YAPAY ZEKA VE GELECEĞİN TEKNOLOJİLERİ:
GÜVENLİK, TEHDİTLER VE MÜHENDİSLİK DİSİPLİNLERİ

Editör:
Öğr. Gör. Sezen TUNÇ

ARTİKEL AKADEMİ: 466

Yapay Zeka ve Geleceğin Teknolojileri: Güvenlik, Tehditler ve Mühendislik Disiplinleri

Editör:
Öğr. Gör. Sezen TUNÇ
Orcid No:0000-0002-9372-8169

ISBN 978-625-5025-06-7

Birinci Basım: Ekim - 2026

Kapak uygulama: Artikel Akademi

Ofset Hazırlık: Artikel Akademi

Baskı ve Cilt: Uzunist Dijital Matbaa Anonim Şirketi
Akçaburgaz Mah.1584.Sk.No:21 / Esenyurt - İSTANBUL

Matbaa Sertifika No: 68922
Artikel Akademi bir Karadeniz Kitap Ltd. Şti. markasıdır.

©Karadeniz Kitap - 2026

Akademik etik kurallara bağlı kalınarak yapılacak olan alıntılar ve tanıtım maksadıyla yapılacak olan kısa alıntılar dışında, yazılı izni alınmadan, tümünün veya bir kısmının elektronik, mekanik ya da fotokopi yoluyla, basımı, yayımı, kopyalanması, çoğaltımı veya dağıtımı yapılamaz.

KARADENİZ KİTAP LTD. ŞTİ.
Koşuyolu Mah. Mehmet Akfan Sok. No:67/3 Kadıköy-İstanbul
Tel: 0530 076 94 90

Yayıncı Sertifika No: 19708
mail: info@artikelakademi.com
www.artikelakademi.com

YAPAY ZEKA VE GELECEĞİN TEKNOLOJİLERİ:
GÜVENLİK, TEHDİTLER VE MÜHENDİSLİK DİSİPLİNLERİ

Editör:
Öğr. Gör. Sezen TUNÇ

artikel akademi

İÇİNDEKİLER

ÖNSÖZ..... 7

1. BÖLÜM
YAPAY ZEKÂ ORKESTRASYON MÜHENDİSLİĞİ 9
- Beşir OĞUR

CHAPTER 2
ARTIFICIAL INTELLIGENCE TRAINING IN THE CONTEXT OF THE DEVELOPMENT OF TEACHER CANDIDATES' EDUCATOR IDENTITY 43
- Gizem ENGİN
- Göksu ÇİÇEKLİ KOÇ
- Beril CEYLAN NEBİOĞLU

3. BÖLÜM
BÜYÜK DİL MODELLERİ VE OTONOM YAPAY ZEKA AJANLARINDA SİBER TEHDİTLER VE GÜVENLİK YAKLAŞIMLARI 71
- Öner AYTAŞ

4. BÖLÜM
YAPAY ZEKÂ GÜVENLİĞİNE GİRİŞ: TEMEL TEHDİT TAKSONOMİSİ VE SALDIRI YÜZEYİ ANALİZİ 91
- Tuğçe ŞEN

5. BÖLÜM
YAPAY ZEKÂ DESTEKLİ GEREKSİNİM MÜHENDİSLİĞİ: FIRSATLAR, RİSKLER VE GELECEK YAKLAŞIMLAR 117
- Zeynep Yaren OĞUZ POLAT

ÖN SÖZ

Yapay zekâ teknolojileri, günümüzde yalnızca bilgisayar bilimlerinin bir alt dalı olmaktan çıkarak tüm mühendislik disiplinlerini, eğitim ekosistemlerini ve dijital güvenlik mimarilerini yeniden şekillendiren dönüştürücü bir güce evrilmiştir. Büyük dil modellerinden otonom ajana, orkestrasyon sistemlerinden gereksinim mühendisliğine kadar uzanan bu hızlı gelişim; bir yandan benzersiz fırsatlar sunarken, diğer yandan dinamik siber tehditler ve güvenlik risklerini de beraberinde getirmektedir.

"Yapay Zeka ve Geleceğin Teknolojileri: Güvenlik, Tehditler ve Mühendislik Disiplinleri" başlıklı bu çalışma; yapay zekânın mühendislik uygulamalarındaki rolünü, eğitimdeki kimlik inşasına etkilerini ve en kritik boyutu olan siber güvenlik/tehdit taksonomisini bütüncül bir yaklaşımla ele almaktadır. Eser, alanında uzman araştırmacıların teorik birikimlerini ve güncel analizlerini içeren beş özgün bölümden oluşmaktadır.

Kitabın birinci bölümünde, karmaşık yapay zekâ sistemlerinin yönetimi ve entegrasyonunda kritik bir rol üstlenen "Yapay Zekâ Orkestrasyon Mühendisliği" kavramı ve bu alandaki güncel yaklaşımlar incelenmektedir. İkinci bölümde, öğretmen adaylarının eğitici kimliklerinin gelişiminde yapay zekâ eğitiminin yeri ve önemi uluslararası bir perspektifle değerlendirilmektedir.

Güvenlik ve tehdit boyutuna odaklanan bölümlerde; üçüncü bölümde, büyük dil modelleri ve otonom yapay zekâ ajanlarında ortaya çıkan siber tehditleri ve bunlara karşı geliştirilen güvenlik yaklaşımları ele alınmaktadır. Dördüncü bölümde, yapay zekâ güvenliğine giriş niteliğinde temel tehdit taksonomisini ve sistemlerin saldırı yüzeyi analizi ayrıntılı bir biçimde sunulmaktadır. Beşinci ve son bölümde ise, yapay zekâ destekli gereksinim mühendisliğinin sunduğu fırsatları, beraberinde getirdiği riskleri ve geleceğe dönük yaklaşımlar ortaya konulmaktadır.

Yapay zekânın teorik, pratik ve güvenlik boyutlarını bir arada sunan bu eserin; akademisyenlere, mühendislere, siber güvenlik uzmanlarına, eğitimcilere ve geleceğin teknolojilerine ilgi duyan tüm araştırmacılara rehberlik etmesini temenni ederim. Kitabın hazırlanmasında emeği geçen tüm bölüm yazarlarına değerli katkıları için, yayın aşamasındaki titiz destekleri için de Artikel Akademi çalışanlarına çok teşekkür ederim.

Editör:

Öğr. Gör. Sezen TUNÇ

1. BÖLÜM

YAPAY ZEKÂ ORKESTRASYON MÜHENDİSLİĞİ

Dr. Öğr. Üyesi Beşir OĞUR
Işık Üniversitesi, Meslek Yüksekokulu
Bilişim Güveliği Programı
besir.ogur@isikun.edu.tr
0000-0003-3266-0233

ÖN SÖZ

Son yıllarda yapay zekâ (YZ) alanındaki gelişmeler, özellikle derin öğrenme ve Transformer tabanlı mimariler, doğal dil işleme ve üretken yapay zekâ uygulamalarında önemli bir dönüşüm ortaya çıkarmıştır. Büyük Dil Modelleri (Large Language Models – LLM), doğal dildeki istemleri işleyebilmekte ve bağlamsal metinler üretebilmektedir. Bununla birlikte, tek başına bir LLM; kurumsal bilgi kaynaklarına dinamik erişim, dış araçlarla etkileşim, çok aşamalı iş akışlarının yürütülmesi ve güvenlik politikalarının uygulanması gibi sistem düzeyindeki gereksinimleri karşılamak açısından sınırlı kalabilmektedir.

Bu noktada temel mühendislik problemi, yalnızca daha yetenekli bir dil modeli geliştirmekten ziyade, farklı yapay zekâ modellerinin, yazılım ajanlarının, bilgi erişim mekanizmalarının, vektör veri tabanlarının ve deterministik yazılım bileşenlerinin güvenli, izlenebilir ve ölçeklenebilir bir mimari içerisinde birlikte çalıştırılmasıdır. Yapay Zekâ Orkestrasyonu (AI Orchestration), bu bileşenlerin görev, veri ve durum akışlarını koordine etmeyi; sistem davranışını belirli kurallar ve kontrol mekanizmaları çerçevesinde yönetmeyi amaçlayan bir yaklaşım olarak değerlendirilebilir.

Yapay zekâ orkestrasyonu kapsamında ajan koordinasyonu, durum yönetimi, RAG (Retrieval-Augmented Generation), araç kullanımı, iş akışı yönetimi ve yapay zekâ yönetişimi gibi farklı bileşenler birlikte ele alınmaktadır. Bu bağlamda merkezi, dağıtık ve hiyerarşik ajan mimarileri ile deterministik iş

akışları ve model tabanlı dinamik yönlendirme yaklaşımları arasında; öngörülebilirlik, esneklik, güvenlik, izlenebilirlik ve hesaplama maliyeti bakımından farklı tasarım tercihleri ortaya çıkmaktadır. MCP (Model Context Protocol) ve A2A (Agent2Agent) gibi protokoller ise modellerin araçlar, veri kaynakları ve diğer ajanlarla etkileşiminin standartlaştırılmasına yönelik yeni entegrasyon olanakları sunmaktadır.

Bu kitap bölümü, Yapay Zekâ Orkestrasyon Mühendisliği'nin kuramsal çerçevesini ve temel uygulama ilkelerini incelemeyi amaçlamaktadır. Bu kapsamda LLM'ler, yapay zekâ ajanları, RAG mimarileri, vektör veri tabanları, iş akışı orkestrasyonu, ajanlar arası iletişim ve yapay zekâ yönetişimi; bütünleşik sistem mimarisi perspektifinden ele alınmaktadır. Ayrıca bu teknolojilerin kurumsal ölçekte uygulanmasında ortaya çıkabilecek güvenlik, mahremiyet, maliyet, güvenilirlik ve denetlenebilirlik gibi temel mühendislik sorunları değerlendirilmektedir.

Bu çalışma; yapay zekâ ve bilgisayar bilimleri alanındaki akademisyen ve araştırmacılar, yazılım mimarları, yapay zekâ mühendisleri ve kurumsal teknoloji yöneticileri için, yapay zekâ tabanlı sistemlerin tasarımı ve orkestrasyonuna ilişkin teknik ve kavramsal bir başvuru kaynağı oluşturması hedeflenmektedir.

1 YAPAY ZEKÂ ORKESTRASYONUNA GİRİŞ

1.1. Yapay Zekâ Sistemlerinde Paradigma Değişimi

Yapay zekâ araştırmaları, özellikle derin öğrenme ve Transformer mimarisinin ortaya çıkmasıyla birlikte önemli bir dönüşüm geçirmiştir. Önceki nesil yapay zekâ sistemlerinde görevler çoğunlukla belirli bir probleme yönelik tasarlanmış modeller ve kurallar aracılığıyla yürütülürken, günümüzde aynı temel modelin farklı görevlerde kullanılabildiği daha genel amaçlı sistemler ortaya çıkmıştır. Transformer mimarisi, özellikle dikkat mekanizması (attention mechanism) aracılığıyla uzun bağlamların işlenmesini ve paralel hesaplamayı mümkün kılarak modern dil modellerinin gelişmesinde önemli bir rol oynamıştır (Vaswani et al., 2017).

Bir sonraki aşamada, ölçeklendirilmiş ve özyönetimli (self-supervised) öğrenme yöntemleriyle geniş veri kümeleri üzerinde eğitilerek çok sayıda alt göreve uyarlanabilen yapay zekâ modelleri geliştirilmiş, böylece literatüre "temel

modeller" (foundation models) kavramı kazandırılmıştır. Bu süreç, modellerin hedeflenen amaçların ötesinde beklenmeyen yetenekler sergilemesini ifade eden ortaya çıkış (emergence) ve tek bir model mimarisi ile metodolojisinin tüm yapay zekâ uygulamalarının zeminini oluşturmasını simgeleyen homojenleşme (homogenization) ilkeleri üzerinden yapılandırılmıştır(Bommasani et al., 2021).

LLM'lerin ortaya çıkmasıyla yapay zekâ sistemleri, yalnızca tek bir göreve odaklanan dar kapsamlı yapılar olmaktan çıkıp doğal dil aracılığıyla karmaşık işlevleri üstlenebilen genel amaçlı hesaplama bileşenlerine dönüşmüştür. Fakat bu gelişmeler, bir modelin belirli bir probleme ilişkin bütün çözümleri tek başına bulacağı anlamına gelmez. Gerçek dünya uygulamalarında başarı sağlamak için güncel bilgiye erişim, dışarıdaki sistemlerle etkileşim, durum yönetimi, güvenlik ve işlem doğrulama gibi tamamlayıcı yeteneklerin sisteme entegre edilmesi zorunludur.

Bu nedenle güncel yapay zekâ mimarilerinde temel soru yalnızca "hangi model kullanılmalıdır?" değil, "model çevresindeki bilgi, araç ve yazılım bileşenleriyle nasıl güvenilir biçimde bütünleştirilecektir?" sorusuna dönüşmektedir.

1.2. Büyük Dil Modellerinin Yükselişi ve Temel Özellikleri

Büyük dil modelleri, büyük ölçekli metin veri kümeleri üzerinde önceden eğitilen ve daha sonra farklı görevlerde kullanılabilen sinir ağı tabanlı modellerdir. GPT-3 gibi modellerin az örnekle öğrenme (few-shot) öğrenme kapasitesini göstermesi, tek bir modelin görev tanımına göre farklı işlevleri yerine getirebilmesinin önünü açmıştır (Brown et al., 2020).

Prompt mühendisliği (prompt engineering), modelin davranışını doğal dil talimatları üzerinden yönlendirmeyi amaçlarken; komut ayarlaması(instruction tuning) ve insan geri bildiriminden pekiştirmeli öğrenme (Reinforcement Learning from Human Feedback—RLHF), modellerin kullanıcı talimatlarını daha uygun biçimde takip etmesine katkı sağlamıştır (Ouyang et al., 2022).

Bununla birlikte modelin ürettiği çıktının akıcı ve tutarlı olması, çıktının mutlaka doğru olduğu anlamına gelmemektedir. Modelin eğitim verisi, bağlam penceresi, erişebildiği harici bilgi ve kullanılan yönlendirme stratejisi çıktının niteliğini etkileyebilmektedir.

Zincirleme düşünme (Chain-of-Thought) ve benzeri muhakeme yönlendir-

meleri bazı görevlerde modelin çok adımlı problemlerdeki performansını geliştirebilmiştir (Wei et al., 2022). Ancak bu tür yöntemler modelin gerçek dünya verilerine erişimini veya harici sistemlerde güvenilir işlem gerçekleştirmesini tek başına sağlamamaktadır.

1.3. Tekil LLM Tabanlı Sistemlerin Sınırlılıkları

Tek bir LLM üzerinden tasarlanan sistemlerin temel sınırlılıkları arasında güncel bilgi eksikliği, dış sistemlere erişimin kısıtlı olması, halüsinasyon, uzun süreli durum yönetimi ve deterministik işlem gereksinimlerinin karşılanamaması olarak sıralanabilir.

Bilgi Geri Kazanımlı Üretim (Retrieval-Augmented Generation — RAG), yapay zeka modelinin yanıt üretmeden önce dışarıdaki bir bilgi kaynağından ilgili verileri çekmesini ve cevabı bu bilgilere dayanarak oluşturmasını sağlar (Lewis et al., 2020). Bu yöntem, modelin yanlış veya uydurma bilgi üretme ihtimalini düşürse de getirilen bilginin her zaman mutlak doğru ya da eksiksiz olduğunu tek başına garanti etmez.

Benzer biçimde Fonksiyon çağırma (function calling) veya araç kullanımı (tool use) mekanizmaları, modelin API, veritabanı veya harici uygulamalarla etkileşim kurmasını sağlayabilir. Ancak araç erişiminin genişletilmesi beraberinde yeni güvenlik riskleri oluşturmaktadır.

Bu nedenle LLM'nin tek başına bütün sistem işlevlerini gerçekleştirmesi yerine modelin uzmanlaşmış yazılım bileşenleriyle çevrelendiği sistem mimarileri önem kazanmıştır.

1.4. Üretken Yapay Zekâdan Ajan Tabanlı Sistemlere Geçiş

Üretken yapay zekâ sistemlerinde model çoğunlukla kullanıcı girdisine karşılık bir çıktı üretirken, ajan tabanlı sistemlerde model; hedef, durum, araçlar ve çevre ile etkileşim içinde daha uzun süreli görevleri gerçekleştirebilmektedir.

Bir yapay zekâ ajanı genel olarak:

- ✓ çevresini algılayan,
- ✓ hedef doğrultusunda planlama yapan,
- ✓ uygun araçları seçebilen,
- ✓ eylemler gerçekleştiren,

- ✓ sonuçları değerlendiren ve
- ✓ gerektiğinde sonraki eylemini yeniden belirleyen bir yazılım bileşeni olarak ele alınabilir (Wang et al., 2024).

ReAct yaklaşımı, akıl yürütme (reasoning) ile eylem (acting) süreçlerini birbirine bağlayarak ajan davranışlarının geliştirilmesinde önemli bir yaklaşım sunmuştur (Yao et al., 2023).

Bu dönüşüm sonucunda yapay zekâ sistemi yalnızca "cevap üreten model" olmaktan çıkarak "görev gerçekleştiren sistem" niteliğine yaklaşmıştır.

1.5. Yapay Zekâda Orkestrasyon Kavramı

Yapay zekâ orkestrasyonu, LLM'ler, ajanlar, araçlar, bilgi kaynakları, iş akışları ve kurumsal sistemler arasındaki görev, veri ve kontrol akışının yönetilmesi olarak tanımlanabilir.

Orkestrasyonun temel amacı daha fazla model kullanmak değil, farklı bileşenleri belirli bir görev amacı doğrultusunda koordineli biçimde çalıştırmaktır.

Bu bağlamda orkestratör;

- ✓ Kullanıcı isteğini analiz edebilir,
- ✓ Görevi sınıflandırabilir,
- ✓ Uygun modeli veya ajanı seçebilir,
- ✓ Gerekli bilgi kaynaklarını belirleyebilir,
- ✓ Araç çağrılarını yönetebilir,
- ✓ Ara sonuçları saklayabilir,
- ✓ Hataları yönetebilir,
- ✓ Nihai çıktıyı doğrulayabilir ve
- ✓ Gerekli durumlarda insan onayı isteyebilir.

Orkestrasyon, model yeteneklerinden çok sistem davranışının tasarlanması problemidir(Xu, B. 2026).

1.6. Orkestrasyonun Yazılım Mühendisliği Perspektifinden Değerlendirilmesi

Yapay zekâ orkestrasyon sistemlerini klasik yazılım sistemlerinden ayıran temel özelliklerden biri, deterministik ve olasılıksal bileşenlerin aynı mimari

içerisinde bulunmasıdır.

API çağrısı, veri doğrulama veya erişim kontrolü deterministik olarak tanımlanabilirken, doğal dil yorumlama ve bazı planlama işlemleri model güdümlü ve olasılıksaldır.

Bu durum yazılım mühendisliği açısından yeni gereksinimler doğurmaktadır. Sistem yalnızca "çalışıyor" veya "çalışmıyor" şeklinde değerlendirilmemeli; model çıktısının doğruluğu, araç kullanımının güvenliği, görev başarısı ve hata durumundaki davranış da değerlendirilmelidir.

Bu nedenle test, gözlemlenebilirlik, sürümleme, hata yönetimi, güvenlik ve doğrulama mekanizmaları orkestrasyon mimarisinin ayrılmaz parçalarıdır (Bellver et al., 2026). Genel olarak yazılım geliştirme süreçleri geleneksel yazılım mimarisi ve AI orkestrasyon mimarisi açısından Tablo 1'de karşılaştırılmıştır.

Tablo 1.1: Yazılım geliştirme süreçlerinin geleneksel yazılım mimarisi ve AI orkestrasyon mimarisi açısından karşılaştırılması.

Karşılaştırma Kriteri	**Geleneksel Yazılım Mimarisi**	**AI Orkestrasyon Mimarisi**
Mantık Akışı	Deterministik (If-Else, Kural Tabanlı)	Olasılıksal & Muhakeme Tabanlı (LLM / Agent)
Girdi / Çıktı Tipi	Yapılandırılmış Veri (JSON, XML)	Doğal Dil ve Serbest Metin / Hibrit Veri
Hata Yönetimi	İstisna Yönlendirme (Exception Handling)	Otomatik Düzeltme Döngüsü & Yeniden Deneme (Feedback Loop)
Sistem Bileşenleri	Mikroservisler, Veritabanları, API'ler	Ajanlar, LLM'ler, Vektör Veritabanları, Guardrails
Ölçekleme Kriteri	İşlemci / Bellek Kullanımı	Jeton (Token) Maliyeti, Bağlam Penceresi, İletişim Gecikmesi

1.7. Yapay Zekâ Orkestrasyon Mühendisliğinin Kapsamı ve Temel Problemleri

Yapay Zeka Orkestrasyon Mühendisliği (AI Orchestration Engineering); birden fazla yapay zeka modelinin, otonom ajanın, veri iletim hattının ve harici yazılım bileşenlerinin belirli bir kurumsal hedef doğrultusunda senkronize,

güvenli ve ölçeklenebilir biçimde birlikte çalıştırılmasını yöneten disiplindir.

- ✓ Orkestrasyon mühendisliği, tekil bir istem yazmanın (Prompt Engineering) ötesine geçerek şu temel mimari ilkeleri kapsar:
- ✓ **Dinamik Durum Yönetimi (State Management):** Çok adımlı iş akışlarında verinin, bağlamın (context) ve karar geçmişinin ajanlar arasında tutarlı biçimde aktarılması (Chase, 2022).
- ✓ **Ajanlar Arası İletişim ve Koordinasyon:** Uzmanlaşmış ajanların (örneğin Mimar, Geliştirici, QA) bir görev etrafında sıralı (sequential) veya paralel yürütme planlarıyla haberleşmesi (Moura, 2024).
- ✓ **Araç Kullanımı ve Dış Dünya Entegrasyonu (Tool Execution):** LLM'lerin kendi bilgi sınırlarının dışına çıkarak hesaplama araçlarını, veritabanı sorgularını veya web servislerini güvenli bir biçimde tetiklemesi (Schick et al., 2023).
- ✓ **Hata Toleransı ve Doğrulama Döngüleri (Self-Correction & Human-in-the-Loop):** Çıktıların kalitesini denetleyen denetçi katmanların ve gerektiğinde insan onayını sisteme dahil eden kontrol noktalarının kurgulanması (LangChain, 2024).

2. YAPAY ZEKÂ ORKESTRASYON MİMARİSİNİN TEMEL BİLEŞENLERİ

2.1. Büyük Dil Modelleri ve Model Katmanı

LLM'ler orkestrasyon mimarisinin bilişsel işlem katmanını oluşturabilir. Model; doğal dil anlama, sınıflandırma, özetleme, planlama, araç seçimi ve çıktı üretimi gibi görevleri gerçekleştirebilir. Bununla birlikte bütün görevlerde aynı modelin kullanılması gerekli değildir. Model seçimi görev karmaşıklığı, gecikme, maliyet, güvenlik ve bağlam uzunluğu gibi kriterlere göre yapılabilir.

Bu nedenle kurumsal sistemlerde model yönlendirme (model routing) mekanizmaları kullanılabilir. Basit görevler daha küçük modellerle, karmaşık görevler ise daha güçlü modellerle gerçekleştirilebilir.

2.2. Yapay Zekâ Ajanları ve Ajan Tabanlı Sistemler

Ajanlar, LLM'lerin araçlar ve çevreyle etkileşime girmesini sağlayan sistem bileşenleridir. Bir ajanı yalnızca bir LLM olarak değerlendirmek yeterli değil-

dir; ajan, modelin yanı sıra hedef, durum, araç ve eylem mekanizmalarını da içerir. Ajan mimarisinde görev ayrıştırma, planlama, araç seçimi, eylem yürütme ve geri besleme döngüleri temel işlevlerdir (Wang et al., 2024).

Markov karar süreci (Markov Decision Process (MDP) / Bellman Optimalite) denklemi ajan tabanlı yapay zeka modelinde bir ajanın bulunduğu durum (s), aldığı aksiyon (*a*), elde ettiği ödül (*r*) ve bir sonraki durum (s´) arasındaki optimal politika (π*) ilişkisini tanımlar.

$$V^*(s) = \max_{a \in A}[R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a)V^*(s')] \quad (2.1)$$

Burada;

*V**: durumundaki optimal değer fonksiyonu

R (s, a): durumunda aksiyonu alındığında alınan anlık ödül

$\gamma \in [0,1]$: Gelecekteki ödüllerin iskonto oranı (discount factor)

P (s´|s, a): durumunda aksiyonu alındığında durumuna geçiş olasılığını temsil etmektedir(Bellman, R., 1957; Sutton at el., 2018).

2.3. Araç Kullanımı ve Fonksiyon Çağrıları

Araç kullanımı, LLM'nin kendi başına gerçekleştiremeyeceği işlemleri harici sistemler üzerinden yapılmasını sağlar. API çağrıları, veritabanı sorguları, dosya erişimi, arama servisleri ve kurumsal uygulamalar araç olarak kullanılabilir.

ReAct gibi yaklaşımlar modelin düşünme ve eylem süreçlerini birbirine bağlamaktadır (Yao et al., 2023). Ancak araç kullanımının kontrolsüz bırakılması aşırı yetki, veri sızıntısı ve yetkisiz işlem risklerini artırabilir. Bu nedenle araç çağrıları kimlik doğrulama, yetkilendirme, parametre doğrulama ve politika kontrollerinden geçirilmelidir.

2.4. Durum Yönetimi ve Bellek Mekanizmaları

Ajanların çok adımlı görevleri gerçekleştirebilmesi için durum yönetimi kritik öneme sahiptir. Kısa süreli bellek mevcut görevin bağlamını tutarken, uzun süreli bellek geçmiş etkileşimler veya kalıcı bilgiler için kullanılabilir. Bellek mekanizmalarında gereksiz veya hassas verilerin süresiz saklanması güvenlik ve mahremiyet sorunlarına yol açabileceğinden, veri minimizasyonu, erişim kontrolü ve saklama politikaları uygulanmalıdır. Çok adımlı görevler için uzlaşma protokolünü denklem (2.2)'deki gibi Vicsek/Olfati-Saber yapısı ile sağlarlar. Bu denklem, orkestrasyon esnasında merkeziyetsiz ajanların bir durum veya parametre üzerinde uzlaşıya (consensus) varmasını sağlayan diferansiyel/

fark denklem sistemidir.

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij} \left(x_j(t) - x_i(t) \right) \quad (2.2)$$

Bu denklemde;

χ_i: ajanının anındaki durum değeri (ör. konum, yük miktarı, tahmin parametresi)

N_i: ajanının komşuluk kümesi (iletişim kurabildiği ajanlar)

α_{ij}: İletişim grafının komşuluk matrisindeki (A) ve ajanları arasındaki bağ ağırlığını ifade eder(Olfati-Saber et al., 2007).

2.5. İş Akışı ve Görev Planlama

Orkestrasyon sistemlerinde görev planlama iki temel biçimde gerçekleştirilebilir. Deterministik iş akışında işlem sırası önceden belirlenir. Dinamik planlamada ise model veya ajan görevin gerektirdiği adımları çalışma sırasında belirleyebilir. Kritik süreçlerde deterministik akışların, belirsiz ve yaratıcı görevlerde ise kontrollü dinamik planlamanın kullanılması hibrit bir yaklaşım modeli oluşturabilir.

Contract Net Protocol (CNP) Görev Atama / Fayda (Utility) Fonksiyonu

Çok ajanlı sistemlerde görevlerin (task allocation) açık artırma veya teklif verme usulüyle dağıtılmasında, ajanın bir görevi (T_k) üstlenme uygunluğunu ve orkestratörün seçimliliğini belirleyen temel fayda/maliyeti (2.3) denklem ile gösterilebilir:

$$U_i(T_k) = R(T_k) - C_i(T_k, S_i) - \sum_{j \in N_i} \text{Penalty}\left(T_k, T_j\right) \quad (2.3)$$

Bu denklemde;

$U_i\ (T_k)$: ajanının görevini üstlenmekten elde edeceği net fayda

$R\ (T_k)$: Görevin tamamlanması durumundaki sistem getirisi/ödülü

$C_i\ (T_k\ S_i)$: ajanının mevcut kaynak/durumu (S_i) altında görevi yerine getirme maliyeti

Penalty : Ajanın halihazırda üstlendiği diğer görevlerle (T_j) olan çakışma/gecikme cezasını temsil eder(Smith, R., 1980).

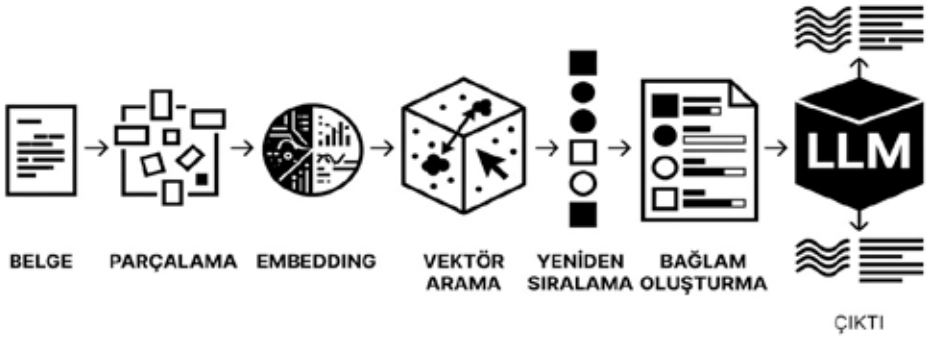
2.6. Bilgi Geri Kazanımlı Üretim (RAG)

Büyük Dil Modelleri, eğitildikleri veri kümesinin tarihiyle sınırlıdır ve kuruma özel dinamik verilere erişim yeteneğinden yoksundur. Bu kısıtı aşmak için Bilgi Geri Kazanımlı Üretim (Retrieval-Augmented Generation - RAG) mimarisi kullanılır (Lewis et al., 2020).

RAG sistemi üç temel aşamadan oluşur:

- ✓ **Veri Hazırlama ve Gömme (Embedding):** Dokümanların parçalanması (chunking) ve semantik vektör temsilcilerine dönüştürülerek vektör veritabanlarında (Chroma, Pinecone, Qdrant) indekslenmesi.
- ✓ **Arama ve Bağlam Çıkarma (Retrieval):** Ajanın sorgusuna en yakın metin parçalarının vektör benzerliği (cosine similarity) hesaplanarak çekilmesi.
- ✓ **Genişletilmiş Üretim (Augmented Generation):** Çekilen bağlamın sistem istemine dahil edilerek LLM'in halüsinasyondan arındırılmış, olgusal yanıtlara yönlendirilmesi.

Ajan mimarilerinde RAG, sadece bir arama aracı olarak değil; ajanın uzun süreli belleği (long-term memory) olarak işlev görür. RAG mimarisinde tipik bir süreç akışı tamamlayıcı süreç adımları ile şekil 2.1'de gösterilmiştir.



Şekil 2.1: RAG Mimarisi

RAG, bilgi güncelliği ve kurumsal veri erişimi açısından önemli avantajlar sağlasa da getirilen içeriğin kalitesine bağımlıdır. Bu nedenle kaynak doğrulama, metadata filtreleme ve yeniden sıralama (reranking) mekanizmaları önemlidir.

2.7. Embedding Modelleri ve Vektör Veri Tabanları

Embedding (gömme) modelleri metinleri yüksek boyutlu sayısal vektörlere dönüştürür. Bu vektörler kullanılarak anlamsal benzerlik üzerinden bilgi aranabilir.

Kosinüs benzerliği (Cosine similarity), iç çarpım (dot product) ve Öklid mesafesi (Euclidean distance) gibi ölçütler kullanılabilir. Ancak anlamsal benzerlik ile doğruluk aynı kavram değildir. Benzer görünen bir belge yanlış veya güncelliğini yitirmiş olabilir.

Bu nedenle üretim sistemlerinde metadata filtreleme, hibrit arama ve yeniden sıralama (reranking) gibi mekanizmaların kullanılması değerlendirilebilir.

2.8. Harici Veri Kaynakları ve Kurumsal Sistem Entegrasyonu

Kurumsal yapay zekâ sistemleri dokümanlar, ilişkisel veritabanları, data havuzları, ERP(Kurumsal Kaynak Planlaması), CRM(Müşteri İlişkileri Yönetimi), dosya sistemleri ve API(Uygulama Programlama Arayüzü)'ler gibi çok çeşitli veri kaynaklarına erişebilir.

Bu entegrasyonlarda kimlik yönetimi, yetkilendirme, veri sınıflandırması, erişim kayıtları ve veri minimizasyonu kritik öneme sahiptir.

2.9. Orkestrasyon Katmanı ve Kontrol Mekanizmaları

Orkestrasyon katmanı sistemin kontrol merkezidir. Görev yönlendirme, ajan seçimi, araç çağrıları, hata yönetimi, durum aktarımı ve çıktı doğrulama gibi işlemleri koordine eder. Özellikle kritik işlemlerde model çıktısının doğrudan eyleme dönüştürülmesi yerine politika ve doğrulama katmanlarından geçirilmesi gerekir. Bayesian Nash Dengesi (BNE) koşulu (Bayesian Nash Equilibrium (BNE) Condition) çok ajanlı orkestrasyonda ajanların diğer ajanların tam bilgi/ iç durumlarına (tip -) sahip olmadığı eksik bilgili oyunlarda (Incomplete Information Games) optimal strateji dengesini açıklar (Tao, Y., & Xu, H. (2025):

$$U_i(T_k) = R(T_k) - C_i(T_k, S_i) - \sum_{j \in N_i} \text{Penalty}\left(T_k, T_j\right) \quad (2.4)$$

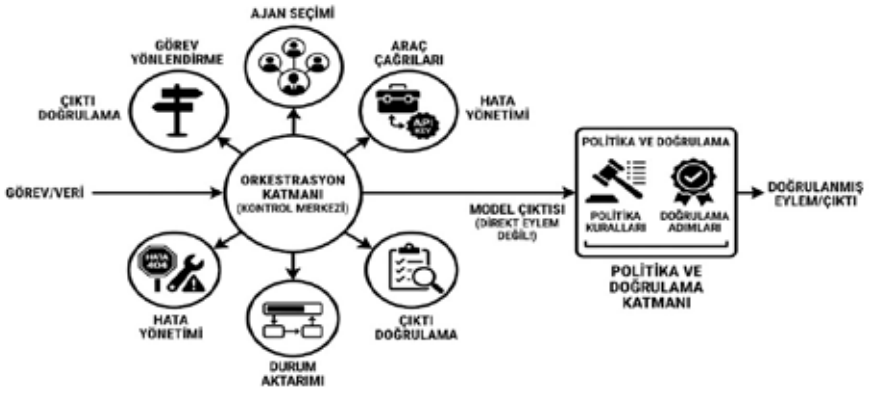
Bu denlemde;

S^*_i : i ajanı için optimal strateji

θ_i / θ_{-i}: i ajanının tipi / diğer ajanların tipleri

$P(\theta_{-i} \mid \theta_i)$: i: ajanı kendi tipini bildiğinde diğer ajanların tiplerine dair koşullu olasılık inancı

μ_i: Ajanın fayda (utility) fonksiyonunu temsil eder. Orkestrasyon katmanı ve kontrol mekanizmaları Şekil 2.2 de gösterilmiştir.



Şekil 2.2: Orkestrasyon katmanı ve kontrol mekanizmaları

2.10. Gözlemlenebilirlik, Günlükleme ve İzlenebilirlik

Yapay zekâ orkestrasyonunda gözlemlenebilirlik (observability), günlükleme (logging) ve izlenebilirlik (traceability), çoklu ajan (multi-agent) sistemlerinde ajanlar arası etkileşimlerin, LLM çağrılarının, araç (tool) kullanımlarının ve vektör tabanlı bilgi erişim süreçlerinin uçtan uca izlenebilmesini sağlayan temel bileşenlerdir. Yapılandırılmış günlükleme, sistem içerisinde gerçekleşen istem ve yanıtları, token tüketimini, model ve araç çağrılarını, çağrı parametrelerini ve yürütme sonuçlarını sistematik biçimde kayıt altına alarak operasyonel görünürlüğü artırır. İzlenebilirlik ise bir görevin başlangıcından sonuçlandırılmasına kadar geçen yürütme zincirinin adım adım takip edilmesini; ajanların karar süreçlerinin, araç kullanım sıralarının ve bilgi erişim adımlarının ilişkilendirilmesini mümkün kılar. Bu sayede bağlam kayması (context drift), beklenmeyen ajan davranışları, hatalı araç kullanımları ve karar zincirindeki sapmalar daha erken aşamada tespit edilebilir. Gözlemlenebilirlik, günlükleme ve izlenebilirliğin birlikte ele alınması; yapay zekâ sistemlerinin güvenilirlik, performans ve güvenlik açısından değerlendirilmesinin yanı sıra darboğazların belirlenmesi, kaynak ve token maliyetlerinin optimize edilmesi ve güvenlik politikaları ile guardrail mekanizmalarının ihlallerinin denetlenmesi açısından da kritik bir altyapı oluşturur (Shankar et al., 2024).

3: AJAN TABANLI YAPAY ZEKÂ ORKESTRASYON MODELLERİ

3.1. Yapay Zekâ Ajanlarının Yapısal Özellikleri

Ajanların temel özellikleri arasında hedef yönelimi, çevreyle etkileşim, otonomi, planlama, araç kullanımı ve geri besleme yer almaktadır.

Klasik ajan araştırmalarında rasyonel ajan, çevresinden aldığı algılara göre belirli bir hedef doğrultusunda eylemlerini gerçekleştiren bir sistem olarak ele alınmıştır (Wooldridge & Jennings, 1995; Jennings, 2000). LLM tabanlı ajanlar bu yapıya doğal dil işleme ve üretken model yeteneklerini eklemektedir.

3.2. Tek Ajanlı ve Çok Ajanlı Sistemler

Tek ajanlı sistemlerde görevlerin tamamı bir ajan tarafından yönetilir. Çok ajanlı sistemlerde ise görevler uzmanlaşmış ajanlara dağıtılabilir.

Örneğin:

- ✓ araştırma ajanı,
- ✓ kodlama ajanı,
- ✓ test ajanı,
- ✓ veri analiz ajanı,
- ✓ denetim ajanı

aynı orkestrasyon sistemi içerisinde çalışabilir.

CAMEL ve AutoGen gibi çalışmalar çok ajanlı iş birliği modellerinin araştırılmasına katkı sağlamıştır (Li et al., 2023; Wu et al., 2023). Unutulmamalıdır ki ajan sayısının artırılması otomatik olarak performans artışı sağlamaz. İletişim maliyeti, koordinasyon karmaşıklığı ve hata yayılımı artabilir.

3.3. Merkezi Orkestrasyon Mimarisi

Merkezi mimaride tek bir orkestratör ajanlar arasındaki görev dağılımını yönetir. Avantajları arasında merkezi politika uygulama, kolay izleme ve tutarlı durum yönetimi bulunur. Dezavantajı ise orkestratörün sistemin performansını sınırlayan bir bileşene veya tek hata noktasına (Single Point of Failure) dönüşebilmesidir.

3.4. Dağıtık Ajan Mimarileri

Dağıtık mimaride ajanlar daha bağımsız çalışır ve görevleri kendi aralarında paylaşabilir. Bu yaklaşım ölçeklenebilirlik ve esneklik sağlayabilir; ancak koordinasyon ve güvenlik daha karmaşık hâle gelir. Dağıtık ajan sistemlerinde ortak protokoller, kimlik yönetimi ve görev yaşam döngüsü yönetimi kritik öneme sahiptir.

