



LOTUS AI

***AI – AGENT’LER HAKKINDA
ARAŞTIRMA PROJESİ***

Hazırlayan : Kübra Nur BABACAN

İÇİNDEKİLER

AI AGENT	3
AI Agent Nedir ?	3
Temel Özellikleri Nelerdir ?.....	3
AI Agent Türleri Nelerdir ?.....	4
AI Agent İle İş Süreçleri Nasıldır ?.....	5
Uygulama Alanları	6
AI Agentler İle Büyük Dil Modelleri (LLM) Arasında ki Farklar.....	7
AI Agentlerin Çalışma Prensipleri	8
AI Agentlerin Mimarisi	11
Uygulama Adımları.....	13
Multiagent Nedir ?	13
Multiagentlerin Mimarisi	14
Multiagentlerin Sistem Yapıları Nelerdir ?	14
Multiagentlerin Davranışları Nelerdir ?	15
Multiagentlerin Avantajları Nelerdir ?	16
Multiagentlerin Dezavantajları Nelerdir ?	16
Multiagentlerin Kullanım Alanları	17
AI Agent Çerçeveleri	17
AI Agent Arayüzleri	21
Kaynakça	23

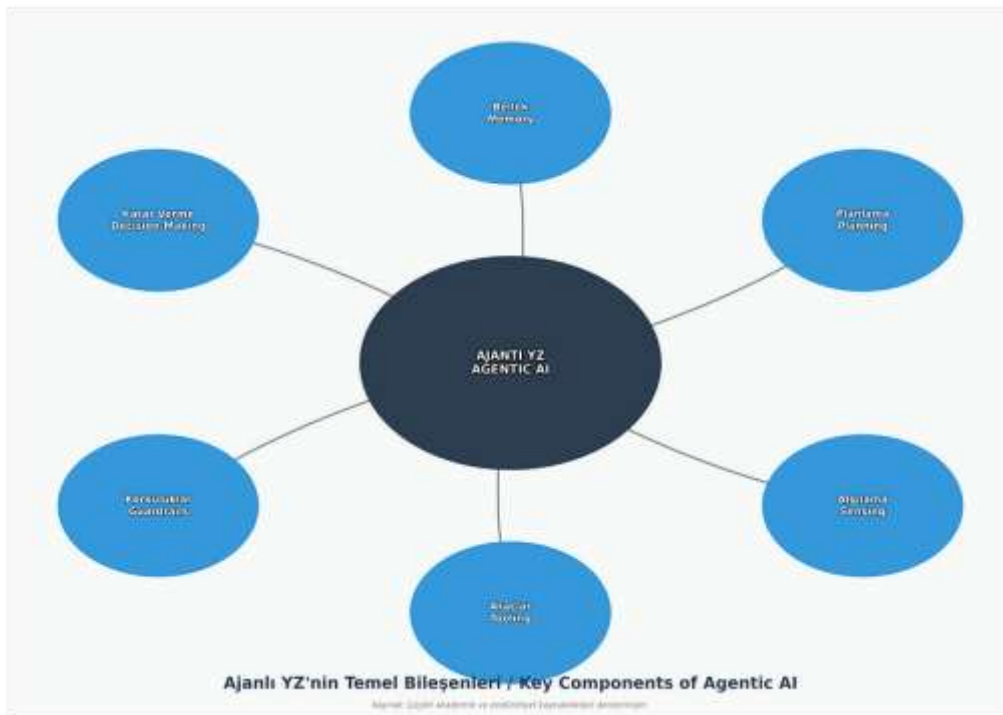
AI AGENT

- **AI Agent Nedir ?**

AI agent , yapay zeka algoritmalarını kullanarak yapılması gereken görevleri ya da işleri otomatik olarak gerçekleştirebilen ve çevreden aldığı bilgilere göre tepki gösteren bir yazılımdır. Agentler, programlanan talimatlar ve algoritmalar ile hareket ederek karmaşık problemleri çözebilmektedirler. Aynı zamanda kullanıcı ile etkileşimde oldukları için veri analizi de yapabilmektedirler.

AI agentler kullanıldıkları alana göre farklılık göstermektedir. Örnek verecek olursak müşteri hizmetlerinde kullanılan AI agent'ten müşterinin sorularını yanıtlamasını ve sorunlarını çözmesi için bir sistem kurulması gerekmektedir. Finans sektöründe ise piyasada ki verilerin analiz edilmesi ile doğru yatırım önerileri için yol gösterilmelidir.

- **Temel Özellikleri Nelerdir ?**



AI agentler , yapay zeka ve makine öğrenmesi teknolojilerinin birleşmesiyle geliştirilmiş sistemlerdir. Temel özelliklerini inceleyecek olursak şu şekildedir :

- ✓ **Algılama Yetenekleri Vardır :**

Çevrelerindeki durumu algılayabilen sensörler ve veri toplama araçları ile gerçek zamanlı verileri sürekli işleyerek yaşanan duruma uygun tepkiler verebilmektedirler.

✓ **Otomatik Karar Verme Yetenekleri Vardır :**

Belirli algoritmalar ile veri analizi teknikleri kullanılarak tanımlanmış kurallar , öğrenilmiş modeller ve tahmin algoritmaları ile kararlar alabilmektedirler.

✓ **Öğrenme Kabiliyeti Yetenekleri Vardır :**

Beklenmedik durumlarda veya değişken koşullar olması durumunda makine öğrenmesi teknikleri ile öğrenilmiş veya deneyimli algoritmalarla öğrenerek performansını artırırken yeni duruma da adapte olmalarını sağlamaktadırlar.

✓ **Öz-yönetim Yetenekleri Vardır :**

İnsan müdahalesinden en az etkileşim ile bağımsız çalışırken iş süreçlerinde bireyden oluşan hatayı azaltarak verimlilik , zaman ve sürekliliği artırmaktadırlar.

✓ **Esneklik Yetenekleri Vardır :**

Çeşitli iş süreçlerini yönetmek ve farklı türlerde ki problemleri çözmek için AI agentlerden yararlanılırken esneklik yeteneği kullanım yönünden çok yönlülük ve geniş bir uygulama haline getirmektedir.

