’ın Pathways sistemine dayanır, yani birden fazla görev üzerinde öğrenme yeteneğine sahiptir ve bu görevlerde daha verimli hale gelir.

**GPT-4**, OpenAI tarafından geliştirilen en son nesil dil modelidir. GPT-4, önceki sürümlerine kıyasla daha büyük, daha güçlü ve çok daha karmaşık dilsel görevlerde olağanüstü performans gösteren bir modeldir.

GPT-4, bağlamları çok daha derinlemesine anlar ve uzun metinlerde bile doğru tahminler yapar. Ayrıca, karmaşık diyaloglarda ve çok adımlı süreçlerde daha doğru sonuçlar üretir.

**LLama**, daha küçük boyutlu ve verimli olduğu için mikroişlemcilerde kullanılabilir.

**PaLM**, geniş görev kapsamına sahip olsa da mikroişlemcilerde doğrudan çalıştırılması zordur, ancak bulut tabanlı entegrasyonlarla kullanılabilir.

**GPT-4**, güçlü dil işleme yetenekleriyle öne çıkar, ancak büyük boyutlu olduğu için mikroişlemcilerde çalıştırılması zor olsa da, bulut servisleri ile entegre edilebilir.

\*\*\*\*\*

Bir sistem için **en uygun büyük dil modelini (LLM) seçmek**, sistemin donanım kaynakları, uygulama senaryosu ve performans gereksinimlerine dayalı bir değerlendirme sürecini gerektirir.

**Uygulama Senaryosu**

Öncelikle LLM'nin hangi amaca hizmet edeceğini belirlemek önemlidir. Farklı senaryolar için farklı LLM'ler daha uygun olabilir:

* **Metin Üretimi**: Eğer model, metin üretimi, yazı tamamlama veya içerik oluşturma gibi görevler için kullanılacaksa, dil üretim yeteneklerine sahip ve geniş bağlamları anlayabilen modeller gereklidir (örneğin, **GPT-4**).
* **Doğal Dil Anlama**: Eğer sistem doğal dil anlama, metin sınıflandırma, duygusal analiz gibi görevler üzerine odaklanıyorsa, dil anlama yetenekleri güçlü modeller seçilmelidir (örneğin, **LLama**).
* **Çoklu Görev Performansı**: Modelin birden fazla görevde kullanılacağı durumlarda (örneğin, dil anlama, özetleme, çeviri), **PaLM** gibi çok yönlü ve çoklu görev performansını optimize eden modeller tercih edilebilir.

**Donanım Kaynakları ve Performans İhtiyaçları**

Donanım kaynakları, hangi LLM'nin kullanılabileceğini belirler. LLM'lerin performansı doğrudan işlem gücü, bellek ve enerji tüketimi gibi donanım sınırlamalarıyla ilişkilidir:

* **Bellek Kullanımı**: Büyük dil modelleri, genellikle çok fazla bellek kullanır. Eğer sisteminizin bellek kapasitesi sınırlıysa, küçük veya orta ölçekli modeller tercih edilmelidir (örneğin, **LLama 7B**).
* **İşlem Gücü**: Mikroişlemciler veya gömülü sistemler gibi sınırlı işlem gücüne sahip donanımlarda, daha küçük ve optimize edilmiş modeller seçilmelidir. Eğer güçlü bir sunucu veya GPU gibi yüksek işlem kapasiteniz varsa, daha büyük modeller kullanılabilir.
* **Enerji Tüketimi**: Düşük enerji tüketimine sahip cihazlar, verimli modelleri tercih etmelidir. Mobil cihazlar veya IoT sistemlerinde daha az güç tüketen modeller (örneğin, **LLama**) uygun olabilir.

**Model Boyutu**

Modelin parametre boyutu, hem işlemci kapasitesini hem de performansı etkileyen en kritik faktörlerden biridir. Genellikle model ne kadar büyükse, o kadar yüksek doğruluk ve dil işleme kapasitesi sunar, ancak daha fazla kaynak gerektirir. Uygulama ve sisteminizin sınırlarına göre bu boyut belirlenmelidir:

* **Küçük Modeller (7B - 13B Parametreler)**: LLama gibi küçük modeller, sınırlı kaynaklar için uygundur ve çoğu dil işleme görevini kabul edilebilir doğrulukla yerine getirir.
* **Orta Boy Modeller (30B - 65B Parametreler)**: Daha fazla işlem kapasitesine sahip sistemlerde, orta boyutlu modeller daha yüksek doğruluk ve performans sunabilir.
* **Büyük Modeller (100B+ Parametreler)**: Eğer çok güçlü sunucularınız veya GPU'larınız varsa, GPT-4 veya PaLM gibi devasa modeller kullanılabilir. Bu modeller çok güçlü performans sağlar, ancak yüksek donanım gereksinimlerine ihtiyaç duyar.

**Gecikme (Latency) ve Yanıt Süresi**

Eğer LLM, gerçek zamanlı uygulamalarda kullanılacaksa (örneğin, sohbet botları, sesli komut işleme), yanıt süresi çok önemlidir:

* **Gerçek Zamanlı Uygulamalar**: Gecikme süresi düşük olmalıdır. Bu nedenle daha küçük ve daha hızlı modeller kullanılmalı, modelin optimize edilmiş ve hızlı yanıt verebilmesi sağlanmalıdır.
* **Yüksek Hesaplama Gücü Gerektiren Görevler**: Daha fazla hesaplama gücü gerektiren görevlerde daha büyük modeller seçilebilir, ancak bu durumda yanıt süresi uzayabilir.