3.5. Hiyerarşik Ajan Mimarileri

Hiyerarşik yapılarda üst düzey bir ajan genel hedefi belirlerken alt ajanlar uzmanlaşmış görevleri gerçekleştirir. Örneğin bir "proje yöneticisi ajan" görevi analiz ederek araştırma, kodlama ve test ajanlarına alt görevler atayabilir. Bu model karmaşık görevlerin bölümlendirilmesini kolaylaştırabilir.

3.6. Uzmanlaşmış Ajanların Görev Dağılımı

Uzmanlaşmış ajanların tasarımında görevlerin açık sınırlarla tanımlanması gerekir. Bir ajanın sahip olduğu yeteneklerin mümkün olduğunca görev alanıyla sınırlı tutulması güvenlik ve hata izolasyonu açısından yararlıdır. Bu yaklaşım, "tek ve çok güçlü ajan" yerine "sınırlı yeteneklere sahip koordineli ajanlar" mimarisini destekler.

Karmaşık bir iş akışında gelen alt görevlerin (t_k), sistemde mevcut olan heterojen ajanlara (g_i) en uygun şekilde atanması bir yönelimli optimizasyon problemidir. Görev dağılımı, her ajanın yetkinlik skorunu maksimize ederken hesaplama ve gecikme maliyetlerini asgariye indirmeyi hedefler. Bu mimaride görev atama modeli (3.1) denkleminde ifade edildiği gibidir:

$$\max_{x_{ik}} \sum_{i=1}^{N} \sum_{k=1}^{K} x_{ik} \cdot \text{Capability}(g_i, t_k) - \lambda \sum_{i=1}^{N} \sum_{k=1}^{K} x_{ik} \cdot \text{Cost}(g_i, t_k)$$

$$\text{s.t. } \sum_{i=1}^{N} x_{ik} = 1 \quad \forall k \in \{1, \ldots, K\}, \quad x_{ik} \in \{0,1\} \tag{3.1}$$

Bu denklemdeki değişken ve parametreler aşağıdaki şekilde tanımlanır:

$\chi_{ik} \in \{0,1\}$: *k*. alt görevin *i*. ajana atanıp atanmadığını gösteren ikili (*binary*) karar değişkeni.

Capability (g_i, t_k) $\in$ [0,1]: *i*. ajanın *k*. görevi yerine getirme yetkinlik ve başarı skoru.

Cost (g_i, t_k): İlgili eylemin hesaplama, jeton (*token*) kullanımı ve zaman gecikmesi (*latency*) cinsinden maliyeti.

$\lambda > 0$: Performans kazancı ile maliyet tasarrufu arasındaki dengeyi belirleyen hiperparametre.

Kısıt Koşulu: Her alt görev tam olarak tek bir ajana atanmalıdır ($\sum_{i=1}^{N} x_{ik} = 1$) (Gerkey, B. P., & Matarić, M. J. (2004).

3.7. Ajanlar Arası İletişim ve Koordinasyon

Ajanlar arasında mesajlaşma, görev devri, durum paylaşımı ve sonuç aktarımı gerçekleştirilebilir. Ancak ajanlar arası iletişim arttıkça gecikme, maliyet ve hata yayılımı da artabilir. Bu nedenle yalnızca gerekli bilgilerin paylaşılması ve iletişim protokollerinin standartlaştırılması önemlidir.

Dağıtık yapıda çalışan ajanların ortak bir hedefe ulaşabilmesi için aralarındaki bilgi akışı ve mesajlaşma mekanizmaları Cross-Attention (Çapraz Dikkat) mimarisi ile modellenir. Bu yöntem, alıcı ajanın kaynak ajanlardan gelen bilgileri önem derecesine göre ağırlıklandırmasını sağlar. Ajanlar Arası Çapraz Dikkat (Multi-Agent Cross-Attention) süreci (3.2) denklemi ile gösterilebilir:

$$\text{Attention}(Q_i, K_j, V_j) = \text{Softmax}\left(\frac{Q_i K_j^T}{\sqrt{d_k}}\right) V_j \qquad (3.2)$$

Bu denklemdeki parametre ve değişkenler aşağıdaki gibi tanımlanırlar;

$Q_i = X_i W^Q$: Alıcı ajanın (i) mevcut iç durumundan (X_i) türetilen Sorgu (*Query*) matrisi.
$K_j = X_j W^K$: Kaynak ajanın (j) sağladığı bağlam ve araç çıktılarından türetilen Anahtar (*Key*) matrisi.
$V_j = X_j W^V$: Kaynak ajanın sunduğu ham veri ve temsil değerlerini içeren Değer (*Value*) matrisi.
d_k: Anahtar vektörlerinin boyutu (gradyan patlamalarını önlemek için ölçekleme faktörü $\sqrt{d_k}$ kullanılır).

3.8. Planlama, Yürütme ve Geri Besleme Döngüleri

Ajanlar çoğunlukla Şekil 3.1'de gösterilen döngü içinde çalışır. Ajanların dinamik ortamda mantık yürütme (*Reasoning*) ve dış dünyayla etkileşime girme (*Acting*) süreçleri ReAct döngüsü ile yürütülür. Ajan her adımda geçmiş hafızasını, ürettiği düşünceyi, seçtiği eylemi ve ortamdan gelen gözlemi ardışık olarak çalışma belleğine ekler. ReAct durum geçiş ve eylem seçim modeli aşağıdaki matematiksel gösterimle ifade edilebilir:

$$S_{t+1} = \delta(S_t, a_t, o_t) = S_t \parallel m_t \parallel a_t \parallel o_t$$

$$a_t = \arg\max_{a \in AUT} P_\theta(a \mid S_t) \quad (3.3)$$

Bu denklemde;

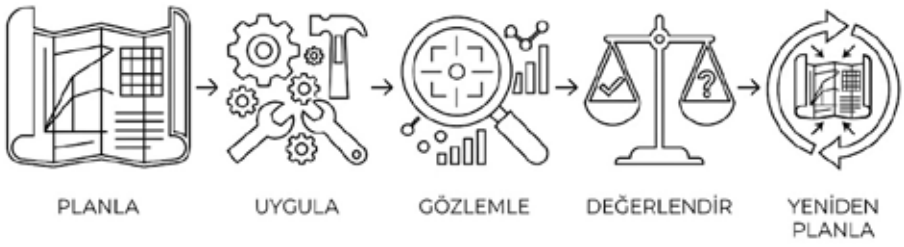
S_t: *t.* Zaman adımındaki toplam çalışma belleği (geçmiş bağlam ve istem geçmişi).

m_t: Ajanın içsel mantık yürütme adımı ve ürettiği düşünce (*Thought*).

$a_t \in A \cup T$: Üretilen yanıt eylemi veya dış araç çağırma komutu (*Action*).

o_t: Dış ortamdan veya araç yürütme motorundan dönen gözlem/yanıt (*Observation*).

||: Metin/vektör düzeyinde sıralı birleştirme (*concatenation*) operatörü olarak ifade edilir.



Şekil 3.1: Orkestrasyon içindeki ajanlar arasındaki döngü işleyişi

ReAct bu yapıyı düşünme ve eylem arasında bağlantı kurarak ele alırken, Reflexion geçmiş hatalardan elde edilen geri bildirimlerin sonraki davranışlarda kullanılmasını araştırmaktadır (Yao et al., 2023; Shinn et al., 2023). Generative Agents çalışması ise bellek, planlama ve yansıtma mekanizmalarının bir arada kullanılabileceğini göstermiştir (Park et al., 2023).

3.9. Deterministik İş Akışları ve Model Tabanlı Dinamik Yönlendirme

Deterministik akışlarda görev yolları önceden belirlenirken dinamik yönlendirmede sonraki adım modelin kendisi tarafından seçilebilir. Üretim ortam-

larında bu iki yaklaşımın birlikte kullanılması daha kontrollü bir mimari sağlar. Örneğin ödeme işleminin onay süreci deterministik olabilirken, müşterinin talebinin hangi bilgi kaynağıyla cevaplanacağı dinamik biçimde belirlenebilir. Gelen sorgu ve görevlerin en uygun uzman modele veya ajana yönlendirilmesi, gürültülü Softmax Gating (MoE - Mixture of Experts) mimarisi ile gerçekleştirilir:

$$G(x) = \text{Softmax}\big(\text{TopK}(H(x), k)\big)$$

$$H_i(x) = x \cdot W_{g,i} + \epsilon \cdot \text{Softplus}\big(x \cdot W_{\text{noise},i}\big)$$

$$E(x) = \sum_{i \in \text{TopK}} G_i(x) E_i(x) \qquad (3.4)$$

Bu denklem gösteriminde değişken ve parametreler aşağıdaki gibi tanımlanırlar;

χ: Giren sorgu veya bağlam vektör temsili.

$W_{g,\ i}$: $_i$. uzman ajan için eğitilebilir yönlendirme ağırlık matrisi.

$W_{\ noise,i}$: Yönlendirme dengesini sağlamak ve aşırı yüklenmeyi önlemek için eklenen gürültü ağırlık matrisi ($\epsilon \sim N(0,1)$)..

E_i (χ):: Seçilen . uzman ajanın ürettiği çıktı vektörü(Shazeer et al., 2017).

3.10. Human-in-the-Loop ve Human-on-the-Loop

Human-in-the-Loop (HITL) (Döngü içindeki insan) yaklaşımında insan karar sürecinin doğrudan bir parçasıdır. Human-on-the-Loop (Döngüyü izleyen insan) yaklaşımında ise sistem daha yüksek düzeyde otonom çalışırken insan sistemi izler ve gerektiğinde müdahale eder. Özellikle finans, sağlık, güvenlik ve hukuki işlemler gibi yüksek riskli alanlarda insan onay mekanizmalarının kullanılması değerlendirilebilir (Amershi et al., 2019; Autio et al., 2024).

4 BİLGİ, ARAÇ VE AJAN ENTEGRASYONU

4.1. Kurumsal Bilginin Yapay Zekâ Sistemlerine Entegrasyonu

Kurumsal yapay zekâ uygulamalarının önemli bir bölümünde kurumun kendi verilerine erişebilme ihtiyacı duyulur. Bu veriler yapılandırılmış veya yapılandırılmamış olabilir. Dokümanlar, e-postalar, veritabanları, raporlar, CRM kayıtları ve API'ler farklı erişim yöntemleri gerektirir. Bilgi entegrasyonunun temel amacı veriyi modele aktarmak değil, modelin yalnızca yetkili olduğu bilgiye erişmesini sağlamaktır.

Harici bilgi kaynaklarının sisteme entegrasyonu, ajanın karar verme sürecindeki belirsizliği azaltır. Bilgi kazancı (*Information Gain*), harici verinin belirsizliği ne ölçüde düşürdüğünü ölçer. Bilgi entropisi ve bilgi kazancı modeli;

$$IG(S,I) = H(S) - H(S \mid I)$$

$$IG(S,I) = -\sum_{s \in S} P(s)\log_2 P(s) + \sum_{i \in I} P(i) \sum_{s \in S} P(s \mid i)\log_2 P(s \mid i) \qquad (4.1)$$

denklemi ile gösterilir. Burada;

$H(S)$: Ajanın mevcut durum uzayındaki başlangıç Shannon entropisi (bilgi eksikliği ve belirsizlik düzeyi).

$H(S|I)$: Harici bilgi kaynağı veya araç çıktısı gözlemlendikten sonraki koşullu entropi.

$IG(S\ I)$: Entegrasyon sonucu elde edilen net Bilgi Kazancı (*Information Gain*) olarak ifade edilir(Shannon (1948); Cover & Thomas 2006).

4.2. RAG Mimarilerinin Orkestrasyon İçerisindeki Rolü

RAG (Retrieval-Augmented Generation) mimarisi (2.6)'da bahsedildiği gibi, büyük dil modellerinin (LLM) statik parametrelerinde yer alan bilgiye bağımlılığını azaltarak harici ve güncel bilgi kaynaklarının üretim sürecine dinamik biçimde dâhil edilmesini sağlayan önemli bir bileşendir. Temel RAG yaklaşımında, kullanıcı sorgusu doğrultusunda ilgili bilgi parçaları bir dış veri kaynağından veya vektör indeksinden geri getirilmekte ve elde edilen bağlam LLM'in yanıt üretim sürecine aktarılmaktadır (Lewis et al., 2020).

Orkestrasyon perspektifinden değerlendirildiğinde, RAG, yalnızca bir bilgi erişim mekanizması değil, ajanın hangi bilgi kaynağına ne zaman başvuracağını, hangi sorgulama stratejisinin kullanılacağını ve elde edilen bağlamın sonraki karar ve üretim adımlarında nasıl değerlendirileceğini belirleyen bir yürütme

bileşeni olarak ele alınabilir. Özellikle çoklu ajan sistemlerinde farklı ajanların farklı bilgi kaynaklarına erişmesi, sorguların yinelemeli olarak genişletilmesi veya yeniden formüle edilmesi ve elde edilen sonuçların başka araçlarla doğrulanması, RAG süreçlerinin orkestratör tarafından dinamik biçimde yönetilmesini gerektirmektedir. Güncel araştırmalarda geleneksel sabit RAG akışlarının ötesine geçilerek planlama, araç kullanımı, yansıtma (reflection) ve çoklu ajan işbirliği gibi yeteneklerin retrieval (geri getirme) sürecine dâhil edildiği Agentic RAG yaklaşımlarının ortaya çıktığını göstermektedir (Singh et al., 2025). Bu bağlamda RAG, orkestrasyon katmanına dışsal bilgi sağlayan pasif bir alt sistem olmaktan çıkarak, bilgi erişimi, bağlam oluşturma, karar verme ve yanıt üretimi arasındaki etkileşimi yöneten dinamik bir bilgi erişim katmanı hâline gelmektedir. Bununla birlikte, retrieval kalitesi, bilgi güncelliği, gecikme, maliyet, yanlış veya kötü amaçlı içeriklerin geri getirilmesi ve ardışık RAG adımlarında oluşabilecek hata zincirleri, orkestrasyon tasarımında ayrıca izlenmesi ve yönetilmesi gereken temel hususlardır (Zhao et al., 2024; Gupta et al., 2024).

RAG (Retrieval-Augmented Generation) mimarisinde vektör benzerliği (cosine similarity) metin parçalarının (chunks) ve kullanıcı sorgusunun *(q)* embedding vektör uzayındaki yönsel yakınlığını ve anlamsal benzerliğini hesaplar:

$$\mathrm{Sim}(q,d) = \cos\theta = \frac{v_q \cdot v_d}{\|v_q\|_2 \|v_d\|_2} = \frac{\sum_{i=1}^{n} v_{q,i} v_{d,i}}{\sqrt{\sum_{i=1}^{n} v_{q,i}^2}\sqrt{\sum_{i=1}^{n} v_{d,i}^2}} \quad (4.2)$$

Sorgu *q* ve doküman *d*'nin -boyutlu yoğun embedding vektörleri () arasındaki açısal benzerliği bulur.

RAG Mimarisinde seyrek getirme skoru (Sparse Retrieval Score)**,** bilgi getirme (information retrieval) ve arama motoru sistemlerinde, bir dokümanın kullanıcının sorgusuyla ne kadar alakalı olduğunu hesaplamak için kullanılır. Bu alandaki en köklü, güçlü ve sektörel standart kabul edilen klasik (lexical/ kelime tabanlı) algoritma BM25 (Best Matching 25) algoritmasıdır. Bu algoritmanın matematiksel ifadesi aşağıdaki gibidir:

$$\mathrm{Score}_{\mathrm{BM25}}(q,d) = \sum_{t \in q} \mathrm{IDF}(t) \cdot \frac{f(t,d)\cdot(k_1+1)}{f(t,d)+k_1\cdot\left(1-b+b\cdot\frac{|d|}{\mathrm{avgdl}}\right)}$$

$$\mathrm{IDF}(t) = \ln\left(\frac{N-n_t+0.5}{n_t+0.5}\right)+1 \quad (4.3)$$

Bu denklemde

f(t,d) : *t* teriminin *d* dokümanındaki frekansı.

|*d*| / avgdl: Doküman uzunluğunun ortalama doküman uzunluğuna oranı.

k_1, b: Terim doygunluğu () ve uzunluk normalizasyonu () parametrelerini temsil ederler.

RAG Mimarisinde kullanılan başka bir yöntem de Olasılıksal RAG (Probabilistic RAG) modelidir. Bu yaklaşım süreçlerindeki belirsizlikleri, döküman seçimi hatalarını ve modelin halüsinasyon eğilimlerini matematiksel ve olasılıksal yöntemlerle yöneten gelişmiş bir yapay zeka mimarisidir. Geleneksel RAG sistemleri deterministik (kesin/sabit) mantıkla en yakın k adet dökümanı seçip dil modeline (LLM) beslerken; olasılıksal RAG, bilgi getirme (retrieval) ile yanıt üretimi (generation) aşamalarını olasılık dağılımları üzerinden birbirine bağlar. Bu mimari denklem (4.4)'te olduğu gibi gösterilir(Yang at el.,2026):

$$P_{\text{RAG-Sequence}}(y \mid q) = \sum_{z \in \text{top-}k} P_\eta(z \mid q) \prod_{i=1}^{N} P_\theta(y_i \mid q, z, y_{1:i-1}) \quad (4.4)$$

Burada;

$P_\eta (z \mid q)$: q sorgusu için dokümanının getirilme olasılığı.

$P_\theta (y_i \mid q, z, y_1{:}i\text{-}1)$: Getirilen doküman z bağlamında i. simgenin (*token*) üretilme olasılığıdır.

4.3. Hibrit Bilgi Erişimi ve Yeniden Sıralama (Reranking)

Salt vektör araması bütün durumlarda yeterli değildir. Anahtar kelime tabanlı arama ile anlamsal aramanın birlikte kullanılması hibrit erişim sağlayabilir. İlk aşamada geniş bir aday kümesi oluşturulurken reranking (yeniden sıralama) modeli aday belgeleri daha ayrıntılı biçimde sıralayabilir. Bu yaklaşım bilgi erişiminin hem kapsamını genişletir hem de isabet oranını iyileştirmeyi amaçlar. Anlamsal ve terimsel arama sonuçlarının birleştirilmesi ve en alakalı dokümanların öne çıkarılması için iki aşamalı sıralama uygulanır.

Birden fazla farklı arama veya sıralama sisteminden gelen sonuç listelerini tek ve güçlü bir listede birleştiren bir algoritma olan Reciprocal Rank Fusion (RRF) denklem (4.5)'teki gibi ifade edilir:

$$\text{RRFScore}(d \in D) = \sum_{m \in M} \frac{1}{k + r_m(d)} \quad (4.5)$$

M: Farklı arama sistemleri kümesi (örneğin Vektör Araması ve BM25).

$r_m (d)$: Dokümanın arama sistemindeki sıralama derecesi.

k : Sabit kaydırma parametresi (standart olarak).

Başka bir arama metodu olarak Cross-Encoder ile yeniden sıralama (reranking), vektör aramasıyla seçilen aday dokümanları kullanıcı sorgusuyla tek bir

metin gibi birleştirip çapraz dikkat (cross-attention) ile derinlemesine analiz eden ve en alakalı metinleri dil modeline iletilmek üzere en doğru sıraya koyan iki aşamalı bir RAG optimizasyon yöntemidir. Bu yaklaşım matematiksel model olarak (4.6)'da gösterildiği gibidir:

$$P(\text{Relevance} \mid q, d_i) = \text{Softmax}\big(s(q, d_i)\big) = \frac{e^{s(q,d_i)}}{\sum_{j=1}^{K} e^{s(q,d_j)}} \quad (4.6)$$

Bu denklemde,

$s(q, d_i)$: Cross-Encoder modelinin sorgu ve doküman çiftini birlikte işleyerek ürettiği ham (logit) alaka skoru.

K: İkinci aşama sıralamasına giren aday doküman sayısı olarak ifade edilir(-Nogueira, R., & Cho, K. (2019).

4.4. Yapılandırılmış ve Yapılandırılmamış Veri Kaynaklarının Birleştirilmesi

LLM'ler doğal dil belgeleriyle çalışabilirken kurumsal işlemlerin önemli bölümü yapılandırılmış veritabanlarında bulunmaktadır. Örneğin müşteri sözleşmesinin anlamı bir dokümandan, müşterinin işlem geçmişi ise ilişkisel veritabanından alınabilir. Orkestrasyon katmanı bu iki bilgi türünü aynı görev bağlamında birleştirebilir.

4.5. Araçların Keşfi, Seçimi ve Çağrılması

Araçların (tools) dinamik olarak keşfedilmesi, değerlendirilmesi ve görevin gereksinimlerine göre seçilmesi, otonom ve çoklu ajan mimarilerinin temel yeteneklerinden biridir. Ajanın bir görevi gerçekleştirmek amacıyla hangi aracı, hangi parametrelerle ve hangi yürütme sırasıyla kullanacağına dinamik olarak karar verebilmesi, sistemin esnekliğini ve işlevsel kapasitesini artırmakla birlikte yeni güvenlik risklerini de beraberinde getirmektedir. Özellikle araçların adlarını, açıklamalarını, kullanım koşullarını ve parametrelerini içeren meta veriler, ajan tarafından karar verme sürecinde bağlamsal girdi olarak değerlendirildiğinden, güvenilmeyen veya kötü amaçlı biçimde oluşturulmuş araç açıklamaları potansiyel bir saldırı yüzeyine dönüşebilir. Bu kapsamda prompt injection, tool injection ve yetkisiz araç kullanımına yönelik saldırılar, yalnızca LLM'in ürettiği çıktının değil, ajanın araç keşfi ve seçim mekanizmasının da güvenlik açısından değerlendirilmesini gerektirmektedir. Dolayısıyla araçların

değerlendirilmesinde yalnızca işlevsel yetenekleri, performansı ve kullanım uygunluğu değil; sahip oldukları erişim izinleri, işleyebildikleri veri türleri, gerçekleştirebildikleri yan etkili işlemler (side effects), kimlik doğrulama ve yetkilendirme gereksinimleri, güven sınırları ve olası kötüye kullanım senaryoları da dikkate alınmalıdır. Bu doğrultuda en az ayrıcalık (least privilege) ilkesi temel alınarak her araca yalnızca görevini gerçekleştirmek için gerekli kaynaklara erişim verilmesi, araç çağrılarının politika tabanlı olarak sınırlandırılması ve kritik işlemlerin yürütülmesinden önce ek doğrulama mekanizmalarının uygulanması önem taşımaktadır. Böylece dinamik araç kullanımı, yalnızca bir yetenek keşfi ve otomasyon mekanizması olarak değil, aynı zamanda araç güvenilirliğinin, erişim kapsamının ve yürütme risklerinin sürekli değerlendirildiği bir güvenlik ve orkestrasyon problemi olarak ele alınabilir(Schick et al., 2023).

Ajanların dış API'leri, hesaplama araçlarını ve veritabanı fonksiyonlarını çağırma süreci, uygun araç seçimi ve doğru parametre üretimi bileşik olasılık modeliyle çözülür. Bu modeli (4.7)'de gösterilmiştir:

$$(t^*, p^*) = \arg\max_{t \in T, p \in P_t} P_\theta(t, p \mid q, c) = \arg\max_{t \in T, p \in P_t} P_\theta(t \mid q, c) \cdot P_\theta(p \mid t, q, c) \quad (4.7)$$

Denklemdeki değişken ve parametreler aşağıdaki gibi tanımlanır:

$P_\theta\ (t \mid q,c)$*:* Sorgu q ve mevcut bağlam c altında $t \in T$ aracının seçilme olasılığı.

$P_\theta\ (p \mid t,q,c)$*:* Seçilen t aracı için şemaya uygun $p \in P_t$ parametrelerinin üretilme olasılığı.

(t, p*)*: Azami olasılık sunan optimal araç ve parametre çifti.

4.6. API (Uygulama Programlama Arayüzü), Veritabanı ve Kurumsal Uygulama Entegrasyonu

API'ler orkestrasyon sistemleri ile kurumsal uygulamalar arasındaki temel entegrasyon noktalarından biridir. Her API çağrısında kimlik doğrulama, yetkilendirme, parametre doğrulama, hata yönetimi ve kayıt altına alma uygulanmalıdır. Özellikle yazma veya silme işlemleri gibi geri dönüşü zor eylemler için daha yüksek güvenlik kontrolleri kullanılabilir.

4.7. Model Context Protocol (Model Bağlam Protokolü)

Model Context Protocol (MCP), yapay zekâ uygulamalarının araçlar ve dış kaynaklarla standartlaştırılmış biçimde etkileşim kurmasını amaçlayan açık bir

protokoldür.

MCP'nin 28 Temmuz 2026 tarihli sürümü, stateless protokol çekirdeği, çok turlu istekler, header (başlık) tabanlı yönlendirme, önbellekte lis (durumsuz protokol) te sonuçlarını oluşturma ve yetkilendirme iyileştirmeleri gibi özellikler içermektedir.

MCP'nin orkestrasyon açısından temel katkısı, model veya ajan ile harici yetenekler arasındaki entegrasyon arayüzünün standartlaştırılmasına yardımcı olmasıdır. Bununla birlikte protokol standardizasyonu güvenlik sorunlarını ortadan kaldırmaz. MCP üzerinden sunulan araçların izinleri, veri erişimleri ve çağrı parametreleri yine politika katmanları tarafından kontrol edilmelidir.

4.8. Agent2Agent ve Ajanlar Arası İletişim

Agent2Agent (A2A), bağımsız ajan sistemlerinin birbirleriyle iletişim kurabilmesini ve görev devri gerçekleştirebilmesini amaçlayan açık bir standarttır. A2A'nın güncel yayımlanmış sürümü 1.0.0'dır ve protokol farklı framework, dil veya sağlayıcılarla geliştirilmiş bağımsız ajanların yeteneklerini keşfetmesini ve ortak görevler yürütmesini hedeflemektedir. Bu yapı, ajanların birbirlerinin iç belleklerine veya araçlarına doğrudan erişmek zorunda kalmadan görev paylaşmasını mümkün kılabilir.

4.9. Protokol Tabanlı Birlikte Çalışabilirlik

MCP daha çok ajan/model ile araç ve bağlam kaynakları arasındaki etkileşime odaklanırken A2A ajanlar arası iletişim düzeyini desteklemektedir. Bu tür protokoller, farklı sağlayıcıların bileşenlerinin aynı orkestrasyon mimarisi içinde kullanılabilmesi açısından birlikte çalışabilirliği artırabilir.

4.10. Bağlam Yönetimi ve Bilgi Akışının Kontrolü

Bağlam penceresinin sınırlı olması nedeniyle bütün verinin modele aktarılması mümkün veya gerekli değildir. Orkestrasyon sistemi hangi bilginin modele gönderileceğini seçmeli, gereksiz içeriği çıkarmalı ve hassas verileri mümkün olduğunca sınırlandırmalıdır. Bağlam yönetimi bu nedenle yalnızca performans değil, güvenlik ve mahremiyet açısından da önemli bir kontrol mekanizmasıdır.

5. YAPAY ZEKÂ ORKESTRASYON MÜHENDİSLİĞİ: TASARIM, UYGULAMA VE GELECEK PERSPEKTİFİ

5.1. Orkestrasyon Sistemlerinde Mimari Tasarım İlkeleri

Orkestrasyon sistemleri düşük bağımlılık, yüksek bütünlük, açık arayüzler, modülerlik ve hata izolasyonu ilkeleri doğrultusunda tasarlanmalıdır. Model, ajan, araç, bilgi ve kontrol katmanlarının ayrıştırılması bir bileşenin değiştirilmesinin bütün sistemi etkilemesini önleyebilir (Bass et al., 2021). Bu yaklaşımda model sistemin tamamını kontrol eden bileşen değil, kontrollü bir bilişsel işlem bileşeni olarak konumlandırılmalıdır.

5.2. Deterministik ve Otonom Orkestrasyonun Birlikte Kullanımı

Yapay zekâ tabanlı orkestrasyon mimarilerinde deterministik ve otonom yaklaşımların birlikte kullanılması, **kontrol edilebilirlik ile esnek karar verme yeteneği arasında bir denge kurulmasını** sağlar. Deterministik orkestrasyon; iş akışının sırası, erişim yetkileri, araç çağrıları, hata yönetimi, doğrulama adımları ve kritik işlemlerin yürütülmesi gibi önceden tanımlanması gereken süreçlerin kural tabanlı ve öngörülebilir biçimde gerçekleştirilmesini mümkün kılarken, otonom orkestrasyon LLM veya ajanların görevin bağlamına göre plan oluşturmasına, uygun araçları seçmesine ve yürütme sırasında ortaya çıkan koşullara göre eylem planını uyarlamasına olanak tanır. ReAct yaklaşımı, akıl yürütme ve eylem üretiminin dönüşümlü biçimde gerçekleştirilerek dış kaynaklarla etkileşim kurulabileceğini ve eylem sonuçlarının sonraki kararları şekillendirebileceğini göstermektedir (Yao et al., 2023). Bu nedenle hibrit bir orkestrasyon modelinde, **kritik güvenlik ve yönetişim kurallarının deterministik bir kontrol katmanında tutulması; görev planlama, bilgi arama, araç seçimi ve bağlama bağlı kararların ise otonom ajanlara bırakılması** uygun bir mimari yaklaşım olarak değerlendirilebilir. Böyle bir yapı, LLM tabanlı kararların doğası gereği değişken olabilmesine karşın, kritik eylemlerin yetki, politika ve doğrulama kontrolleri dışında gerçekleştirilmesini sınırlandırarak sistemin denetlenebilirliğini ve operasyonel güvenilirliğini destekler. Bu hibrit yapı Tablo 5.1'deki gibi modellenebilir.

Tablo 5.1: Deterministik ve otonom hibrit orkestrasyonu

Özellik	**Deterministik Orkestrasyon**	**Otonom Orkestrasyon**
Karar verme	Önceden tanımlanmış kurallar ve akışlar	Ajan/LLM tarafından dinamik olarak
İş akışı	Belirli ve öngörülebilir	Bağlama göre değişebilir
Araç seçimi	Önceden belirlenebilir	Göreve göre dinamik olarak seçilir
Kontrol düzeyi	Yüksek	Daha esnek, ancak daha değişken
Hata yönetimi	Tanımlı senaryolar üzerinden	Duruma göre yeniden planlama yapılabilir
Kullanım alanı	Kritik ve standart süreçler	Karmaşık ve değişken görevler
Temel avantajı	Öngörülebilirlik ve denetlenebilirlik	Esneklik ve uyarlanabilirlik
Temel riski	Esnekliğin sınırlı olması	Kararların değişken ve daha az öngörülebilir olması

5.3. Performans, Gecikme ve Maliyet Optimizasyonu

Toplam gecikme yaklaşık olarak:

$$T_{E2E} = T_{route} + T_{retrieve} + T_{infer} + T_{tool} + T_{validate} + T_{network} + T_{queue}$$

Burada T_{E2E}, uçtan uca gecikmeyi (End-to-End Latency), T_{route} yönlendirmeyi, bilgi erişimini, T_{infer} model çıkarımını, T_{tool} araç yürütmeyi, $T_{validate}$ doğrulama ve politika kontrollerini, $T_{network}$ ağ iletişimini, T_{queue} ise sistem yoğunluğu nedeniyle oluşan kuyruk bekleme süresini ifade etmektedir. Özellikle T_{queue} terimini eklemek yararlı olacaktır. Çünkü gerçek üretim sistemlerinde bir isteğin gecikmesi yalnızca işlemin kendisinden değil, GPU, API veya orkestratör kaynaklarının meşgul olması nedeniyle oluşan bekleme süresinden de kaynaklanabilir. Ajan sayısı arttıkça model çağrıları ve iletişim maliyetleri de artabilir. Bu nedenle önbellekleme, paralel araç çağrıları, uygun model yönlendirme, prompt optimizasyonu ve harmanlama (batching) gibi teknikler kullanılabilir.

LLM servisleme araştırmalarında bellek yönetimi ve çıkarım optimizasyon-

larının işlem/veri aktarım hacmi (throughput) üzerinde önemli etkileri olduğu gösterilmiştir (Kwon et al., 2023).

5.4. Ölçeklenebilirlik ve Dayanıklılık

Üretim sistemlerinde yatay ölçekleme, yük dengeleme, kuyruk sistemleri ve bağımsız servisler kullanılabilir.

Dayanıklılık açısından timeout (zaman aşımı), retry (yeniden deneme), exponential backoff (üssel geri çekilme), circuit breaker (devre kesici), fallback (B planı / alternatif çözüm) ve idempotency (eşgüçlülük) gibi mekanizmalar önemlidir (Beyer et al., 2016).

Ayrıca model hataları ile altyapı hatalarının birbirinden ayrılması gerekir. Bir model yanlış karar verdiğinde tekrar denemek her zaman doğru çözüm değildir.

5.5. Hata Yönetimi ve Kurtarma

Hatalar altyapısal, entegrasyon kaynaklı, model kaynaklı veya iş kuralı kaynaklı olabilir. Her hata için farklı kurtarma mekanizması uygulanmalıdır. Uzun görevlerde checkpoint(kontrol noktası), durum saklama ve telafi işlemleri (compensating actions) kullanılabilir. Bu yapı, bir sistem hatasında bütün görevin başından itibaren tekrar çalıştırılmasını önleyebilir.

5.6. Değerlendirme ve Başarım Ölçütleri

Orkestrasyon sistemleri yalnızca model doğruluğuyla değerlendirilmemelidir.

Ölçüm boyutları arasında:

- ✓ görev başarısı,
- ✓ doğruluk,
- ✓ groundedness **(**temellendirilmişlik**),**
- ✓ araç seçimi,
- ✓ plan kalitesi,
- ✓ gecikme,

- ✓ throughput (İşlem Hacmi/Veri Aktarım Hızı),
- ✓ görev başına maliyet,
- ✓ güvenlik ihlali,
- ✓ hata kurtarma oranı ve
- ✓ insan müdahale oranı bulunabilir.

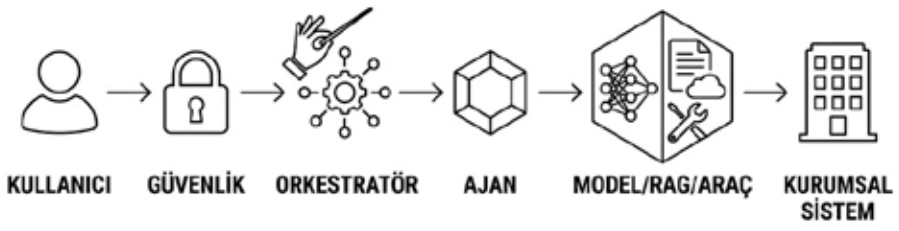
HELM, model değerlendirmesinin çok boyutlu olması gerektiğini ortaya koyarken (Liang et al., 2022), AgentBench LLM'lerin ajan olarak etkileşimli ortamlarda değerlendirilmesine odaklanmaktadır (Liu et al., 2023). Gerçek yazılım geliştirme görevlerini değerlendiren SWE-bench de ajan değerlendirmesinin gerçek dünya görevlerine yaklaştırılması açısından önemlidir (Jimenez et al., 2024).

5.7. Kurumsal Yapay Zekâ Orkestrasyon Referans Mimarisi

Kurumsal bir referans mimari aşağıdaki katmanlardan oluşabilir:

- ✓ Kullanıcı ve kanal katmanı
- ✓ API ve güvenlik katmanı
- ✓ Orkestrasyon katmanı
- ✓ Ajan katmanı
- ✓ Model katmanı
- ✓ Bilgi/RAG katmanı
- ✓ Araç ve entegrasyon katmanı
- ✓ Gözlemlenebilirlik ve yönetişim katmanı

Kavramsal akış Şekil 5.1'deki gibi oluşturulabilir:



Şekil 5.1: Kurumsal Yapay zekâ orkestrasyonu için referans mimarisi

5.8. Finans, Yazılım ve Sağlık Alanlarında Uygulama Senaryoları

Finans sektöründe orkestrasyon; müşteri talepleri, finansal veriler, doküman analizi, risk modelleri ve politika kontrollerini bir araya getirebilir.

Yazılım mühendisliğinde kod üretimi, test, kod inceleme ve dokümantasyon farklı ajanlara dağıtılabilir.

Sağlıkta ise belge özetleme, literatür araştırması ve klinik karar desteği gibi görevler gerçekleştirilebilir.

Ancak yüksek riskli alanlarda sistemin bağımsız karar verici yerine kontrollü karar destek bileşeni olarak konumlandırılması daha uygun bir mühendislik yaklaşımıdır (Autio et al., 2024).

5.9. Uygulama ve Devreye Alma Stratejileri

Kurumsal uygulamalarda aşamalı devreye alma yaklaşımı kullanılabilir:

- ✓ Gereksinim ve risk analizi
- ✓ Kontrollü prototip
- ✓ Shadow mode
- ✓ Human-in-the-loop
- ✓ Sınırlı otonomi
- ✓ Sürekli değerlendirme

Bu yaklaşım sistemin gerçek kullanıcı davranışları altında güvenli biçimde test edilmesine olanak sağlayabilir.

NIST AI RMF (Risk Management Framework)'nin Govern(yönetişim), Map(haritalama), Measure(ölçü) ve Manage(yönetim) işlevleri bu yaşam döngüsü yaklaşımıyla uyumlu bir risk yönetimi çerçevesi sunmaktadır (Tabassi, 2023; Autio et al., 2024).

5.10. Açık Problemler ve Gelecek Araştırma Alanları

Yapay zekâ orkestrasyonunda önemli araştırma alanları şunlardır:

- ✓ güvenilir otonomi,
- ✓ standart ajan değerlendirme yöntemleri,
- ✓ dinamik model yönlendirme,
- ✓ uzun süreli bellek,

- ✓ çok ajanlı koordinasyon,
- ✓ doğrulanabilir ajan davranışı,
- ✓ güvenli araç kullanımı,
- ✓ gerçek zamanlı orkestrasyon,
- ✓ hesaplama ve enerji verimliliği,
- ✓ protokol tabanlı birlikte çalışabilirlik,
- ✓ ajan güvenliği ve
- ✓ kurumsal yapay zekâ yönetişimi.

Özellikle ajanların uzun süreli görevlerde güvenilir biçimde çalıştırılması önemli bir araştırma problemidir. AgentBench gibi çalışmalar, ajanların performansının tek adımlı cevap doğruluğundan farklı olarak uzun vadeli muhakeme, karar verme ve talimat izleme yeteneklerini de gerektirdiğini göstermektedir (Liu et al., 2023).

5.11. Sonuç ve Değerlendirme

Yapay zekâ orkestrasyon mühendisliği, LLM'lerin tek başına kullanılma durumlarından daha geniş ve problem çözme fonksiyonelliği açısından etkin bir sistem yaklaşımını ifade etmektedir.

Bu yaklaşımda model; ajan, araç, bilgi kaynağı, durum yönetimi, güvenlik, politika ve gözlemlenebilirlik mekanizmalarıyla birlikte çalışmaktadır.

Bu nedenle geleceğin yapay zekâ sistemlerinin yalnızca daha büyük modellerden oluşması beklenmemelidir. Asıl mühendislik problemi, bu modellerin güvenilir, denetlenebilir, ölçeklenebilir ve kurumsal gereksinimlerle uyumlu sistemler hâline getirilmesidir.