• **AI Agent Türleri Nelerdir ?**

Verimlilik artışı ve karar verme süreçlerinin iyileştirilmesi gibi kritik alanlarda önemli faydalar sağlayan AI agentlerin türlerini inceleyecek olursak şu şekildedir :

1. Reaktif AI Agentler

Çevresel uyarıcılarla doğrudan etkileşime girerek sınırlı bir şekilde tepki vermektedirler. Önceden tanımlanmış senaryolarla karmaşık düşünme süreçlerinin dışında anlık veriler ile çalışmaktadırlar. Örnek verecek olursak belirli bir sensörden gelen sinyal ile çalışan makine durdurulabilmektedir.

2. Proaktif AI Agentler

Bağımsız hedefler doğrultusunda hareket etmektedirler. Aynı zamanda kullanıcının ihtiyaçlarını öngörerek buna uygun kararlar alabilmektedirler. Uzun süreli hedeflerde stratejik planlama yaparak öncelik kullanabilen özellikleri vardır. Örnek verecek olursak müşteri hizmetleri deneyimini analiz etmek için müşteri verilerini analiz edilerek uygun çözümler gerçekleştirilebilmektedirler.

3. Hibrit Al Agentler

Hem reaktif hem de proaktif al agentlerin özelliklerini birlikte taşımaktadırlar. Karmaşık ve değişken ortamlarda etkili çalışmaktadırlar. Gerçek zamanlı veri işlemleri ile acil durumlarda anında yanıt vererek geniş kapsamlı hedeflere ulaşmak için de planlama yapabilmektedirler.

Çeşitli sektör ve ihtiyaçlara göre bu türler avantaj sağlarken uygulandıkları alana göre değerlendirilip uygulanmalıdır.

• **Al Agent İle İş Süreçleri Nasıldır ?**

Çeşitli iyileştirmeler ile kurumların daha verimli , daha hızlı ve etkin çalışmalarını sağlayan agentlerin iş süreçlerini geliştirme yolları :

- ✚ Tekrar eden veya rutinleşen işler için iş süreçlerini otomatikleştirerek iş sürecinin hız kazanmasını sağlamaktırlar. Bu ise iş performansında iş yükünün azalmasını ve önemli bir artışa sebep olabilmektedir. Yani **verimlilik ve performans artışı** sağlamaktadır.

Örnek verecek olursak veri giriş işlemi zaman alıcı bir işlemdir. Aynı zamanda çok veri olduğu zaman veri girişi yapan bireyin hata yapması da olabilmektedir. Bu durumda al agent ile bu durum otomatik olarak gerçekleştirilerek hata önlenmiş , zamandan tasarruf edilmiş olur ve verinin analizi üzerinde daha önemli kararlar alınabilir.

- ✚ Yapılan işler bazen uzun zaman alabilmektedir. İşlemleri daha kısa sürede hatasız şekilde yürütmek **zaman ve maliyet tasarrufu** sağlamaktadır. Bu yüzden insan hatasının oluşabileceği hata riskini ortadan kaldırmak , verimliliği artırmak ve kaliteli hizmet için zaman ve maliyet tasarrufu gereklidir.

- ✚ Geniş veri setlerinde veya karmaşık işlemlerin olduğu durumlarda farklı iş süreçlerini otomatikleştirerek manuel işlemler gerçekleştirilmesi ile görevlerde büyük oranda iyileşme sağlamaktadırlar. Bu sayede **iş süreçlerinde otomasyon** gerçekleştiriyorlar.

- ✚ Agentlerin veri analizi ve öğrenme yetenekleri sayesinde analizler ile doğru kararlar alınmasını sağlamaktadır. Yani **karar alma süreçlerini iyileştirmektedirler**. Risk yönetimi , stratejik planlama ve Pazar analizleri gibi kritik durumlarda iş süreçlerini yönetmede önemlidirler. Agentlerin yaptığı iyileştirmeler veya düzenlemeler ile iş süreçlerinde esneklik sağlanması , uyarlanabilme sağlanması ve maliyeti azaltması ile avantajlı hale gelebilmektedirler.

Uygulama Alanları

✓ Müşteri Hizmetleri

Sıkça sorulan soruları yanıtlayarak müşteri isteklerini işlemek ve destek taleplerini yönetmek için yaygın bir şekilde kullanılmaktadır. Chatbotlar'ın 7/24 destek sağlaması ile müşterilerin memnuniyetini artırmada önemli bir göreve sahiptirler.

✓ Pazarlama ve Satış

Müşterilerin verileri analiz edilerek kişiye uygun pazarlama kampanyaları oluşturabilmektedirler. Aynı zamanda müşteri davranışlarının tahmin edilmeye çalışılması ile satış süreçlerini yönetmek ve çapraz satış fırsatlarını belirlemek için de kullanılabilmektedirler.

✓ İnsan Kaynakları

Adayların taranması , özgeçmişlerinin analiz edilmesi ve en uygun adayları belirlenmesin de büyük oranda kullanılmaktadır. Ayrıca çalışanların performans değerlendirmesi ve kariyer gelişim süreçlerini yönetmek için de tercih edilmektedirler.

✓ Üretim ve Lojistik

Envanter yönetimlerinde , tedarik zinciri optimizasyonlarında ve lojistik planlamalarda kullanılarak süreçleri daha verimli hale getirmektedirler. Ayrıca üretim hatlarında ki arızaları önceden tahmin etmeye çalışarak bakım süreçlerinin iyileştirilmesini sağlayarak üretim sürekliliğini sağlamaktadırlar.

Temel Bileşenleri :

- **Mimari :** Bir AI yazılım aracının mimarisi işlemleri etkinleştirmek için bir metin istemi, API ve veri tabanları olabilmektedir.
- **Aracı İşlevi :** Toplanan verilerin aracının amacını destekleyecek şekilde eylemlere nasıl çevrildiğini açıklamasıdır. Bu aşamada geliştiriciler bilgi türünü, yapay zeka yeteneklerini, bilgi tabanı vb. durumlarla bildirim mekanizmalarını ve gerekli diğer teknolojileri dikkate almaktadırlar.
- **Aracı Programı :** Aracı işlevinin uygulanmasıdır. Yapay zeka aracısını belirlenen şekilde geliştirmeyi, eğitmeyi ve dağıtmayı içermektedir.