**Veri Miktarı ve Kalitesi**

Modelin nasıl eğitildiği ve veri setlerinin büyüklüğü, belirli dil görevlerinde nasıl performans göstereceğini etkiler. Eğer çok dilde performans veya spesifik görevlerde yüksek başarı arıyorsanız, bu alanlarda eğitilmiş modelleri seçmek önemlidir:

* **Genel Dil İşleme**: GPT-4 gibi modeller çok geniş veri kümeleri üzerinde eğitilmiştir ve genelleştirilebilir performans sağlar.
* **Özel Alan Uygulamaları**: Eğer model tıbbi, hukuki ya da teknik bir alanda kullanılacaksa, bu alanda eğitim almış bir model daha iyi sonuçlar verebilir. Belirli alanlarda özelleştirilmiş bir model seçmek daha uygun olacaktır.

**Modelin Özelleştirilebilmesi**

Kimi zaman, uygulamanız için dil modelini özelleştirmeniz gerekebilir:

* **İnce Ayar (Fine-Tuning)**: Eğer dil modeli belirli bir alan veya uygulamaya göre ayarlanacaksa, bu ince ayar yapabilme özelliğine sahip bir model seçilmelidir. LLama gibi açık kaynaklı modeller, bu noktada avantaj sağlar.
* **API Tabanlı Kullanım**: Eğer modelin eğitilmesi veya özelleştirilmesi mümkün değilse, GPT-4 gibi API tabanlı çözümlerle dış kaynaklı kullanımlar tercih edilebilir.

**Gizlilik ve Güvenlik**

Modelin kullanıldığı senaryoya bağlı olarak gizlilik ve güvenlik gereksinimleri de değerlendirilmelidir:

* **Yerel Çalışma (On-Premise)**: Eğer modelin yerel sistemlerde çalışması gerekiyorsa (örneğin, hassas veri işlemesi için), açık kaynaklı ve yerel olarak çalıştırılabilen modeller (örneğin, LLama) tercih edilebilir.
* **Bulut Tabanlı Kullanım**: Gizlilik gereksinimlerinin daha düşük olduğu durumlarda, GPT-4 veya PaLM gibi bulut tabanlı modeller kullanılabilir. Ancak bu durumda verilerin bulutta işlenmesi gerektiği unutulmamalıdır.

**Maliyet**

Modellerin kullanımı ve barındırılması, maliyet açısından da önemli bir faktördür:

* **Bulut Maliyetleri**: Eğer büyük modeller bulut tabanlı olarak kullanılacaksa, bu hizmetlerin maliyetini göz önünde bulundurmalısınız. Daha büyük ve güçlü modellerin kullanım maliyeti daha yüksek olabilir.
* **Yerel Barındırma Maliyetleri**: Yerel sistemlerde çalıştırılacak modeller, özellikle büyük modeller için yüksek donanım maliyetleri doğurabilir. Küçük veya optimize edilmiş modeller, bu maliyeti düşürebilir.

**Prompt engineering** (istek mühendisliği), büyük dil modelleri (LLM'ler) ile çalışırken doğru ve etkili yanıtlar almak için giriş cümlelerini (prompts) dikkatle tasarlama sürecidir.

**Fine-tuning** (ince ayar), önceden eğitilmiş bir makine öğrenimi modelinin, belirli bir uygulama ya da görev için daha iyi performans göstermesi amacıyla yeniden eğitilmesi işlemidir. Bu yöntem, özellikle büyük dil modelleri (LLM'ler) gibi geniş bilgi tabanlarına sahip modellerin belirli alanlarda özelleştirilmesi ve daha odaklanmış sonuçlar üretmesi için kullanılır. Fine-tuning, derin öğrenme ve dil modellerinin, özellikle doğal dil işleme (NLP) alanında etkili bir şekilde uygulanabilirliğini artırır.

**Low-Rank Adaptation (LoRA)**, büyük dil modellerinin (LLM) ince ayar sürecini optimize etmek için kullanılan bir yöntemdir. LoRA, özellikle büyük ve karmaşık modellerin, daha düşük maliyetle ve daha az veri ile özelleştirilmesine olanak tanır.

**Retrieval-Augmented Generation (RAG)**, bilgi alma (retrieval) ve metin oluşturma (generation) süreçlerini birleştiren bir doğal dil işleme (NLP) yaklaşımıdır. RAG, özellikle büyük dil modellerinin (LLM'ler) daha doğru ve bağlama uygun metinler oluşturmasına olanak tanır. Bu yöntem, modelin eğitiminde kullanılan geniş veri kümesine ek olarak, gerçek zamanlı olarak dış kaynaklardan bilgi alarak metin üretimini iyileştirmeyi amaçlar.

**Bilgi Alma (Retrieval) Aşaması**: Bu aşamada, model belirli bir sorguya karşılık gelen ilgili bilgiyi veya belgeleri dış veri kaynaklarından alır

**Metin Oluşturma (Generation) Aşaması**: Alınan bilgiler, bir dil modeli tarafından kullanılarak bağlam içerisinde uygun ve anlamlı bir metin oluşturulmasına yardımcı olur.