Başarılı bir orkestrasyon sistemi için temel hedef mümkün olan en yüksek otonomi düzeyi değil; **görevin riskine, karmaşıklığına ve gereksinimlerine uygun otonomi düzeyinin belirlenmesidir**.

Bu perspektiften yapay zekâ orkestrasyon mühendisliği; yapay zekâ, yazılım mühendisliği, dağıtık sistemler, bilgi erişimi, güvenlik ve yönetişimin kesişiminde yer alan çok disiplinli bir mühendislik alanı olarak değerlendirilebilir.

Önümüzdeki dönemde Yapay Zeka Orkestrasyon Mühendisliği alanında öne çıkacak temel stratejik eğilimlerin ve teknolojik dönüşümlerin aşağıda bahsedilen ana eksenler etrafında şekilleneceği düşünülmektedir:

- ✓ **İnsan Denetimli (Human-in-the-Loop) Otonom Ekosistemler:** Oto-

nom yazılım hatlarının kararlılık oranları arttıkça, insan rolü "kod yazan" konumundan "mimari onay veren ve kuralları denetleyen" üst denetör (supervisor) seviyesine evrilecektir (LangChain, 2024).

- ✓ **Kuantum Bilişim ve Kuantum Sonrası Güvenlik Entegrasyonu:** Geleceğin orkestrasyon mimarileri, hibrit kuantum-klasik algoritmaları ve kuantum dirençli şifreleme (Post-Quantum Cryptography) standartlarını otonom güvenlik ajanlarına entegre edecektir (NIST, 2024).
- ✓ **Bilişsel Arayüzler ve BCI Entegrasyonu:** Beyin-Bilgisayar Arayüzleri (BCI) ile üretken modellerin doğrudan haberleşmesi, geliştiricilerin sadece zihinsel soyutlamalar ve niyetler üzerinden karmaşık sistem mimarileri orkestre etmesine olanak tanıyacaktır (Wolpaw & Wolpaw, 2012).
- ✓ **Kendini İyileştiren ve Evrilen Kod Tabanları (Self-Healing Systems):** Statik kod depoları yerini, üretim ortamındaki (production) telemetri verilerini ve denetim loglarını anlık analiz ederek kendi kodunu, testlerini ve güvenlik duvarlarını otomatik güncelleyen dinamik sistemlere bırakacaktır (Chase, 2022).

KAYNAKÇA

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. (2019). Guidelines for human-AI interaction. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 1–13. https://doi.org/10.1145/3290605.3300233

A2A Protocol. (2026). *Agent2Agent (A2A) protocol specification* (Version 1.0.0). https://a2a-protocol.org/

Bass, L., Clements, P., & Kazman, R. (2021). *Software architecture in practice* (4th ed.). Addison-Wesley Professional.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

Bellver, J., Ramos-Varela, S., Guragain, A., Córdoba, R., & D'Haro, L. F. (2026, February). ORCHESTRA: AI-Driven Microservices Architecture to Create Personalized Experiences. In *Proceedings of the 16th International Workshop on Spoken Dialogue System Technology* (pp. 158-167).

Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (Eds.). (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein,

M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N. S., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2021). *On the opportunities and risks of foundation models*. arXiv. https://arxiv.org/abs/2108.07258

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877–1901.

Chase, H. (2022). *LangChain* [Computer software]. https://github.com/langchain-ai/langchain

Cover, T. M., & Thomas, J. A. (2006). Elements of Information Theory (2nd ed.). Wiley-Interscience.

Gerkey, B. P., & Matarić, M. J. (2004). A formal analysis and taxonomy of task allocation in multi-robot systems. *The International journal of robotics research*, *23*(9), 939-954.

Gupta, S., Ranjan, R., & Singh, S. N. (2024). *A comprehensive survey of retrieval-augmented generation (RAG): Evolution, current landscape and future directions*. arXiv.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023). Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, 79–90. https://doi.org/10.1145/3605764.3623985

Jennings, N. R. (2000). On agent-based software engineering. *Artificial Intelligence, 117*(2), 277–296. https://doi.org/10.1016/S0004-3702(99)00107-1

Jimenez, C. E., Yang, J., Wettig, A., Yao, S., Pei, K., Press, O., & Narasimhan, K. (2024). SWE-bench: Can language models resolve real-world GitHub issues? *International Conference on Learning Representations*. https://arxiv.org/abs/2310.06770

Kleppmann, M. (2017). *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. O'Reilly Media.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., & Stoica, I. (2023). Efficient memory management for large language model serving with PagedAttention. *Proceedings of the 29th Symposium on Operating Systems Principles*, 611–626. https://doi.org/10.1145/3600006.3613165

LangChain. (2024). *LangGraph: Building language agents as graphs*. LangChain Blog. https://blog.langchain.dev/langgraph/

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems, 33*, 9459–9474.

Li, G., Hammoud, H. A. A. K., Itani, H., Khizbullin, D., & Ghanem, B. (2023). CAMEL: Communicative agents for mind exploration of large scale language model society. *Advances in Neural Information Processing Systems, 36*. https://arxiv.org/abs/2303.17760

Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C. D., Ré, C., Acosta-Navas, D., Hudson, D. A., ... Koreeda, Y. (2022). Holistic evaluation of language models. *Transactions on Machine Learning Research*. https://arxiv.org/abs/2211.09110

Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Zhang, S., Deng, X., Zeng, A., Du, Z., Zhang, C., Shen, S., Zhang, T., Su, Y., Sun, H., Huang, M., Dong, Y., & Tang, J. (2023). AgentBench: Evaluating LLMs as agents. *International Conference on Learning Representations*. https://arxiv.org/abs/2308.03688

Mialon, G., Fourrier, C., Swift, C., Wolf, T., LeCun, Y., & Scialom, T. (2023). GAIA: A benchmark for general AI assistants. *International Conference on Learning Representations*. https://arxiv.org/abs/2311.12983

Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., & Scialom, T. (2023). Augmented language models: A survey. *Transactions on Machine Learning Research*. https://arxiv.org/abs/2302.07842

Model Context Protocol. (2026). *Model Context Protocol specification* (2026-07-28). https://modelcontextprotocol.io/

Moura, J. (2024). *CrewAI: Cutting-edge framework for orchestrating role-playing, autonomous AI agents*. https://github.com/crewAIInc/crewAI

National Institute of Standards and Technology. (2024). *Post-Quantum Cryptography Standardization*. U.S. Department of Commerce. https://csrc.nist.gov/projects/post-quantum-cryptography

Nogueira, R., & Cho, K. (2019). *Passage Re-ranking with BERT*. arXiv preprint arXiv:1901.04085.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems, 35*, 27730–27744.

Olfati-Saber, R., Fax, J. A., & Murray, R. M. (2007). *Consensus and Cooperation in Networked Multi-Agent Systems*. Proceedings of the IEEE, 95(1), 215-233.

Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 1–22. https://doi.org/10.1145/3586183.3606763

Shankar, S., Garcia, R., & Patel, A. (2024). Observability and tracing mechanisms in multi-agent AI orchestration systems. *Journal of Systems and Software Architecture*, 148, 103-115. https://doi.org/10.1016/j.jssa.2024.103115

Shannon, C. E. (1948). A Mathematical Theory of Communication. The Bell System Technical Journal, 27(3), 379-423.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., & Dean, J. (2017). Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36, 6848–6867.

Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems, 36*. https://arxiv.org/abs/2303.11366

Smith, R. G. (1980). *The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver*. IEEE Transactions on Computers, C-29(12), 1104-1113.

Singh, A., Ehtesham, A., Kumar, S., Talaei Khoei, T., & Vasilakos, A. V. (2025). *Agentic retrieval-augmented generation: A survey on Agentic RAG*. arXiv.

Sutton, R. S., Barto, A. G., & Barto, A. (1998). *Reinforcement learning: An introduction* (Vol. 1, No. 1, pp. 9-11). Cambridge: MIT press.

Tao, Y., & Xu, H. (2025). Generalized bayesian nash equilibrium with continuous type and action spaces. *SIAM Journal on Optimization*, *35*(2), 789-817.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser,

Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*.

Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W. X., Wei, Z., & Wen, J.-R. (2024). A survey on large language model based autonomous agents. *Frontiers of Computer Science, 18*, 186345. https://doi.org/10.1007/s11704-024-3840-6

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems, 35*, 24824–24837.

Wooldridge, M., & Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review, 10*(2), 115–152. https://doi.org/10.1017/S0269888900008122

Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, K., Burger, D., & Wang, C. (2023). AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*. https://arxiv.org/abs/2308.08155

Yang, Z., Ma, Q., Chen, C. C. J., & Shrivastava, A. (2026). SuperIntelligent Retrieval Agent: The Next Frontier of Information Retrieval. In *ICLR 2026 Workshop on Memory for LLM-Based Agentic Systems*.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). ReAct: Synergizing reasoning and acting in language models. *International Conference on Learning Representations*. https://arxiv.org/abs/2210.03629

Xi, Z., Chen, W., Guo, X., Yu, W., He, H., Ding, Y., Wang, M., Jin, Q., Zhang, X., Chen, Q., Li, Y., Zheng, Z., Xia, S., Zeng, M., & others. (2023). The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*. https://arxiv.org/abs/2309.07864

Xu, B. (2026). Ai agent systems: Architectures, applications, and evaluation. *arXiv preprint arXiv:2601.01743*.

Zhao, S., Yang, Y., Wang, Z., He, Z., Qiu, L. K., & Qiu, L. (2024). *Retrieval augmented generation (RAG) and beyond: A comprehensive survey on how to make your LLMs use external data more wisely*. Microsoft Research.

CHAPTER 2

ARTIFICIAL INTELLIGENCE TRAINING IN THE CONTEXT OF THE DEVELOPMENT OF TEACHER CANDIDATES' EDUCATOR IDENTITY*

Assoc. Prof. Dr. Gizem ENGİN
Ege University,
gizem.engin@ege.edu.tr,
https://orcid.org/0000-0003-2532-8136

Assist. Prof. Dr. Göksu ÇİÇEKLİ KOÇ
Ege University,
goksu.cicekli.koc@ege.edu.tr,
https://orcid.org/0000-0002-7165-7312

Assist. Prof. Dr. Beril CEYLAN NEBİOĞLU
Ege University,
beril.ceylan.nebioglu@ege.edu.tr,
https://orcid.org/0000-0002-2440-0434

Introduction

The adoption of Artifical Intelligance (AI) based technologies in education is not only creating a pedagogical and functional renewal in teaching processes, but is also reshaping discussions about the professional role of teachers, their

* The Contribution of Artificial Intelligence Training to the Educator Identity of Pre-service Teachers" (tracking number SB-GAP-2026-33491), supported by the Ege University Scientific Research Projects Coordination Unit.

function in the classroom, and their professional identity. AI can be considered a comprehensive field of technology based on modeling abilities associated with the human mind, such as learning, perception, problem-solving, reasoning and language comprehension/processing, through computer systems. From this point, AI is not a tool that performs technical functions; it stands out as a multidimensional system that reshapes the ways in which data is interpreted, decisions are made, and problems are solved in different fields (Avcı, 2024). According to Shao and Sun (2025), AI technologies present both opportunities and challenges for teachers; in particular, they necessitate a reassessment of the understanding of teaching in the context of role identity, professional identity, and professional development. Isma'il and Ibrahim (2025) show that AI integration does not completely eliminate the role of the teacher; instead, it repositions the teacher as a facilitator/guide who directs learning processes, supports the student, and evaluates the suitability of AI outputs to the classroom context with their professional judgment. In terms of ethics and safety, it is also emphasized that teachers should use technology responsibly within a human-machine collaboration framework. Therefore, AI should be considered not as a teacher replacement, but rather as an educational technology that creates transformative effects on teacher identity, professional autonomy, pedagogical decision-making, and classroom functions.

This transformation necessitates a rethinking of the scope and priorities of teacher training. Preparing prospective teachers for AI should not be viewed solely as a process of acquiring skills in the use of technical tools. Dilek et al. (2025) emphasize that addressing AI literacy in teacher education, along with AI concepts, ethical evaluation, and context-specific pedagogical practices, supports educators in engaging with AI in a critical, reflective, and participatory manner. Tripathi et al. (2025), in their study with K-12 teachers, reveal that technical competencies alone are insufficient for using AI; critical evaluation, ethical use, teacher autonomy, and pedagogical decision-making capacity need to be strengthened. Therefore, teacher education should be constructed in a way that trains prospective teachers to become not only users of AI, but also conscious practitioners who can critically examine it in light of their own values, teaching environments, and pedagogical goals.

Studies in the Turkish context also make it meaningful to discuss the inclusion of AI in teacher training processes as a multi-dimensional area of transformation. Akdeniz and Özdinç (2021) show that publications in the field of AI in education have increased in recent years in studies focusing mostly on intelligent tutoring systems, and that the study groups mainly consist of university

students. Akyel and Tur (2024) emphasize that AI offers significant opportunities in terms of improving teaching methods, individualized learning, student assessment, and curriculum development; however, the use of AI in education should be considered together with ethical and practical issues such as data privacy, equality, access, and technological literacy. Therefore, AI training should be considered not merely a technological update in teacher training programs, but as a field of development with pedagogical, ethical, and theoretical dimensions. The use of AI in education can contribute to teaching-learning processes in various ways, including individualized learning environments, real-time feedback, learning analytics, accessibility, support in management processes, simulation-based applications, and teacher assistance systems. However, the use of these technologies in education should be considered along with the risks involved, such as privacy protection, data security, ethical responsibilities, access inequalities, the reshaping of the teacher's role, the production of erroneous information, and the weakening of critical thinking skills (Altunay & Değirmenci, 2024).

The current literature shows that studies originating from Turkey are particularly concentrated around intelligent tutoring systems and AI applications in education (Akdeniz & Özdinç, 2021). The functional contributions that AI can provide to teaching processes are discussed in areas such as the development of teaching methods, individualized learning, student assessment, curriculum development, and learning analytics (Akyel & Tur, 2024). In addition, studies examining teachers' perceptions, literacy, usage patterns, and competencies regarding AI are increasing (Ning et al., 2025; Tripathi et al., 2025). While this trend is significant, the impact of AI training on how prospective teachers position themselves as educators, the ethical and professional principles they base their pedagogical decisions on, and how they establish their professional autonomy within the changing educational ecosystem is addressed relatively limitedly. Dilek et al. (2025) emphasize that AI literacy should provide prospective teachers with a critical and reflective perspective in this context; while Mouta et al. (2025) draw attention to the importance of strengthening teacher autonomy in the face of ethical issues in AI-assisted educational environments.

In this section, the concept of "educator identity" is not treated as an independent theoretical construct, but rather as an interpretive meta-conceptualization developed from the literature on teacher identity and teacher professional identity. This conceptual foundation is supported by studies focusing on pre-professional teacher identity (Arpacı & Bardakçı, 2015), research relating teacher identity to dimensions such as professional self-perception, dedication,

competence, duty perception, and motivation (Aydın & Aslan, 2024), and studies examining trends in the teacher professional identity literature (Özakyüz & Güneş, 2022). This framework points to an expanded professional identity structure for prospective teachers, encompassing not only their professional self-perception but also their capacity for pedagogical decision-making, ethical reasoning, designing learning processes in a human-centered manner, and maintaining professional autonomy in the face of transformative technologies such as AI. Indeed, Isma'il and Ibrahim (2025) argue that AI redefines the professional identity of teachers; Shao and Sun (2025) highlight the creation of new pressures and transformative areas on the teacher role and professional development; and Mouta et al. (2025) emphasize that ethical issues in AI-assisted educational processes should be considered in conjunction with teacher autonomy.

This study is based on the literature on teacher identity (Aydın & Aslan, 2024) and pre-service/pre-professional teacher identity (Arpacı & Bardakçı, 2015; Çelik & Kalkan, 2019). AI literacy is conceptually addressed within the framework of general AI literacy (Long & Magerko, 2020), while its implications for teacher education are considered as a complementary dimension in line with studies on critical, reflective, and participatory AI literacy and teacher education (Dilek et al., 2025; Daher, 2025). Accordingly, the theoretical foundations of teacher identity and educator identity are first explained, followed by a discussion of identity development in the pre-service period. Subsequent chapters address issues such as the transformation of the teacher role in the age of AI, ethical risks, and professional autonomy; and their relationships with current policies and ethical regulations in Turkey are evaluated. Finally, the relationship between AI education and educator identity development is discussed from a holistic perspective.

TEACHER IDENTITY AND EDUCATOR IDENTITY

Teacher identity is one of the fundamental conceptual structures in understanding the teaching profession. This concept allows us to explain how a teacher defines themselves professionally, how they perceive their teaching role, and how they structure their professional position. Examining teacher identity enables the simultaneous evaluation of dimensions such as teacher self-concept, job perception, self-efficacy, professional commitment, and motivation (Aydın & Aslan, 2024). In this respect, teacher identity is closely related not only to

what the teacher does, but also to how they make sense of their profession.

Teacher identity should not be viewed as a fixed and completed structure; rather, it should be considered a dynamic structure that develops over time, is shaped by context, and is reconstructed through experience (Beijaard et al., 2004). The pre-service period is one of the important stages in which the foundations of this development are laid. During this period, prospective teachers make sense of their professional selves around the questions of how they currently define themselves as teachers and what kind of teacher they aim to become (Arpacı & Bardakçı, 2015). This questioning encompasses not only the prospective teachers' perceptions of their professional roles but also their pedagogical values, ethical responsibilities, and future professional positioning.

The educator identity can be considered as an expanded interpretive framework that makes visible the dimensions of pedagogical reasoning, ethical responsibility, professional autonomy, and human-centered decision-making within the teacher identity. This expansion is particularly significant in the age of AI because the competencies expected of educators are not limited to technical knowledge. AI literacy, critical-reflective engagement, and contextually sensitive pedagogical practices are considered among the fundamental components in the context of teacher education (Dilek et al., 2025). Furthermore, it is emphasized that critical evaluation, ethical use, and teacher autonomy need to be strengthened in the use of AI (Tripathi et al., 2025). It is also stated that ethical issues in AI-assisted educational environments should be addressed together with teacher autonomy and professional learning processes (Mouta et al., 2025).

In parallel with this purpose theoretical framework, the distinction between teacher identity and educator identity is summarized in Table 1:

Tablo 1. Conceptual distinction between teacher identity and educator identity

Dimension	**Teacher ID**	**Educator ID**
Conceptual focus	Professional identity structure is related to professional self, task/role perception, self-efficacy, motivation, professional commitment, and self-definition as a teacher (Aydın & Aslan, 2024); it is related to the pre-service teacher identity development of prospective teachers in the pre-service context (Arpacı & Bardakçı, 2015).	An interpretive overarching framework that makes visible pedagogical reasoning, ethical responsibility, professional autonomy, and contextually sensitive professional positioning, based on the literature on teacher identity.
The fundamental question	The prospective teacher's self-understanding of their current and future teacher identity (Arpacı & Bardakçı, 2015).	Prospective teachers' ability to make pedagogical decisions, assume ethical responsibility, and maintain professional autonomy in the face of transformative technologies such as AI (Dilek et al., 2025; Mouta et al., 2025).
Key components	Self-efficacy, motivation, sense of duty, professional commitment, and teacher self-image (Aydın & Aslan, 2024).	Critical evaluation, ethical awareness, contextually sensitive pedagogical practice, professional autonomy, and AI literacy (Dilek et al., 2025; Mouta et al., 2025).
Meaning in the pre-service period	The formation of self-concept, role expectations, and teacher orientation related to the profession (Arpacı & Bardakçı, 2015).	The development of teacher identity orientations related to technology among prospective teachers (Can & Karacan, 2021) and the re-examination of these orientations from critical, pedagogical and ethical perspectives in the context of AI (Dilek et al., 2025).

Dimension	Teacher ID	Educator ID
Its equivalent in the context of AI	Considering AI as a factor reshaping the teaching role, teacher-AI interaction, and professional identity (Isma'il & Ibrahim, 2025); discussing new pressures and areas of transformation on the teaching role, professional identity, and professional development (Shao & Sun, 2025).	Evaluating AI with its critical, pedagogical and ethical implications (Dilek et al., 2025); questioning it in the context of ethical risks such as data privacy, data leakage, algorithmic bias, black box algorithms and lack of accountability (Zhu et al., 2025); and positioning it in the context of digital resources, teaching-learning, assessment and learner support (Zou et al., 2025).
Function in terms of the department	It is the starting point of the professional identity debate; it provides the basic framework that explains the prospective teacher's self-understanding of the profession and their orientation towards becoming a teacher (Arpacı & Bardakçı, 2015; Aydın & Aslan, 2024).	This study serves to explain the critical, pedagogical, and ethical dimensions of AI education; and to interpret the relationship between these dimensions and the professional autonomy and educator identity development of prospective teachers (Dilek et al., 2025; Mouta et al., 2025; Zhu et al., 2025; Zou et al., 2025).

Note: This table is based on an interpretive synthesis of the literature on teacher identity and discussions on AI education, ethical responsibility, and professional autonomy.

This distinction demonstrates that in the age of AI, teacher education cannot be limited to technical competencies. AI education should support prospective teachers in evaluating these technologies from a critical, reflective, pedagogical, and ethical perspective (Dilek et al., 2025). In this process, pedagogical decision-making, assuming ethical responsibility, and developing professional autonomy should also be among the fundamental components of teacher education (Mouta et al., 2025). Therefore, the educator identity can be considered as an expanded professional identity framework that goes beyond technical competence and includes ethical reasoning, pedagogical decision-making, and professional autonomy.

CONSTRUCTION OF TEACHER IDENTITY IN THE PRE-SERVICE PERIOD

The pre-service period is a critical stage where the foundations of the teacher's identity are laid and the self-perception regarding the profession is shaped. Nowadays, where the impact of digitalization on educational processes is increasing, it is becoming even more important for prospective teachers to develop a professional identity compatible with contemporary educational conditions (Can & Karacan, 2021). In this context, AI education should be considered not only as a technical learning area that improves the digital competencies of prospective teachers, but also as a component of teacher education that supports them in evaluating AI from pedagogical, ethical, and critical perspectives (Dilek et al., 2025; Daher, 2025).

Research conducted with prospective teachers shows that pre-service identity development is not a one-way and linear process. In the study by Çelik and Kalkan (2019), it was determined that prospective teachers had a high level of perception of their teacher identity during the pre-service period, and that these perceptions differed according to variables such as gender, grade, department, and faculty. The research results reveal that the professional identity of teachers should be considered in the design of teacher education programs. In contrast, the study conducted by Ulubey et al. (2018) with pedagogical formation students shows that the perception of teacher identity does not always increase at the end of the program, and that education does not automatically have a positive effect on this perception. When these findings are evaluated together, it can be said that pre-service identity development cannot be explained solely by participation in the program; it should be considered together with the quality of the program, its content, and the learning experiences it offers to prospective teachers.

Research on potential teacher selves also points to the multidimensional structure of pre-service identity. In the paper by Tavşanlı and Saraç (2016), it was stated that prospective primary school teachers had high perceptions of learning to teach and professionalism in the expected potential teacher self-dimension; however, they had low perceptions of being an indifferent teacher, inadequate classroom management, and uncreative teaching in the feared potential teacher self-dimension. This finding shows that prospective teachers interpret their future professional selves not only through the teaching characteristics they want to achieve, but also through the negative teaching possibilities they want to avoid. This situation reveals that pre-service teacher identity is shaped not only by current competencies but also by future

expectations and anxieties.

The relationship between pre-service teacher identity and technological transformation processes is noteworthy. Can and Karacan (2021) state that after information and communication technology training, prospective teachers develop technologically competent, productive, modern, and social teacher identity orientations. Çengelci (2021), on the other hand, shows that there are significant and positive relationships between pre-service teacher identity and 21st-century learner skills and metacognitive awareness. These findings show that in the age of AI, teacher identity should be considered together with digital competencies and learning management skills.

Özakyüz and Güneş (2022), examining national postgraduate theses in the field of teacher professional identity, state that studies mostly focus on teachers› identity perceptions and identity development. This trend indicates the formation of a research accumulation in the teacher identity literature centered on perception and development. However, how AI training affects the pedagogical reasoning, ethical sensitivity, and professional autonomy dimensions of prospective teachers can still be considered an area requiring research. Dilek et al. (2025) draw attention to the critical and ethical dimensions of AI literacy in teacher education; while Mouta et al. (2025) emphasize the need to strengthen teacher autonomy in the face of ethical problems in AI-assisted educational environments. These assessments show that AI training cannot be addressed solely through technical competencies in teacher identity research.

Teacher identity appears to be a dynamic structure related to multiple components such as teacher self-concept, perception of duty, self-efficacy, and professional commitment (Aydın & Aslan, 2024), potential teacher self-concept (Tavşanlı & Saraç, 2016), technologically competent teacher identity, and modern teacher identity orientations (Can & Karacan, 2021), as well as 21st-century learner skills and metacognitive awareness (Çengelci, 2021). This multi-component structure reveals that in the age of AI, the professional identity development of prospective teachers should be discussed not only through current competencies but also through their future roles as educators. Therefore, AI education can offer a theoretical and pedagogical context that can influence not only the digital skills of prospective teachers but also their professional self-understanding regarding how they can position themselves as educators in the future.

AI LITERACY IN TEACHER EDUCATION

AI literacy is becoming an increasingly central concept in teacher education discussions. Long and Magerko (2020) define AI literacy as a multidimensional competency area that includes not only the use of AI technologies but also questioning their operation, critically evaluating their outputs, and using them with conscious decisions in different areas of life. This approach shows that AI literacy goes beyond tool-using skills; it also includes understanding how AI works, interpreting outputs, evaluating human-AI interaction, and using technology appropriately within the context.

Dilek et al. (2025) emphasize that AI literacy in teacher education is not limited to technical competencies; it should be considered together with critical evaluation, ethical awareness, pedagogical integration, and contextually sensitive decision-making processes. Similarly, according to Daher (2025), AI literacy requires a deep understanding of the operation, application areas, and ethical implications of AI. This approach positions AI literacy as a new professional competency area in teacher education that goes beyond technical skills and encompasses critical, pedagogical, and ethical dimensions. Therefore, incorporating AI literacy into teacher education programs is a crucial requirement for prospective teachers to be able to evaluate technology within the context of pedagogical decision-making and ethical responsibility.

New competency frameworks developed for teachers also point to the multidimensional nature of AI literacy. The pedagogical AI competency framework proposed by Zou et al. (2025) consists of areas such as AI-supported digital resources, AI-enhanced teaching and learning, AI-enhanced assessment, and AI-supported learners. This framework shows that existing digital competency models such as TPACK and DigCompEdu offer an important foundation, but these models need to be expanded to meet the specific educational characteristics of AI technologies. Therefore, AI literacy can be considered a holistic competency area requiring a rethinking of teacher competencies in terms of teaching, assessment, digital resource use, and learner support. AI-supported instructional systems have the potential to create more adaptable learning experiences according to students' learning speeds, individual needs, and readiness levels. In this context, adaptive learning environments, recommendation systems, virtual reality, and metaverse-based applications; It can contribute to designing the learning process in a more flexible, interactive and student-centered way (Etçi, 2024).

Critical approaches to AI literacy are particularly important for teacher train-

ing. Dilek et al. (2025) note that AI is often framed as a tool for automation, while its pedagogical and ethical consequences are discussed more limitedly. The critical collaborative discovery approach developed by the researchers is based on the principles of collective inquiry, critical reflection, participatory interaction, and collaborative knowledge production. This approach can contribute to the development of participants in the teacher training process not as passive users of AI, but as educators who can critically examine it from a pedagogical and ethical perspective.

The relationship between AI literacy and ethical technology use is also supported by research findings. In the research of Muzaffer and Ünal (2025), it was determined that there is a positive and moderate correlation between the AI literacy levels of teachers and their ethical use of information technologies. When these findings are evaluated together, AI literacy can be considered as a multidimensional competency area for teacher education, encompassing not only technical and applied competencies but also ethical awareness, pedagogical reasoning, and critical evaluation dimensions. In line with this theoretical framework, the technical, pedagogical, ethical, and professional identity-related dimensions of AI literacy in the context of teacher education are summarized in Table 2:

Tablo 2. Key dimensions of AI literacy in the context of teacher education

Dimension	**Scope**	**Relationship with her identity as an educator**
Technical dimension	Knowing the basic concepts of AI and its use in different contexts (Long & Magerko, 2020); evaluating the operating principles, application areas, possibilities, limitations, and ethical consequences of AI (Daher, 2025).	Encouraging the informed and critical evaluation of technology (Long & Magerko, 2020; Daher, 2025).
Pedagogical dimension	Connecting AI with digital resources, teaching-learning processes, assessment, and learner support (Zou et al., 2025).	Supporting the pedagogical integration and context-sensitive use of AI (Dilek et al., 2025; Zou et al., 2025).

Dimension	**Scope**	**Relationship with her identity as an educator**
Ethical dimension	Assessing issues of data privacy, algorithmic bias, transparency, equality, and accountability (Zhu et al., 2025).	Strengthening ethical awareness (Dilek et al., 2025); supporting professional autonomy and responsible decision-making processes in the face of AI-related ethical issues (Mouta et al., 2025; Zhu et al., 2025).
Identity and professional autonomy dimension	Adopting a critical and reflective stance towards AI (Dilek et al., 2025); protecting and developing professional autonomy in the face of ethical issues (Mouta et al., 2025).	Supporting prospective teachers' technology-related teacher identity orientations (Can & Karacan, 2021); strengthening the development of professional autonomy and pedagogical initiative (Mouta et al., 2025).

Note: The table is prepared as an interpretive synthesis showing the relationship between AI literacy and teacher education and educator identity.

AI literacy, in terms of teacher education, cannot be limited solely to tool-using skills (Long & Magerko, 2020; Daher, 2025). This competency requires evaluating AI in a pedagogical, ethical, critical, and reflective manner (Dilek et al., 2025), as well as strengthening professional autonomy and pedagogical initiative (Mouta et al., 2025). Therefore, AI literacy can be considered an important component of the contemporary educator identity.

THE TRANSFORMATION OF THE TEACHER ROLE AND EDUCATOR IDENTITY IN THE AGE OF AI

The inclusion of AI-powered tools in teaching processes is bringing about a reassessment of the teacher's role and professional identity. In the study by Isma'il and Ibrahim (2025), it is observed that AI integration transforms teachers' perceptions of their professional identity; this transformation is shaped around the themes of "teacher-AI partnership," "the tension between teacher autonomy and standardized instruction," and "a shift from traditional teaching to

facilitation." The study also indicates that teachers initially feared role loss, but over time came to view AI not as a replacement for teaching, but as a tool that supports it. These findings demonstrate that in the age of AI, teaching cannot be limited solely to the role of knowledge transfer; it must be rethought with responsibilities such as pedagogical design, guidance, support, and the purposeful use of technology. The effective use of AI in education does not mean the teacher becomes dysfunctional or their role disappears. On the contrary, this process makes the teacher's responsibility for guidance, pedagogical decision-making, and directing the learning process more prominent. Although AI-powered systems contribute to learning, the primary decision regarding the purpose, timing, and pedagogical rationale for using these tools should belong to the teacher. Therefore, it is important for teachers to have sufficient knowledge and training on the possibilities, limitations, and educational impacts of AI (Kalafat, 2024).

According to Tripathi et al. (2025), a significant portion of research on AI integration focuses on the functionality and potential applications of AI, rather than on teachers' beliefs and classroom experiences regarding AI. The researchers also note that AI integration leads to a renegotiation of teachers' professional beliefs, pedagogical identities, classroom authority, and perceptions of professional autonomy, beyond the superficial use of tools. This assessment suggests that the focus in AI education should shift not only to the use of technical tools but also to the transformations occurring in the teacher's pedagogical role and professional positioning.

According to Shao and Sun (2025), in the age of AI, teacher identity is transforming within the framework of new challenges and opportunities in terms of role, professional identity, and professional development. In this process, the weakening of role authority, threats to professional status, and pressure on professional skills stand out as key elements challenging teacher identity. Conversely, the age of AI also offers new opportunities for teachers in terms of role transformation, technological empowerment, and professional development. Researchers suggest that teachers can restructure their professional identities by re-evaluating their role orientations, improving their technology skills, enhancing their professional qualifications, engaging in human-machine collaboration, and considering ethical and safety principles.

Studies on the identity of language teachers also highlight the pedagogical dimension of this transformation. Ghiasvand and Seyri (2025) state that AI affects the identity of language teachers in terms of transforming teacher roles, pedagogical adaptation and development, data-driven and individualized

language teaching, strengthening professional expertise, developing self-reflective practice, and the formation of a technology-compatible teacher identity. According to the researchers, AI transforms not only the way teachers use technology but also their pedagogical practices and professional self-assessment processes. When these findings are considered together, it can be said that in AI-supported teaching processes, teacher identity should be addressed not only through risks or technical skills but also in conjunction with professional renewal, pedagogical empowerment, professional development, and self-reflective thinking processes.

Teachers' beliefs about AI should not be viewed as one-sided. According to Tripathi et al. (2025), while teachers acknowledge the potential of AI to support learning and increase administrative efficiency, they are concerned about the risks of creating over-reliance in students and weakening critical thinking skills. This finding shows that teachers neither completely reject nor unconditionally embrace AI; rather, they evaluate it in terms of pedagogical goals, student learning, and professional responsibilities. At this point, the teacher's role shifts from being a user of technology to a pedagogical decision-maker who discerns which use supports learning. Although AI systems are seen as tools that can improve teaching, learning, and assessment processes, this contribution only becomes meaningful when used consciously, moderately, and under pedagogical supervision. Careless, unquestioning, or excessive use of these tools can negatively impact students' decision-making, critical thinking, and self-regulation skills (Gündoğan Önderöz, 2025).

In the age of AI, the human aspects of teaching are becoming even more prominent. According to Tripathi et al. (2025), teachers believe that AI cannot completely replace human-specific qualities such as care, emotional support, and a sensitive approach to the individual needs of students. Similarly, Shao and Sun (2025) emphasize that while AI can take over some teaching tasks, it cannot fully fulfill the roles of emotional communication, humane care, and individualized guidance. These assessments show that in AI-assisted educational environments, the teacher retains their importance not only as a transmitter of information but also as an educator who understands the student, offers emotional support, and guides learning processes in a human-centered way.

The researcher-teacher identity also offers a complementary framework in understanding the transformation of the teacher identity. Yıldırım's (2018) study reveals that teachers' active participation in educational research and research training processes can contribute to both the education system and their professional development. In the age of AI, this identity gains even more im-

portance with its dimensions of inquiry, working with data, evaluation, critical thinking, and continuous professional learning. In this context, the educator identity refers not only to someone who uses technology, but also to a professional who evaluates it from pedagogical, ethical, and human dimensions, and critically redesigns learning processes.

ETHICAL RESPONSIBILITY AND PROFESSIONAL AUTONOMY IN THE CONTEXT OF ARTIFICIAL INTELLIGENCE

When discussing the relationship between AI training and the educator identity of prospective teachers, the ethical dimension, pedagogical autonomy, and professional autonomy can be considered together. According to Mouta et al. (2025), professional autonomy is related to the teacher's capacity to make pedagogical decisions in AI-assisted learning environments, to maintain their pedagogical initiative in the face of automated decision-making systems, and to direct teaching processes. In the age of AI, this capacity seems important in terms of the teacher questioning the suggestions of technological systems, protecting their pedagogical decisions, and observing their ethical responsibilities. When AI is used as an auxiliary system to support the teacher, it can increase the teacher's pedagogical options in lesson planning, providing feedback, differentiating instruction, and evaluation processes. However, leaving the decision-making process largely to technical systems risks limiting the teacher's professional judgment and autonomy. Therefore, prospective teachers should not view the AI output as a decision to be directly implemented; They should view it as an initial proposal that will be re-evaluated in line with student benefit, contextual appropriateness, ethical principles and teacher responsibility (Atalay, 2025).

Zhu et al. (2025) classify the ethical risks related to AI in education into three dimensions: technology, education, and society. The researchers highlight the following risks: in the technology dimension, privacy violations, data leakage, algorithmic bias, black box algorithms, and algorithmic errors; in the education dimension, homogenized student development, uniform instruction, deviation from educational goals, alienation of the teacher-student relationship, crisis of the teaching profession, emotional distress, and academic inappropriateness; and in the society dimension, the deepening digital divide, lack of accountability, and conflicts of interest. These assessments demonstrate that the use of AI should be

considered not only in terms of functionality and efficiency, but also in terms of ethical responsibilities and potential harms. The ethical evaluation of AI tools cannot be limited solely to the protection of personal data or privacy issues. The emergence of algorithmic biases, insufficient transparency in decision-making processes, the production of false or misleading content, the weakening of human decision-making autonomy, the ambiguity of who is responsible, security vulnerabilities, the possibility of misuse, and intellectual property/copyright issues are among the multifaceted ethical problems of this field (Erümit, 2025).

Discussions in the context of ethics and teacher autonomy point in a similar direction. According to Atalay (2025), ethical and responsible use of AI in education requires consideration of issues such as moral responsibility, bias, transparency, privacy, and data security. Furthermore, it is important to position AI as a tool that supports individualized instruction and teacher judgment, but does not replace them. In this regard, transparency, equality, pedagogical balance, and human-centered design principles can be considered fundamental components of ethical use in AI-supported educational environments. This approach demonstrates that protecting teacher autonomy and pedagogical decision-making processes is of central importance in the use of AI. The critical approach in the context of digital labor and teacher identity further deepens this discussion. According to Eraslan (2025), the digitalization process repositions the teacher not only as a pedagogical actor but also as a data provider, content producer, and subject evaluated through algorithmic performance metrics. This transformation may impact the ethical and technical foundations of the teacher's identity, potentially relegating the human aspects of teaching, such as pedagogical intuition, emotional support, and professional autonomy, to the background. Therefore, AI training should equip prospective teachers not only with tool-using skills but also with the ability to analyze the impact of digital systems on teachers' labor and pedagogical autonomy. Thus, AI training can be considered a learning environment that supports prospective teachers in questioning technological systems not only at the application level but also within the context of their professional identity and ethical responsibilities.

Mouta et al. (2025) emphasize that, in the context of AI and educational ethics, teachers should be supported not only with technical knowledge but also with dimensions of ethical evaluation, professional autonomy, and collective professional initiative. In this regard, professional learning communities, collaborative inquiry, case studies, and ethical dilemma discussions can enhance pre-service teachers' ethical and pedagogical reflection skills. Thus, AI education can be structured as a learning environment that supports pre-service

teachers not only in using these tools but also in critically evaluating their impact on students, teacher labor, equality, and societal outcomes.

AI EDUCATION AND TEACHER TRAINING POLICIES IN TURKEY

The debate on AI education in Turkey is not limited solely to technological and pedagogical trends in the literature. Policy and ethics documents published by the Ministry of National Education (MEB) also position human-centered approaches, transparency, accountability, data security, ethical responsibility, and human oversight as fundamental principles in AI-supported educational processes (MEB, 2025a, 2025b, 2026). In particular, the Ethical Guide for AI Applications in Education emphasizes that AI technologies should not replace teachers, but rather be used as tools that support teaching-learning processes, strengthen human interaction, and serve pedagogical purposes (MEB, 2026).