AI agentler , kendi başlarına görevleri gerçekleştirebilen, karar alabilen ve belirli hedeflere ulaşmak için hareket edebilen yazılım sistemlerinin geleneksel yazılımlardan farklı olarak katı ve önceden belirlenmiş kuralları takip etmek yerine daha esnek ve adapte olabilen bir yapıya sahip olması onları öne çıkarmaktadır. Büyük dil modellerinin (LLM) teknolojilerini kullanmaktadırlar. Bunlar arasında örnek verecek olursak OpenAI'nin GPT modelleri, Anthropic'in Claude'u ve Google'ın Gemini'si gibi gelişmiş dil anlama sistemleri bunlara örnek verilebilmektedir. Bu modeller ile agentlar bilgiyi işleme ve karar verme yetenekleri kazanarak dönüşler sağlamaktadır.

Geleneksel yazılımlar belirli görevleri yerine getirmek için programlanırken, AI agentlar ise daha geniş hedefleri anlayabilmek ve bu hedeflere ulaşmak için kendi planlarını oluşturabilmektedir. Örneğin, belirli bir kişiye takvim daveti göndermek yerine, bir AI agent ile kullanıcının takvimindeki uygunluğuna göre otomatik bir şekilde toplantı planlanabilmektedir.

AI AGENTLER İLE BÜYÜK DİL MODELLERİ (LLM) ARASINDA Kİ FARKLAR

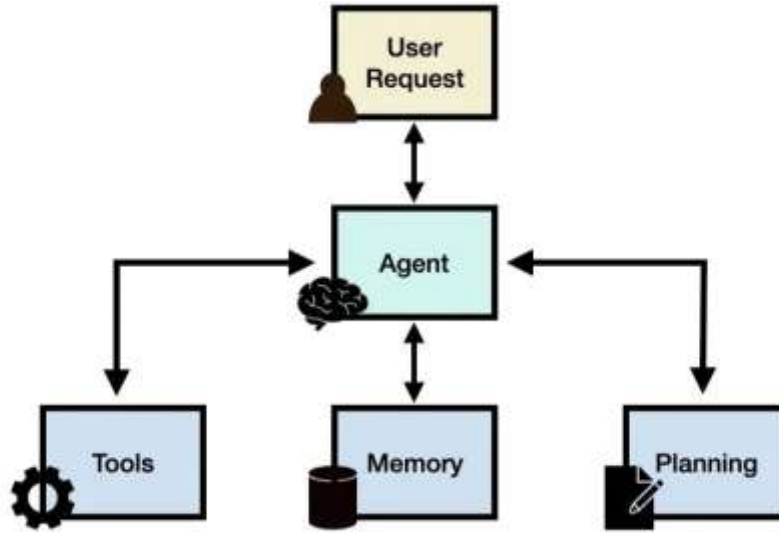
AI agentler ile büyük dil modelleri (LLM) birbirlerinden çok farklıdır. Büyük dil modelleri daha çok eğitim verilerine dayanarak metinler üretirken dünya ya da kullanıcılar ile etkileşime girme ihtimalleri olmadığı gibi güncelleme yetenekleri de olmadıkları için farklıdır.

AI agentler ise buna göre daha kapsamlıdır. Şu şekildedir :

- **Etkileşim** : Dışarıdan etkileşime girebilir ve aynı zamanda bilgiler edinebilmektedirler.
- **Hafıza** : Önceki kurulan etkileşimleri hatırlayarak bunları öğrenebilir aynı zamanda güncelleyebilmektedirler.
- **Aksiyon** : İstenilen duruma yalnızca yanıt vermeyip buna uygun bazı eylemler gerçekleştirebilmektedirler.

- **Adaptasyon** : Değişen her tür koşula uyum sağlayarak çalışabilmektedirler.

AI AGENTLERİN ÇALIŞMA PRENSİBİ



Gelişmiş çözümler olarak işlev gördükleri için planlama – yürütme – eylemlerden öğrenme yeteneklerine sahiptirler. Çalışma prensipleri ise birkaç temel bileşenden oluşarak devam etmektedir. inceleyecek olursak :

- **1 - Planlama / Hedef Belirleme**

Hedefleri tanımlayarak başladıkları için bu hedefleri yönetebilir ve görevlere göre ayrıntılı planlar oluşturabilmektedirler. Bu süreçte “ Düşünce Zinciri (Chain of Thought) ” yaklaşımına benzeyerek problemleri alt görevlere bölünmesini sağlayıp adım adım çözüm yolları oluşturabilmektedirler.

- **2 - Araçlarla Etkileşim**

AI agentler çeşitli araçlarla etkileşime girebilme özellikleri vardır. Bu özellikleri sayesinde internete , veritabanlarına ve API'lere erişim sağlayarak agentlerin bilgi toplama ve gerçekleştirme yeteneklerini artırmaktadır. Örnek araçlar :

- API entegrasyonları

- E – Posta mesajlaşma sistemleri
- Web tarayıcıları
- Dosya sistemleri
- Veritabanı sorguları
- Takvim ve planlama araçları
- CRM ve ERP sistemleri

• 3 - Hafıza ve Bilgi Toplama

Bilgileri depolayarak gerekli zamanlarda bu özel bilgileri kullanabilmektedirler. Bu sayede iş akışlarında kolaylık sağlamaktadır. Örneğin bir şirketin özel veritabanına erişerek buna uygun daha doğru yanıtlar ve yönlelmeler yapabilmektedir.

Hafıza sisteminin uzun vadeli ve kısa vadeli olmak üzere iki şekilde çalışma sistemleri vardır.

- **Kısa Vadeli Hafıza** : Burada mevcut olan bir konuşma ya da görev olan bilgileri saklayarak buna bağlı olarak AI agentin bir anda işleyeceği bilgi miktarlarını belirlemektedir.
- **Uzun Vadeli Hafıza** : Bu hafıza da ise kullanıcı tercihleri , önceki etkileşimler ve öğrenilen bilgileri uzun süre saklayarak korumaktadır.

Kısa vadeli hafıza da context window olarak bilinen bir yapı bulunmaktadır. yapıyı detaylıca inceleyecek olursak ;

Bağlam Penceresi (Context Window)

AI agentlerde etkinliği belirleyen en önemli faktörlerden birisidir. Bir agentin aynı anda işleyebileceği veya hatırlayabileceği maksimum bilgi miktarını ifade etmektedir.