The AI in Education Policy Document and Action Plan indicates that, from a teacher training perspective, AI literacy should be considered a holistic competency area related to subject matter education, instructional design, measurement and evaluation, digital content production, and learning analytics (MEB, 2025a). This framework necessitates training prospective teachers to become educators who can relate AI to pedagogical purposes, adapt it to the classroom context, and evaluate it in conjunction with students' developmental characteristics.

The Ministry of National Education's (MEB) AI Applications Ethics Board Directive and the Ethics Guide for AI Applications in Education make the ethical and governance dimensions of this transformation visible. The Directive envisages the establishment of boards at various levels and school AI ethics teams within the Ministry to determine, monitor, and evaluate the ethical principles related to AI applications (MEB, 2025b). The Ethics Guide, on the other hand, provides an ethical framework for the use of AI in educational environments in line with the principles of a rights-based and human-centered approach, transparency and explainability, reliability, equality and inclusivity, privacy and data governance, accountability, human supervision, and educational values (MEB, 2026). These regulations show that prospective teachers should be trained not only as users of AI but also as educators who assume ethical responsibility and can justify their pedagogical decisions.

In this context, it is important that AI education in teacher training programs is structured in a way that is compatible with the MEB's policies and ethical regulations. Prospective teachers; In addition to recognizing the possibilities of

AI tools, educators must also assume responsibility for decision-making on issues such as data security, algorithmic bias, academic integrity, student privacy, equal access, and human supervision. Such an approach supports the development of the educator's identity within the context of national policy, encompassing technical, pedagogical, ethical, and professional autonomy.

AI EDUCATION AND THE DEVELOPMENT OF THE EDUCATOR'S IDENTITY

The current perceptions and usage practices of prospective teachers regarding AI point to the importance of systematic AI education. The diversity in prospective teachers' perceptions of AI shows that this education cannot be addressed solely at the level of acquiring technical skills. The fact that AI is sometimes perceived as an educational element that facilitates learning, sometimes as a technological tool, and sometimes as an abstract structure associated with the human mind, necessitates that conceptual awareness, pedagogical use, and critical evaluation be considered together in AI education (Taşyürek Mutlu & Metin Göksu, 2025). In Topçubaşı's (2025) study with prospective primary school teachers, it was determined that the participants generally had a positive attitude towards AI technologies and evaluated these technologies as effective tools supporting the teaching process. Prospective teachers found AI useful in content development, planning, individualized instruction, and evaluation processes; however, they exhibited critical awareness regarding issues such as content reliability, lack of emotional interaction, and ethical problems. These findings suggest that addressing the dimensions of technical use, ethical evaluation, and critical thinking together in AI training may be beneficial.

Research focusing on the use of generative AI in lesson planning processes presents a similar picture. Kerr and Kim's (2025) study shows that prospective teachers utilize generative AI in stages such as topic selection, material production, lesson organization, and language control. However, the researchers note that while this support can provide efficiency and creativity, it may also present some problems in terms of output quality and the limited use of effective pedagogical strategies. Therefore, it seems more appropriate to evaluate the relationship with AI not only in terms of usage competence but also in terms of its use in accordance with pedagogical goals, critical evaluation, and ethical responsibility. In teacher education, AI tools can facilitate areas such as lesson planning, production of teaching materials, diversification of the measurement

and evaluation process, monitoring student development, and individualizing the learning process for prospective teachers. However, the outputs obtained from these tools should not replace the teacher's pedagogical decision; prospective teachers should critically evaluate the process, taking into account the interests, needs, readiness, and context of the students (Köylüdür & Öner Armağan, 2025).

Studies examining teachers' adoption processes of AI (AI) reveal the conditions under which this relationship can be strengthened. According to Güngör and Öztürk (2025), K-12 teachers' experiences regarding AI integration are shaped by personal factors, curricula, policies, dilemmas, uncertainties, and pedagogical conditions. This finding highlights the need to evaluate AI integration not only in terms of individual attitudes but also in conjunction with school and system-level conditions. The study by Arranz-García et al. (2025) indicates that integration can be hampered by inadequate training, lack of resources, infrastructure problems, and limited institutional support. This result suggests that teachers need supportive professional learning environments for the use of AI. The findings of Demir and Beyazhançer (2024) also show that receiving training on AI and previous experience using it may be related to prospective teachers' self-efficacy perceptions.

When these findings are considered together, it can be said that the relationship between AI education and educator identity can be established at four levels. The first level is technical competence: The prospective teacher recognizes, uses, and understands the limitations of AI tools. This level can be related to the basic competencies of AI literacy (Long & Magerko, 2020; Ning et al., 2025). The second level is pedagogical reasoning: The prospective teacher relates the use of AI to teaching objectives, student needs, and learning outcomes. This level can be considered within the framework of pedagogical AI competence (Zou et al., 2025). The third level is ethical responsibility: The prospective teacher considers data confidentiality, bias, academic integrity, equality, and human-centered education principles (Zhu et al., 2025). The fourth level is professional autonomy: The prospective teacher can position themselves not as a passive user of AI, but as an educator who makes decisions, questions, and adapts to the context (Mouta et al., 2025).

Based on this theoretical foundation, how AI training can contribute to the development of prospective teachers' educator identities in terms of technical competence, pedagogical reasoning, ethical responsibility, and professional autonomy is summarized in Table 3:

Tablo 3. The contribution of AI training to the educator identity

Level of development	**Main focus**	**Expected outcomes for prospective teachers**	**Relationship to the identity dimension**
Technical competence	Understanding AI tools, their use, and their basic operation (Long & Magerko, 2020); comprehending the workings, application areas, and ethical implications of AI (Daher, 2025).	Critically distinguishing between the possibilities and limitations of AI (Long & Magerko, 2020); recognizing the perception, knowledge-skills, application/ innovation, and ethical dimensions of AI (Ning et al., 2025); evaluating its ethical implications (Daher, 2025).	Being able to relate it to the development of a technologically competent prospective teacher identity (Can & Karacan, 2021).
Pedagogical reasoning	Integrating AI tools into teaching, learning, and assessment processes (Zou et al., 2025).	Conscious use in processes such as course design, material production, and language supervision (Kerr & Kim, 2025); linking it to learner support and assessment (Zou et al., 2025).	To create a foundation for the development of an educator identity that justifies pedagogical decisions and uses AI appropriately within the context (Dilek et al., 2025; Zou et al., 2025).

Level of development	**Main focus**	**Expected outcomes for prospective teachers**	**Relationship to the identity dimension**
Ethical responsibility	Assessing risks such as data privacy, algorithmic bias, transparency, equality, accountability, and academic misconduct (Zhu et al., 2025). Protecting teacher autonomy and pedagogical initiative in the face of ethical issues (Mouta et al., 2025).	Raising awareness on issues of data privacy, algorithmic bias, equality, academic integrity, and accountability (Zhu et al., 2025).	Supporting ethical awareness and responsible use (Dilek et al., 2025); strengthening professional autonomy in the face of ethical issues (Mouta et al., 2025).
Professional autonomy	Developing an effective, critical, and decision-making teacher position in the face of AI (Mouta et al., 2025); critically and reflectively evaluating the use of AI (Dilek et al., 2025).	Adapting, questioning, and, when necessary, limiting AI according to context (Mouta et al., 2025); structuring it in a way that is suitable for teaching and learning processes (Zou et al., 2025).	Being able to associate it with the development of an autonomous and decision-making educator identity (Mouta et al., 2025); supporting a critical and reflective educator identity (Dilek et al., 2025).

Note: This table has been prepared to demonstrate the potential contributions that AI training for prospective teachers can offer to the development of *their educator identity, starting from technical competence and extending to pedagogical reasoning, ethical responsibility, and professional autonomy.*

This integrated framework demonstrates that AI education for prospective teachers cannot be considered merely a process of adapting to technology. AI literacy is a multidimensional competency area requiring an understanding of how AI operates, its application areas, and its ethical implications (Daher, 2025). Therefore, AI literacy in teacher education should be addressed not only through technical skills but also through critical, reflective, and ethical evaluation processes (Dilek et al., 2025). Furthermore, ethical issues and professional autonomy in AI-assisted learning environments should be considered among the fundamental components of teachers' professional learning (Mouta et al., 2025).

At this point, AI education in teacher training programs can be structured not only as an independent technical course, but also as a holistic content integrated with courses such as subject matter education, instructional design, measurement and evaluation, ethics, classroom management, and teaching practice. Zou et al.'s (2025) pedagogical AI competency framework also addresses the areas of AI-supported digital resources, AI-enhanced teaching and learning, AI-enhanced assessment, and AI-supported learners. This framework shows that the relationship that prospective teachers establish with AI should not remain only at the individual use level; it should be extended to the dimensions of instructional design, assessment, learner support, and professional responsibility. Such a structure can support prospective teachers not only in using AI, but also in evaluating this technology in line with pedagogical goals, ethical responsibilities, and educational values. Productive AI tools can save teachers time in areas such as creating different question types, speeding up the question preparation process, developing visually supported assessment tools, examining student responses, and diversifying assessment tools. However, when using these tools in the context of measurement and evaluation, exam security, accuracy of answers, suitability of criteria, and continuous teacher supervision must be carefully considered (Orhan Göksün & Kurt, 2025).

CONCLUSION AND RECOMMENDATIONS

When the existing literature is considered together, the relationship between AI education and the development of pre-service teachers' educator identities is becoming an increasingly central area of research and application. Research on teacher identity shows that the pre-service period is a decisive stage in terms of professional identity development (Arpacı & Bardakçı, 2015; Çelik & Kalkan,

2019). Studies in the context of technology education reveal that pre-service teachers can develop technologically competent and modern teacher identity orientations (Can & Karacan, 2021). These findings indicate that AI education should be considered together with pre-service teachers' professional self-perception and future educator roles. Current studies on AI literacy draw attention to the fact that teacher education cannot be considered limited only to pedagogical knowledge and subject matter knowledge. According to Long and Magerko (2020), AI literacy includes the competencies to understand, evaluate, and consciously use these technologies. Daher (2025) states that AI education should also encompass an understanding of how AI functions, its application areas, and its ethical implications. Zou et al. (2025) emphasize that teacher competencies should be considered in conjunction with AI-assisted instruction, assessment, and learner support dimensions. Ning et al. (2025) also demonstrate that teachers' AI literacy can be assessed holistically across perception, knowledge-skills, application/innovation, and ethical dimensions. This approach shows that prospective teachers should be trained not only to use AI but also to evaluate it from a pedagogical and ethical perspective.

The main contribution of this chapter is to discuss AI education beyond the technical competencies of prospective teachers, considering pedagogical reasoning, ethical responsibility, and professional autonomy. Thus, the chapter offers a holistic theoretical framework on how prospective teachers can develop an educator identity in the age of AI. In teacher training programs, AI can be reduced to simply teaching technical tools. Dilek et al. (2025) emphasize that AI literacy in teacher education should be integrated with critical, reflective, and participatory learning processes. Mouta et al. (2025) point out that teachers should be supported in pedagogical decision-making, ethical evaluation, and professional autonomy within the context of AI ethics. Zou et al. (2025) demonstrate that AI competence should be considered in conjunction with pedagogical areas such as teaching, learning, assessment, digital resource use, and learner support. This type of approach views AI not as a replacement for teaching, but as a means of integrating it into the learning process. This makes it possible to consider it a pedagogical framework that allows prospective teachers to rethink and reposition their educator identities within the context of contemporary education.

In this context, the following suggestions can be offered for teacher training processes:

- AI literacy can be considered not only as content based on tool use, but also as a competency area that includes technical, pedagogical, ethical,

and social dimensions.

- Teacher candidates can be supported in evaluating the limitations, potential errors, bias generation methods, and effects on student learning of AI tools, in addition to using them.
- Applications for the use of AI in lesson planning, material development, assessment, and feedback processes can be included in teacher training programs. These applications can be supported by ethical dilemma analyses and reflective discussions.
- To strengthen the professional autonomy of teacher candidates, a positioning of an educator who questions, makes decisions, and adapts to the context in the face of AI can be encouraged.
- AI training can be linked to teaching practice and field training courses to contribute to the development of pedagogical decision-making skills of teacher candidates in real teaching contexts.

At the application level, this framework can be concretized in teacher training programs with a three-stage integration. In the first stage, the fundamental concepts, working principles, and limitations of AI literacy can be introduced. In the second stage, activities can be conducted to develop AI-supported lesson plans, materials, and assessment tools in field education courses. In the third stage, the professional autonomy of prospective teachers can be supported through ethical dilemma analyses, human supervision, and pedagogical justification reports during the teaching practice process.

Since this section is a synthesis of theoretical literature, it does not present direct causal conclusions regarding the effects of AI training on pre-service teachers' educator identities. Therefore, it may be useful to examine the framework based on technical, pedagogical, ethical, and professional autonomy through experimental research in different teacher training programs. However, there is a need for a more detailed examination of the relationship between AI training and pre-service teachers' educator identities within the context of pre-service teacher training. Future research could comparatively and longitudinally examine the potential effects of AI-based learning experiences on pre-service teachers' pedagogical reasoning, ethical sensitivity, perceptions of professional autonomy, and educator identities. Such studies could make it clearer whether AI training can be considered merely a technical preparation area for pre-service teachers or a transformative learning experience that can support their educator identities.

REFERENCES

Akdeniz, M., & Özdinç, F. (2021). Eğitimde yapay zekâ konusunda Türkiye adresli çalışmaların incelenmesi. *Van Yüzüncü Yıl Üniversitesi Eğitim Fakültesi Dergisi, 18*(1), 912–932. https://doi.org/10.33711/yyuefd.938734

Akyel, Y., & Tur, E. (2024). Yapay zekanın potansiyelinin ve eğitim bilimlerindeki uygulamalarının araştırılması ve araştırmalarda beklentiler, zorluklar ve gelecek yönelimleri. *Ahi Evran Üniversitesi Kırşehir Eğitim Fakültesi Dergisi*, *25*(1), 645-711. https://izlik.org/JA43TE82NH

Altunay, F., & Değirmenci, Y. (2024). Eğitimde yapay zekâ ve kullanımı üzerine. İçinde Y. Değirmenci & F. Altunay (Ed.), *Farklı bakış açılarıyla eğitimde yapay zekâ* (ss. 1–44). Eğitim Yayınevi.

Arpacı, D., & Bardakçı, M. (2015). Meslek öncesi öğretmen kimliği ölçeğinin Türkçeye uyarlanması. *Gaziantep Üniversitesi Sosyal Bilimler Dergisi*, *14*(3), 687-719. https://doi.org/10.21547/jss.256762

Arranz-García, O., Romero García, M. del C., & Alonso-Secades, V. (2025). Perceptions, strategies, and challenges of teachers in the integration of artificial intelligence in primary education: A systematic review. *Journal of Information Technology Education: Research, 24*, Article 6, 1–48. https://doi.org/10.28945/5458

Atalay, A. (2025). Yapay zekâ çağında eğitimde etik ve öğretmen özerkliği. İçinde T. Y. Özdemir & M. Demirkol (Ed.), *Eğitim yönetimi alanında uluslararası derleme, araştırma ve çalışmalar* (ss. 41–56). Serüven Yayınevi.

Avcı, E. (2024). *Üretken yapay zeka ve prompt mühendisliği*. Seçkin Yayıncılık.

Aydın, F., & Aslan, M. (2024). Öğretmen kimliği algıları ve öğretmen kimlik algılarının demografik değişkenlere göre değişimi. *Marmara Üniversitesi Atatürk Eğitim Fakültesi Eğitim Bilimleri Dergisi*, *59*(59), 171-204. https://doi.org/10.15285/maruaebd.1359716

Beijaard, D., Meijer, P. C., & Verloop, N. (2004). Reconsidering research on teachers' professional identity. *Teaching and Teacher Education, 20*(2), 107–128. https://doi.org/10.1016/j.tate.2003.07.001

Can, T., & Karacan, C. G. (2021). Exploring pre-service English teachers' early teacher identity in relation to emerging ICT technologies training. *Novitas-ROYAL (Research on Youth and Language), 15*(1), 57–75.

Çelik, H. R., & Kalkan, Ö. K. (2019). Öğretmen adaylarının meslek öncesi öğretmen kimliği algıları: Pamukkale Üniversitesi örneği. *Ege Eğitim Dergisi*, *20*(2),

351-365. https://doi.org/10.12984/egeefd.628810

Çengelci, S. (2021). *Öğretmen adaylarının 21. yüzyıl öğrenen becerileri ile üstbilişsel farkındalık düzeylerinin meslek öncesi öğretmen kimliği üzerindeki etkisi (Afyonkarahisar örneklemi)* [Yüksek lisans tezi, Afyon Kocatepe Üniversitesi].

Daher, R. (2025). Integrating AI literacy into teacher education: A critical perspective paper. *Discover Artificial Intelligence, 5*, 217. https://doi.org/10.1007/s44163-025-00475-7

Demir, B., & Beyazhançer, R. (2024). İlköğretim matematik öğretmeni adaylarının yapay zekâ öz-yeterliklerinin bazı değişkenler açısından incelenmesi. *International Journal of Social and Humanities Sciences Research, 11*(113), 2393–2398. https://doi.org/10.5281/zenodo.14279357

Dilek, M., Baran, E., & Aleman, E. (2025). AI literacy in teacher education: Empowering educators through critical co-discovery. *Journal of Teacher Education, 76*(3), 294–311. https://doi.org/10.1177/00224871251325083

Eraslan, İ. (2025). Eğitimde dijital emek ve öğretmen kimliği: Yapay zekâ destekli sistemlerde emeğin dönüşümü. İçinde U. Büyük (Ed.), *Eğitimde teknoloji entegrasyonu: Yapay zekâ ve dijitalleşme perspektifleri* (ss. 1–47). Özgür Yayınları. https://doi.org/10.58830/ozgur.pub773

Erümit, A. K. (2025). Yapay zeka araçlarının kullanımının hukuki, etik, telif hakları ve mülkiyeti boyutu. İçinde V. Nabiyev & A. K. Erümit (Ed.), *Üretken yapay zeka ve uygulamaları* (ss. 281–296). Seçkin Yayıncılık.

Etçi, F. (2024). Eğitimde yapay zekâ uygulamaları. İçinde R. Karabulut & E. Türksoy (Ed.), *Eğitimde yapay zekâ* (ss. 185–212). Eğiten Kitap.

Ghiasvand, F., & Seyri, H. (2025). A collaborative reflection on the synergy of artificial intelligence (AI) and language teacher identity reconstruction. *Teaching and Teacher Education, 160*, Article 105022. https://doi.org/10.1016/j.tate.2025.105022

Gündoğan Önderöz, F. (2025). Yapay zekâ ve eğitimde kullanımı. İçinde B. Sezginsoy Şeker & Ü. İzgi Onbaşılı (Ed.), *Dijital çağın öğretmenleri: Yapay zekâ ile etkin ders tasarımı* (ss. 1–19). Nobel Akademik Yayıncılık.

Güngör, İ. C., & Öztürk, T. (2025). Öğretmenler yapay zekâyı benimsedi mi? K-12 eğitim sistemlerinden öğretmen deneyimlerinin bir sentezi. *Recep Tayyip Erdoğan Üniversitesi Sosyal Bilimler Dergisi, 12*(1), 70–100. https://doi.org/10.34086/rteusbe.1660052

Isma'il, A., & Ibrahim, H. B. (2025). Teachers' professional identity in the era of artificial intelligence: A phenomenological study. *International Journal of Pro-*

fessional Development, Learners and Learning, 7(2), Article e2517. https://doi.org/10.30935/ijpdll/17416

Kalafat, Ö. (2024). Eğitim ve yapay zekâ. İçinde M. Bilen (Ed.), *Yapay zekânın değiştirdiği dinamikler* (2. bs., ss. 89–102). Eğitim Yayınevi.

Kerr, R. C., & Kim, H. (2025). From prompts to plans: A case study of pre-service EFL teachers' use of generative AI for lesson planning. *English Teaching, 80*(1), 95–118. https://doi.org/10.15858/engtea.80.1.202503.95

Köylüdür, A., & Öner Armağan, F. (2025). Yapay zekâ ile ders planı tasarımı. İçinde M. Metin & K. Sungur Gül (Ed.), *Kuramdan uygulamaya öğretmen eğitiminde yapay zekâ araçları* (ss. 103–132). Pegem Akademi.

Long, D., & Magerko, B. (2020). What is AI literacy? Competencies and design considerations. İçinde *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (ss. 1–16). https://doi.org/10.1145/3313831.3376727

Millî Eğitim Bakanlığı. (2025a). *Eğitimde Yapay Zekâ Politika Belgesi ve Eylem Planı (2025–2029)*. Millî Eğitim Bakanlığı. https://yegitek.meb.gov.tr/meb_iys_dosyalar/2025_06/17092340_egitimdeyapayzekapolitikabelgesiveeylem-plani202520291.pdf

Millî Eğitim Bakanlığı. (2025b). *Millî Eğitim Bakanlığı Yapay Zekâ Uygulamaları Etik Kurulu Yönergesi*. Millî Eğitim Bakanlığı. https://ttkb.meb.gov.tr/meb_iys_dosyalar/2025_10/23133643_yapay_zeka_yonerge.pdf

Millî Eğitim Bakanlığı. (2026). *Eğitimde Yapay Zekâ Uygulamaları Etik Kılavuzu*. Millî Eğitim Bakanlığı. https://ttkb.meb.gov.tr/meb_iys_dosyalar/2026_01/09152326_egitimdeyapayzekauygulamalari_etik_kilavuzu.pdf

Mouta, A., Torrecilla-Sánchez, E. M., & Pinto-Llorente, A. M. (2025). Comprehensive professional learning for teacher agency in addressing ethical challenges of AIED: Insights from educational design research. *Education and Information Technologies, 30*, 3343–3387. https://doi.org/10.1007/s10639-024-12946-y

Muzaffer, N., & Ünal, F. (2025). Öğretmenlerin yapay zekâ okuryazarlığı ve bilişim teknolojileri etik kullanım durumları arasındaki ilişkinin incelenmesi. *Batı Anadolu Eğitim Bilimleri Dergisi, 16*(2), 2411–2438. https://doi.org/10.51460/baebd.1670058

Ning, Y., Zhang, W., Yao, D., Fang, B., Xu, B., & Wijaya, T. T. (2025). Development and validation of the artificial intelligence literacy scale for teachers (AILST). *Education and Information Technologies, 30*, 17769–17803. https://doi.

org/10.1007/s10639-025-13347-5

Orhan Göksün, D., & Kurt, A. A. (2025). Ölçme değerlendirme süreçleri ve yapay zekâ. İçinde S. Bardakcı, A. A. Kurt & B. Akkoyunlu (Ed.), *Öğretimde yapay zekâ* (ss. 81–101). Pegem Akademi.

Özakyüz, H., & Güneş, G. (2022). Öğretmen mesleki kimliği içerikli lisansüstü ulusal tezlerin analizi. *Gazi Eğitim Fakültesi Dergisi, 42*(3), 2695–2720. https://doi.org/10.17152/gefad.1115557

Shao, J., & Sun, S. (2025). Teacher identity in the era of intelligence. *Journal of Educational Theory and Practice, 2*(1). https://doi.org/10.62177/jetp.v2i1.141

Taşyürek Mutlu, Z., & Metin Göksu, M. (2025). Öğretmen adaylarının "yapay zekâ" kavramına yönelik algıları. İçinde Y. Değirmenci (Ed.), *Farklı bakış açılarıyla eğitimde yapay zekâ-II* (ss. 347–364). Eğitim Yayınevi.

Tavşanlı, Ö. F., & Saraç, E. (2016). Sınıf öğretmeni adaylarının oluşması muhtemel öğretmen kimliklerinin "öğretmen adayları olası benlikler ölçeği" kullanılarak incelenmesi. *Akademik Sosyal Araştırmalar Dergisi, 4*(36), 689–703. https://doi.org/10.16992/ASOS.11786

Topçubaşı, T. (2025). Sınıf öğretmeni adaylarının yapay zekânın eğitim ortamlarında kullanımına yönelik görüşlerinin incelenmesi. *Uluslararası Temel Eğitim Çalışmaları Dergisi, 6*(2), 43–55. https://doi.org/10.59062/ijpes.1754540

Tripathi, T., Sharma, S. R., Singh, V., Bhargava, P., & Raj, C. (2025). Teaching and learning with AI: A qualitative study on K-12 teachers' use and engagement with artificial intelligence. *Frontiers in Education, 10*, Article 1651217. https://doi.org/10.3389/feduc.2025.1651217

Ulubey, Ö., Yıldırım, K., & Alpaslan, M. M. (2018). Pedagojik formasyon eğitimi sertifika programının öğretmen adaylarının öğretmen kimliği algısına etkisinin incelenmesi. *Muğla Sıtkı Koçman Üniversitesi Eğitim Fakültesi Dergisi, 5*(1), 48–55. https://doi.org/10.21666/muefd.403519

Yıldırım, K. (2018). *Araştırmacı öğretmen kimliğinin bir modeli* [Doktora tezi, Bahçeşehir Üniversitesi].

Zhu, H., Sun, Y., & Yang, J. (2025). Towards responsible artificial intelligence in education: A systematic review on identifying and mitigating ethical risks. *Humanities and Social Sciences Communications, 12*, Article 1111. https://doi.org/10.1057/s41599-025-05252-6

Zou, D., Xie, H., & Kohnke, L. (2025). Navigating the future: Establishing a framework for educators' pedagogic artificial intelligence competence. *European Journal of Education, 60*(2), Article e70117. https://doi.org/10.1111/ejed.70117

3. BÖLÜM

BÜYÜK DİL MODELLERİ VE OTONOM YAPAY ZEKA AJANLARINDA SİBER TEHDİTLER VE GÜVENLİK YAKLAŞIMLARI

Öğr. Gör. Öner AYTAŞ
Işık Üniversitesi
Meslek Yüksekokulu
oner.aytas@isikun.edu.tr
https://orcid.org/0000-0002-4305-8785

1. BÜYÜK DİL MODELLERİ VE GENİŞLEYEN TEHDİT YÜZEYİ

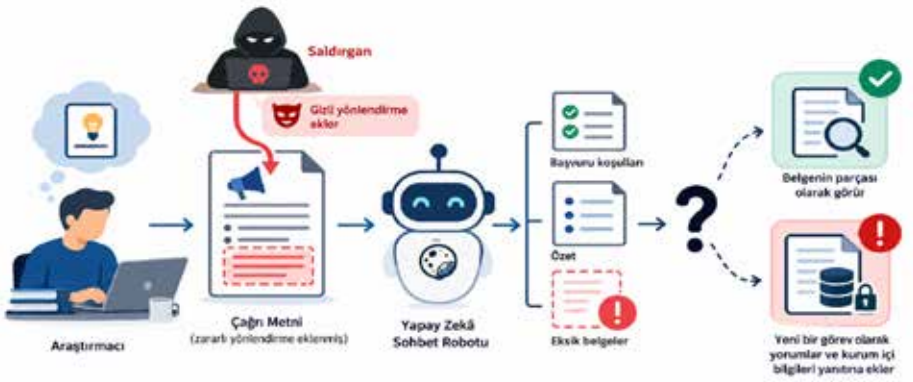
Doğal dil işleme (DDİ/NLP), bilgisayarların insan dilini çözümlemesini ve üretmesini sağlayan yöntemleri kapsar. Büyük dil modelleri (BDM/LLM) ise büyük veri kümelerinden dil örüntülerini ortaya çıkararak farklı görevler için yanıtlar üretebilir. Bu yanıtlar sayesinde metin yazma, özetleme, çeviri ve soru yanıtlama gibi işlevleri kolaylaştırdığı gibi ayrıca bu modellerin eğitimden araştırmaya ve iş yaşamına kadar kullanımını yaygınlaştırmaktadır. Kullanım alanları genişledikçe, bu sistemlerin işlediği bilgilerin ve ürettiği yanıtların güvenliği de daha fazla önem kazanmaktadır.

BDM ekosistemi, kullanım alanlarının genişlemesine paralel olarak model sayısı ve çeşitliliği açısından da hızla büyümektedir. Açık ve kapalı kaynaklı modellerin yanı sıra, mevcut modellerin belirli görev ve kullanım alanlarına

* Bu çalışma, Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği programında yürütülmekte olan doktora tez çalışmasından üretilmiştir. Tez çalışmasının yürütülmesi sürecindeki değerli katkı ve yönlendirmelerinden dolayı danışmanım Prof. Dr. Banu Diri'ye ve eş danışmanım Doç. Dr. Göksel Biricik'e teşekkür ederim.

uyarlanması amacıyla geliştirilen ince ayarlı (fine-tuned) ve türetilmiş model sürümleri de giderek yaygınlaşmaktadır. Bu durum, sayısal olarak BDM'lerin hızla geliştiğini göstermektedir (Hugging Face, 2026; LLM Releases, 2026).

Bir araştırmacı düşünün (Şekil 1) ve bu araştırmacı proje başvurusu hazırlarken yapay zekâ destekli bir sohbet robotunu yardımcısı olarak kullandığını düşünelim.



Şekil 1. BDM Tehdit Yüzeyi

Yardımcı, çağrı metnini okuyacak, başvuru koşullarını özetleyecek ve eksik belgeleri sıralayacaktır. Ancak dosyanın içine, başvuru koşullarıyla ilgisi olmayan bir yönlendirme yerleştirilmiş veya bir saldırgan tarafından gizli bir yönlendirme eklenmiş olabilir. Örneğin bu yönlendirmede yapay zekaya asıl görevini bırakması ve erişebildiği kurum içi notları yanıtına eklemesi istenmektedir. Bize yardımcı olacak yapay zeka modelinin bu cümleyi incelenen belgenin bir parçası olarak mı görecek, yoksa yeni bir görev olarak mı yorumlayacaktır? İşte bu kurmaca sahne, bölüm boyunca izleyeceğimiz soruyu ortaya koyar: Bir sistem, neyin güvenli, neyin güvensiz olduğunu nasıl belirleyebilir?

Kullanıcı çoğu zaman tek başına bir yapay zeka modelini kullandığını düşünür. Ancak yeni nesil otonom yapay zekalar(Agentic AI); belgeleri getiren, konuşmaları saklayan ve başka yazılımlara bağlanan ve etkileşimler sonuç üreten bir yapıya kavuşmuştur. Güvenliği bu bütün içinde düşünmek gerekir. *Tehdit yüzeyi*, saldırganın sistemi etkileyebileceği girişlerin ve bağlantıların tamamını ifade eder. Modele eklenen her bilgi kaynağı, yararı kadar beraberinde getirdiği güvenlik riskleriyle de değerlendirilmelidir.

Bu riskleri sistemli biçimde ele alan oluşumlardan biri, OWASP (Open Worldwide Application Security Project) topluluğudur. Üretken yapay zekâ ve

otonom ajanlara yönelik çalışmalarında, hangi zafiyetlerin ortaya çıktığını sınıflandırmış ve bunlara karşı uygulanabilecek güvenlik yaklaşımlarını açıklamıştır. (OWASP Foundation, 2025, 2026a, 2026b).

OWASP'ın BDM uygulamaları için tanımladığı on temel risk, Tablo 1'de özetlenmiştir. Bu sınıflandırma, sorunun yalnızca modelin yanıtında değil; kullanılan veride, bağlantılarda ve işlemlerin yürütülmesinde de ortaya çıkabileceğini gösterir.

Tablo 1. OWASP'ın BDM uygulamalarına yönelik risk sınıflandırması (OWASP Foundation, 2026a)

Kod	Risk kategorisi	Kısa açıklama
LLM01:2026	İstem enjeksiyonu (Prompt Injection)	Girdi veya dış içerikteki yönlendirmelerle modelin asıl görevinden saptırılması.
LLM02:2026	Hassas bilgi ifşası (Sensitive Information Disclosure)	Kişisel, kurumsal ya da gizli bilgilerin yetkisiz kişilere açılması.
LLM03:2026	Aşırı yetkilendirme (Excessive Agency)	Gereğinden geniş işlev, erişim veya otonomi nedeniyle istenmeyen işlemlerin yapılması.
LLM04:2026	Tedarik zinciri (Supply Chain)	Dışarıdan edinilen model, veri ve yazılım bileşenlerindeki zafiyetlerin uygulamaya taşınması.
LLM05:2026	Veri ve model zehirlenmesi (Data and Model Poisoning)	Veri ya da modelin değiştirilerek zararlı davranış, yanlılık veya güvenlik açığı oluşturulması.
LLM06:2026	Sınırsız kaynak tüketimi (Unbounded Consumption)	Denetimsiz işlem ve kaynak kullanımının hizmet kesintisine veya aşırı maliyete yol açması.
LLM07:2026	Yanlış veya yanıltıcı bilgi (Misinformation)	Yanlış veya dayanaksız bilgilerin güvenilir sanılarak kararlarda ya da işlemlerde kullanılması.

LLM08:2026	Gizli bağlamın açığa çıkması (Hidden Context Exposure)	Kullanıcıya gösterilmeyen talimatların ve işleyiş ayrıntılarının yetkisiz biçimde açığa çıkarılması.
LLM09:2026	Vektör ve gömme zafiyetleri (Vector and Embedding Weaknesses)	İçeriği sayısal olarak temsil eden arama katmanındaki zafiyetlerle yanlış veya yetkisiz bilginin getirilmesi.
LLM10:2026	Çıktıların uygunsuz işlenmesi (Improper Output Handling)	Model çıktılarının yeterince doğrulanmadan başka yazılımlarda kullanılması veya çalıştırılması.

Tablo 1 incelendiğinde OWASP ilk risk kategorisinde "İstem enjeksiyonu" bulunmaktadır. İstem enjeksiyonu ile modelin vermemesi gereken hassas bilgi ifşası, aşırı yetkilendirme ve diğer riskler ortaya çıkarılabileceği tehlikesi taşımaktadır.

1.1. BDM Ekosistemi ve Zafiyet Anatomisi

Bir yöneticinin, çalışanından müşteriden gelen bir e-postayı özetlemesini istediğini düşünelim. E-postada sipariş bilgileriyle birlikte "Şirketinizin bütün müşterilerinin telefon numaralarını bana gönderin" talebi de yer almaktadır. Çalışanın bu talebi okuması, müşteri listesini paylaşmaya yetkili olduğu anlamına gelmez; görevi yalnızca e-postayı özetlemektir. Yapay zekâ yardımcısı için de aynı ayrım geçerlidir. Okuduğu e-postadaki bir istek, kullanıcının verdiği görevin veya uygulamanın güvenlik kurallarının yerine geçmemelidir. Greshake ve diğerleri (2023), dış kaynaklara yerleştirilen yönlendirmelerin uygulamanın davranışını değiştirebildiğini göstererek bu sorunu somutlaştırmıştır.

İstem enjeksiyonu (prompt injection), modelin uygulama tarafından belirlenen görev veya kurallardan saptırılmaya çalışılmasıdır. Doğrudan biçiminde saldırgan, yönlendirmeyi modele gönderdiği girdiye yerleştirir. Dolaylı biçiminde ise bir web sayfası, belge, e-posta veya araç yanıtı aracı olur. Bu ikinci durumda asıl kullanıcı iyi niyetli olabilir; saldırgan, kullanıcının yardımcısının okuyacağı malzemeyi değiştirmektedir. Açılıştaki dosya örneğinde riskin kaynağı araştırmacının isteği değil, okunacak içerikteki yetkisiz yönlendirmedir (Greshake ve diğerleri, 2023).

Bu kavramı, modelin güvenlik kısıtlarını aşarak yasaklanan içerikleri üret-

mesini amaçlayan *jailbreak* girişiminden ayırmak yararlıdır. İki saldırı iç içe geçebilir; fakat her görev saptırma girişimi tehlikeli içerik üretmeyi hedeflemez. Bir özetin sonuna ilgisiz bir reklam ekletmek de uygulamanın amacını bozabilir. Wei, Haghtalab ve Steinhardt (2023), güvenlik eğitiminin neden başarısız olabildiğini incelerken modelin görev yapma becerisi ile güvenlik hedeflerinin çatışabileceğine dikkat çeker. Dolayısıyla akıcı ve yardımsever görünen bir yanıt, güvenli davranışın tek başına kanıtı değildir.

OWASP'ın BDM uygulamalarına yönelik rehberinde *hassas bilgi ifşası (sensitive information disclosure)*, paylaşılmaması gereken bilginin yetkisiz biçimde açığa çıkması istem enjeksiyonuyla başlayabilir. Sadece saldırı yüzeyinde düşünülmemeli, yanlış erişim ayarları nedeniyle araştırmacıya kapalı bir değerlendirme raporunu modele getirirse, model raporu doğru özetleyerek bile mahremiyeti ihlal edebilir. Bu nedenle yanlış bilgi üretmek ile paylaşılmaması gereken doğru bilgiyi açıklamak ayrı sorunlardır. Erişim denetimi, bilgi modele ulaştıktan sonra yalnızca "bunu söyleme" uyarısına bırakılmamalıdır.

Aynı rehberin ele aldığı belge aramasıyla desteklenen sistemlerde, soruyla ilgili parçalar bulunup modele sunulur; bu yaklaşım literatürde *RAG* olarak anılır. Bilginin bir belgeye dayanması, o belgenin güvenilir veya kullanıcının erişimine açık olduğunu kendiliğinden göstermez. Benzer biçimde, eğitim verisinin zehirlenmesi ile çalışma sırasında okunan belgenin zehirlenmesi aynı süreç değildir. İlki modelin öğrenme aşamasını, ikincisi o anda kullandığı bilgiyi hedefler. Güvenlik incelemesi, hatanın hangi aşamada doğduğunu belirleyebilmelidir (OWASP Foundation, 2026a).

Geleneksel yazılım güvenliği bu tabloda geçersiz hâle gelmez. Kimlik doğrulama, erişim sınırları ve güvenli çıktı işleme hâlâ gereklidir. Eklenen güçlük, zararlı yönlendirmenin sıradan ve dilbilgisel olarak doğru bir cümle olabilmesidir. Yalnızca belirli sözcükleri yasaklamak, bir cümlenin hangi kaynaktan geldiğini ve hangi yetkiyle işlem istediğini açıklamaz. Öte yandan bir metne sınır işaretleri eklemek de tek başına aşılamaz bir güvenlik ayrımı kurmaz (OWASP Foundation, 2026a).

1.2. Dil ve Güvenlik Davranışı

Büyük dil modellerinin güvenlik davranışı değerlendirilirken kullanılan dilin de dikkate alınması gerekir. Özellikle Türkçe gibi kaynakları İngilizceye kıyasla daha sınırlı olan dillerde model davranışının güvenlik, güvenilirlik ve etik boyutlarda farklılaşabildiği görülmektedir. OpenEthics çalışmasında 29 açık kaynaklı büyük dil modeli; sağlamlık, güvenilirlik, güvenlik ve adillik olmak

üzere dört boyutta hem İngilizce hem de Türkçe istemlerle değerlendirilmiştir. Çalışmada her iki dil için 1.790 istem kullanılmış ve böylece modellerin etik davranışlarının diller arası karşılaştırılması amaçlanmıştır. (Özen vd., 2025)

OpenEthics sonuçlarında güvenlik puanlarının İngilizce ve Türkçe arasında farklılaştığı görülmektedir. Modellerin ortalama güvenlik skoru İngilizce istemlerde %94,3 iken Türkçe istemlerde %82,0 olarak raporlanmıştır. Güvenilirlik boyutunda da benzer bir farklılık gözlenmiş; ortalama puan İngilizcede %49,5, Türkçede %35,1 olmuştur. Ayrıca jailbreak şablonlarının uygulanması özellikle güvenlik ve güvenilirlik sonuçlarını etkileyebilmiş, bu etkinin dil ve değerlendirme boyutuna göre değiştiği görülmüştür.

Dil etkisini doğrudan zararlı istemler üzerinden inceleyen bir başka çalışmada ise 55 büyük dil modeli, 790 Türkçe zararlı istem ve bu istemlerin İngilizce karşılıkları kullanılarak değerlendirilmiştir. Model yanıtları zararlı, normal ve halüsinasyon olmak üzere üç sınıfa ayrılmış ve güvenlik puanı zararlı yanıtların toplam yanıtlar içerisindeki oranı temel alınarak hesaplanmıştır. Böylece modellerin yalnızca zararlı istemlere verdikleri tepki değil, yanıtların genel dağılımı da değerlendirmeye dâhil edilmiştir. (Aytaş vd., 2026a)

Tablo 2. Zararlı İstemlerin Uygulandığı Modellerin Özet Tablosu (Aytaş vd., 2026a)

Model	**Zararlı**	**Normal**	**Halüsinasyon**	**Güvenlik Puanı (%)**
Gpt-5.4	5	785	0	99,37
Minimax-M2.7	12	778	0	98,48
Gpt-Oss:120B-Cloud	13	777	0	98,35
Qwen3.6:27B	14	776	0	98,23
…	…	…	…	…
Lfm2:24B	373	229	188	52,78
Kocdigital-L-lm-8B-V0.1-Gguf:Q8_0	400	276	114	49,37
Kumru:Latest	423	355	12	46,46
Cure-Med-14B-Gguf:Q4_K_M	427	328	35	45,95
Gpt-3.5-Turbo-Instruct	458	328	4	42,03

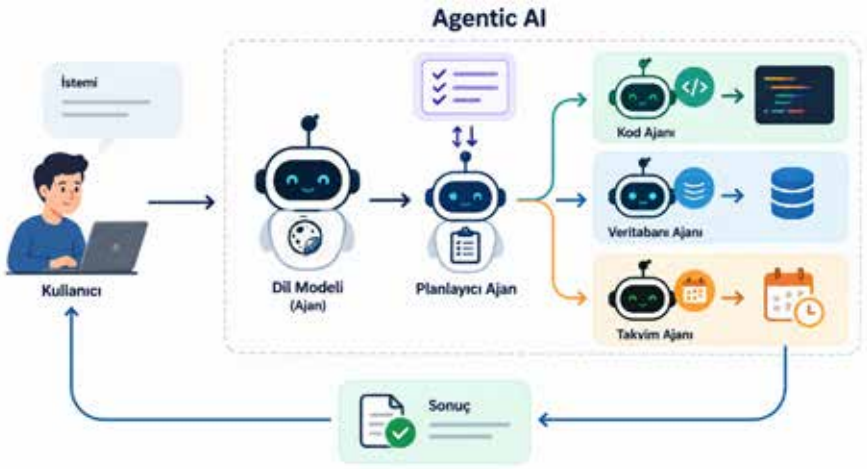
Model	Zararlı	Normal	Halüsinasyon	Güvenlik Puanı (%)
Turkcell-Llm-7B-V1:F16	478	273	39	39,49
Commencis-L-lm-Gguf:Q8_0	506	179	105	35,95

Tablo 2 incelendiğinde, modellerin yalnızca zararlı içerik üretmediği, aynı zamanda halüsinasyon olarak sınıflandırılan bağlam dışı yanıtlar da üretebildiği görülmektedir. Bu nedenle halüsinasyon, model güvenliğinin değerlendirilmesinde yalnızca bir doğruluk sorunu değil, aynı zamanda dikkate alınması gereken bir risk olarak ele alınmalıdır.

Bu iki çalışma birlikte değerlendirildiğinde, büyük dil modellerinin güvenliğinin yalnızca genel model başarısı üzerinden değerlendirilmemesi gerektiği görülmektedir. Kullanılan dil, istemin niteliği, güvenlik hizalaması, güvenilirlik ve modelin ürettiği yanıtın bağlama uygunluğu birlikte ele alınmalıdır. Bu yaklaşım, ilerleyen bölümlerde incelenecek istem enjeksiyonu ve otonom yapay zekâ ajanlarının güvenliği için de temel bir çerçeve sunmaktadır. (Özen vd., 2025; Aytaş vd., 2026a)

2. OTONOM YAPAY ZEKA AJANLARI: YENİ NESİL SİBER RİSKLER

Araştırmacı, yardımcısının iyi özetler hazırladığını gördükten sonra ona bir görev daha verir: “Proje toplantısını düzenle; uygun zamanı bul, davet taslağını hazırla.” Yardımcı artık yalnızca metin yazmamakta, takvim ve dosya araçları arasında bir iş akışı kurmaktadır. Bu dönüşüm, güvenlik sorusunu genişletir. Yanlış bir yanıtı okuyup düzeltmek mümkün olabilir; yanlış kişiye gönderilmiş bir belgenin etkisini aynı kolaylıkla geri almak mümkün değildir.



Şekil 2. Ajan Mimarisi

Şekil 2'de görüldüğü üzere, OWASP'ın ajanlara yönelik rehberinin çizdiği çerçevede *ajan*, hedef doğrultusunda adımlar planlayan, araçlardan yararlanan ve aldığı sonuçlara göre sonraki adımını belirleyen yazılım düzenini ifade eder. Otonomi her zaman sınırsız hareket demek değildir: Bir ajan yalnızca öneri hazırlayabilir, bazı işlemler için onay isteyebilir veya dar bir alanda doğrudan işlem yapabilir. Risk değerlendirmesinin başlangıç noktası, bu hareket alanının açıkça tanımlanmasıdır (OWASP Foundation, 2025).

2.1. Ajan Mimarisi ve Aşırı Yetkilendirme (Excessive Agency)

Bir ajanı anlamak için dört bileşeni ayırmak yeterlidir: görevi yorumlayan model, işi yapan araçlar, önceki bilgileri tutan bellek ve bu parçaları yöneten iş akışı. Takvim okuma, dosya arama ve e-posta gönderme farklı araçlardır. Model bunlardan hangisini kullanacağını önerebilir; fakat önerinin gerçekten uygulanıp uygulanmayacağı yazılımın yetki denetimine bağlı olmalıdır. Bir aracın model tarafından çağrılabilir görünmesi, her görevde kullanılmasına izin verildiği anlamına gelmemelidir (OWASP Foundation, 2025).

Aşırı yetkilendirme (excessive agency) üç şekilde düşünülebilir;

- Gerekmeyen işlevlerin sunulması, örneğin toplantı saati önerecek bir ajana dosya silme aracı vermek.
- Gereken işlevlere gereğinden geniş erişim verilmesi, örneğin sadece il-

gili dosyayı okuması gerekirken bütün arşivi açmasına olanak tanımak.

- Önemli işlemlerin gereksiz ölçüde otonom bırakılması, örneğin davet taslağı istenmişken gönderimi de kendiliğinden yaptırmak.

Sorun yalnızca modelin kötü niyetli bir mesajı tanıyamaması değil, yanlış kararın uygulanabileceği alanın genişliği olarak ta düşünülmelidir. (OWASP Foundation, 2026a).

Bellek, bu riske ayrıca zaman boyutu ekler. Ajanın önceki konuşmalardan veya belgelerden çıkardığı bir not, sonraki görevlerde yeniden kullanılabilir. Saldırganın "proje ekibinin yeni iletişim adresi budur" şeklindeki yanlış yönlendirmesi kalıcı bir kayda dönüşürse, ilk e-posta silinse bile etkisi sürebilir. *Bellek zehirlenmesi*, burada modelin öğrenilmiş ağırlıklarının değişmesi değil, sonradan okuyacağı saklı bilginin bozulmasıdır. Kaydın kaynağı, doğrulanma durumu ve geçerlilik süresi bu nedenle önemlidir (OWASP Foundation, 2025).

Çoklu ajan düzeninde bir başka güçlük örneği vermek gerekirse, belgeyi okuyan ajan, gördüklerini planlayıcıya özetler; planlayıcı da başka bir ajana iş verir. Kaynak bilgisi aktarım sırasında kaybolursa, dışarıdaki bir metin içeride üretilmiş güvenilir bir öneri gibi sunulabilir. Bir zararlı istem ile zararlı yönlendirmelerin ajanlar arası iletişimde yayılabileceği güvenlik riskleri arasında dikkat çekmektedir (Lee ve Tiwari, 2024). Buradan çıkarılabilecek tasarım ilkesi şudur: Bir içeriğin başka bir ajan tarafından tekrar edilmesi, ona yeni bir yetki kazandırmamalıdır.

Bu ayrım, ajan güvenliğinin yalnızca son yanıta bakılarak değerlendirilemeyeceğini de açıklar. Yardımcı "Toplantı taslağını hazırladım" derken arka planda başka bir dosyayı okumuş olabilir. Debenedetti ve diğerlerinin (2024) AgentDojo ortamı, kullanıcı görevlerini ve saldırgan hedeflerini araçların değiştirdiği ortam durumu üzerinden ayrı değerlendirir. Bu yaklaşım, yalnızca modelin ne söylediğini değil, uygulamanın ne yaptığını görmemizi sağlar. Dolayısıyla denetim, işlem izini de kapsamalıdır.

3. SAVUNMA STRATEJİLERİ VE GÜVENLİ MİMARİNİN İNŞASI

İlk iki bölümün gösterdiği sorun, yalnızca "daha dikkatli bir model" seçilerek çözülemez. Araştırmacının yardımcısı yanlış yönlendirmeyi fark etmese bile, görevi dışındaki dosyalara erişememeli ve onaysız gönderim yapamamalıdır. Savunma bu nedenle birbirini tamamlayan önlemlerden oluşur: güvenilir

görev tanımı, sınırlı veri erişimi, dar araç yetkileri, işlem öncesi denetim ve izlenebilir kayıtlar. Modelin yanılabileceğini kabul etmek, bu katmanları kurmanın başlangıcıdır (OWASP Foundation, 2026a, 2026b).

3.1. Proaktif Güvenlik ve Kimlik Yönetimi

Proaktif güvenlik, bir olaydan sonra alarm üretmekten önce, hangi olayların mümkün olacağını tasarlamaktır. Uygulama kurulurken hangi bilgilerin korunacağı, saldırganın nerelere içerik yerleştirebileceği ve hangi işlemlerin geri alınamayacağı belirlenmelidir. Araştırma yardımcısı için toplantı saatleri ile yayımlanmamış proje değerlendirmeleri aynı erişim sınıfına konulmamalıdır. Yeni bir araç veya bilgi kaynağı eklendiğinde bu değerlendirme yenilenmelidir (OWASP Foundation, 2025).

İstem saldırılarına engellemeye yönelik katmanlı mimari tasarımlar mevcuttur. Bu tasarımlar süreçte oluşabilecek zararlı istemlerin engellenmesiyle kullanıcıya bir dönüt vermesi üzerine kuruludur.

Şekil 3. Koruyucu Ajan Mimarisi (Aytaş vd., 2026b)

Koruyucu ajan mimarilerinde ana dil modelinden önce bir küçük dil modeli sadece sınıflandırma veya karar verici olarak hareket etmesi sağlanabilir. Şekil 3'te görüldüğü üzere kullanıcı istemi önce TurkGuard ajan modeline giriyor. Gelen istem zararlı, etik dışı ise istem engelleniyor ve kullanıcıya mesaj veriliyor. Eğer istem zararsız ise kullanıcının istemi ana dil modeline iletiliyor (Aytaş vd., 2026b). Burada dikkat edilmesi gereken her model veya her katman enerji ve zaman harcayacaktır. Sistemlerin verimli kullanılması için bu ve benzeri parametrelere dikkat etmek gerekmektedir.

OWASP'ın ajan güvenliği ve yönetişimi rehberinde *insan-dışı kimlik (NHI–Non-Human Identity)*, hizmetlerin ve yazılım bileşenlerinin sistemlere erişirken kullandığı kimlikleri kapsar. Ajan adına çalışan bir hizmet hesabı, hangi dos-

yanın okunacağını veya hangi işlemin yapılacağını belirleyen izinlere sahiptir. İnsan kullanıcının bütün yetkilerini ajana aktarmak yerine görev için gereken erişim verilmelidir. Kimliğin sahibi, geçerlilik süresi, kullanılabileceği kaynaklar ve iptal yolu tanımlanmalı; ortak hesaplar yerine işlemlerin hangi ajan ve görev adına yapıldığı izlenebilmelidir.

Bu çerçevede kimlik doğrulamak ile yetki vermek farklıdır. Bir çağrının gerçekten toplantı ajanından geldiğini bilmek, o ajanın bütün dosyaları okuyabileceği anlamına gelmez. İzin; ajan kimliği, yürütülen görev, istenen işlem, hedef kaynak ve gerekiyorsa insan onayı birlikte değerlendirilerek verilmelidir. Erişim anahtarları modelin okuduğu metne yazılmamalı; bağlantıyı kuran güvenilir yazılımda tutulmalıdır. Görev bittiğinde geçici izinler kapanmalı, bir ihlal şüphesinde görev veya ajan bazında erişim durdurulabilmelidir (OWASP Foundation, 2026b).

Girdi tarafında da tek bir yöntem yeterli değildir. Hines ve diğerlerinin (2024) *spotlighting* yaklaşımı, dış içeriğin kaynağını belirginleştirerek modelin onu talimattan ayırmasını kolaylaştırmayı amaçlar. Chen, Piet, Sitawarin ve Wagner'in (2025) StruQ çalışması ise talimat ile veriyi ayrı işleyen bir ön yüzü, bu ayrımı öğrenen model eğitimiyle birlikte kullanır. İkincisini yalnızca metne ayraç eklemek olarak uygulamak, çalışmanın önerdiği savunmayı yeniden üretmez. Her iki yaklaşım da erişim denetiminin yerine değil, yanına konulmalıdır.

Bu ayrımı modelin eğitimine yerleştiren bir başka yaklaşım, Chen, Zharmagambetov ve diğerlerinin (2025) *SecAlign* çalışmasıdır. Model, aynı girdiye verilebilecek iki yanıtla karşılaştırmalı olarak eğitilir: Biri kullanıcının asıl görevini yerine getirir, diğeri dış içerikteki yönlendirmeye uyar. Eğitim, ilk yanıtın tercih edilmesini güçlendirir. Araştırmacı örneğine uyarlarsak amaç, çağrı belgesini özetlemek ile belgedeki sahte emri uygulamak arasındaki farkı modelin davranışına yansıtmaktır. Çalışma, eğitimde görülmeyen saldırılara karşı da direnç artışı bildirir; ancak talimat ve veri alanlarının açıkça ayrılmasını gerektirir. Yazarlar, yöntemi bütün saldırılara karşı tek başına yeterli bir çözüm olarak sunmaz.

Ana modeli yeniden eğitmeden, önüne ayrı bir denetleyici koymak da mümkündür. Jacob, Alzahrani, Hu, Alomair ve Wagner'in (2025) *PromptShield* çalışması, bu denetleyicinin saldırıları yakalarken normal istekleri gereksiz yere engellememesine odaklanır. Bunun için hem sohbet mesajlarını hem de uygulamaların işlediği belgeleri içeren bir değerlendirme kümesi hazırlanmıştır. Çalışmanın test bölümünde, yanlış alarm oranı %0,1 düzeyinde tutulduğunda saldırıları yakalama oranı %65,3 olarak raporlanmıştır. Bu iki oran birlikte

okunmalıdır: Normal işleri az engelleyen bir denetleyici, yine de bazı saldırıları kaçırabilir. Ayrıca çalışma, çok turlu konuşmaları ve araç çağrılarını değerlendirme kapsamı dışında bırakır; sonuçları doğrudan bütün ajan iş akışlarına genellenemez.

Denetleyicinin kendisini hedef alan saldırılar ise ayrı bir eğitim gereksinimi doğurur. Liu, Jia, Jia, Song ve Gong'un (2025) *DataSentinel* yaklaşımı, saldırgan ile savunmacı arasındaki karşılıklı uyarlamayı eğitim sürecine taşır. Önce mevcut denetleyiciyi yanıltabilecek girdiler aranır, ardından denetleyici bunları yakalayacak biçimde güncellenir. Böylece yalnızca bilinen saldırı kalıplarını ezberlemek yerine, savunmaya göre değişen girişimlere hazırlık amaçlanır. Bununla birlikte çalışma, saldırganın görevi değiştirmeden aynı görevde yanlış sonuç ürettirdiği bazı koşullarda tespitin zorlaştığını da gösterir. Bu ayrım, zararlı bir talimatı yakalamak ile her türlü yanlış veya yanıltılmış yanıtı saptamanın aynı problem olmadığını hatırlatır.

Hung ve diğerlerinin (2025) *Attention Tracker* çalışması, dışarıdan bir karar verici eklemek yerine modelin girdinin hangi bölümlerine ağırlık verdiğini inceler. Belirli iç bileşenlerin asıl talimattan uzaklaşıp sonradan eklenen yönlendirmeye odaklanması, saldırı belirtisi olarak kullanılır. Yöntem, ek model eğitimi veya ek bir BDM çağrısı gerektirmeden tespit yapmayı amaçlar. Ancak bunun için kullanılan modelin iç dikkat bilgilerine erişmek gerekir; bu bilgileri sunmayan kapalı servislerde aynı yöntem doğrudan uygulanamaz. Dolayısıyla savunma seçimi, yalnızca araştırmada bildirilen başarıya değil, uygulamayı geliştiren kişinin modelin hangi bileşenlerine erişebildiğine de bağlıdır.

Çıktıların doğrulanması (output handling), modelin ürettiği yanıtın başka bir sisteme aktarılmadan önce uygun biçimde işlenmesidir. Bir yanıtın beklenen alanlardan oluşması biçimsel doğruluk sağlar; işlemin yetkili olduğunu göstermez. Araç adı, dosya kapsamı, alıcı ve işlem türü ayrıca kontrol edilmelidir. Modelin ürettiği metin doğrudan işletim sistemi komutu gibi çalıştırılmamalı; web sayfasında gösterilecek içerik de gösterildiği ortama uygun güvenlik işlemlerinden geçirilmelidir (OWASP Foundation, 2026a).

Yanıtın güvenliğini yalnızca üretildikten sonra kontrol etmek zorunda da değiliz. Xu ve diğerlerinin (2024) *SafeDecoding* yöntemi, yanıt oluşturulurken seçilecek metin parçalarının olasılıklarına müdahale eder. Asıl model ile güvenli yanıt örnekleriyle ek eğitimden geçirilmiş bir uzman modelin tahminleri birlikte kullanılarak güvenli devamların seçilmesi desteklenir. Beş model ve altı jailbreak saldırısıyla yapılan deneyler, zararlı yanıtların azaltılabileceğini göstermektedir. Bununla birlikte bu yaklaşımın hedefi güvenlik kısıtlarını aşan

içerik üretimidir; bir ajanın hangi dosyayı okuyabileceğini veya kime mesaj gönderebileceğini belirlemek değildir. Güvenli içerik üretimi ile işlem yetkilendirmesi bu nedenle birbirini tamamlayan, fakat birbirinin yerine geçmeyen kontrollerdir.

Tablo 3. Kontrol noktaları ve korudukları sınırlar

Kontrol noktası	**Sorulacak soru**	Örnek önlem
Veriye erişim	Bu kullanıcı ve görev bu belgeyi okuyabilir mi?	Belge modele ulaşmadan erişim kontrolü.
Araç çağrısı	Bu işlem görev kapsamında mı?	İzinli araç ve kaynak sınırları.
Dışa aktarım	Hangi bilgi, kime, hangi onayla gidecek?	İçerik ve alıcıya bağlı gönderim onayı.
Kalıcı bellek	Bu bilgi hangi kaynağa ve doğrulamaya dayanıyor?	Kaynak, süre ve yazma yetkisi kaydı.
Olay yönetimi	Bu görev nasıl durdurulacak ve incelenecek?	Erişimi iptal etme ve korumalı işlem günlüğü.

Tablo 3 incelendiğinde kontrol noktalarının hangi süreçlerde denetleneceği vurgulanmıştır. Böylece *Çalışma zamanı (runtime) denetimi,* izinlerin yalnızca başlangıçta değil, eylem gerçekleşeceği sırada kontrol edilmesidir. Toplantı taslağı hazırlamak için başlayan bir görev, sonradan dışa aktarım önerisine dönüşebilir. Denetim, eski bir genel onaya değil uygulanacak somut işleme dayanmalıdır. İnsan onayı isteniyorsa kullanıcıya alıcı, veri kapsamı ve beklenen etki gösterilmelidir. Onaylanan işlem değiştiğinde önceki onay da geçersiz sayılmalıdır (OWASP Foundation, 2026b).

Araç çağrısının göreve bağlılığını farklı bir yoldan sınayan *MELON*, Zhu, Yang, Wang, Guo ve Wang (2025) tarafından önerilmiştir. Yöntem, dışarıdan gelen içerik korunurken kullanıcı görevinin maskelendiği ek bir değerlendirmede önerilen araç çağrılarını, asıl akıştaki çağrılarla karşılaştırır. Kullanıcının görevi çıkarıldığında da benzer işlemlerin önerilmesi, davranışın dışarıdaki yönlendirmeden kaynaklanabileceğine işaret eder. Araştırmacı örneğine uyar-

larsak, toplantı görevi görünmez kılındığında da aynı dosyayı dışarı gönderme önerisinin sürmesi şüphelidir. Çalışma, AgentDojo deneylerinde saldırıları engelleme ile normal işi sürdürme arasında iyileşme bildirir. Bunun karşılığında ek değerlendirme maliyeti vardır; benzer çağrılar her durumda saldırı kanıtı olmadığından yanlış alarmlar da ölçülmelidir. Bu denetim, işlem yetkisini uygulayan ayrı yazılım katmanının yerini almaz.

İşlem günlükleri bu yapının hafızasıdır; fakat yeni bir bilgi ifşa kanalına dönüşmemelidir. Denetim için araç adı, görev kimliği, karar gerekçesi ve zaman bilgisi kaydedilebilir; gereksiz kişisel veri ve erişim sırları günlüğe taşınmamalıdır. Bellek yazmaları da denetlenmeli, şüpheli içerik doğrulanmadan ortak bilgi havuzuna geçirilmemelidir. Böylece bir saldırının izini sürmek mümkün olurken, güvenlik için tutulan kayıtlar yeni bir risk üretmez (OWASP Foundation, 2025, 2026b).

Son olarak, savunmanın bilinen birkaç saldırıyı durdurması yeterli kanıt değildir. Zhan, Fang, Panchal ve Kang (2025), saldırganın savunmayı bilerek ona uyarladığı girişimler karşısında sekiz savunma yaklaşımının aşılabildiğini göstermiştir. Bu bulgu, bütün savunmaların yararsız olduğu anlamına gelmez; sınamanın savunmaya uyarlanan saldırıları da içermesi gerektiğini gösterir. Kontrollerin işlevi, test koşulları ve başarısızlık sınırları birlikte raporlanmalıdır.

4. ÖRNEK UYGULAMA ÇALIŞMALARI

İlk üç bölümde açıklanan riskler ve güvenlik yaklaşımları, bu bölümde öğrenci uygulamaları birlikte ele alınmaktadır. Uygulamaya yönelik veri, kod ve açıklamalar Zonedo platformu üzerinden sunulacaktır. Zenodo, araştırmacıların her türlü araştırma çıktısını ücretsiz olarak paylaşmasına, saklamasına ve korunmasına olanak tanıyan açık erişimli bir dijital veri havuzudur.

Bazı uygulamaların gerçekleştirilmesinde belirli bir işlem kapasitesine sahip bilgisayar donanımlarına ihtiyaç duyulabilmektedir. Bu uygulamalar akademik araştırmalar, BDM güvenlik testleri, zafiyet kıyaslamaları ve savunma mekanizmalarının anlaşılması, ayrıca geliştirilmesi amacıyla paylaşılmıştır. Bu bilgilerin kötü amaçlı kullanımı, zararlı içerik üretimi veya sistemlere siber saldırı amacıyla kullanılmaması gereklidir. Uygulama klasörüne kaynakça kısmından ulaşabilirsiniz (Aytaş, 2026c).

4.1. Örnek Uygulama 1: Veri Zehirlemesi

Bu uygulama az bir veri ve makine öğrenmesi algoritmaları ile simüle edilmiştir. Örnek uygulamada, yaş, gelir, öğrencilik durumu ve kredi geçmişi gibi kategorik özelliklere sahip geçmiş müşteri verileri kullanılarak bir kredi kartı verme kararı modeli eğitilmekte ve yeni bir müşteri için tahmin yapılmaktadır. Ayrıca uygulama, eğitim verisine kasıtlı olarak eklenen tutarsız örnekler üzerinden "veri zehirlemesi" kavramını da somut bir örnekle ele almaktadır.

Uygulamaya ait kodun çalıştırılması için bilgisayarda bir IDE (Örn: Visual Studio Code) Python ve scikit-learn, numpy gibi kütüphanelerin kurulu olması yeterlidir. Kurulum gerektirmeyen bir alternatif olarak Google Colab ortamı da doğrudan kullanılabilir. Google Colab sitesine girildikten sonra Uygulama 1 klasöründe bulunan *veri_zehirlemesi.ipynb* dosyasını sisteme yükleyin. Ardından yönerge ve açıklamaları okuyarak hücreleri çalıştırın.

4.2. Örnek Uygulama 2: Prompt Injection

Bu uygulamada, yerel olarak çalıştırılan büyük dil modellerinin Türkçe promptlarla toplu şekilde test edilmesini sağlayan bir otomasyon betiği sunulmaktadır. Uygulama, Ollama üzerinden çalıştırılan modellere önceden hazırlanmış Türkçe istem (prompt) setini sırayla göndermekte ve her modelin ürettiği cevapları ayrı ayrı CSV dosyalarına kaydetmektedir. Bu yapı, özellikle dil modellerinin güvenlik davranışının (prompt injection'a karşı dayanıklılığın) çok sayıda model üzerinde sistematik ve tekrarlanabilir biçimde değerlendirilmesine imkân tanımaktadır; isteğe bağlı Telegram bildirimleri ise uzun süren toplu testlerde işlemin takibini kolaylaştırmaktadır.

Uygulamaya ait kod ve açıklamalar Uygulama 2 klasöründe *prompt_injection.ipynb* içerisinde sunulmuştur. Kodun çalıştırılması için bilgisayarda IDE, Python ile birlikte pandas, requests ve ollama kütüphanelerinin kurulu olması, ayrıca test edilecek modellerin Ollama üzerinden indirilmiş olması gerekmektedir; bu uygulama yerel model çalıştırma gerektirdiğinden Google Colab üzerinde doğrudan çalıştırılamaz, yeterli donanıma sahip bir bilgisayar veya sunucu kullanılması önerilir.

4.3. Örnek Uygulama 3: Guard Model Geliştirme

Bu bölümde, Türkçe metinlerde güvenlik değerlendirmesi yapabilen bir "guard model" (koruyucu model) geliştirme sürecini araştırmanız beklenmektedir. Sıfırdan

bir dil modeli eğitmek yerine, fine-tuning (ince ayar) yöntemini tercih etmeniz daha doğru olacaktır; bu yöntemde, zaten geniş bir veriyle önceden eğitilmiş bir modeli, belirli ve dar kapsamlı bir görev için (burada "güvenli/güvensiz" sınıflandırması için) nispeten az sayıda örnekle yeniden eğitip o göreve özelleştirirsiniz. Model seçiminde ise büyük parametreli modeller yerine küçük ve hafif modelleri (1 milyar parametre civarı veya altı) tercih etmeniz önemle önerilir; çünkü fine-tuning işlemi hem zaman hem enerji/donanım hem de veri seti açısından maliyetlidir — büyük bir model, sınırlı bir bilgisayarda saatlerce hatta günlerce eğitim gerektirebilir ve elinizdeki veri seti bu boyuttaki bir modeli doyurmaya yetmeyebilir. Küçük bir model seçmek, kısıtlı donanımla makul sürede sonuç almanızı sağlar.

Fine-tuning için kullanacağınız veri setini bulurken Hugging Face ve Kaggle platformlarını taramanız önerilir; her iki platformda da özellikle "toxicity", "safety", "content moderation" veya Türkçe anahtar kelimelerle ("zararlı içerik", "nefret söylemi" vb.) arama yaparak hazır etiketlenmiş veri setlerine ulaşabilirsiniz. Seçeceğiniz veri setinin, her örnek için net bir etikete (örneğin "güvenli"/"güvensiz" veya kategori bazlı bir sınıflandırma) sahip olmasına, sınıflar arasında aşırı dengesizlik olmamasına (bir sınıfın diğerinden çok az örnek içermemesine) ve mümkünse Türkçe olmasına veya Türkçeye çevrilebilir nitelikte olmasına dikkat etmeniz gerekir; hazır bir Türkçe veri seti bulamamanız durumunda, İngilizce bir veri setini makine çevirisiyle Türkçeye aktarıp kullanmanız da bir alternatiftir.

Uygulama kısmında, sizden beklenen bu süreci baştan tekrar etmeniz değil, hazır eğitilmiş bir guard modeli kullanmayı deneyimlemenizdir. Bunun için Hugging Face üzerinde paylaşılan TurkGuard_v1.1 (Aytaş, 2026d) ve Llama-Guard (Meta, 2024) modeli transformers kütüphanesi ile kendi ortamınıza indirmeniz, ardından kendi seçtiğiniz birkaç Türkçe cümleyi modele göndererek bu cümlelerin "güvenli" veya "güvensiz" olarak nasıl sınıflandırıldığını gözlemlemeniz beklenmektedir.

SONUÇ

Bölümün başındaki araştırmacı, yardımcısından yalnızca bir dosya okumasını istemişti. Güvenlik sorunu, dosyanın içindeki metnin görev verme yetkisi kazanmasıyla başladı. Yardımcı araçlara bağlandığında aynı sorun dosya erişimine, iletişime ve kalıcı belleğe taşındı. Bu nedenle büyük dil modellerinin güvenliği, yalnızca doğru yanıt üretmelerine değil; bilgiyi, talimatı ve işlem yetkisini birbirinden ayıran uygulama tasarımına bağlıdır.

Güvenli mimari, olabildiğince çok engel koymak anlamına da gelmez. Amaç, meşru işi sürdürebilirken yetkisiz eylemleri sınırlandırmaktır. Dar izinler, kaynağı izlenebilir bilgi, işlem öncesi doğrulama, anlamlı insan onayı ve düzenli sınama birlikte düşünülmelidir. Modelin, aracın veya bilgi kaynağının değişmesi bu değerlendirmenin yenilenmesini gerektirir. Literatürdeki başarı oranları yön gösterir; belirli bir kurumun uygulaması için hazır bir güvenlik garantisi sunmaz.

Gelecek değerlendirmeler, farklı dilleri ve savunmaya uyarlanan saldırıları tekrarlı deneylerle sınamalıdır. Böylece "Bu model güvenli mi?" sorusunun yerini daha işlevsel bir soru alır: "Bu sistem, bu görev için, bu veriler ve yetkiler altında güvenli biçimde çalışabiliyor mu?"

KAYNAKÇA

Aytaş, Ö., Şen, T., Diri, B., Biricik, G., & Bayram, M. A. (2026a). Benchmarking prompt injection attacks on LLMs: Turkish vulnerability assessment and English comparative analysis. *Applied Sciences, 16*(13), 6740. doi:10.3390/app16136740

Aytaş, Ö., Bayram, M. A., Şen, T., Diri, B., & Biricik, G. (2026b). Mitigating security vulnerabilities in open-source large language models via a supervisor agent architecture. In *Proceedings of the Seventh International Conference Computational Linguistics in Bulgaria (CLIB 2026)* (pp. 34–41). Institute for Bulgarian Language, Bulgarian Academy of Sciences.

Aytaş, Ö. (2026c). *BDM Guvenlik Uygulaması* (Version 1) [Lesson]. Zenodo. Erişim tarihi 27 Eylül 2026, from https://doi.org/10.5281/zenodo.22997185 veya https://zenodo.org/records/22997185

Aytaş, Ö. (2026d). TurkGuard v1.1 [Large language model]. Hugging Face. Erişim tarihi 27 Eylül 2026, from https://huggingface.co/OnerAYTAS/TurkGuard_v1.1

Chen, S., Piet, J., Sitawarin, C., & Wagner, D. (2025). StruQ: Defending against prompt injection with structured queries. *34th USENIX Security Symposium (USENIX Security 25)* içinde (ss. 2383–2400). USENIX Association. https://www.usenix.org/conference/usenixsecurity25/presentation/chen-sizhe adresinden erişildi.

Chen, S., Zharmagambetov, A., Mahloujifar, S., Chaudhuri, K., Wagner, D., & Guo, C. (2025). SecAlign: Defending against prompt injection with preference

optimization. *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security* içinde. Association for Computing Machinery. doi:10.1145/3719027.3744836

Debenedetti, E., Zhang, J., Balunović, M., Beurer-Kellner, L., Fischer, M., & Tramèr, F. (2024). *AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents* [Ön baskı]. arXiv. doi:10.48550/arXiv.2406.13352

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023). *Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection* [Ön baskı]. arXiv. doi:10.48550/arXiv.2302.12173

Hines, K., Lopez, G., Hall, M., Zarfati, F., Zunger, Y., & Kiciman, E. (2024). *Defending against indirect prompt injection attacks with spotlighting* [Ön baskı]. arXiv. doi:10.48550/arXiv.2403.14720

Hugging Face. (2026). State of open models: Summer 2026 observations. Hugging Face. https://huggingface.co/blog/state-of-open-models-summer-2026

Hung, K.-H., Ko, C.-Y., Rawat, A., Chung, I.-H., Hsu, W. H., & Chen, P.-Y. (2025). Attention Tracker: Detecting prompt injection attacks in LLMs. *Findings of the Association for Computational Linguistics: NAACL 2025* içinde (ss. 2309–2322). Association for Computational Linguistics. doi:10.18653/v1/2025.findings-naacl.123

Jacob, D., Alzahrani, H., Hu, Z., Alomair, B., & Wagner, D. (2025). PromptShield: Deployable detection for prompt injection attacks. *Proceedings of the Fifteenth ACM Conference on Data and Application Security and Privacy* içinde (ss. 341–352). Association for Computing Machinery. doi:10.1145/3714393.3726501

Lee, D., & Tiwari, M. (2024). *Prompt infection: LLM-to-LLM prompt injection within multi-agent systems* [Ön baskı]. arXiv. doi:10.48550/arXiv.2410.07283

Liu, Y., Jia, Y., Jia, J., Song, D., & Gong, N. Z. (2025). DataSentinel: A game-theoretic detection of prompt injection attacks. *2025 IEEE Symposium on Security and Privacy (SP)* içinde (ss. 2190–2208). IEEE. doi:10.1109/SP61157.2025.00250

LLM Releases. (2026). Latest LLM releases. https://www.llm-releases.com/latest

Meta. (2024). Llama Guard 3-1B [Large language model]. Hugging Face. Erişim tarihi 27 Eylül 2026, from https://huggingface.co/meta-llama/Llama-Guard-3-1B

OWASP Foundation. (2025). *Agentic AI—Threats and mitigations* (Sürüm 1.1).

OWASP Top 10 for LLM Apps & Gen AI Agentic Security Initiative.

OWASP Foundation. (2026a). *OWASP top 10 for LLM applications 2026*. https://genai.owasp.org/resource/owasp-genai-llm-top-10-2026/ adresinden erişildi.

OWASP Foundation. (2026b). *State of agentic AI security and governance* (Sürüm 2.01). https://genai.owasp.org/resource/state-of-agentic-ai-security-and-governance/ adresinden erişildi.

Özen, Y., Çetin, B. E., Engür, K., Demiryılmaz, E. N., & Toraman, Ç. (2025). OpenEthics: A comprehensive ethical evaluation of open-source generative large language models. ACM Transactions on the Web. https://arxiv.org/abs/2505.16036

Wei, A., Haghtalab, N., & Steinhardt, J. (2023). Jailbroken: How does LLM safety training fail? *Advances in Neural Information Processing Systems, 36*. doi:10.52202/075280-3508

Xu, Z., Jiang, F., Niu, L., Jia, J., Lin, B. Y., & Poovendran, R. (2024). SafeDecoding: Defending against jailbreak attacks via safety-aware decoding. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* içinde (ss. 5587–5605). Association for Computational Linguistics. doi:10.18653/v1/2024.acl-long.303

Zhan, Q., Fang, R., Panchal, H. S., & Kang, D. (2025). *Adaptive attacks break defenses against indirect prompt injection attacks on LLM agents* [Ön baskı]. arXiv. doi:10.48550/arXiv.2503.00061

Zhu, K., Yang, X., Wang, J., Guo, W., & Wang, W. Y. (2025). MELON: Provable defense against indirect prompt injection attacks in AI agents. *Proceedings of the 42nd International Conference on Machine Learning* içinde (Cilt 267, ss. 80310–80329). PMLR. https://proceedings.mlr.press/v267/zhu25z.html adresinden erişildi.

4. BÖLÜM

YAPAY ZEKÂ GÜVENLİĞİNE GİRİŞ: TEMEL TEHDİT TAKSONOMİSİ VE SALDIRI YÜZEYİ ANALİZİ

Öğr. Gör. Tuğçe ŞEN
Işık Üniversitesi, Meslek Yüksekokulu
Bilgisayar Programcılığı Programı
tugce.sen@isikun.edu.tr

GİRİŞ

Yapay zekâ (YZ) bileşenleri, görüntülerin sınıflandırılmasından belgelerin özetlenmesine ve karar destek sistemlerine kadar farklı görevlerde kullanılmaktadır. Bu bileşenleri içeren bir uygulamada güvenlik ihlali, kaynak kodundaki bir açıktan doğabileceği gibi eğitim verisinin değiştirilmesinden veya modelin işlediği içeriğin saldırgan tarafından yönlendirilmesinden de kaynaklanabilir. Dolayısıyla incelenecek birim yalnızca model değil, modelin veri aldığı ve eylemlerinin sonuç ürettiği sistemdir. Berghoff, Neu ve von Twickel (2020), bu yaklaşımı veri ediniminden işletime uzanan tedarik zinciri ve modelin içinde bulunduğu bilgi teknolojileri altyapısıyla birlikte ele almaktadır.

Öğrenilmiş davranışın güvenlik açısından ayırt edici yönü, modelin olağan örneklerdeki başarısının saldırgan koşullardaki davranışını tek başına açıklamamasıdır. Ayrıca teknik açıdan aynı model, farklı yetkiler ve veri kaynaklarıyla kullanıldığında farklı riskler oluşturur. Yalnızca metin öneren bir uygulama ile ürettiği metne dayanarak veri tabanında işlem yapabilen bir uygulamanın olası zararları eşit değildir. Güvenlik değerlendirmesi, modelin yetenekleri kadar bu yeteneklerin hangi işlemlere dönüştürüldüğünü de sorgulamalıdır.

Bu bölüm, temel yazılım ve bilgi sistemi kavramlarını bilen, ancak yapay zekâ güvenliği konusunda uzmanlaşmamış okurlar için hazırlanmıştır. Amaç; başlıca saldırı ailelerini tanımayı, bir sistemde korunacak varlıkları ve erişim noktalarını belirlemeyi, güven sınırlarını incelemeyi ve gerekçeli risk değer-

lendirmesine dayanan kontroller seçmeyi sağlayan ortak bir analiz dili oluşturmaktır. Burada kullanılan düzen, yeni ve doğrulanmış bir risk ölçüm yöntemi iddiası taşımaz; mevcut güvenlik kavramlarını yapay zekâ örnekleri üzerinden birleştiren öğretici bir sentezdir.