- **Context Window Büyüklüğü** : AI agentlerin bağlam penceresi genellikle GPT-4 gibi modellerde 8K , 16K veya 32K token (yaklaşık 6000 – 24000 kelime) gibi değerlere ulaşabilmektedir. Daha büyük bağlam pencereleri ise agentların daha fazla bilgiyi aynı anda işlenmesine olanak tanımaktadır.
- **Bilgi Erişimin Genişletilmesi** : agentlerin daha uzun olacak şekilde belgelerin işlenmesini , karmaşık olan görevlerin yönetilmesini ve geçmişte ki konuşmanın daha uzun süre olacak şekilde hatırlanmasına imkan sağlamaktadır.

- **Verimli Bilgi Yönetimi** : Eğer ki bağlam pencereleri sınırlı olursa bu durumda verimli bilgi üretebilmek adına verimli bilginin yönetilmesi gerekirken bunun teknikleri de büyük öneme sahiptir. Bunlar arasında ;

I. Özet Teknikler

Uzun konuşmaların özetlenmesini sağlayarak bağlam penceresini verimli kullanmaktadır.

II. Bilgi İndeksleme

Önemli olan bilgilerin daha hızlı şekilde erişilebilir yapılandırılmasıdır.

III. Retrieval – Augmented Generation (RAG)

Harici olan veritabanlarından önemli ve ilgili bilgileri alarak bağlama eklenmesi yeteneğidir.

IV. Vektör Veritabanları

Modern AI agentlerinde vektör veritabanları kullanılmaktadır. Sebebi ise bu sayede şematik aramalar yaparak , bağlam penceresinde aşan bilgileri erişebilmektedir. Bu ise agentin hafızasını bağlam penceresinin ötesine olacak şekilde genişletebilmesidir.

Bağlam penceresi genişliği ne kadar artarsa agentlerin performansı ise daha da iyileşebilmektedir. Çünkü daha fazla geçmiş bilgi birikimini ve kullanıcı etkileşimlerini daha iyi hatırlayabilmektedir. Bu durumun böyle bir iyi yanı varken tabii ki dezavantaj bir durumu da bulunmaktadır. bu ise çok büyük bağlam pençelerinde işlem maliyetini artırırken aynı zamanda yanıt verme sürelerinde gecikme meydana gelebilmektedir. Bu sebeplerde ötürü en ideal bağlam penceresi boyutu agentin kullanım senaryosu ve gerektirdiği hafıza ihtiyaçlarına göre belirlenmelidir.

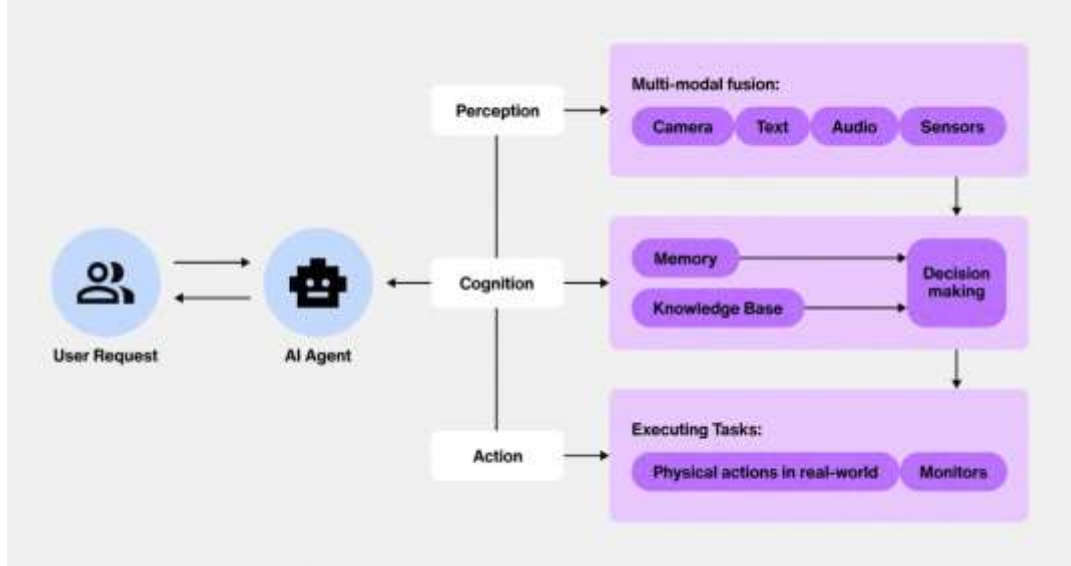
• 4 - Eylem Gerçekleştirme

AI agentlerin birçok görevi bulunurken belirli görevler için eğitilmiş diğer agentlerle da iletişim kurabilmektedirler. Bu ise otomasyon işlemlerinde kolaylık sağlamaktadır. Örnek verecek olursak ,

Bir AI agent kullanıcısı başka bir kişiyle belirli bir tarih ve zaman diliminde toplantı planlaması ayarlatıp toplantı ile ilgili maddeler belirleyip bunu diğer kişiye gönderilmesini talep ederse agent diğer kişinin takvimini kontrol eder ve kişinin

uygunluğunu doğrulayıp toplantı davetini oluşturarak maddeleri de hazırlayarak ilgiye kişiye gönderebilmektedir.

AI AGENTLERİN MİMARİSİ



1) Kullanıcı Giriş Mekanizması

Kullanıcıların agentlerle etkileşime girmesini sağlayan arayüzdür. Bu arayüz metin tabanlı bir sohbet arayüzü , ses komutları veya grafiksel bir kullanıcı arayüzü olabilmektedir.

2) Orkestratör

Agentin beynini oluşturmaktadır. Görev akışlarını yönetirken sistemin farklı parçaları arasında ki koordinasyonu sağlamaktadır. En büyük görevleri ise

Kullanıcı isteklerini almaktadır.

AI modelleri ile planlar oluşturmaktadır.

Oluşturulan plana göre görevlendirmeler yürütülmektedir.

Görevleri ise dinamik bir şekilde yöneterek yeni oluşan görevleri de ele almaktadır.

3) AI Modeli

Agentlerin düşünme mekanizmasıdır. Genellikle GPT-4 , Claude veya Gemini gibi büyük dil modellerini kapsamaktadır. Model kullanıcıların girdilerini anlayarak buna uygun planlar oluşturur ve içerik üretmek için hazırlamalar başlatarak bunları kullanmaktadır. Büyük dil modelleri isteğe bağlı olarak Ollama işlemi kullanılarak self - host edilebilmektedir.