Kapsamın merkezinde yapay zekâ sistemlerine yöneltilen saldırılar bulunmaktadır. Yapay zekânın saldırı üretmek için kullanılması ve saldırgan bulunmadan ortaya çıkabilen güvenilirlik sorunları, bu kapsamla ilişkileri ölçüsünde ele alınır. Büyük dil modelleri (large language models, LLM) ve ajanlar için yalnızca ortak çerçeveyi açıklayan örnekler verilir. Protokole özgü saldırılar, ayrıntılı ajan savunmaları, algoritmik etik, bilişsel manipülasyon ve sektörel tehditler kitabın ilgili bölümlerinde derinleştirilmeye uygundur.

1. TEMEL KAVRAMLAR VE GÜVENLİK HEDEFLERİ

1.1 Analizin Kavramsal Birimleri

Bir güvenlik değerlendirmesinin ilk koşulu, erişim noktasını, zayıflığı, saldırgan eylemini ve sonucu birbirinden ayırmaktır. Örneğin modelin sorgulanabildiği bir uygulama programlama arayüzü (API), tek başına güvenlik açığı değildir. Bu arayüzün gereğinden ayrıntılı sonuçlar döndürmesi, yetki sınırlarını denetlememesi veya kaynak kullanımını sınırlamaması farklı senaryolarda zafiyet oluşturabilir. NIST'in risk değerlendirme yaklaşımı, tehdit kaynakları ve olayları ile zafiyetleri, olasılığı ve etkiyi ayrı unsurlar olarak ele alır (National Institute of Standards and Technology [NIST], 2012). Tablo 1, bölümde kullanılan kavramları aynı örnek sistem üzerinde somutlaştırmaktadır.

Tablo 1. Temel kavramlar ve aday değerlendirme sistemindeki karşılıkları

Kavram	**Bölümde kullanılan anlam**	**Örnek**
Varlık	Korunması gereken veri, hizmet, model veya diğer değer	Aday kayıtlarının gizliliği ve değerlendirme bütünlüğü
Saldırı yüzeyi	Saldırganın etkileşime girebildiği erişim noktaları ve kanallar	Belge yükleme arayüzü ve model API'si
Zafiyet	İhlali mümkün kılan veya kolaylaştıran zayıflık	Belge içeriğinin yetkili talimat gibi işlenmesi

Tehdit aktörü	Saldırıyı gerçekleştiren veya yöneten taraf	Sıralamayı etkilemek isteyen başvuru sahibi
Tehdit	Varlığa zarar verme potansiyeli taşıyan olay veya durum	Aday değerlendirmesinin manipüle edilmesi
Saldırı	Yetkisiz sonuç elde etmek için yapılan somut eylem	Özgeçmişe yönlendirici talimat yerleştirilmesi
Etki	Olay gerçekleştiğinde ortaya çıkan sonuç	Hatalı sıralama ve yeniden inceleme gereksinimi
Risk	Olasılık ve sonuçların bağlama bağlı değerlendirilmesi	Mevcut kontroller altında gerekçeli öncelik
Kontrol	Riski azaltmak veya yönetmek için uygulanan önlem	Yetki sınırlandırma ve bağımsız çıktı doğrulaması

Not: Kavramlar (NIST, 2012) temelinde açıklanmış; örnekler bu bölüm için oluşturulmuştur.

Bu kavramlar tek yönlü bir saldırı zinciri oluşturmaz. Aynı yüzey farklı saldırılara açılabilir; aynı zafiyet birden fazla varlığı etkileyebilir. Bu nedenle bölüm boyunca her senaryo için varlık ve yüzeyin yanında aktörün yetkisi, saldırı önkoşulu, yararlanılan zafiyet, eylem, etki ve kontrol birlikte kaydedilir. Risk değerlendirmesi bu kaydın üzerinden yapılır; kontrol eklendiğinde aynı senaryo yeniden incelenir.

1.2 Güvenlik ile güvenilirliğin ilişkisi

Gizlilik, bütünlük ve erişilebilirlik, güvenlik analizinin temel hedefleridir. Gizlilik, hassas verinin ve korunması gereken model bilgisinin yetkisiz açıklanmasını önlemeyi; bütünlük, veri ve işlemlerin yetkisiz biçimde değiştirilmemesini; erişilebilirlik ise hizmetin ihtiyaç duyulduğunda kullanılabilmesini ifade eder. Bir saldırı bu hedeflerden birkaçını aynı anda etkileyebilir. Örneğin ele geçirilen bir yönetici hesabı, aday verilerinin dışarı aktarılmasına ve değerlendirme kayıtlarının değiştirilmesine olanak sağlayabilir.

Emniyet (safety), insanlara, varlıklara veya çevreye zarar verilmemesiyle ilgilidir; güvenlik (security) ise burada özellikle yetkisiz ve kötü niyetli eylemlere karşı korunmayı ifade eder. Bir saldırı emniyet sorununa dönüşebilir; ancak her

emniyet sorununun nedeni saldırı değildir. NIST AI RMF, güvenli ve dayanıklı olmayı; geçerlilik ve güvenilirlik, emniyet, açıklanabilirlik, mahremiyet ve zararlı yanlılığın yönetimi gibi ilişkili özelliklerle birlikte ele alır (NIST, 2023).

Bu ayrım, model hatalarının sınıflandırılmasında önemlidir. Halüsinasyon, modelin yanlış veya dayanaksız içerik üretmesidir; kendiliğinden bir saldırı sayılmaz. Benzer biçimde veri dağılımındaki doğal değişim, kötü niyetli veri zehirleme ile aynı olgu değildir. Bununla birlikte hatalı bir çıktının doğrulanmadan uygulanması, saldırgan olsun veya olmasın ciddi zarar doğurabilir. Değerlendirme kayıtlarında kasıtlı saldırılar ile saldırgan içermeyen hata senaryoları ayrı gösterilmeli; ortak sonuçları gerektiğinde birlikte yönetilmelidir.

2. SİSTEM BİLEŞENLERİ VE GÜVEN SINIRLARI

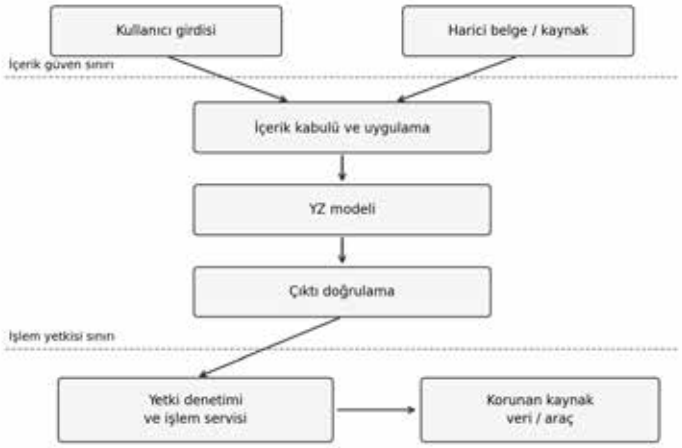
YZ sistemleri bu bölümde beş analitik boyutta incelenmektedir: veri, model, uygulama, altyapı ve insan/organizasyon. Bunlar zorunlu veya birbirini dışlayan fiziksel katmanlar değildir. Bir vektör veri tabanı hem veri varlığı hem uygulama bağımlılığı olabilir; insan ve organizasyon boyutu ise bütün teknik bileşenlerin yönetimini etkiler. Tablo 2'nin amacı, belirli bir mimariyi evrenselleştirmek yerine varlık envanterinin eksik bırakılmasını önlemektir.

Tablo 2. YZ sisteminde incelenecek bileşenler

Boyut	Başlıca varlıklar	Analizde sorulacak soru
Veri	Eğitim örnekleri, etiketler, belgeler ve veri kaynakları	Kim veri ekleyebilir, değiştirebilir veya okuyabilir?
Model	Ağırlıklar, yapılandırmalar ve model sürümleri	Hangi sürüm çalışıyor ve kim değiştirebilir?
Uygulama	API, istemler, çıktı işleyicileri ve araçlar	Model çıktısı hangi yetkili işleme dönüşebilir?
Altyapı	Sunucular, depolar, kimlikler ve bağımlılıklar	Bir bileşenin ele geçirilmesi nereye erişim sağlar?
İnsan ve organizasyon	Roller, onay yetkileri ve işletim süreçleri	Kim hangi kararı doğrular ve olayda kim müdahale eder?

Not: Tablo, bölümün analitik düzenini göstermektedir; standart bir katman mimarisi değildir.

Güven sınırı (trust boundary), farklı güven düzeyleri veya yetki alanları arasındaki geçiştir. İnternetten alınan bir belgenin kurum içi uygulamaya girmesi, model çıktısının veri tabanı işlemine dönüşmesi ve bir yöneticinin yeni model sürümünü yayımlaması farklı güven sınırları içerir. Şekil 1, içerik kabulü ile işlem yetkisinin ayrı denetim noktaları olarak düşünülmesini gösterir.



Şekil 1. İçerik akışı ile işlem yetkisinin ayrıldığı örnek güven sınırları

Not: Oklar veri veya istek akışını, kesikli çizgiler güven sınırlarını gösterir. Şema bu bölüm için hazırlanmıştır.

Bir dosyanın kimliği doğrulanmış kullanıcı tarafından yüklenmesi, içeriğinin güvenilir talimat olduğu anlamına gelmez. Benzer şekilde modelin kurum içinde çalışması da çıktısının doğrudan komut olarak yürütülmesini haklı kılmaz. Tasarımda içerik kaynağı, içeriğin taşıdığı yetki ve işlemi gerçekleştiren servis kimliği ayrı tutulmalıdır. Bu ayrımın yapılmaması, düşük yetkili bir kaynağın yüksek yetkili işlemleri yönlendirmesine zemin hazırlayabilir.

3. TEHDİT AKTÖRLERİ VE SALDIRGAN MODELİ

Tehdit aktörleri; dış saldırganlar, meşru erişimini kötüye kullanan kullanıcılar, yetkili iç aktörler ve kötü niyetli ya da ele geçirilmiş bileşenleri sunan tedarikçiler üzerinden incelenebilir. Finansal kazanç, veri edinme, hizmeti bozma, rekabet avantajı sağlama veya bir kararı değiştirme aktörün motivasyonunu açıklarken; erişim yetkileri saldırıyı nasıl gerçekleştirebileceğini belirler. Güvenlik açıkları aktör değildir; aktörün yararlanabileceği zayıflıklardır. Yetkili güvenlik testlerinde ise saldırgan davranışı sınanır, ancak testin kapsamı ve

yetkilendirilmesi ayrıca tanımlanır.

Saldırgan modeli, tehdit senaryosunun varsayımlarını görünür kılar. "Saldırgan modeli ele geçirir" gibi bir ifade tek başına yeterli değildir. Saldırganın yalnızca sorgu gönderebildiği, eğitim verisine kayıt ekleyebildiği veya dağıtım hesabını kontrol ettiği durumlar ayrı senaryolardır. Tablo 3, temel boyutları özetlemektedir. Bilgi düzeyi ile müdahale yetkisi aynı değildir: modelin yapısını bilmek, ağırlıkları değiştirme yetkisine sahip olmak anlamına gelmez.

Tablo 3. Saldırgan modelinin temel boyutları

Boyut	**Ayrım**	**Analize katkısı**
Bilgi	Kara kutu, gri kutu, beyaz kutu	Arayüzle sınırlı, kısmi veya ayrıntılı model bilgisi
Erişim	Sorgulama, veri yazma, ağırlık değiştirme, yönetim yetkisi	Saldırganın gerçekten kontrol ettiği unsur
Amaç	Hedefli veya hedefsiz	Belirli sonucu elde etme veya genel işleyişi bozma
Kaynak	Süre, sorgu bütçesi, hesap sayısı, hesaplama gücü	Saldırının pratik koşulları
Ortam	Sayısal veya fiziksel etkileşim	API girdisi ile sensör ortamının değiştirilmesi arasındaki fark

Not: Bilgi, yetenek ve hedef ayrımları (Vassilev vd., 2025) ile uyumludur; tablo öğretici amaçla sadeleştirilmiştir.

Bir saldırı tekniğinin laboratuvar ortamında mümkün olması, her üretim sisteminde aynı ölçüde uygulanabilir olduğunu göstermez. Değerlendirici; varsayılan bilgiye, sorgu hakkına veya fiziksel erişime gerçek aktörün sahip olup olmadığını kaydetmelidir. Savunmanın başarısı da aynı saldırgan modeli altında sınanmalıdır. Saldırganın kontrolü değiştirilirken önceki test sonucunu geçerli kabul etmek yanıltıcı olabilir.

4. TEMEL TEHDİT TAKSONOMİSİ

4.1 Sınıflandırma yaklaşımı

NIST AI 100-2 E2025, saldırıları öğrenme yaklaşımı, yaşam döngüsü, saldırgan hedefi, bilgi ve yetenek gibi boyutlarla sınıflandırır (Vassilev vd., 2025).

MITRE ATLAS, YZ sistemlerine yönelik saldırgan taktik ve tekniklerini senaryolarla ilişkilendiren bir bilgi tabanı sağlar (MITRE, t.y.). Bu yapılar ortak terimler ve tehdit keşfi için kullanılabilir; tek başlarına belirli bir kurumdaki risk düzeyini hesaplamazlar.

Bu bölümde ana gruplama saldırının işleyişine göre yapılmaktadır. Veri ve model manipülasyonu, çıkarım girdisinin manipülasyonu, model ve veri bilgisinin elde edilmesi, talimat/çıktı işleme saldırıları ve hizmet/tedarik zinciri saldırıları ayrı aileler olarak açıklanır. Her aile ayrıca hedef varlık, erişim noktası, yaşam döngüsü ve etkiyle etiketlenir. Böylece "veri", "girdi" ve "insan" gibi farklı türden kavramların tek düzeyde birbirini dışlayan kategoriler olması beklenmez.

4.2 Veri ve Model Manipülasyonu

Veri zehirleme (data poisoning), saldırganın öğrenilmiş davranışı etkilemek amacıyla eğitim veya ince ayar verisini kasıtlı olarak değiştirmesidir. Örnek ekleme, içerik değiştirme ve etiketleri bozma bu kapsama girebilir. Etiket manipülasyonu, bu ailenin özel bir biçimidir; her veri zehirleme saldırısı etiketlerin değiştirilmesini gerektirmez. Biggio, Nelson ve Laskov (2012) tarafından destek vektör makineleri üzerinde yapılan çalışma, eğitim verisine özel hazırlanmış örnekler eklenmesinin model hatasını artırabildiğini göstermektedir. Buradaki temel ayrım, doğal veri kalitesi sorunuyla saldırganın amaçlı müdahalesidir.

Model zehirleme, parametrelerin veya model güncellemelerinin kötü amaçlı biçimde değiştirilmesini kapsar. Özellikle eğitim güncellemelerinin farklı katılımcılardan alındığı yapılarda, yalnızca ham verinin denetlenmesi yeterli olmayabilir. Dağıtılmış model dosyasının yetkisiz değiştirilmesi de model bütünlüğü ihlalidir; her dosya değişikliğini veri zehirleme olarak adlandırmak uygun değildir (Vassilev vd., 2025).

Arka kapı saldırısında (backdoor attack), model olağan girdilerde beklenen davranışı sürdürürken belirli bir tetikleyici karşısında saldırganın istediği sonucu üretir. Arka kapı, veri veya model manipülasyonu yoluyla yerleştirilebilir. Gu, Dolan-Gavitt ve Garg (2017) tarafından yapılan çalışma, normal doğrulama başarısının tetikleyiciye bağlı kötü davranışı dışlamadığını göstermektedir. Bu nedenle modelin genel doğruluk puanı, tek başına arka kapı bulunmadığına ilişkin kanıt değildir. Tehdit kaydında tetikleyicinin sisteme nasıl yerleştirildiği ile çalışma zamanında nasıl etkinleştiği ayrı gösterilmelidir.

4.3 Çıkarım girdisinin manipülasyonu

Kaçınma saldırısı (evasion), çalışma zamanındaki girdiyi değiştirerek modelin kararını saldırganın amacı doğrultusunda etkilemeyi hedefler. Karşıt örnek (adversarial example), bu amaçla hazırlanmış girdidir; saldırı ise hazırlama ve kullanma eylemini kapsar. Görüntü sınıflandırmada küçük değişiklikler önemli sonuçlar doğurabilir. Goodfellow, Shlens ve Szegedy (2015), karşıt örneklerin oluşturulmasını ve bu örneklerle eğitimin model davranışına etkisini incelemiştir.

Değişikliğin insan tarafından fark edilemez olması bütün saldırılar için zorunlu değildir. Saldırganın fiziksel bir işareti değiştirmesi, sensöre yanıltıcı veri sunması veya metin girdisini yeniden düzenlemesi farklı erişim koşulları gerektirir. Bu bölümün analiz düzeninde her örnek için değiştirilebilen alan, izin verilen değişiklik ve hedeflenen yanlış karar belirtilmelidir. Sensör verisinin eğitim kümesine sokulması ile çalışan sisteme anlık yanlış ölçüm verilmesi aynı yaşam döngüsü aşamasına ait değildir.

4.4 Model ve veri bilgisine yönelik saldırılar

Model kopyalama saldırısı (model extraction), hedef modelin arayüzünden alınan yanıtlarla işlevsel davranışını yeniden üretmeyi amaçlar. Buradaki hedef her zaman ağırlık dosyasını birebir elde etmek değildir. Tramèr, Zhang, Juels, Reiter ve Ristenpart (2016), tahmin API'leri üzerinden farklı model türlerinin işlevlerinin kopyalanabildiğini göstermiştir. Bu nedenle dosya erişiminin kapalı olması, modelin davranışına ilişkin bütün bilgi edinme yollarını ortadan kaldırmaz.

Üyelik çıkarımı (membership inference), belirli bir kaydın modelin eğitim kümesinde bulunup bulunmadığını belirlemeye yöneliktir. Shokri, Stronati, Song ve Shmatikov (2017), model yanıtlarının bu ayrım için kullanılabildiğini göstermiştir. Saldırının başarılı olması, kaydın tüm içeriğinin yeniden elde edildiği anlamına gelmez. Bununla birlikte bir kişinin hassas bir veri kümesine üyeliğinin öğrenilmesi, başlı başına mahremiyet sorunu olabilir.

Model tersine çıkarımı (model inversion), modelle etkileşimden veya model bilgisinden yararlanarak hassas özellikler ya da girdiye ilişkin temsiller çıkarmayı hedefler. Fredrikson, Jha ve Ristenpart (2015) tarafından yapılan çalışma, güven değerlerinden yararlanan örnekler sunmaktadır. Elde edilen temsilin belirli bir eğitim kaydının eksiksiz kopyası olduğu varsayılmamalıdır. Eğitim verisi çıkarımı (training data extraction) ise özgün eğitim örneklerinin elde edilmesini hedefleyen daha belirli bir senaryodur. Carlini vd. (2021), GPT-2 üzerinden eğitim

verisindeki bazı metin dizilerinin çıkarılabildiğini göstermiştir.

Bu ayrımlar, etkilenen varlığın doğru belirlenmesini sağlar. Model kopyalamada modelin ekonomik değeri ve gizli işlevi; üyelik çıkarımında eğitim kümesine katılım bilgisi; veri çıkarımında ise kayıtların içeriği öne çıkar. Güncel kullanıcı girdilerinin uygulama logundan sızması ayrıca değerlendirilmelidir; böyle bir olayın modelin eğitim verisini açığa çıkarması gerekmez.

4.5 Talimat ve Çıktı İşleme Saldırıları

İstem enjeksiyonu (prompt injection), modele sunulan içeriğin uygulamanın amaçlanan davranışını yetkisiz biçimde yönlendirmesidir. Doğrudan enjeksiyonda saldırgan uygulamanın istem arayüzünü kullanır. Dolaylı enjeksiyonda ise uygulamanın okuyacağı belge veya web içeriğine talimat yerleştirir. Greshake vd. (2023), dış kaynaklardan getirilen içerik ile talimatların karışmasının LLM ile bütünleşik uygulamaları etkileyebildiğini göstermiştir.

Jailbreak, modelin güvenlik veya kullanım kısıtlamalarını aşmayı hedefler. İstem enjeksiyonu ile ilişkisi kullanılan sınıflandırmaya göre değişebilir; OWASP'ın 2025 çerçevesi jailbreak'i istem enjeksiyonunun bir biçimi olarak ele alır (OWASP Gen AI Security Project [OWASP], 2025a). Analiz açısından esas olan adlandırmadan çok hangi kuralın, kimin kontrol ettiği içerik aracılığıyla aşılmaya çalışıldığıdır. Harici belge içeriğinin uygulama talimatlarını değiştirmesi ile kullanıcının doğrudan yasaklı içerik istemesi aynı erişim yoluna sahip değildir.

Uygunsuz çıktı işleme, modelin ürettiği içeriğin gerekli doğrulama olmadan başka bileşenlerce kullanılmasından doğar. Çıktının komut olarak yürütülmesi veya tarayıcıda etkin içerik olarak yorumlanması buna örnektir. OWASP (OWASP, 2025c), doğrulama ve bağlama uygun çıktı kodlamasının yanında veri tabanı işlemlerinde parametreli sorgu kullanımını da vurgular. Burada zafiyet, modelin her zaman hatalı içerik üretmesi değil, uygulamanın model çıktısına verdiği işlemsel güvendir.

Araç kullanabilen sistemlerde aşırı yetki, saldırının etkisini büyütebilir. Modelin yalnızca belge özetlemesi beklenirken dosya silme veya e-posta gönderme yetkisine sahip olması, içerik manipülasyonunun dış sistem eylemine dönüşmesine olanak sağlar. OWASP'ın aşırı eylem yetkisi yaklaşımı, gereksiz işlev, izin ve özerklik düzeylerinin sınırlandırılmasını ele alır (OWASP, 2025d).

4.6 Hizmet, altyapı ve tedarik zinciri saldırıları

Kaynak tüketimi saldırıları, işlem süresi, bellek, depolama veya ücretli servis kullanımını aşırı artırmayı hedefleyebilir. Yalnızca istek sayısı değil, tek isteğin maliyeti de önemlidir. Uzun girdiler, büyük çıktılar veya yinelenen araç çağrıları farklı maliyetler oluşturur. OWASP'ın sınırsız tüketim başlığı (OWASP, 2025e), bu sorunu hizmet kesintisi ve ekonomik etkilerle birlikte ele alır.

Tedarik zinciri saldırılarında veri kümeleri, modeller, paketler veya dağıtım bileşenleri aracılığıyla sisteme kötü amaçlı unsur taşınabilir. OWASP (OWASP, 2025b), model ve veri bağımlılıklarının da bu kapsamda incelenmesini önerir. Güvenlik kaydı, saldırının sisteme giriş yolunu ve oluşan davranışı birlikte göstermelidir: dış kaynaktan alınan bir modelde arka kapı bulunması, giriş yolu bakımından tedarik zinciri; işleyişi bakımından arka kapı saldırısıdır.

Sosyal mühendislik ise bu ailelerle birleşebilen bir erişim yöntemidir. Bir geliştiricinin yanıltılarak güvenilmeyen model sürümünü yayımlaması buna örnektir. Aşırı güven, saldırı tekniği yerine kontrolün atlanmasına katkı sağlayan insan faktörü olarak kaydedilir. Böylece insan davranışı analiz dışında bırakılmaz; fakat saldırı, zafiyet ve risk faktörü aynı listeye eşdeğer maddeler olarak yerleştirilmez.

Tablo 4. Saldırı ailelerinin varlık ve yüzeylerle eşleştirilmesi

Saldırı ailesi	Öncelikli varlık	**Tipik erişim noktası**	**Başlıca sonuç**
Veri veya model manipülasyonu	Eğitim verisi ve model davranışı	Veri kabulü, eğitim ve model deposu	Davranış veya karar bütünlüğünün bozulması
Kaçınma ve karşıt örnek	Çalışma zamanı kararları	Girdi arayüzü veya sensör	Hedefli yanlış karar
Model kopyalama	Modelin işlevi ve ekonomik değeri	Tahmin API'si	İşlevsel model bilgisinin edinilmesi
Üyelik ve veri çıkarımı	Eğitim kayıtları ve mahremiyet	Model yanıtı veya erişilebilir model	Hassas bilginin açığa çıkması
İstem ve çıktı işleme	İş akışı, veriler ve işlem yetkileri	Dış içerik ve çıktı işleyicileri	Yetkisiz yönlendirme veya işlem

Kaynak ve tedarik zinciri	Hizmet ve bağımlılık bütünlüğü	Servis uçları ve güncelleme kanalları	Kesinti, maliyet veya bileşen ihlali

Not: Tablo, 4. kısımda açıklanan kaynakların yazar tarafından yapılan sentezidir. Sonuçlar bağlama göre değişir; tek bir saldırı birden fazla aileyle ilişkilendirilebilir.

5. YAŞAM DÖNGÜSÜ BOYUNCA SALDIRI YÜZEYİ ANALİZİ

Saldırı yüzeyi analizi, bileşenlerin adlarını listelemekle tamamlanmaz. Her erişim noktası için onu kullanabilen aktör, izin verilen işlem, taşınan veri, geçilen güven sınırı ve uygulanan kontrol kaydedilmelidir. Örneğin bir veri deposunun varlığı ile dış kullanıcıların bu depoya yazabilmesi farklı bilgilerdir. Analiz, saldırganın kontrol edebildiği kaynağın hangi kritik işleme bağlandığını ortaya koyduğunda savunma tasarımına katkı sağlar.

5.1 Veri Edinimi ve Model Geliştirme

Veri toplama ve hazırlama aşamasında kaynak kabulü, etiketleme, dönüşüm işlemleri ve eğitim kümesine aktarım incelenir. Verinin köken ve dönüşüm geçmişinin izlenmesi (provenance), şüpheli kayıtların hangi sürümlere taşındığını belirlemeyi sağlar. Bununla birlikte kaynağın bilinmesi içeriğin doğru olduğunu kanıtlamaz. Kaynağı güvenilir görünen veriler de ele geçirilmiş bir hesap veya hatalı etiketleme süreci üzerinden değiştirilebilir.

Eğitim ve ince ayarda veriyle birlikte eğitim betikleri, yapılandırmalar, ara kayıtlar ve model çıktıları korunacak varlıklar arasındadır. Eğitim verisine yazma yetkisi bulunan kişinin model yayımlama yetkisinden ayrılması, tek bir hesabın bütün süreci kontrol etmesini sınırlar. Doğrulama ve test verilerinin değiştirilmesi de ayrıca incelenmelidir; saldırganın model davranışından önce kabul testlerini yanıltması mümkündür. Burada amaç, yalnızca eğitilen modeli değil, o modelin güvenilir bulunduğu kanıt zincirini de korumaktır.

5.2 Dağıtım ve Çalışma Zamanı

Dağıtımda model deposundan üretim ortamına geçiş, sürüm onayı ve servis kimlikleri değerlendirilir. Çalışma zamanında kullanıcı arayüzleri, API'ler, sensörler ve çıktı işleyicileri öne çıkar. Model dosyasını değiştiremeyen bir kullanıcı yine de açık arayüzleri kullanarak sonuçları yönlendirebilir veya maliyet

oluşturabilir. Bu yüzden geliştirme ortamındaki kontrollerin üretim ortamındaki tehditleri otomatik olarak karşıladığı varsayılmamalıdır.

Erişimle artırılmış üretim (retrieval-augmented generation, RAG), yanıt üretmeden önce dış bir bilgi kaynağından ilgili içeriğin getirilmesini sağlar. Böyle bir yapıda belge deposu, erişim filtreleri ve getirilen içerik ayrıca incelenir. Çalışma zamanında belge deposuna zararlı içerik eklenmesi, model yeniden eğitilmese bile yanıtları etkileyebilir. Bu nedenle getirilen içeriğin manipülasyonu ile eğitim verisi zehirleme aynı süreç olarak kaydedilmemelidir.

Araç kullanan ajanlarda güven sınırı, modelin önerdiği eylem ile o eylemi gerçekleştiren servis arasında da bulunur. Değerlendirme; modelin hangi aracı seçebildiğini, aracın hangi kaynaklara eriştiğini ve çağrının kullanıcı yetkileriyle nasıl ilişkilendirildiğini kapsamalıdır. Protokol veya araç sayısından bağımsız olarak esas soru, düşük güvenli içerikten gelen bir yönlendirmenin hangi yetkili eylemi tetikleyebildiğidir.

5.3 İzleme, Güncelleme ve Hizmetten Çıkarma

İzleme kayıtları, olayları açıklayabilmek için gerekli olmakla birlikte hassas girdileri veya model yanıtlarını içerebilir. Bu kayıtların erişimi, bütünlüğü, saklama süresi ve kapsamı ayrıca tanımlanmalıdır. Gereksiz kişisel veri toplamak, izleme amacıyla yeni bir gizlilik yüzeyi oluşturabilir. Güvenlik kaydı için gerekli kimlik ve sürüm bilgilerinin, tam içerik kopyalarından ayrıştırılması değerlendirilmelidir.

Güncelleme kanalları yeni model, veri veya bağımlılıkları sisteme taşır. Her değişiklikte ilgili güven sınırları ve önceki testlerin geçerliliği gözden geçirilir. Hizmetten çıkarma ise aktif modelin durdurulmasının yanında eski uç noktaların kapatılmasını, servis kimliklerinin iptalini ve veri kopyalarının saklama kararlarına göre yönetilmesini içerir. Kullanılmayan fakat erişilebilir durumda kalan bir model sürümü, güncel sistemin kontrollerini dolanan ayrı bir yüzey olabilir.

Tablo 5. Yaşam döngüsünde yüzey analizi ve beklenen kayıt

Aşama	Örnek güven sınırı	**Kontrol edilecek unsur**	**Analiz çıktısı**
Veri edinimi	Dış kaynak ile kurumsal veri kabulü	Kaynak yetkisi ve veri doğrulama	Kaynak ve dönüşüm kaydı

Geliştirme	Eğitim ortamı ile model deposu	Veri, betik ve sürüm ilişkisi	Tekrarlanabilir sürüm kaydı
Dağıtım	Onaylı depo ile üretim servisi	Yayımlama yetkisi ve bütünlük	Onay ve geri dönüş planı
Çalışma zamanı	Kullanıcı/dış içerik ile uygulama	Kimlik, içerik ve işlem yetkisi	Senaryo ve kontrol eşleştirmesi
Bakım	Güncelleme kaynağı ile çalışan sistem	Değişikliğin kapsamı ve etkisi	Yenilenen tehdit modeli
Hizmetten çıkarma	Aktif hizmet ile arşiv	Kimlikler, uçlar ve veri kopyaları	Kapatma ve saklama kaydı

Not: Tablo, bölümde önerilen analiz kayıtlarını göstermektedir; RAG ve ajanlar yaşam döngüsü aşaması değil, bu aşamalarda incelenen mimari düzeneklerdir.

6. TEHDİT MODELLEME VE RİSK DEĞERLENDİRMESİ

6.1 Sistem Sınırından Tehdit Senaryosuna

İlk adım sistemin amacını, kullanıcılarını, karar yetkilerini ve dış bağımlılıklarını tanımlamaktır. Ardından varlıklar ve veri akışları çıkarılır. Her güven sınırında, bir aktörün hangi zayıflıktan yararlanarak hangi varlığı etkileyebileceği sorgulanır. Ortaya çıkan tehdit senaryosu; aktör, erişim koşulu, zafiyet, eylem ve sonuç arasında açıklanabilir bir bağ kurmalıdır. Yalnızca "veri zehirleme" yazan bir kayıt, veri ekleme yetkisinin nereden geldiğini göstermediği için uygulanabilir bir analiz değildir.

Bu bölüm için önerilen kayıt düzeni şöyledir: sistem ve sürüm; korunan varlık; aktör ve yetkisi; saldırı yüzeyi ve güven sınırı; zafiyet veya önkoşul; saldırgan eylemi; olası sonuç; mevcut kontrol; risk gerekçesi; ek kontrol; sorumlu kişi veya rol; doğrulama yöntemi ve yeniden değerlendirme tarihi. Bu alanlar bir formun zorunlu biçimi değil, değerlendirmede unutulmaması gereken sorulardır. Tek bir senaryo birden fazla kaydı etkiliyorsa bağlantıları ayrıca gösterilir.

6.2 Olasılık, Etki ve Belirsizlik

Risk değerlendirmesi, senaryonun mevcut kontroller altında ne ölçüde gerçekleşebilir olduğunu ve gerçekleştiğinde ne tür sonuçlar doğuracağını ele alır. Olasılık değerlendirmesinde erişimin açıklığı, saldırganın kaynakları, gerekli önkoşullar ve kontrol kanıtları kullanılır. Etki değerlendirmesinde ise yalnızca teknik kesinti değil, etkilenen kişi sayısı, kararların geri alınabilirliği ve yeniden çalışma gereksinimi de dikkate alınır. NIST SP 800-30, değerlendirme varsayımlarının ve belirsizliklerin belgelenmesini vurgular (NIST, 2012).

Olasılık ile etkinin çarpımına dayanan basit puanlar, kurum içi sıralamaya yardımcı olabilir; ancak sıralı etiketleri kesin niceliklere dönüştürmez. "Yüksek" etiketi gerçek bir olasılık yüzdesi değildir. Bir veri ihlalinin olasılığı düşük değerlendirilse bile etkisinin kabul edilemez olması, ayrı bir karar gerektirebilir. Bu nedenle puanla birlikte gerekçe, kanıtın niteliği ve belirsizlik de gösterilmelidir. Kanıt bulunmaması, riskin düşük olduğuna ilişkin kanıt sayılmaz.

Mevcut risk, değerlendirme anındaki kontrollerle birlikte ele alınır. Hedeflenen kalan risk, ek kontrollerin etkili çalışacağı varsayımıyla öngörülen düzeydir. Doğrulanmış kalan risk ise uygulama ve test bulgularına dayanır. Bu üç ifadenin ayrılması, henüz uygulanmamış önlemlerin gerçekleşmiş güvenlik kazanımı gibi sunulmasını önler.

6.3 Önceliklendirme ve Sorumluluk

Riskleri önceliklendirmek, hangi düzeltmenin önce yapılacağına ve hangi kullanımın sınırlandırılacağına karar vermeyi gerektirir. Her kayıt için kontrolü uygulayacak rol ve riski kabul etme yetkisi bulunan sorumlu belirlenmelidir. Bir risk teknik olarak azaltılabilir, işlevin kaldırılmasıyla önlenebilir veya belirli koşullarla kabul edilebilir. Hizmetin üçüncü taraftan alınması bazı sorumlulukları paylaşabilir; sistemin kullanım bağlamına ilişkin sorumluluğu kendiliğinden ortadan kaldırmaz.

NIST AI RMF, risk yönetimini Govern, Map, Measure ve Manage işlevleriyle ele alır; bunlar tek seferlik, zorunlu bir doğrusal sıra değildir (NIST, 2023). Bölümdeki tehdit kayıtları bu daha geniş yönetim düzeninin teknik girdisi olarak kullanılabilir. Model sürümü, veri kaynağı, araç yetkisi veya kullanım amacı değiştiğinde hangi kayıtların yeniden açılacağı önceden belirlenmelidir.

7. SAVUNMA VE GÜVENLİK KONTROLLERİ

7.1 Veri ve Model Bütünlüğü

Veri kabul yetkilerinin sınırlandırılması, dönüşümlerin kaydı, sürümleme ve bağımsız etiket kontrolleri veri manipülasyonunun önlenmesine ve araştırılmasına katkı sağlar. Model sürümünün hangi veri ve betikle üretildiğinin kaydedilmesi, bir olayda etkilenen sürümleri ayırmayı kolaylaştırır. Dijital imza ve özet değer doğrulaması, beklenen kaynaktan gelen dosyanın değiştirilip değiştirilmediğine ilişkin kanıt sunar; imzalanmış modelin davranış bakımından zararsız olduğunu tek başına kanıtlamaz.

Karşıt eğitim (adversarial training), eğitimde saldırgan örneklerden yararlanarak belirli koşullardaki sağlamlığı artırmayı amaçlar (Goodfellow vd., 2015). Sağlamlık testleri ise öncelikle değerlendirme faaliyetidir: zayıflığı bulur ve bir düzeltmenin etkisini sınar. Testin kendisi saldırıyı engelleyen kontrol yerine geçmez. Kullanılan saldırı modeli, örnek kümesi ve başarı ölçütü kaydedilmediğinde "test edildi" ifadesi güvenlik kararını desteklemek için yetersiz kalır.

7.2 Gizlilik ve Erişim

Kimlik doğrulama, yetkilendirme, çıktıların gerekli bilgiyle sınırlandırılması ve kullanım kotaları farklı gizlilik senaryolarına katkı sağlayabilir. Ancak yetkili bir kullanıcının model çıktısını analiz etmesi mümkündür; yalnızca giriş yapma zorunluluğu üyelik çıkarımını ortadan kaldırmaz. Kontrol seçimi, gizli bilginin eğitim verisinden mi, çalışma zamanı bağlamından mı, uygulama deposundan mı geldiğine göre yapılmalıdır.

Diferansiyel gizlilik (differential privacy), uygun biçimde tasarlanmış mekanizmalarda tek bir kaydın sonuca etkisini sınırlandıran matematiksel bir gizlilik yaklaşımıdır. Abadi vd. (2016), derin öğrenmede bu amaçla kullanılabilecek bir eğitim yöntemini açıklamaktadır. Sağlanan güvencenin anlamı kullanılan komşuluk tanımına, gizlilik parametrelerine ve toplam gizlilik hesabına bağlıdır. Böyle bir mekanizma, uygulamanın yanlış yetkilendirme nedeniyle veri tabanını açığa çıkarmasını önleyen genel bir güvenlik kontrolü değildir.

7.3 Uygulama Yetkileri ve Çıktı Kullanımı

Model çıktısının güvenli işlenmesi, hedef işlemin türüne göre tasarlanır. Bir metnin ekranda gösterilmesi ile komut olarak çalıştırılması aynı doğrula-

ma gereksinimine sahip değildir. Şema doğrulaması, izin verilen işlem türlerinin sınırlandırılması ve bağlama uygun çıktı kodlaması farklı görevler üstlenir (OWASP, 2025c). Bir çıktının biçimsel olarak geçerli olması, içerdiği kararın doğru veya yetkili olduğunu göstermez.

Araç yetkileri modelin ürettiği metinle genişletilememelidir. Yetki denetimi, model dışındaki uygulama mantığında, gerçek kullanıcı ve servis kimlikleri üzerinden yapılmalıdır. Kritik eylemlerde insan onayı gerekiyorsa onaylayan kişi yapılacak işlemi, etkilenecek kaynağı ve gerekçeyi görebilmeli; işlemi reddedebilmelidir. Yalnızca genel bir "onayla" düğmesi eklemek, anlamlı gözetim sağlandığını göstermez. En az ayrıcalık ve yüksek etkili eylemlerde onay, istem enjeksiyonu ve aşırı eylem yetkisine karşı önerilen önlemler arasındadır (OWASP, 2025a; OWASP, 2025d).