Ollama : Büyük dil modellerinde doğrudan bilgisayarda çalıştırma sürecini basitleştirmek için tasarlanmış bir araçtır. Açık kaynaklıdır. Yerel bir model yöneticisi ve çalışma zamanı olarak hareket ederken model dosyalarını indirirken onlarla etkileşime girilebilecek yerel bir kurulmasına kadar birçok şeyi halledebilmektedir.

4) Araç Entegrasyonları

Agentlerin dış dünya ile etkileşim kurmasını sağlamaktadır.

Dosya sistemleri entegrasyonları

Web hizmetleri

Veritabanı erişimi

API bağlantıları

Diğer yazılım sistemleri

5) Bellek Bileşeni

Agentlerin bilgilerin depolanmasını ve hatırlanmasını sağlayan sistemlerdir.

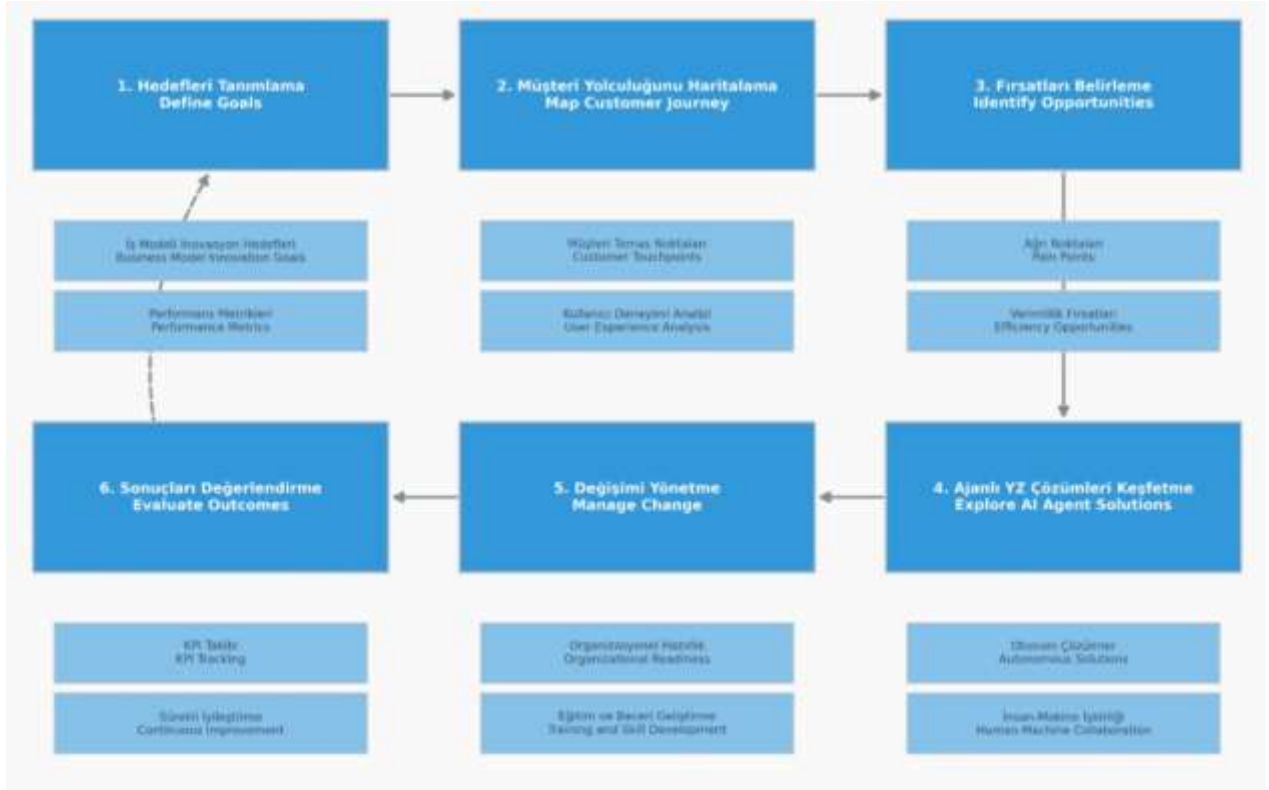
Kısa vadeli bağlam hafızası

Uzun vadeli bağlam hafızası

Bilgi tabanı erişimi

UYGULAMA ADIMLARI

Bir şema ile uygulama adımlarını gösterecek olursak şu şekildedir :



- **Multiagent Nedir ?**

AI agentlerde (tek ajanlı akıllı sistemler) otonom şekilde planlama yapılırken araçları çağırmak ve yanıtlar üretilmesi için çevreleri ile etkileşime girmektedirler. Multiagentlerde ise (çoklu ajan sistemleri) birbirlerinin hedeflerini , hafızalarını ve eylem planlarını modellemek için çevrede ki tüm ajanları da dahil ederek farklılık göstermektedirler. Çoklu ajan sistemlerinde her varlık belli bir dereceye kadar özerk bir ajandır. Bu özerklik genellikle ajanın planlamasını , araç çağırmasını ve genel akıl yürütmesiyle görülmektedir. Çoklu ajan sisteminde ki özerk ajanlar özerk kalırken aynı zamanda ajan yapıları ile işbirliği yaparak daha koordine olmaktadırlar. Karmaşık sorunları çözmek için ajan iletişimindedirler. Bu tür ajan etkileşimi çoklu ajan yani **takviyeli öğrenme** olarak da tanımlanabilmektedir . Bu öğrenme biçimiyle paylaşılan bilgiler , sensörler veya eylemler aracılığıyla edinilen anlık bilgileri içermektedir.

AI agentler kendi başlarına güçlüdür. Alt görevler oluşturup araçları kullanabilecekleri gibi etkileşim yoluyla da öğrenebilmektedirler. Multiagentlerde ise kolektif davranış biçimi ; doğruluk , uyarlanabilirlik ve ölçeklenebilirlik potansiyelini artırmaktadır. Multiagentler daha büyük paylaşılan kaynak havuzu , optimizasyon ve otomasyon nedeniyle tek aracı sistemlerinden daha iyi performans gösterme eğilimindedirler. Birden fazla aracının aynı politikaları öğrenmesi yerine , zaman karmaşıklığını ve verimliliği optimize etmek için öğrenilen deneyimler paylaşabilmektedirler.



- **Merkezi Ağlar :**

Multiagentler çeşitli mimariler altında çalışabilmektedirler. Merkezi ağlarda bulunan merkezi bir birim bilgi tabanlarını içermektedir. Bu bilgi tabanları ajanları birbirlerine bağlayarak bilgileri denetlemektedir. Bunun güçlü yanlarından birisi ise ajanlar ile tekdüze bilgiler arasında oluşan iletişimin kolaylığıdır. Zayıf yanlarından birisi ise merkezi birime bağımlı kalıp başarısız olursa tüm ajan sisteminin de başarısız olmasıdır.

- **Merkezi Olmayan Ağlar :**

Ajanlar küresel bir bilgi tabanları yerine komşu ajanları ile bilgiler paylaşmaktadır. Merkezi olmayan ağın bazı avantajı sağla olması ve modüler olmasıdır. Burada bir ajan başarısız olursa merkezi bir birim olmadığı için genel sisteminde başarısız olmasını engellemektedir. Bir zorluk gerçekleştiği zaman diğer iş birliği yaptığı ajanlar ile fayda sağlayacak şekilde koordine olup çalışmaktadırlar.

Multiagentlerin Sistem Yapıları Nelerdir ?

1. Hiyerarşik Yapı

Ağaca benzeyen bir şekil ile farklı özerklikteki araçları dahil etmektedir. Basit bir yapıda oldukları için bir tane aracı karar alma yetkisine sahip olabilmektedir. Aynı zamanda tekdüze özelliği ile de sorumluluk birden fazla aracıya da dağıtılabilmektedir.

2. Halonik Yapı

Bu mimaride ajanlar holarşiler halinde gruplandırılmaktadır. Burada bir holonun bileşenleri olmazsa çalışamazdır. Örnek verecek olursak insan vücudunda çalışan organlar olmazsa işlev göremez yani bu da bize insan vücudunun bir holon yapıda olduğunu göstermektedir. Aynı zamanda holon multiagentlerde lider ya da tekil gibi görünürken birden fazla alt ajana sahip olabilmektedirler. Bu ajanlar ise diğer holonlarla birlikte rol oynayabilmektedir. Genel olarak bu yapı kendi kendine organize olurken alt ajanların yardımı veya işbirliği ile de istenilen hedefe ulaşılmış olunur.

3. Koalisyon Yapı

Grupta bulunan tekil bir ajanın düşük performans göstermesi sonucunda ajanlar geçici süreliğine birleşerek performansı artırmaktadır. İstenilen performans durumuna ulaşıldığında ise koalisyonlar dağılmaktadır. Bu şekilde çalışarak hedefe faydalı ve doğru şekilde ulaşmış olmaktadır.

Multiagentlerin Davranışları Nelerdir ?

- Multiagentlerde bulunan kolektif davranış insanların davranışına da benzemektedir. Yani “ **akın etme** ” özellikleri vardır. Bu sistemde ajanlar bir hedefi paylaşarak davranışlarını koordineli bir şekilde yönetmektedir. Sürüler halinde hareket ederken yönlerini şu sezgisel yöntemlerle yapmaktadırlar :

Ayrılma : Yakınında bulunan ajanlar ile çarpışmamak için önlem alıyorlar.

Hızalama : Yakınında bulunan ajanın j-hızına uyum sağlamak için bu girişime giriyorlar.

Bağlılık : Diğer etkenler ile yakın kalma çabasıyla işlevlerine devam etmektedirler.

Bunlara en güzel örnek demiryolu sistemleri gibi ulaşım ağlarını yöneten multiagent sistemleridir.

- Multiagentlerde ajanlar mekânsal olarak konumlandırılabilir. Bu durum da “ **sürü halinde** ” yaşamayı örnek göstermektedir. Teknik açıdan sürü halinde olmaları merkezi olmayan kontrole sahip her grubun kendi kendine örgütlenmesini

ve toplanmasını sağlamaktadır. Bunun faydasına değinecek olursak ise bir operatörün bir ajan sürüsünü yönetmek için eğitilebilmesini göstermektedir.

Multiagentlerin Avantajları Nelerdir ?

✓ **Esneklik :**

Ajan ekleme , kaldırma gibi uyarılmalar ile değişen ortama uyum sağlamaktadırlar.

✓ **Ölçeklenebilirlik :**

Birkaç ajanın birlikte bulunarak çalışması daha büyük bir bilgi havuzu oluşmasını sağlamaktadır. Bu işbirliği ile multiagentler tek ajanlı sistemlere göre karmaşık ve zor görevlerin daha kolay çözülmesini sağlarlar.

✓ **Alan uzmanlığı :**

Tek ajanlı sistemlerde çeşitli görevlerde görevi yerine getirmek için tek ajana ihtiyaç duyulurken multiagentlerde her ajan belli bir alan uzmanlığına sahiptir.

✓ **Daha iyi performans :**

Bir ajan için ne kadar çok eylem planı mevcutsa, o kadar çok öğrenme ve düşünmenin gerçekleşmektedir buna bağlı olarak daha iyi performans göstermektedirler.

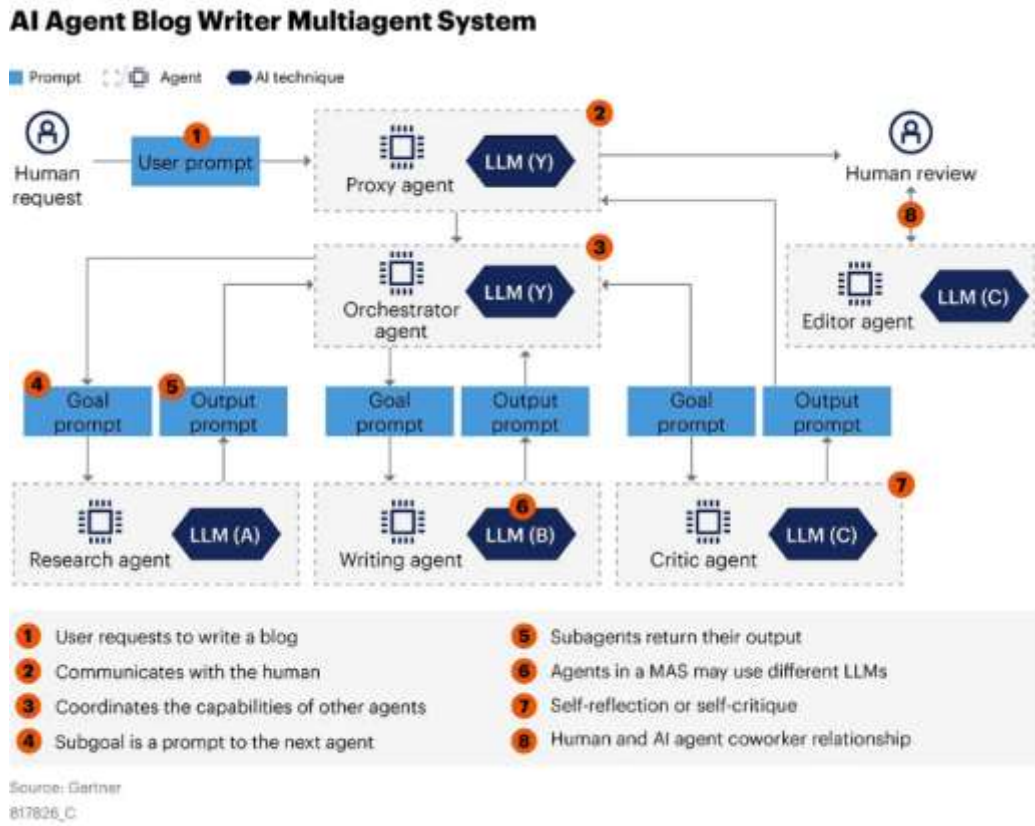
Multiagentlerin Dezavantajları Nelerdir ?