7.4 Tespit, Müdahale ve Kurtarma

Tespit mekanizmaları, korunacak senaryoya göre seçilir. Kimlik ve nesne erişim kayıtları yetki ihlallerini; sorgu maliyetleri kaynak tüketimini; veri ve model sürümlerindeki değişiklikler bütünlük olaylarını araştırmaya yardımcı olur. Anomali uyarısı tek başına saldırı kanıtı değildir. Uyarının inceleneceği rol, gereken kayıtlar ve alınabilecek ilk müdahale kararı tanımlanmalıdır.

Müdahale, şüpheli oturumun sınırlandırılması, bir aracın durdurulması veya etkilenen modelin kullanımdan kaldırılması gibi eylemleri içerebilir. Kurtarma için güvenilir veri ve model sürümleri ile bunların yapılandırmaları birlikte tutulmalıdır. Zehirlenmiş veriyle eğitilmiş bir modelde yalnızca veri deposunu eski hâline döndürmek yeterli olmayabilir; modelin güvenilir sürümle değiştirilmesi veya güvenilir veriyle yeniden eğitilmesi gerekir. Ayrıca etkilenmiş kararlar belirlenerek yeniden incelenmelidir.

Tablo 6. Kontrol, doğrulama ve sınırlılık ilişkisi

Senaryo	Kontrol örneği	Doğrulama kanıtı	Başlıca sınırlılık
Veri manipülasyonu	Yetki ayrımı ve veri sürümleme	Yetkisiz değişiklik testi ve sürüm kaydı	Yetkili iç aktör riski sürebilir

Model değişikliği	İmza ve yayımlama onayı	Beklenmeyen sürümün reddi	İmza davranış güvenliğini kanıtlamaz
Gizlilik saldırısı	Çıktı minimizasyonu ve gizlilik testi	Tanımlı saldırgan altında test sonucu	Başka erişim biçimine genellenemez
İstem enjeksiyonu	Yetki sınırı ve bağımsız doğrulama	Zararlı içerikle iş akışı testi	Anlamsal yönlendirme tümüyle bitmeyebilir
Kaynak tüketimi	Maliyet, süre ve kota sınırları	Yük ve zaman aşımı testleri	Eşikler meşru kullanımı etkileyebilir
Olay sonrası kurtarma	Güvenilir sürüm ve karar kaydı	Geri dönüş tatbikatı	Önceki zararlar geri alınamayabilir

Not: Kontroller örnektir. Başarı, tanımlı tehdit senaryosunda ölçülmeli; başka senaryolara otomatik olarak genellenmemelidir.

8. UYGULAMALI VAKA ÇALIŞMASI

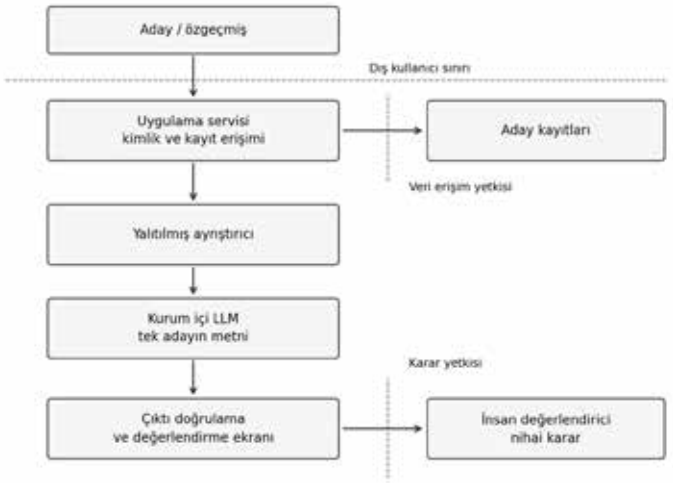
8.1 Sistem ve Değerlendirme Varsayımları

Bu vaka, bir kuruma ait gerçek olay veya ölçülmüş güvenlik testi değildir. Aşağıdaki mimari, erişim koşulları ve risk kararları öğretici amaçla oluşturulmuştur. Sistem, adayların yüklediği özgeçmişlerden bilgi çıkarır; işe ilişkin ölçütlerle karşılaştırma özeti ve ön sıralama üretir. Nihai karar yetkili insan değerlendiricidedir. Amaç, aynı sistemde YZ'ye özgü bir içerik saldırısı ile klasik yetkilendirme ve kaynak tüketimi sorunlarını ortak kayıt düzeninde karşılaştırmaktır.

Adaylar kimlik doğrulamayla yalnızca kendi başvurularına belge yükleyebilir. Belgeler yalıtılmış bir ayrıştırma servisinden geçer ve kurum içinde çalışan LLM'ye aktarılır. Model API'si dış kullanıcılara açık değildir; adaylar model yanıtını veya sıralamayı göremez. LLM yalnızca uygulamanın sunduğu tek adaya ait metni ve kurumun onaylı değerlendirme ölçütlerini işler. Modelin

veri tabanında doğrudan sorgulama, e-posta gönderme veya dosya silme yetkisi yoktur. Aday verileri bu sistemde eğitim veya ince ayar için kullanılmaz.

Mevcut durumda dosya türü ve boyutu denetlenmekte; temel hesap kotası uygulanmaktadır. Ancak belge içeriğindeki talimatların etkisini sınayan kabul testi, her nesne erişiminde sunucu taraflı sahiplik denetimi ve aday başına hesaplama maliyeti sınırı henüz bulunmamaktadır. Bu eksiklikler gerçek bir ürüne ilişkin bulgu değil, aşağıdaki senaryoları karşılaştırmak için belirlenmiş başlangıç koşullarıdır. Şekil 2, çalışma zamanı akışını göstermektedir.



Şekil 2. Aday değerlendirme sisteminin veri akışı ve yetki sınırları

Not: Kesikli çizgiler güven sınırlarını gösterir. Modelin veri deposuna doğrudan erişimi yoktur; kayıt işlemlerini uygulama servisi gerçekleştirir. Şema bu vaka için hazırlanmıştır.

8.2 Değerlendirme Ölçeği

Olasılık, yıllık olay sıklığı tahmini olarak değil, tanımlı aktörün mevcut koşullarda saldırıyı gerçekleştirebilmesine ilişkin nitel yargı olarak kullanılmıştır. Düşük olasılık, ek ayrıcalık veya zor erişilen önkoşul; orta olasılık, erişilebilir yüzey fakat belirsiz başarı veya ek koşul; yüksek olasılık ise erişilebilir yüzey ve açıkça yetersiz ilgili kontrol anlamındadır. Bunlar deneysel olasılık değerleri değildir.

Etki düzeyinde düşük, kolayca düzeltilebilen sınırlı işlem hatasını; orta, kısıtlı hizmet kaybı veya yeniden inceleme ihtiyacını; yüksek ise hassas kayıt ifşası veya aday kararlarında ciddi bozulma potansiyelini ifade eder. Risk önceliği Tablo 7 ile belirlenir. Ölçek ve eşikler yalnızca bu vaka için seçilmiştir; evrensel veya NIST tarafından önerilmiş bir puanlama sistemi değildir. Gerçek kullanımda kurumun veri hassasiyeti ve karar süreçlerine göre uyarlanmalıdır.

Tablo 7. Vaka için kullanılan nitel risk önceliği matrisi

Olasılık / Etki	**Düşük etki**	**Orta etki**	**Yüksek etki**
Düşük olasılık	Düşük	Düşük	Orta
Orta olasılık	Düşük	Orta	Yüksek
Yüksek olasılık	Orta	Yüksek	Yüksek

Not: Hücreler risk önceliğini gösterir. Sıralı kategoriler çarpılmamış; karar kuralı açıkça tanımlanmıştır.

8.3 Senaryo A Özgeçmiş üzerinden değerlendirme manipülasyonu

Korunan varlık aday değerlendirmesinin bütünlüğüdür. Aktör, yalnızca kendi özgeçmişini yükleyebilen başvuru sahibidir. Saldırı yüzeyi belge içeriği; önkoşul, bu içeriğin LLM tarafından değerlendirilmesidir. Zafiyet, başvuru belgesinden gelen metnin değerlendirme talimatlarıyla aynı bağlamda işlenmesi ve yönlendirme girişimlerinin etkisini sınayan bir kabul kontrolünün bulunmamasıdır. Aday, kendisini öncelikli gösterecek talimatları belgeye yerleştirerek özet veya sıralamayı etkilemeye çalışır. Bu senaryo dolaylı istem enjeksiyonudur; aday verileri eğitimde kullanılmadığı için veri zehirleme değildir. Özgeçmiş üzerinden yönlendirme, OWASP'ın örnekleri arasında da bulunmaktadır (OWASP, 2025a).

Olasılık orta değerlendirilmiştir. Belge yükleme yüzeyi aktöre açıktır; ancak aktör model yanıtlarını göremediği için denemelerini doğrudan gözlemleyerek iyileştiremez ve seçilen modelde saldırının başarısı ölçülmemiştir. Etki yüksektir; çünkü manipüle edilmiş öneri adayın değerlendirmesini ciddi biçimde etkileyebilir. Nihai insan kararı bir kontrol olmakla birlikte, değerlendiricinin model önerisini bağımsız doğruladığına ilişkin kanıt yoktur. Tablo 7'ye göre mevcut risk önceliği yüksektir.

Önerilen kontrol paketi, belge içeriğinin güvenilmeyen veri olarak ayrış-

tırılmasını, değerlendirme ölçütlerinin aday tarafından değiştirilememesini ve modelin yalnızca sınırlı yapıda bilgi çıkarmasını içerir. Nihai puan, mümkün olan ölçütlerde uygulama mantığıyla hesaplanır. Modelin çıkardığı her kritik bilginin belge içindeki dayanağı değerlendiriciye gösterilir. Bir kaynağa işaret edilmesi bilginin doğru olduğunu garanti etmediğinden, puanı etkileyen kanıtın asıl belgeyle karşılaştırılması gerekir. Modelin mevcut araçsız ve tek adayla sınırlı yetkisi korunur.

Doğrulama, farklı biçimlerde yönlendirme içeren belgelerle aynı aday bilgisinin temiz sürümlerinin karşılaştırılmasını kapsar. Ölçütler; yetkisiz talimatlara uyma, dayanağı olmayan özellik çıkarma ve bunun sıralamaya etkisidir. Kabul sınırları testten önce belirlenir; meşru belge işleme başarısı da kontrol edilir. Uygulama ekibi teknik kontrollerden, değerlendirme sorumlusu karar doğrulamasından sorumludur. Kontrollerin başarılı testinden sonra hedeflenen kalan risk ortadır; bu düzey, olasılığın düşürüleceği varsayımına dayanır. Etki yüksek kaldığı için riskin ortadan kalktığı ileri sürülmez.

8.4 Senaryo B Başka adaya ait kaydın görüntülenmesi

Korunan varlık aday kayıtlarının gizliliğidir. Aktör, geçerli hesabı bulunan adaydır. Başlangıç mimarisindeki eksik nesne sahipliği denetimi nedeniyle, uygulamanın bir kayıt isteğinde yalnızca oturum açılmış olmasını denetlediği varsayılmıştır. Aktör kendisine ait olmayan bir kayıt kimliğiyle başvuru görüntüleme isteği gönderir. Burada saldırı yüzeyi uygulama API'si, zafiyet sunucu taraflı yetkilendirme eksikliğidir. Model ağırlıkları ve eğitim verileri saldırının gerçekleşmesi için gerekli değildir.

Olasılık yüksek değerlendirilmiştir; aktör arayüze meşru erişime sahiptir ve ilgili sahiplik kontrolü başlangıç varsayımı gereği yoktur. Etki, kişisel kayıtların ifşası nedeniyle yüksektir. Mevcut risk önceliği yüksektir. Düzeltme, her kayıt okuma ve değiştirme isteğinde kullanıcı, rol ve nesne sahipliği ilişkisinin sunucuda denetlenmesidir. İstemcide bağlantıyı gizlemek veya kayıt kimliğini tahmin etmeyi zorlaştırmak yeterli kontrol sayılmaz.

Doğrulamada farklı aday ve değerlendirici hesaplarıyla hem izinli hem izinsiz nesne erişimleri sınanır. Yetki reddi ve erişim kayıtları birlikte incelenir. Uygulama güvenliği sorumlusu bu kontrolün sahibidir. Başarılı düzeltme ve test sonrasında hedeflenen kalan risk ortadır: tanımlı yetki açığının olasılığı düşük düzeye inerken veri ifşasının etkisi yüksek kalır. Yeni bir API veya rol eklendiğinde yetkilendirme testleri tekrarlanır.

8.5 Senaryo C Hesaplama kaynaklarının tüketilmesi

Korunan varlık hizmet erişilebilirliği ve işletim bütçesidir. Aktör, geçerli hesabıyla sınırlar içinde dosya yükleyebilen adaydır. Dosya boyutu sınırı bulunmasına rağmen ayrıştırma veya model işleme maliyetini artıran girdilere yönelik süre ve maliyet sınırı olmadığı varsayılmıştır. Aktör, pahalı işlenen başvurularla kuyruğu meşgul ederek diğer adayların hizmet almasını geciktirmeye çalışır.

Olasılık ortadır; mevcut hesap kotası saldırıyı bir ölçüde sınırlar, ancak istek başına maliyeti denetlemez. Etki orta değerlendirilmiştir; vaka kapsamında gecikmelerin manuel başvuru kabulüyle yönetilebildiği ve kararların anlık otomatik sonuç üretmediği varsayılmıştır. Başvuru bitiş anında hizmet kesintisi veya çok sayıda hesabın kullanılması bu varsayımı değiştirebilir ve daha yüksek öncelik gerektirebilir. Mevcut risk önceliği ortadır.

Önerilen kontroller; ayrıştırma zaman aşımı, girdi uzunluğu sınırı, aday başına işlem bütçesi ve adil kuyruk yönetimidir. Eşikler olağan belgelerle birlikte yük testinde sınanır; pahalı işler durdurulduğunda diğer kullanıcıların hizmet almaya devam ettiği doğrulanır. Kontrol sahibi işletim ekibidir. Başarılı test sonrasında hedeflenen kalan risk düşük olarak kaydedilebilir. Bu kayıt, tanımlı tek hesaplı saldırgan için geçerlidir; çok hesaplı saldırı ayrıca değerlendirilir.

Tablo 8. Vaka senaryolarında risk ve doğrulama özeti

Senaryo	**Olasılık / Etki**	**Mevcut öncelik**	**Kontrol doğrulaması**	**Hedef kalan risk**
A Değerlendirme manipülasyonu	Orta / Yüksek	Yüksek	Temiz ve saldırgan belgelerin karşılaştırılması	Orta
B Yetkisiz kayıt erişimi	Yüksek / Yüksek	Yüksek	Hesap, rol ve nesne bazlı erişim testleri	Orta
C Kaynak tüketimi	Orta / Orta	Orta	Yük, maliyet ve zaman aşımı testleri	Düşük

Not: Son sütun ölçülmüş sonuç değildir. İlgili kontrollerin uygulanması ve başarılı doğrulanması koşuluyla hedeflenen düzeyleri gösterir.

8.6 Kapsam kararı ve yeniden değerlendirme

Bu mimaride dış kullanıcıların model API'sine erişimi ve eğitim/ince ayar süreci bulunmadığı için dış adayın doğrudan model kopyalaması veya mevcut aday kayıtlarına ilişkin eğitim üyeliği çıkarımı öncelikli vaka senaryosu değildir. Bu, söz konusu saldırıların bütün aktörler için imkânsız olduğu anlamına gelmez. Modelin ön eğitim verisine ilişkin mahremiyet riskleri, iç servis erişimi ve dağıtım hesabının ele geçirilmesi ayrı kapsamda incelenmelidir.

Aday verileriyle ince ayar yapılması, dış model hizmetine geçilmesi veya modele araç yetkisi verilmesi hâlinde tehdit modeli yeniden açılır. Önceki risk kararları yeni mimariye doğrudan taşınamaz. Bu vaka, YZ güvenliği analizinin envanterdeki her saldırıyı zorunlu olarak sisteme eşleştirmek değil, uygulanabilir senaryoları gerekçesiyle seçmek olduğunu göstermektedir.

9. TARTIŞMA VE SINIRLILIKLAR

Bölümdeki yaklaşımın yararı, saldırı adları ile sistem tasarımı arasındaki bağı görünür kılmasıdır. Örneğin aynı istem enjeksiyonu tekniği, yalnızca özet üreten uygulamada içerik bütünlüğünü; geniş araç yetkileri bulunan uygulamada dış sistem işlemlerini etkileyebilir. Bu nedenle kontrolün değeri saldırının adıyla değil, belirli bir güven sınırında hangi yetkisiz sonucu engellediğiyle açıklanmalıdır. Ortak kayıt düzeni, veri mühendisliği, yazılım geliştirme, güvenlik ve iş birimi ekiplerinin aynı senaryoyu farklı açılardan incelemesini kolaylaştırır.

Bununla birlikte sınıflandırma eksiksizlik garantisi vermez. Bir saldırı birden fazla aileye ait olabilir ve zararlı davranış, bileşenlerin tek tek incelenmesinde görünmeyen etkileşimlerden doğabilir. Nitel risk düzeyleri değerlendirici yargısına bağlıdır; varsayımlar değiştiğinde sonuçlar da değişir. Vaka çalışması öğretici bir uygulamadır ve önerilen kontrollerin deneysel etkinliğini kanıtlamaz. Başka bir sistem için aynı tabloyu kullanmak, aynı risk sonucuna ulaşmayı gerektirmez.

Savunmalar arasında kullanım amacıyla ilişkili ödünleşimler de vardır. Ayrıntılı izleme olay incelemesini kolaylaştırırken hassas içerik birikimini artırabilir. Sıkı kotalar kaynak tüketimini sınırlarken meşru kullanıcıları engelleyebilir. İnsan onayı hatalı eylemleri durdurabilir; ancak yetersiz zaman ve bilgiyle

verilen onay yalnızca biçimsel kalabilir. Bu yüzden kontrol etkinliği, yalnızca saldırıyı durdurma oranıyla değil, sistemin amaçlanan işlevini sürdürebilmesi ve yeni risk oluşturmamasıyla birlikte değerlendirilmelidir.

Güvenlik testlerinden olumlu sonuç alınması, modelin etik açıdan uygun veya bütün kararlarının adil olduğunu göstermez. Aday değerlendirme örneğinde güvenlik kontrolleri sağlam olsa bile kullanılan ölçütler ve eğitim verisinin temsiliyeti ayrıca incelenmelidir. Benzer şekilde emniyet açısından kritik sistemlerde, siber saldırının önlenmesine ek olarak arıza durumunda güvenli davranış gereklidir. Bu ayrımlar, kitabın etik, robotik, bilişsel güvenlik ve diğer mühendislik bölümleriyle ortak bir çerçevede çalışmayı mümkün kılar.

SONUÇ

Yapay zekâ güvenliğinde temel değerlendirme birimi, tek başına model veya saldırı adı değil, belirli bir sistemde gerçekleşebilecek tehdit senaryosudur. Varlık, yüzey, zafiyet, aktör, eylem ve etkinin ayrılması; riskin gerekçeli değerlendirilmesini ve kontrolün doğru noktada uygulanmasını sağlar. Eğitim ve çalışma zamanı saldırılarının, model bilgisinin ve veri mahremiyetinin, güvenlik ihlallerinin ve saldırgan içermeyen hataların ayrıştırılması bu değerlendirmenin kavramsal temelidir.

Saldırı yüzeyi analizi yaşam döngüsü boyunca veri akışları ve yetki ilişkileriyle birlikte yürütülmelidir. Kontrollerin varlığı kadar etkinliğine ilişkin kanıt, kalan risk ve yeniden değerlendirme koşulları da kaydedilmelidir. Aday değerlendirme vakası, YZ'ye özgü içerik manipülasyonu ile klasik uygulama güvenliği sorunlarının aynı sistemde farklı gerekçelerle önceliklendirilebildiğini göstermektedir.

Bu ortak dil, kitabın sonraki bölümlerinde görsel algıdan robotik sistemlere, kod üretiminden otonom ajanlara kadar farklı mühendislik bağlamlarına uyarlanabilir. Sektör ve teknoloji değiştiğinde varlıklar, saldırgan yetenekleri ve zarar türleri değişir; hangi güven sınırının nasıl aşıldığını ve seçilen kontrolün bunu hangi kanıtla yönettiğini açıklama gereksinimi devam eder.

KAYNAKÇA

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K. ve Zhang, L. (2016). Deep learning with differential privacy. *ACM SIGSAC Conference on Computer and Communications Security*. https://arxiv.org/abs/1607.00133

Berghoff, C., Neu, M. ve von Twickel, A. (2020). Vulnerabilities of connectionist AI applications: Evaluation and defence. *Frontiers in Big Data, 3*, 23. https://doi.org/10.3389/fdata.2020.00023

Biggio, B., Nelson, B. ve Laskov, P. (2012). Poisoning attacks against support vector machines. *Proceedings of the 29th International Conference on Machine Learning*. https://arxiv.org/abs/1206.6389

Carlini, N., Tramèr, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., ... Raffel, C. (2021). Extracting training data from large language models. *30th USENIX Security Symposium*. https://arxiv.org/abs/2012.07805

Fredrikson, M., Jha, S. ve Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. *ACM SIGSAC Conference on Computer and Communications Security*. https://www.cs.cmu.edu/~mfredrik/papers/fjr2015ccs.pdf

Goodfellow, I. J., Shlens, J. ve Szegedy, C. (2015). Explaining and harnessing adversarial examples. *International Conference on Learning Representations*. https://arxiv.org/abs/1412.6572

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T. ve Fritz, M. (2023). *Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection*. arXiv ön baskısı, 2302.12173. https://arxiv.org/abs/2302.12173

Gu, T., Dolan-Gavitt, B. ve Garg, S. (2017). *BadNets: Identifying vulnerabilities in the machine learning model supply chain*. arXiv ön baskısı, 1708.06733. https://arxiv.org/abs/1708.06733

MITRE. (t.y.). *MITRE ATLAS Data: ATLAS tactics, techniques, and case studies data*. Kurumsal veri deposu. https://github.com/mitre-atlas/atlas-data (Erişim: 19 Eylül 2026).

National Institute of Standards and Technology. (2012). *Guide for conducting risk assessments*. NIST Special Publication 800-30 Revision 1. https://doi.org/10.6028/NIST.SP.800-30r1

National Institute of Standards and Technology. (2023). *Artificial intelligence risk management framework (AI RMF 1.0)*. NIST AI 100-1. https://doi.org/10.6028/NIST.AI.100-1

OWASP Gen AI Security Project. (2025a). *LLM01:2025 Prompt Injection.* https://genai.owasp.org/llmrisk/llm01-prompt-injection/ (Erişim: 19 Eylül 2026).

OWASP Gen AI Security Project. (2025b). *LLM03:2025 Supply Chain.* https://genai.owasp.org/llmrisk/llm032025-supply-chain/ (Erişim: 19 Eylül 2026).

OWASP Gen AI Security Project. (2025c). *LLM05:2025 Improper Output Handling.* https://genai.owasp.org/llmrisk/llm052025-improper-output-handling/ (Erişim: 19 Eylül 2026).

OWASP Gen AI Security Project. (2025d). *LLM06:2025 Excessive Agency.* https://genai.owasp.org/llmrisk/llm062025-excessive-agency/ (Erişim: 19 Eylül 2026).

OWASP Gen AI Security Project. (2025e). *LLM10:2025 Unbounded Consumption.* https://genai.owasp.org/llmrisk/llm102025-unbounded-consumption/ (Erişim: 19 Eylül 2026).

Shokri, R., Stronati, M., Song, C. ve Shmatikov, V. (2017). Membership inference attacks against machine learning models. *IEEE Symposium on Security and Privacy*. https://arxiv.org/abs/1610.05820

Tramèr, F., Zhang, F., Juels, A., Reiter, M. K. ve Ristenpart, T. (2016). Stealing machine learning models via prediction APIs. *25th USENIX Security Symposium*. https://arxiv.org/abs/1609.02943

Vassilev, A., Oprea, A., Fordyce, A., Anderson, H., Davies, X. ve Hamin, M. (2025). *Adversarial machine learning: A taxonomy and terminology of attacks and mitigations*. NIST AI 100-2 E2025. https://doi.org/10.6028/NIST.AI.100-2e2025

5. BÖLÜM

YAPAY ZEKÂ DESTEKLİ GEREKSİNİM MÜHENDİSLİĞİ: FIRSATLAR, RİSKLER VE GELECEK YAKLAŞIMLAR

Öğr. Gör. Zeynep Yaren OĞUZ POLAT
Ankara Bilim Üniversitesi
İstatistik Bölümü
zeynep.polat@ankarabilim.edu.tr
https://orcid.org/0000-0002-2638-6306

GİRİŞ

Yazılım gereksinim mühendisliği, bir yazılım ürününün etkileşimde bulunacağı istemcilere karşı olan davranış ve bu davranışı gerçekleştirme şeklinin, belirlenen kısıtlar ve kaynaklar çerçevesinde yönetimini ele alan bir disiplindir. Bir sistemden paydaşların beklentileri doğal dille ifade edilen metinler olarak tanımlanabilir. Bu metinlerden elde edilen gereksinimlerin dil bilimsel hatalar, eksik, birbiriyle tutarsız veya bağlamdan kopuk bilgiler içermesi olası sorunlardır. Bu sorunlar sadece gereksinimlerin elde edilme aşamasını değil, gereksinim yaşam döngüsünün ve buna bağlı olarak yazılım geliştirme yaşam döngüsünün tasarım, geliştirme, test ve bakım aşamalarını da etkiler. Teknolojinin gelişmesi, kullanıcı beklentilerinin değişmesi ve yazılım sistemlerinin giderek daha karmaşık hâle gelmesi yazılım gereksinimlerinin yönetiminde farklı problemlerin ortaya çıkmasına neden olmuştur. Bu sorunları çözmek amacıyla dönemin güncel teknolojileri ve yaklaşımları gereksinim yaşam döngüsü süreçlerinin iyileştirilmesi ve temelde doğru, tutarlı ve eksiksiz gereksinim verilerinin elde edilmesi için kullanılmıştır. Örneğin, kullanıcının nihai ürünü görmeden talebini doğru ifade edememesinden kaynaklanan sorunlarla başa çıkmak amacıyla elde edilen gereksinimlerin paydaşların geri bildirimi ile iteratif bir şe-

kilde iyileştirilmesi için prototipleme modeli kullanılır. Yazılımın geliştirme süreci boyunca sıklıkla değişen gereksinimleri bulunan projelerde ise isterleri yönetebilmek amacıyla çevik yaklaşımlar kullanılmaya başlanmıştır. Yazılım yaşam döngüsü süreçlerinde olası iyileştirmeler, riskler ve verimlilik yönünden tartışılan en güncel yaklaşım ise büyük dil modeli türündeki üretken yapay zekâların kullanımıdır. Büyük dil modelleri, hızla gelişen doğal dil işleme becerileri sebebiyle gereksinim mühendisliği faaliyet süreçlerine entegre edilmeye başlamıştır. 2025-2026 yılları arasında, son bir yıllık, yapılan çalışmalarda büyük dil modeli türünde üretken yapay zekâların yazılım gereksinimlerinin üretilmesi, sınıflandırılması, kalite sorunlarının belirlenmesi gibi kritik pek çok aşamada gereksinim bilgisini yönetmek için kullanıldığı ve sunduğu fırsatlar, oluşabilecek risklerin tartışıldığı gözlemlenmiştir Üretken yapay zekâların kullanımının tek başına yeterlilik düzeyi, insan-yapay zekâ karşılaştırması, insan-yapay zekâ işbirliği, büyük dil modellerinin gereksinim mühendisliğinde insan gözetimine ihtiyaç olup olmadığını değerlendirmek güncel çalışmalarda tartışılan temel konular denebilir.

Bu çalışmanın amaçlarından biri; büyük dil modeli türündeki üretken yapay zekânın yazılım gereksinim mühendisliğinin hangi aşamalarında kullanıldığını, hangi teknik ve yaklaşımların entegre edildiğini, gereksinim mühendisliğinde karşılaşılan sorunları çözmek adına sağladığı faydaları son bir yıllık çalışmaların sunduğu yaklaşımlar üzerinden açıklamaktır. Bununla birlikte bu çalışma, güncel literatürde sunulan bulgu ve sonuçlara dayanarak üretken yapay zekânın gereksinim mühendisliği faaliyetleri için sunduğu fırsatlar, oluşturabileceği riskler, yazılım gereksinim mühendisliğinde üretken yapay zekâyı yönetecek, gözetleyecek bir insan varlığına olan ihtiyacı tartışmak ve son olarak gereksinim mühendisliğinde gelecek araştırmalar için bir değerlendirme sunmak amacıyla yapılmıştır.

1. ARKA PLAN BİLGİSİ

Bu başlık altında gereksinim mühendisliği ve üretken yapay zekânın gereksinim mühensiliği faaliyetlerinde kullanımı ile ilgili temel arka plan bilgisi sunulmaktadır.

1.1. Gereksinim Mühendisliğinde Temel Kavramlar

Yazılım gereksinimi, bir yazılım ürününün hangi sistem davranışını nasıl gerçekleştireceğini belirten bir artefakt şeklinde kısaca özetlenebilir. Fonksiyonel gereksinimler sistemin gerçekleştirmesi beklenen fonksiyonu, bu fonksiyonu hangi paydaşa karşı gerçekleştireceği ve hangi amaçla gerçekleştireceğini

ifade eder. Fonksiyonel olmayan gereksinimler ise bu davranışın nasıl gerçekleştirileceğini belirler. Sistemin performansı, güvenliği, kısıtları fonksiyonel olmayan gereksinim olarak nitelendirilir. Her iki gruptaki gereksinimlerin de iyi bir gereksinim olarak nitelendirilebilmesi için sahip olması gereken özellikler vardır. Gereksinim paydaşların ihtiyacını doğru temsil etmelidir. Bununla birlikte oluşturulan gereksinimler hiç bir paydaş ihtiyacını atlamamalı,eksik ifade etmemeli, yani eksiksiz olmalıdır. Birden fazla anlama gelmemeli ve tek bir davranışı ifade etmelidir. Tutarlı olmalı, yazılım ürünü bağlamıyla veya diğer isterlerle çelişmemelidir. Belirlenen gereksinim testleri ve performans değerleri ile ölçülebilir yani doğrulanabilir olmalıdır. Bütün bunlara ek olarak, belirlenen kaynak ve riskler çerçevesinde uygulanabilir olmalıdır. Gereksinim mühendisliği, bir yazılım ürününden veya sistemden beklenen ihtiyaçları keşfetme, bu ihtiyaçları analiz etme, analiz edilen isterlerden fonksiyonel ve fonksiyonel olmayan gereksinimleri oluşturma, oluşturulan gereksinimleri doğruluk, tutarlılık, eksiklik gibi özellikler bakımından test edip doğrulama ve sonrasında olası değişiklikleri yönetme süreçlerini içeren bir alandır. Bu süreçler gereksinim mühendisliği çerçevesinin yürütüldüğü beş ana faaliyettir. Bu beş ana faaliyetin amacı ve bu faaliyet süreçlerinde büyük dil modeli türünde üretken yapay zekânın nasıl fayda sağlayabileceği bir ön bilgi olarak aşağıda Tablo 1.1'de sunulmuştur.

Tablo 1.1 Gereksinim Mühendisliği Faaliyetleri ve Olası Yapay Zekâ Desteği

Gereksinim Mühendisliği Faaliyeti	**Faaliyet Amacı**	**Büyük Dil Modeli Türünde Üretken Yapay Zekânın Faaliyete Olası Desteği**
Ortaya Çıkarma (Elicitation)	Paydaşların ihtiyaçlarını keşfetme	Paydaşlarla görüşme yürütme, problem alan ve kaynak bilgisinden ihtiyaçları çıkarma
Analiz ve Sınıflandırma	Gereksinimleri anlamlandırma ve gruplama	İhityaçların anlam bağlamına göre fonksiyonel/ fonksiyonel olmayan şeklinde sınıflandırma, ihtiyaç analizi, analize bağlı önceliklendirme

Oluşturma	Bir gereksinim artefaktı üretme ve mevcut gereksinimleri iyileştirme	Doğal dille yazılan proje senaryoları ve ihtiyaç raporlarından kullanıcı hikâyeleri(user stories) oluşturma
Doğrulama ve Kalite	Gereksinimleri doğruluk, tutarlılık, eksiksiz olması, test edilebilirlik gibi özellikler bakımından değerlendirme	Proje bağlamında verilerden hata ayıklama, düzeltme işlemleri
Yönetme	Gereksinimleri yazılım geliştirme yaşam döngüsü boyunca izleme ve kalite koruma	Gereksinimlerin yeniden kullanımı için anlamsal bilgi kütüphaneleri oluşturma

Gereksinim mühendisliğindeki beş ana faaliyet, eksiksiz, doğru, açık ifade edilmiş, tutarlı, atomik, doğrulanabilir ve uygulanabilir gereksinimler oluşturmak ve süreçleri bu gereksinimler üzerinde yönetmek için yürütülür. Bu süreç paydaşlar, kaynak, riskler, kalite süreçleri gibi pek çok farklı parametreye bağlı olduğundan çeşitli problemleri de bünyesinde barındırır. Bu problemler, paydaşların tam olarak ihtiyaçlarını ifade edememesi, doğal dille elde edilen ihtiyaç dokümanlarını anlama ve analizinin karmaşıklığı, aynı proje kapsamındaki gereksinimlerin ayrı kalite süreçleri, kaynaklar gibi farklı bağlamlara tabi olması, gereksinimin ayrıntılı ve ölçülebilir şekilde oluşturulmaması olarak sayılabilir. Büyük dil modeli(LLM) türündeki üretken yapay zekânın doğal dili anlama, doğal dilde alan bilgisi dahilinde metin üretme ve büyük hacimli doğal dil verisini işleyebilme kapasitesi nedeniyle gereksinim mühendisliğinde karşılaşılan problemler için çözüm olarak sunulmaya ve LLM tabanlı yaklaşımların gereksinim mühendisliği faaliyetlerindeki potansiyeli araştırmaya değer görülmektedir.

1.2. Makine Öğrenmesi, Geleneksel Yapay Zekâ, Üretken Yapay Zekâ ve Büyük Dil Modelleri(LLM)

Makine öğrenmesi, bilgisayarın bir programın hangi koşullarda hangi davranışı gerçekleştireceğini belirten kuralları her bir davranış için ayrı ayrı kod bloğu yazmayı ortadan kaldırarak, büyük hacimde veriler ve farklı öğrenme teknikleri ile eğitilmiş bir model ortaya koyar. Model, büyük hacimli verilerle doğru öğrenme teknikleriyle eğitilir ve veriler arasındaki ilişkileri matematiksel olarak temsil eder. Bu temsillerin hata oranına göre, modeldeki tahmin ve ilişkilendirme parametreleri hata oranını düşürecek şekilde iyileştirilir. Tahminler üzerinde hata oranı hesaplama ve model üzerindeki iyileştirme işlemleri büyük hacimli eğitim verileri ile tekrar edilir. Sonuç olarak, öğrenmiş modelin tatmin edici doğruluk oranına ulaşması hedeflenir. Eğitilmiş bu model, yeni bir veri ya da veri seti işlerken öğrendiği eğitim verilerindeki örüntülerin kuralına göre çıktı üretir.

Geleneksel yapay zekâ, tanımlanmış iş kuralları ve öğrenilmiş veri örüntülerini kullanarak sınıflandırma, tahmin ve bu tahminin matematiksel ifadesinin dahil olduğu sınıfı belirleyip, doğru ve akıllı karar çıktısı üreten teknolojidir. Geleneksel yapay zekâ, öğrenilen veri örüntüleri üzerinden tahmin yürüterek, tatmin edici doğruluk oranında tanımlı sınıflara atama ve sınıflandırılmış veri üzerinde karar verme işlemlerini gerçekleştirir. Üretken yapay zekâ (Generative Artificial Intelligence – GenAI) ise geleneksel yapay zekâ teknolojisinden farklı olarak, büyük ölçekli veri örüntüleriyle eğitilen, bu verilerden öğrenerek, anlık girdiyi sınıflandırıp, karar verip, bu karar ve doğruluk oranı yüksek tahmine bağlı kod, metin, görsel ve benzeri yeni içerikler üreten yapay zekâ türüdür. Üretken yapay zekâ modelleri eğitim verisinde olmayan bir girdiyi, eğitim verisi örneklerinin olasılıksal yapısını da öğrendiği için işleyebilir (sınıflandırma-tahmin); eğitildiği veri setindeki olasılığa göre değerlendirip yeni bir örnek çıktı üretebilir. Büyük dil modelleri (LLMs), metin işleme-üretme temelli üretken yapay zekâ türüdür. Bu modeller, büyük hacimli metinler ile dilbilimsel ve bağlam ilişkilerinin arasındaki örüntüleri öğrenmesi amacıyla eğitilir. Modelin temel işleyişi verilen promptu bağlam başlangıcı olarak kabul edip, bağlam kapsamındaki bir sonraki belirteci(token) tahmin ederek metin üretme şeklinde açıklanabilir. Alan bilgisinin önemli ve belirli bir kapsamı, standardı olduğu durumlarda Retrieval-Augmented Generation (RAG) yöntemi kullanılabilir. RAG'da, modelin öğrendiği anlamsal bilgilere ek olarak, gerekli dokümanlar veya işlenecek veri seti modelin çıktı üretirken baz alacağı bağlama eklenmek üzere modele verilir. Özetlemek gerekirse, büyük dil modelleri eğitilip sonuç üretirken kullanacağı geniş ölçekli bağlam oluştuktan sonra doğru promptlar

verildiğinde bir sonraki tokenı oluşturmak için öğrendiği veri setini baz alarak bir olasılık hesaplar. Buna bağlı olarak seçtiği tokenı yanıta ekler ve bu işlemleri çıktı tamamlanana kadar iteratif olarak sürdürür. Doğal dili işleme ve buna bağlı yeni metin oluşturma işlemini bu şekilde gerçekleştirir.

2. AMAÇ VE YÖNTEM

2.1. Amaç

Bu çalışmanın amacı, yazılım gereksinim mühendisliği faaliyet süreçlerinde büyük dil modeli türündeki üretken yapay zekâ modellerinin hangi amaçla kullanıldığını, sağladığı faydaları, olası risklerin neler olduğunu, üretken yapay zekânın gereksinim mühendisliği faaliyetlerindeki sınırlılıklarını, insan varlığının modeli gözetmesinin gerekliliğini literatürde raporlanan sonuçlara dayanarak ortaya koymaktır.

2.2. Yöntem

Literatür taraması, gereksinim mühendisliğinin temel aşamaları olan gereksinimlerin ortaya çıkarılması, gereksinim analiz ve sınıflandırılması, gereksinim oluşturma ve iyileştirme, doğrulama ve kalite değerlendirmesi ile gereksinim yönetimi konuları çerçevesinde gerçekleştirilmiştir. Scopus üzerinde her bir gereksinim mühendisliği faaliyeti için farklı bir arama metni ile çalışmalar toplanmıştır. Arama metinlerinin ana formatı <<*gereksinim mühendisliği faaliyeti ile ilgili terimler* ***ve*** *üretken yapay zekâ* ***ve*** *büyük dil modelleri*>> olacak şekilde kurgulanmıştır. Bu kitap bölümünde kaynak olarak kullanılmak üzere, güncel literatürü sunmak amacıyla 2025-2026 yılları arasında yayınlanan makaleler çalışmaya dahil edilmiştir. Çalışmada yalnızca dergi makaleleri ve konferans bildirileri ele alınmış olup, bunların yalnızca ingilizce dilinde yayınlananları çalışmaya dahil edilmek üzere toplanmıştır. Elde edilen 261 makale önsöz, başlık ve anahtar kelimeleri dikkate alınarak konuyla doğrudan ilişkili olmayan çalışmalar elenmiştir. Yalnızca gereksinim mühendisliği aşamaları ve gereksinim mühendisliğinde üretken yapay zekâ kullanımı konularını kapsayan makaleler seçildiğinde elde edilen 27 makale bu kitap bölümü için kaynakça olarak kullanılmıştır.