- ✓ Ajan arızaları gerçekleşebilir.
- ✓ Koordinasyon bozukluğu ya da karmaşası yaşanabilir.
- ✓ Öngörülmeven davranışlar gerçekleşebilir.

MULTIAGENTLERİN KULLANIM ALANLARI

- ✚ Toplu taşıma sistemlerinde kullanılmaktadır.
- ✚ Sağlık ve halk sağlığı alanlarında belirli görevlerde kullanılmaktadır.
- ✚ Tedarik zinciri yönetiminde kullanılmaktadır.
- ✚ Savunma sistemlerinde kullanılmaktadır.

Multiagentlerin çalışmasına şema ile örnek verecek olursak şu şekildedir :



Gartner

AI AGENT ÇERÇEVELERİ

AI aracı çerçevesi seçerken “ *karmaşıklık , veri gizliliği ve güvenliği , kullanım kolaylığı , kusursuz olacak şekilde entegrasyon , performans ve ölçeklenebilirlik* ” gibi hususlara dikkat edilmesi gerekmektedir.

AI Aracı Çerçeveleri

❖ AutoGen

Microsoftta açık kaynaklı olarak bulunan bir çerçevedir. Çoklu ajanlar ile AI uygulamalarını oluşturmak veya karmaşık görevleri gerçekleştirmek için kullanılmaktadır.

- ✚ Ölçeklenebilir ve dağıtılabılır bir ajan ağı gerçekleştirmek için kullanılan programlama çerçevesidir.
- ✚ İş akışlarını izleyerek hata ayıklama araçlarına sahiptir.
- ✚ Hem istek – yanıt şeklinde hem de olay odaklı etkileşimi ile asenkron mesajlaşmayı aktif şekilde kullanmaktadır.

❖ CrewAI

AutoGen gibi CrewAI de açık kaynaklı bir çerçevedir. Çoklu ajanları kullanarak AI çözümleri için orkestrasyon çerçevesidir. Bu çerçevenin rol mimarisi AI'yı çalışanlardan oluşan bir ekip olarak ele almaktadır. Ekibin temel bileşenleri ise şu şekildedir :

- ✚ Karmaşık iş akışlarında iş birliği yapılırken araçlara özel roller atanmaktadır. Geliştiriciler ise bir aracının rolünü , hedefini ve geçmişini ana hatlarıyla belirtmek için doğal dili kullanmaktadır.
- ✚ Her bir aracın belirli bir sorumluluğu görevleri vardır. Geliştiriciler ise her bir aracın bir görevi ve beklenen çıktıyı çıkartması için doğal dili kullanmaktadır.
- ✚ Araçların nasıl birlikte çalıştığını ve görevlerini nasıl yürüttüklerini tanımlamak için bir süreç yapıları vardır. Görevleri önceden bir sırayla belirlenmiş ardışık veya yönetici aracının görevlerine göre yürütmesi ve denetlemesi hiyerarşik şekilde olabilmektedir.

❖ LangChain

ChatGPT ve yapay zeka araçları gibi sohbet robotları dahil olmak üzere LLM destekli uygulamalar oluşturmak için kullanılan açık kaynaklı çerçevedir. Modüler bir mimari kullanılmaktadır ve her modül LLM'ler ile birlikte çalışmak için gerekli olan karmaşık kavramları ve adımları kapsayarak soyutlanmaları temsil etmektedir. Daha sonra bu modüller bileşenler bir araya getirilerek yapay zeka uygulamaları oluşturulabilir.

✚ Basit AI ajanlarını basit iş akışlarıyla geliştirmek için kullanışlıdır.

- ✚ Uygulamalarına bellek eklemek için vektör şeklinde veritabanları ve yardımcı programlar ile destek sağlamaktadır. Bu sayede geçmişte ki olaylar ve bağlamlar korunmaktadır.
- ✚ LangSmith platformu ise hataların ayıklanmasında , test edilmesinde ve performansın izlenmesi gibi olanakları sağlamaktadır.

❖ LangGraph

LangChain ekosistemi içerisinde yaşamaktadır. Bu çerçeve multiagentler için olan karmaşık iş akışlarını düzenlemede başarılıdır. Çalışması hakkında biraz bilgi verecek olursak ;

- ✚ Yapay zeka araçlarının belirli görev veya eylemler düğümler olacak şekilde gösterildiğinde bu eylemler arasında ki geçişlerin kenarlar olarak temsil edildiği bir grafik mimarisi uygulanmaktadır.
- ✚ Bir bileşen tüm etkileşimler boyunca kendi görev listesini korumaktadır.
- ✚ Bu yüzden bu mimari döngüsel , koşullu veya doğrusal olmayan iş akışları için uygundur.

❖ LamaIndex

Gen AI ve aracı AI çözümleri oluşturmak için açık kaynaklı bir veri düzenleme çerçevesidir. Önceden oluşturulmuş olan paketlenmiş araçlar ve araçlar ile yakın zamanda tanıtılan iş

akışları ve çoklu aracı sistemleri ile geliştirmek için bir mekanizma sunmaktadır. İş akışını oluşturan temel unsurlar :

- + Bir aracının belirli eylemleri vardır. Bunlar ise adımlardır. İş akışının temel bileşenlerini oluşturmaktadır.
- + Olaylar adımları tetikleyerek adımlar ile iletişim kurmasını sağlayan araçlardır.
- + Bağlam ise iş akışı boyunca paylaşılr böylece adımlar verileri depolayarak alabilir ve iletebilmektedir. Çalışmaları boyunca da durumlarını koruyabilmektedirler.

Olay odaklı olan bu mimari iş akışı adımlarının eş zamanlı olmayan bir şekilde gerçekleşmesini sağlamaktadır. Bu ise bir grafik mimarisinin aksine adımlar arasındaki yolların tanımlanmasına gerek olmadığını ve bunun sonucunda aracı eylemleri ile arasında esnek geçişler sağladığı anlamına gelmektedir. Bu yüzden LamaIndex iş akışları sık sık önceki adımlara geri dönmesi veya birkaç adıma dallanması ile daha dinamik yapay zeka aracı uygulamaları için oldukça uygundur.

❖ Semantic Kernel

Microsoft'un kurumsal düzeyde olan üretken AI uygulamaları oluşturmak için geliştirdiği açık kaynaklı bir geliştirme kitidir. Şu anda ajanlar oluşturmak için temel soyutlamalar sağlamaktadır. 2 tane yerleşik aracı uygulaması bulunmaktadır. birisi sohbet tamamlama aracı iken bir diğeri ise daha gelişmiş bir yardımcı aracıdır.

- + Karmaşık iş akışlarında grup sohbetleri aracılığı ile bu çerçeve kullanılarak birden fazla aracı yönetilebilmektedir.
- + Bu süreç AI ajanlarına atanan görevleri temsil eden ve verileri adımlar arasında nasıl aktığını ana hatlarıyla belirten adımlardan oluşmaktadır.

	<i>DİL</i>	<i>ÖZELLİK</i>	<i>KULLANIM</i>	<i>AVANTAJ</i>
JADE	Java	Fıpa uyumlu, Dağıtık sistemleri desteklemektedir.	Mobil ajanlar , multiajanlar , iletişim protokolleri	Zengin araçlar (GUI desteği)
SPADE	Python	XMPP protokolü ile mesajlaşma ,	IoT ,	Python ile kolay geliştirilebilmektedir.

		Modern ve hafiftir.	Dağıtık karar sistemleri	
GAMA Platform	GAML (kendi dili)	Simülasyon odaklıdır , Coğrafi bilgi sistemleri bulunmaktadır.	Sosyal, çevresel ve biyolojik sistemlerin modellenmesi	Karmaşık ortamların ajanlarla modellenmesine olanak vermektedir.
MESA	Python	Modüler ajan tabanlı simülasyon kütüphanesi bulunmaktadır.	Akademik araştırmalar ve eğitim alanlarında	Basit modellerle başlamak için idealdir.
JASON	Java + AgentSpeak	Rasyonel ajanlar modeli bulunmaktadır.	Karar destek sistemleri , bilgi sistemleri	BDI modeli sayesinde bilişsel ajanlara odaklanmaktadır.
MadKit	Java	Organizasyon temelli ajan sistemleri vardır.	Hiyerarşik veya rol temelli ajan sistemleridir.	

AI AGENT ARAYÜZLERİ

1) Grafiksel Kullanıcı Arayüzleri

Kullanıcının ajanları görsel olarak kullanarak izlenmesini , kontrol edilmesini ve yönlendirilmesini sağlamak için kullanılan arayüzdür.

- **JADE GUI**: Ajanların mesaj alışverişini ve durumunu görsel olarak izlemeyi sağlamaktadır.
- **NetLogo GUI**: Ajan davranışlarını ve ortamlarını simüle ederek görselleştirmeyi sağlamaktadır.
- **GAMA GUI**: Karmaşık ortam simülasyonlarını ve ajan görselleştirmesi için gelişmiş bir arayüz sağlamaktadır.

2) Komut Satırı Arayüzleri

Ajan sistemlerinin başlatılmasından durdurulmasına kadar bu süreçlerin test edilmesi için kullanılmaktadır. Bu arayüz geliştiriciler için daha uygundur.

- JADE ve SPADE iş çerçevelerindeki ajanlar için CLI ile başlatılarak yapılandırılmaktadır.

3) Web Tabanlı Arayüzler

Tarayıcıda ki ajanlara erişim sağlamaktadır. Dağıtık sistemlerde tercih edilmektedir.

- JADE ile entegre edilen HTTP-based web konsolları ,
- SPADE ile geliştirilen XMPP tabanlı web arayüzleri ,
- GAMA modelleri için web üzerinden senaryo izleme modülleri bunlara örnektir.

4) API ve SDK Arayüzleri

Diğer yazılım sistemleri , ajanlar sistemleri ile bu API'ler üzerinde iletişime geçerek programatik erişim sağlamaktadır.

- JADE için Java API ,
- SPADE için Python SDK ,
- Mesa için modüler API arayüzü bunlara örnektir.

5) Sensör ve Aktüatör Arayüzleri

Gerçek dünya uygulamalarında bulunan ajanlar çevreleriyle etkileşim sağlamaktadır.

- ROS (Robot Operating System) arayüzleri ,
- OpenAI Gym ortamları ,
- Unity ML-Agents Toolkit örnek verilebilmektedir.

KAYNAKÇA

- <https://www.ibm.com/think/topics/multiagent-system>
- <https://www.ibm.com/think/insights/top-ai-agent-frameworks>
- <https://www.freecodecamp.org/news/how-to-run-open-source-llms-on-your-own-computer-using-ollama/>
- <https://www.webtures.com/tr/blog/ai-agent-nedir-ai-agentler-ile-is-surecleri-nasil-gelistirilebilir/#:~:text=AI%20Agentler%2C%20%C3%A7e%C5%9Fitli%20g%C3%B6revleri%20yerine,karar%20verme%20gibi%20yeteneklere%20sahiptir.>
- <https://innovance.com.tr/ai-agents-ve-n8n-nedir/#:~:text=AI%20agentlar%C4%B1%2C%20geleneksel%20yaz%C4%B1%C4%B1m%20kodlamas%C4%B1n%C4%B1,daha%20esnek%20%C3%A7al%C4%B1%C5%9Fabilme%20kabiliyeti%20kazand%C4%B1r%C4%B1r.>