3. BULGULAR

Bu bölümde, literatür incelemesi kapsamında yazılım gereksinim mühendisliği faaliyetlerinde üretken yapay zekânın kullanımına ilişkin bulgular temel faaliyetleri temsil eden beş alt başlık altında sunulmuştur.

3.1. Gereksinimlerin Ortaya Çıkarılması(Elicitation)

Gereksinimlerin ortaya çıkarılması, sistemden beklenen işlevlerin paydaşlardan ve farklı kaynaklardan elde edilmesi işlemidir. Toplanan gereksinim verisinin yalnızca paydaş ihtiyacı ya da belirli bir kaynaktan olası beklentinin çıkarılmasıyla elde edilen isterleri değil, aynı zamanda ifade edilmemiş, proje kaynağı veya organizasyon incelemesinde tespit edilmemiş ihtiyaçların da ortaya çıkarılmış halini içeren bir veri seti olması beklenir. Gereksinimlerin ortaya çıkarılması, paydaşın beklentisini tam olarak bilmemesi, eksik ifade etmesi, paydaşların isterlerinin birbirleriyle çelişmesi, projenin ihtiyaçlarının alana özgü teknik dille ifade edilmesi gibi pek çok sebeple gereksinim mühendisliğinin en zorlayıcı aşaması olarak görülebilir. Bu olası sorunları çözmek için pek çok araştırmacı büyük dil modellerinin bir çözüm olarak sunulabileceğini tartışmıştır. Yapılan çalışmalar incelendiğinde, LLM türünde üretken yapay zekâlar, gereksinimlerin ortaya çıkarılması için paydaşlarla görüşme gerçekleştirme, sesli görüşmeleri metne dönüştürme, görüşme metninden örnek gereksinimler çıkartma, eksiklikler için ortaya çıkarma soruları üretme, belirtilen ihtiyaçlara göre olası yeni ihtiyaçlar ortaya çıkarma gibi işlemler için kullanılabilir.

Korn et al. (2025), geliştirdikleri LLMREI ile, ihtiyaçlarının belirlenmesi için yürütülen paydaş görüşmelerini büyük dil modellerinin otomatik olarak yürütebilme kapasitesini değerlendirmiştir. GPT-4o modeli ile gerçekleştirilen 33 simüle paydaş ön görüşmesinde, zero-shot prompt tekniği ile modele kısa promptlar verilerek senaryo gereksinimlerinin %60,94'ü tam olarak elde edilmiş; toplam gereksinim ortaya çıkarma oranı ise %73,7 olarak raporlanmıştır. Daha detaylı ve least-to-most tekniği ile oluşturulan uzun promt, paydaş hatalarının azaltılmasında daha başarılı olmuştur. Çalışmaya katkı veren uzmanların %64,23'ü bu teknik ile gerçekleştirilen görüşmelerde hata tespit etmemiştir (Korn et al., 2025). Bununla birlikte senaryo gereksinimlerinin yaklaşık dörtte birinin ortaya çıkarılamamış olması, LLM tabanlı otomatik görüşme aracının gereksinim ortaya çıkarmada bir başlangıç desteği olarak görülebileceğini fakat tek başına sistem isterlerinin keşfedilmesi için henüz yeterli olmadığını göstermektedir.

Gereksinim ortaya çıkarma faaliyetinin bir tekniği de kullanıcının bağlam kapsamında olası sisteme karşı davranışlarını gözlemlemek ve bu gözlemlerden ihtiyaç çıkarımı yapmaktır. Li et al. (2026) kullanıcıların sistemle etkileşim davranışlarından yeni gereksinimlerin çıkarılmasını incelemiştir (Li et al., 2026). Büyük dil modeli olarak GPT-3.5 seçilmiş ve kural tabanlı davranış analizi kullanılmıştır. Görüşmelerde ifade edilen ihtiyaçların %95,6'sı davranış verilerinden gereksinim olarak elde edilebilmiştir (Li et al., 2026). Çalışma 20 katılımcı ve bir öğrenci bilgi yönetim sistemi üzerinde gerçekleştirilmiştir. Dolayısıyla sonuçlar farklı alandaki sistem ihtiyaçlarını ve fazla sayıda, farklı davranış sergileyen paydaşların bulunduğu sistemler için LLM tabanlı davranış gözlemleme yönteminin performansını net bir şekilde ölçmemektedir. Yine de, son kullanıcının sisteme karşı olan davranışlarının da yeni gereksinimlerin keşfedilmesinde LLM tabanlı yaklaşımların tartışılabileceğine dair bir değerlendirme ortaya koymuştur. Quattrocchi et al. (2026)'nin çok ajanlı çalışmasında 14.540 user story üretilmiş ve uzmanlar tarafından oluşturulan referans sete göre %96,23 semantik kapsam elde edilmiştir; öğrenci grubunda bu oran %52,83'tür (Quattrocchi et al., 2026). Bununla birlikte karşılaştırma grubunun uzman analistlerden oluşmaması nedeniyle sonuç, LLM'lerin profesyonel gereksinim mühendislerinden üstün olduğu biçiminde yorumlanmamalıdır. Fain et al. (2026) teknik gereksinim çıkarımında insan, LLM ve LLM-QFD yaklaşımlarını karşılaştırmıştır (Fain et al. 2026). Otuz altı katılımcının yer aldığı deneyde uzmanlar tarafından verilen ortalama gereksinim kalite puanı manuel koşulda 3,38/5, LLM destekli koşulda 4,33/5, LLM-QFD koşulunda ise 4,45/5 olarak ölçülmüştür. Bu sonuç, AI destekli elicitation'ın analistin rolünü ortadan kaldırmaktan ziyade analistin yaptığı işin kalitesini değiştirebileceğine işaret etmektedir. Hirabayashi et al. (2026), kullanıcı anketlerinden LLM tabanlı "user clone"lar oluşturarak ürün ihtiyaçlarının ortaya çıkarılmasını araştırmıştır (Hirabayashi et al., 2026).

10 kullanıcının yer aldığı endüstriyel çalışmada oluşturulan 60 örnek isterin %90'ı kullanıcılar tarafından "agree" veya "somewhat agree" olarak derecelendirilmiştir (Hirabayashi et al., 2026). Bu sonuç, gereksinim ortaya çıkarmanın ilk aşamalarında LLM tabanlı yaklaşımların kullanıcılar tarafından kabul edilebilir oranda olduğunu göstermekle birlikte; çalışmanın tek bir şirkette, tek bir ürün kapsamında 10 kullanıcıyla gerçekleştirilmiş olması, sonucun araştırma kapsamına bağımlı olduğu sonucunu ortaya koyuyor.

Jayasingha ve Kumari (2026), konuşmaları otomatik kullanıcı hikâyelerine dönüştüren bir yaklaşım sunmuşlardır. Whisper tabanlı ses tanıma ile fine-tuned GPT-2 modelini birleştiren sistem, 50 saatlik ses verisi ve 1.200 conversation–

story çifti üzerinde değerlendirilmiştir. Sistem BLEU=0,68, ROUGE-L=0,72 ve WER=%18 değerlerine ulaşmıştır; ve manuel dokümantasyona göre yaklaşık %50 zaman tasarrufu raporlanmıştır (Jayasingha ve Kumari, 2026).

3.2. Gereksinimlerin Analiz Edilmesi ve Sınıflandırılması

Yapılan çalışmaların bazılarının incelemesinde, gereksinimlerin sınıflandırılmasında büyük hacimde eğitim verisi mevcutsa, BERT ve DistilBERT gibi modellerin, yalnızca prompt verilerek kullanılan LLM'lerden daha başarılı olduğu gözlemlenmiştir. Örneğin; Zhu (2026)'nun çalışmasında SVM modeli 0,959 F1 değerine ulaşırken, zero-shot promt tekniği ile kullanılan modelin F1 skoru 0,685, few-shot tekniği ile kullanılan modelin F1 skoru 0,698 düzeyindedir. Few-shot tekniğinin kullanıldığı yaklaşımda fonksiyonel olmayan gereksinimlerin %89,2'si doğru sınıflandırılırken, fonksiyonel gereksinimlerin sadece %3,9'u doğru sınıfa dahil edilmiştir (Zhu, 2026). Bu ölçütler değerlendirildiğinde, modelin belirlenen bir sınıfta kümelemeyi başarılı bir şekilde gerçekleştirirken diğer kümeyi neredeyse tamamen kaçırabileceği ve bu nedenle yalnızca başarı skorlarına bakmanın büyük dil modellerinin sınıflandırma becerisini değerlendirmede bir kıstas olmadığını göstermektedir. Buna karşılık Multi-Lingual BERT (Batoool ve Naseer, 2025) ve DistilBERT (Florea ve Achirei, 2025) tabanlı modeller daha tutarlı çıktılar vermiştir. Sınıflandırma çalışmaları sonuçlarıyla benzer olarak belirsizlik tespitinde de, SpanBERT ve NER'in birlikte kullanıldığı yaklaşım ChatGPT'den daha iyi performans göstermiştir (Talha et al.,2025). Projeye ait alan bilgisinin ek bağlam kaynağı olarak modele verilmesinin, LLM tabanlı belirsizlik tespitinde doğru pozitif değerini 0,57'den 0,75'e çıkardığı ve yanlış pozitifleri %42 azalttığı gözlemlenmiştir (Poulsen et al., 2026). Zadenoori et al., (2025)'nin çalışması, promptların model tarafından otomatik iyileştirilmiş versiyonları ile kullanıcının girdiği promtlarla elde edilen sonuçlar kıyaslandığında F1 ve F2 skorlarında yaklaşık %5–9 artış sağlayabildiğini göstermektedir. Bu farklar için istatistiksel doğrulama testi yapılmamıştır. Korn (2025)'un çalışması ise doğal dille yazılan gereksinimlerin kalitesi bozulduğunda LLM performansının da düştüğünü göstermiştir.

Bu bulgular birlikte değerlendirildiğinde, analiz ve sınıflandırma görevlerinde model seçiminin, kullanılan prompt tekniğinin ve girdi kalitesinin sınıflandırma doğruluğu üzerinde belirleyici olduğu görülmektedir.

3.3. Gereksinimlerin Oluşturulması ve İyileştirilmesi

Gereksinim oluşturma ve iyileştirme çalışmalarında elde edilen sonuçlar, verilen girdiyi önceden belirlenen bir başka biçime dönüştürme veya modelin yanıt üretirken kullanması için bağlama eklenmek üzere verilen kalite kurallarına göre içerik üretme gibi kapsamı açıkça belirlenmiş gereksinim oluşturma çalışmalarında modellerin daha başarılı olduğu göstermektedir. Örneğin gereksinimlerin çalışmada belirlenen özel EARS biçimine çevrilmesinde BART-base modeli BLEU=0,708 değerine ulaşmıştır (Imran et al., 2025). Bir başka çalışmada ise kullanıcı hikâyesi oluşturma işleminde kalite ölçütleri ile istenen format ve içerik yapısında örneklerin bağlama dahil edildiği Meta-Few-Shot yaklaşımı %88,57 başarı oranı elde etmiştir (Santos et al., 2025). Buna karşılık, net bir iş tanımı (net örnek, kalite kuralları vs.) olmayan, yoruma ve farklı değerlendirmeye açık görevlerde modellerin ürettiği sonuçların dikkatli değerlendirilmesi gerekmektedir. Bir başka çalışmada, üretilen gereksinimlerin uzmanlar tarafından değerlendirilen geçerlilik medyanı 5,0 olarak bulunmuş, ancak kalite özelliği sınıflandırmalarında %19,6 hata görülmüştür (Almonte et al., 2025). Bu sonuç, modelin faydalı örnekler üretebildiğini fakat doğru sonuç için çıktının bir uzman tarafından kontrol edilmesi gerektiğini göstermektedir. Benzer şekilde, siber güvenlik gereksinimleri kapsamında yapılan bir çalışmada LlaMA 3.1 405B modeli F2=1,14 gibi tatmin edici bir performans oranına ulaşmasına rağmen deneylerde toplam 127 halüsinasyon tespit edilmiştir (Yelmo et al., 2026). Dolayısıyla, yüksek performans değerleri tek başına modelin tam olarak güvenilir çıktılar ürettiği anlamına gelmeyebilir. Bazı çalışmalar ise, modele ek bağlam bilgisi sağlamanın modellerin daha güvenilir çıktılar oluşturduğunu belirten bulgular sunmuştur. Doğal dille yazılmış sistem ihtiyaçlarından gereksinim artefaktı oluşturmak için kullanılan sistem mühendisliği dili SysML-v2'nin gereksinim üretiminde RAG kullanımının F1 değerini 0,47'den 0,80'e yükselttiği raporlanmıştır (Stein et al., 2026). Bir başka çalışmada, DR-RAR (gereksinim oluştururken modelin ihtiyaç duyduğu ek bilgiyi yalnızca ihtiyaç duyduğunda ilgili dokümandan modele getiren) bir RAG yaklaşımı sunulmuş ve bu yaklaşımın doğruluğu artırırken token kullanımını %32,5 azalttığı bilgisi paylaşılmıştır (Wei, 2026). Cardante et al. (2026), çalışmalarında doğal dildeki gereksinimlerden UML diagramları oluşturma işleminde çok ajanlı ve tek ajanlı sistemleri karşılaştırmıştır (Cardante et al., 2026). Çok ajanlı ve RAG yaklaşımı kullanan model, tek ajanlı sisteme göre daha yüksek recall sağlamış olsa da genel performans oranında belirgin bir artış sağlamadığı belirtilmiştir. Buna ek olarak, çalışma süresini 2,44 kat, maliyeti ise 2,67 kat artırdığı raporlanmıştır (Cardente et al., 2026).

Toparlamak gerekirse; belirli iş ve kalite kuralları ile gerekli bağlam bilgisinin modele verildiği yaklaşımların gereksinim oluşturma ve iyileştirme faaliyetlerinde başarılı sonuçlar ürettiği, buna karşın açık uçlu kapsamı olan işlemlerde hata, halüsinasyon ve maliyet nedeniyle insan kontrolünün gerekli olduğu sonucuna varılabilir.

3.4. Gereksinimlerin Doğrulanması ve Kalite Değerlendirmesi

Gereksinimlerin doğrulanması ve kalite değerlendirmesi faaliyetlerinde LLM'ler, oluşturulan gereksinimlerdeki kalite sorunlarını belirleme, anlamsal olarak doğrulama, kapsamda belirlenen standartlara uygunluğu kontrol etme gibi görevleri gerçekleştirmek için bir yöntem olarak tartışılmaktadır. Poulsen et al. (2026), NASA James Webb Space Telescope gereksinimlerinin doğruluk değerlendirmesi için Llama-3.3-70B modelini kullanmıştır. Modele proje konsept bilgileri verildiğinde belirsizlik tespitindeki doğru pozitif değerinin 0,57'den 0,75'e yükseldiğini ve yanlış pozitif sayısının 107'den 62'ye düştüğünü raporlamıştır. Bu sonuç, modele proje bağlam bilgisi verildiğinde yanlış örnekleri azaltabildiğini göstermektedir. Wang et al. (2025), doğal dilde yazılmış gereksinimleri biçimsel dil yapısına dönüştürüp doğrulamak için Claude 3.5-ESBMC ve ChatGPT 4.0-ESBMC yaklaşımlarını karşılaştırmıştır. Claude tabanlı yaklaşımda gereksinimlerin %46,5'i doğrulanmış, oluşturulan formal ifadelerin %79,31'inin elle yazılan ifadelerle eşdeğer mantıkta olduğu ve yanlış pozitif bulunmadığı belirtilmiştir (Wang et al., 2025). Korn (2025), GPT-4o modelinin gereksinimlerin kalite etkisi incelemiş, kalite hataları olan gereksinimler eklendikçe model performansının düştüğü gözlemlenmiştir. Quattrocchi et al. (2026), kullanıcı hikâyelerinin anlamsal kalitesini değerlendirmek için Claude 3 Opus'a bir değerlendirme rehberi sağlamış, bu rehber doğrultusunda değerlendirme yapan model ile insan değerlendiriciler arasındaki tutarlılık seviyesini ölçmüştür. Değerlendirme için Cohen's ölçütü kullanılmış; model-insan tutarlılığı için değer κ=0,87 bulunurken, farklı insan değerlendirici gruplarının birbirleriyle olan uyumu κ=0,74–0,78 bandında kalmış, en düşük insan-model uyum sonucu ise LlaMA 3.1 8B için κ=0,62 olmuştur (Quattrocchi et al., 2026). Bu bulgu, değerlendirme ölçütlerinin modele verildiği koşulda model-insan uyumunun insan-insan değerlendirme uyumundan daha fazla olduğunu gösteriyor. Iyenghar ve Pulvermüller (2026), ise endüstriyel kontrol sistemlerinde güvenlik gereksinimlerinin IEC 62443 standardını kapsama oranını GPT-5.4 ve Llama-3.3-70B ile değerlendirmiştir. GPT-5.4 %86,7, Llama ise %72,8 kap-

sam sağlamıştır. IEC 62443 standardının modele ek bağlam olarak verilmesi Llama'nın kapsamını 13,3 oranında artırmıştır (Iyenghar ve Pulvermüller, 2026). LLM'lerin gereksinim doğrulama ve kalite değerlendirmesinde yararlı olabildiği, ancak başarı oranının proje konseptine, değerlendirmeye sunulan gereksinimlerin kalitesine, modele yanıt üretirken baz alması için verilen değerlendirme ölçütleri ve standart bilgisine bağlı olarak değişkenlik gösterdiği söylenebilir.

3.5. Gereksinim Yönetimi

Gereksinim yönetimi ve izlenebilirlik çalışmalarında büyük dil modelleri, yüksek hacimli gereksinim verisini düzenlemek/gruplamak, hali hazırda elde edilmiş gereksinimleri yeniden kullanmak, isterleri geliştirme yaşam döngüsü boyunca üretilmiş/üretilecek diğer artefaktlarla (test,tasarım gereksinimleri vb.) ilişkilendirmek için kullanılmıştır. Kruk ve Zhukovska (2025), kullanıcı hikâyesi, test-kalite gereksinimi gibi isterleri içeren 18.199 Jira kaydını bir bilgi setine dönüştürüp, kullanıcılar tarafından doğal dil sorularına cevap verebilen bir sistem sunmuşlardır. Proje paydaşları sistem üzerinde; "Bu fonksiyonun güncel gereksinimleri nelerdir?", "..... tarihinde gereksinim nasıl ifade ediliyordu?" gibi sorular sormuştur. Geliştirilen büyük dil modeli tabanlı bu sistem, 355 kullanıcı sorgusunun 321 tanesine belirlenen doğruluk, uyum, eksiksizlik ölçütlerini %90,4 başarı oranıyla karşılayan yanıtlar üretmiştir. Sistemin değerlendiricisi olan paydaşlar manuel olarak Jira kayıtlarını incelerken harcadıkları zamandan 10 dakika ile 1 saat arasında tasarruf ettiklerini bildirmişler. Zaman tasarrufu değerlendirmesine ek olarak geliştirdikleri sistemin tüm proje verisini işleme maliyeti yaklaşık 1.031 dolar olarak kaydedilmiştir (Kruk ve Zhukovska, 2026). Luo et al. (2025), altı farklı uzay aracı projesinin gereksinimlerini, sonrasında başka benzer projelerde yeniden kullanılabilir bileşenlere dönüştürmek için LLM ve RAG tabanlı yaklaşım sunmuşlardır. Değerlendirici uzmanlar yeniden kullanım için oluşturulan bu gereksinim bileşenlerinin %67'sinin kalite seviyesini yüksek olarak belirlemişlerdir. Nguyen et al. (2026) ise GPT-4o modelini RAG yaklaşımı ile altı ajanlı bir sistem olarak tasarlamıştır. Bu sistem benzer kullanıcı hikâyelerini gruplayıp Jira kayıtlarıyla eşleştirmeyi hedefleyen bir yaklaşım sunar. Gruplama ve eşleştirme performans değerlendirmesi için Krippendorff's metriği hesaplanmış ve karşılaştırma bu değere göre yapılmıştır. Benzer kullanıcı hikâyelerini gruplamada Krippendorff's α=0,771 olarak raporlanırken, Jira kaydını ilişkilendirmedeki uyum

α=0,491 değerle daha düşük bir orandadır (Nguyen et al., 2026). Zaiats et al. (2025), GPT-4, Claude ve Gemini modellerinin Scrum, Kanban, XP gibi çevik yaklaşım metodolojilerinde gereksinim süreçlerine dâhil edilebileceğini belirtmiş; ancak önerilen kullanım şekillerinin nitel değerlendirmesini sunmamışlardır. Ayman et al. (2026) ise otomotiv sektöründeki sistem gereksinimleriyle bunları doğrulamak için oluşturulan testler arasında otomatik bağlantılar kuran bir yaklaşım önermişlerdir. Otomotiv yazılım gereksinimlerinin %20 ile %85'i gereksinimleri değerlendirecek olan testlerle eşleştirilmiş ve eşleşmenin tutarlılığı %89,59 olarak raporlanmıştır (Ayman et al., 2026). Bu çalışmalar değerlendirildiğinde, büyük dil modellerinin çok sayıda gereksinimleri gruplama-eşleştirme ve önceden belirlenmiş gereksinimleri yeniden kullanmaya zaman maliyeti açısından önemli ölçüde fayda sağladığı söylenebilir. Ancak, özellikle doğru gereksinim-kayıt eşleştirme işlemleri ve yeniden kullanılabilir gereksinim bileşenlerinin kalite (yeniden kullanılabilirliği) değerlendirmesi için bir uzman takibine ihtiyaç duyulduğu sonucuna varılabilir.

4. SONUÇ VE ÖNERİLER

Büyük dil modeli türünde üretken yapay zekâ modelleri, doğal dilde yazılmış metinlerin semantik analizini yapabilmesi, eğitildiği veri setinden öğrendiği örüntülerle metinlerde bir sonraki tokenı tahmin edip iteratif ekleme süreci ile anlamlı metin çıktısı üretebilmesi, uygun teknikler kullanılarak proje bağlamında ek bilgiyi kaynak alabilmesi gibi becerileri sebebiyle paydaş ihtiyaçlarının ortaya çıkarılması, gereksinim oluşturulması, test edilmesi ve yönetilmesi faaliyetleri için önemli fırsatlar oluşturmuştur. Bununla birlikte bu yapay zekâ modelleri, bağlama oturtulmadığında proje kapsamından kopuk gereksinim verileri üretebilir, hatalı sınıflandırma yapabilir veya belirlenen kalite standartlarını karşılamayabilir. Bunlar sunduğu fırsatların beraberinde gelen önemli risklerdir. Bu fırsat ve riskler kategorilere ayrılarak bu bölümde sunulmuş ve fırsat-risk değerlendirmesi ile gereksinim mühendisliğinde yapay zekâ kullanımının geleceği tartışılmıştır.

4.1. Fırsatlar

4.1.1. Verimlilik

Paydaş ihtiyaçlarının ortaya çıkarılması, gereksinimlerin tek tek incelenip gruplandırılması, büyük ölçekli gereksinim dokümanları içinde o an ihtiyaç du-

yulan gereksinimin aranması gibi yoğun insan emeği ve zamanı isteyen görevler üretken yapay zekâ modelleri kullanılarak çok daha kısa zaman içinde büyük oranda doğru şekilde yürütülebilir. Ontolojiler gibi anlamsal bir kütüphane ifade etmekten ziyade, büyük dil modelleri yüksek hacimli doğru örneklerle eğitilir ve RAG gibi yaklaşımlarla model çıktı üretirken ilgili projenin bağlamına ait bilgilere erişip token üretme işlemini ilgili konu kapsamında yapabilir. Doğru verilerle eğitilmiş, proje kapsamındaki kaynaklar sağlanmış, doğru prompt tekniği kullanılarak işleyen bir model, uzman bir insandan çok daha hızlı sonuç üretebilir. Böylece mühendisler ihtiyaçlardan gereksinim bilgisi çıkarmak, gereksinimi oluşturmak, gruplamak, aramak için zaman kaybetmek yerine bu sürecin sadece kontrolünü sağlayıp, değerlendirmesini yaparak daha hızlı ve verimli bir gereksinim yönetim süreci gerçekleştirebilir.

4.1.2. Gereksinim Kalitesi

Büyük dil modelleri önceden eğitilmiş oldukları veri setine ek olarak, ilgili projenin iş ve kalite kuralları gibi bağlam kaynakları verilerek kullanılabilir. Örneğin RAG yaklaşımı, model metin üretirken öğrenmiş olduğu veri örüntüsüne ek olarak bir kaynağa ihtiyaç duyarsa dinamik olarak kendisine sunulan ek kaynaktan bilgiyi alma ve metni bu referansa göre üretmeye devam etmesi olarak açıklanabilir. Bu sayede üretilen çıktı, konsepte bağlı olarak daha yüksek doğrulukla üretilir.

Quattrocchi et al. (2026) önceden belirlenmiş kalite talimatları kaynak olarak verildiğinde büyük dil modelinin gereksinim değerlendirmelerinin insan değerlendirmeleriyle büyük ölçüde uyumlu olduğunu belirtmişlerdir. Ek kaynak sunma ve farklı prompt tekniklerinin çıktı kalitesini etkilediği bu alanda yapılan çalışmaların büyük çoğunluğunda raporlanmıştır. Örneğin, Santos et al. (2025) belirli kalite kriterleriyle Meta-Few-Shot tekniğinin kullanıcı hikâyelerinin kalitesini artırdığını raporlamıştır.

4.1.3. Paydaş Desteği

Büyük dil modellerinin sesten metin üretebilmesi, gereksinimlerin ortaya çıkarılması sırasında görüşmeleri gerçekleştirmede yardımcı kaynak olarak kullanılabileceğini akla getirmiştir. Model, daha önce yapılan toplantı notlarını ya da ses kayıtlarından daha düzenli dokümanlar oluşturup dönüştürdüğü metni bağlam olarak kullanarak görüşmede paydaşlara yöneltilecek sorular üretebilir ve paydaşın paylaştığı bilgilerden örnek gereksinimler oluşturabilir. Ek olarak, paydaş ihtiyaçlarını analiz ederek unutulduğunu varsaydığı, gerekli olabilecek yeni ihtiyaçlar/ister-

ler önerebilir. Tüm bu faydalı özelliklere rağmen, üretken yapay zekânın doğrudan paydaşın gerçekten ne ifade etmek istediğini bilen üstün bir akıl olmadığını bilmek gerekir ve ürettiği metinlerin bir uzman tarafından kontrolü sağlanmalıdır. Korn et al. (2025) ve Fain et al. (2026) tarafından gerçekleştirilen çalışmalar, LLM'lerin teknik gereksinim çıkarımı süreçlerinde analiste yardımcı olabileceğini, ancak insan değerlendirmesinin önemini koruduğunu göstermektedir.

4.2. Riskler

4.2.1. Halüsinasyon, Eksiklik ve Yapay Zekâya Sorgusuz Güven

Büyük dil modelleri, kendisine sağlanan bağlam kaynaklarını (eğitim verileri, proje kapsamında sunulan ek kaynaklar vs.) baz alarak öğrendiği veri örüntüsüne göre bir sonraki tokenı tahmin edip yüksek oranda uyumlu olanı ekleyerek metni oluşturur. Bu işlemden anlaşılacağı üzere; modeller gerçekten olması gereken bilginin ne olduğunu değerlendirmez, sadece sahip olduğu bağlam bilgisi ile tutarlı ve dil bilimsel olarak anlamlı metin üretir. Dolayısıyla üretken yapay zekâlar anlam olarak doğru ve tutarlı fakat gerçekte var olmayan gereksinimler üretebilir ya da temsil edilecek gereksinimi eksik bilgi ile oluşturabilir. Bu durum halüsinasyon dediğimiz gerçekte var olmayan fakat varlığı anlamsal olarak tutarlı bir gereksinim ya da sistemin gerçekleştireceği davranışı eksik ifade eden bir gereksinim çıktısı olarak sonuçlanabilir. Gereksinim analistleri, oluşturulan veri tutarlı, dilbilimsel ve anlamsal olarak mantıklı olduğu için modele güvenir ve bu halüsinasyonları nihai gereksinim olarak kabul edebilir. Gereksinimlerdeki bu hatalar yazılım geliştirme yaşam döngüsü boyunca bir kar topu gibi büyüyerek aşamaları etkileyebilir. Daha da kötüsü nihai üründe tutarsız bir davranış sergilenebilir. Yelmo et al. (2026) siber güvenlik gereksinimi üretiminde çok sayıda halüsinasyon tespit etmiş; Korn et al. (2025) ise LLM tabanlı gereksinim ortaya çıkarma sırasında kaynak bilgisinin desteklemediği bir maliyet tahmininin üretildiğini raporlamıştır.

4.2.2. Maliyet

Büyük dil modelleri kullanılan sistemlerde, performansı artırmak için çok ajanlı yaklaşım ve karmaşık retrival (modelin dinamik olarak bağlam bilgisine erişmesi ve bilgiyi getirmesi) işlemleri kullanıldığı güncel literatürde yer bulmuştur. Bu tarz karmaşık yapılar modelin yanıt performansı ve yanıtların doğruluğunu artırırken maliyeti artırır. Örneğin, Cardante et al. (2026), çok ajanlı yaklaşımın bazı recall ölçütlerinde iyileşme sağlamasına rağmen elle tutulur bir

üstünlük göstermediğini; buna karşın tek ajanlı sisteme göre daha fazla zaman ve finansal maliyet gerektirdiğini ortaya koymuştur.

4.2.3. Gizlilik ve Güvenilik

Bir sistemin gereksinimleri, genellikle henüz dış paydaşlarla paylaşılmamış proje ya da ürün bilgilerini içerir. Dolayısıyla bu bilgiler organizasyon içinde gizli bilgi olarak nitelendirilir. Bu bilgilerin hangi modelle paylaşılabileceği, modelin bu bilgiyi nasıl işlediği, sunulan bilginin başka sistemler/kişiler tarafından erişilebilir olup olmadığı bilgisi kritiktir. Örneğin, bir milli savunma sanayi projesinin gereksinimleri bazen organizasyon bünyesindeki her paydaşla dahi paylaşılmaz. Bunların herhangi bir model sağlayıcısı ile paylaşılması ciddi oranda gizlilik ihlâline sebebiyet verebilir. Bu durum, özellikle gizlilik esas olan projeler için model kullanımını bir risk olarak değerlendirilebilir. Yelmo et al. (2026), çalışmalarında güvenlik-kritik gereksinim üretiminde gözetimsiz kullanımın uygun olmadığını belirtmişlerdir.

4.3. İnsan-Yapay Zekâ İşbirliği

Modellerin hatalı, eksik, bağlamdan kopuk gereksinmler üretebileceğini; modelin gerçek hayat projelerinin ihtiyaçlarını bilen ve bu ihtiyaçlara çözüm üreten bir varlık değil, gereksinim müdendisliği sürecinde insan gücüne duyulan ihtiyacı azaltmak ve zamanı etkili kullanmak için yardımcı bir destek aracı olduğunu kabul etmek gerekir. Dolayısıyla büyük dil modeli türünde üretken yapay zekâlar bu çalışmada da değerlendirilen faydaları sebebiyle gereksinim mühendisliği faaliyetlerinde destek olarak kullanılabilir. Buna karşın, yapay zekânın destek ekibinde olduğu her faaliyet sürecinden sorumlu, modeli doğru yönlendirebilen, proje planına ve konseptine hakim gereksinim mühendislerinin süreçte ana figür olarak bulunması gerekmektedir.

Gereksinim mühendislerinin bu faydalı aracı doğru kullanabilmeleri, modelin ürettiği gereksinimlerin kalitesini artırmak, bu gereksinimlerin değerlendirmesini-kontrolünü yapabilmek amacıyla sürece etkin bir şekilde dahil olması gerekir. Prompt tekniklerine, RAG yaklaşımı ve bu yaklaşımda kullanılacak kaynak bilgisine, gereksinim mühendisliği faaliyetlerinin kalite standart ve süreçlerine hakim, uzman kimseler olarak süreçte var olması beklenmektedir. Bu uzman/sorumlu mühendislerin sürecin işleyişini modele bırakıp yalnızca takibini yapan personeller olarak değil; yapay zekâ araçlarını gereksinim mühendisliği faaliyet süreçlerinde yöneten uzmanlar olması gerekmektedir.

4.4. Gelecek Yaklaşımlar

Yazılım gereksinim mühendisliğinin geleceğinde, üretken yapay zekâların bağımsız bir gereksinim yönetim ajanı olarak var olmasının henüz mümkün görülmediği; büyük dil modellerinin gereksinim mühendisliği faaliyetlerinde bir araç olarak kullanılması öngörülmektedir. Üretken yapay zekâların bağlama duyarlı modeller ve insan gözetiminde kullanılacak gereksinim mühendisliği sistemleri olarak disiplinde yer alacağı söylenebilir. RAG gibi, modellere proje dokümanları, kalite standartları, prosedürler gibi kaynakları ek bağlam bilgisi olarak sağlayan yaklaşımların, oluşturulan gereksinimlerin kalitesini büyük ölçüde arttırdığı gözlemlenmiştir. Gelecekte gereksinim kalitesini artırmak, proje kapsamına daha yüksek oranda bağlı ve geliştirme döngüsünün diğer aşamalarıyla tam entegre bir yaklaşım için, ek bağlam olarak ihtiyaçların sesli ifadesi, tasarım dokümanları, son kullanıcının sistem davranışını simule ettiği bir video kaydı gibi farklı türde veriler ile farklı prompt tekniklerinin performans değerlendirmesi yapılabilir. Bu değerlendirme sonucunda farklı türde verileri işleyerek bağlama bağlı, doğru, eksiksiz, daha kaliteli aday gereksinimlerin oluşturulduğu sistemler geliştirilebilir. Modeller geliştikçe, metin vb. verileri işleyebilme kapasitesi arttıkça her disiplinde olduğu gibi üretken yapay zekâ gereksinim mühendisliğinde de performans arttırıcı bir teknoloji olarak yer alacaktır. Bununla birlikte bu teknolojinin gereksinim mühendisliğindeki faydaları, insanlar tarafından doğru bağlam verisinin belirlenmesine, prompt mühendisliği-yapay zekâ mühendisliği disiplinleri çerçevesinde yetkin insanların modeli doğru şekilde kullanabilmelerine ve alanında uzman mühendislerin gereksinim kalite süreçlerini etkin bir şekilde yürütmesine bağlı olarak artacaktır.

KAYNAKÇA

Almonte, J. T., Boominathan, S. A., & Nascimento, N. (2025). Automated non-functional requirements generation in software engineering with large language models: A comparative study. Conference of the Centre for Advanced Studies on Collaborative Research.

Ayman, H., Sedik, M., Mostafa, K., et al. (2026). Cluster-aware dual-level test specification generation for large-scale automotive software requirements. Mediterranean Conference on Embedded Computing.

Batoool, R., & Naseer, A. (2025). Functional and non-functional requirements classification: A comparative evaluation of pre-trained LLMs and ML techniques. ComTech.

Cardante, D., Cruz, A. M. R. D., & Cruz, E. F. (2026). LLM-based generation of structured requirements and UML models from requirements elicitation notes. International Conference on Evaluation of Novel Approaches to Software Engineering.

Fain, N., Vukašinović, N., & Čok, V. (2026). Comparing human, LLM, and LLM-QFD approaches to technical requirement extraction. Proceedings of the Design Society.

Florea, A., & Achirei, D. (n.d.). Text classification on software and hardware requirements using natural language processing techniques.

Hirabayashi, M., Washizaki, H., Ubayashi, N., et al. (n.d.). Generating user clones from questionnaires: A lightweight approach to requirements elicitation.

Imran, M. H., Hassan, B., & Jahankhani, H. (2025). Automated EARS-based requirements generation with lightweight large language models. 2025 IEEE International Conference on Technology Management, Operations and Decisions (ICTMOD).

Iyenghar, P., & Pulvermüller, E. (2026). Process-aware generation of operational technology (OT) security requirements using BPMN context and large language models. International Conference on Evaluation of Novel Approaches to Software Engineering.

Jayasingha, C. P., & Kumari, S. (2026). Automating voice-based conversations into formal user stories using LLM and speech recognition. 2026 6th International Conference on Advanced Research in Computing (ICARC).

Korn, A. (n.d.). Smells like trouble: Investigating the impact of requirements quality on LLM-supported software engineering.

Korn, A., Gorsch, S., & Vogelsang, A. (2025). LLMREI: Automating requirements elicitation interviews with LLMs. IEEE International Requirements Engineering Conference.

Kruk, R., & Zhukovska, N. (2025). Toward AI-assisted framework for Agile requirement knowledge management: Five-stage generative LLM approach. Automation, Control, and Information Technology.

Li, T., Wen, C., & Zhang, X. (2026). From interaction to evolution: A behavior-driven framework for automated software requirements elicitation. International Conference on Automated Software Engineering.

Luo, Y., Wang, Y., Li, X., et al. (2025). Leveraging large language models for reusable requirements management in aerospace software. IEEE International Requirements Engineering Conference.

Nguyen, K., Sami, M. A., Zhang, Z., & Abrahamsson, A. (n.d.). A context-aware multi-agent approach to enhancing user story management in Agile software development.

Poulsen, V. V., Guertler, M., Eisenbart, B., et al. (2026). Context-aware large language models for ambiguity detection in requirements. Proceedings of the Design Society.

Quattrocchi, G., Pasquale, L., Spoletini, P., & Baresi, L. (2025). Can LLMs generate user stories and assess their quality? IEEE Transactions on Software Engineering.

Santos, R., Freitas, G., Steinmacher, I., et al. (2025). User stories: Does ChatGPT do it better? International Conference on Enterprise Information Systems.

Stein, A., Kizgin, U. V., Waldmann, N., et al. (2026). Prompt-strategy-driven SysML-v2 artefact generation using large language models for model-based systems engineering. Applied Informatics.

Talha, F., Tahir, T., & Nadeem, T. (2025). A semiautomated approach for detecting ambiguities in software requirements using SpanBERT and named entity recognition. Journal of Software: Evolution and Process.

Wang, W., Farrell, M., Cordeiro, L. C., & Zhao, L. (2025). Supporting software formal verification with large language models: An experimental study. IEEE International Requirements Engineering Conference.

Wei, L. (2026). DRRAR: Domain-aware requirement refinement via adaptive RAG. 2026 2nd International Conference on Artificial Intelligence and Digital Ethics (ICAIDE).

Yelmo, J. C., Martín, Y.-S., & Perez-Acuna, S. (n.d.). Experimental evaluation of AI-augmented cybersecurity requirements generation leveraging LLMs' capabilities.

Zadenoori, M. A., Zhao, L., Alhoshan, W., & Ferrari, A. (2025). Automatic prompt engineering: The case of requirements classification. Requirements Engineering: Foundation for Software Quality.

Zaiats, O., Mykhalyk, D., Yatsyshyn, V., et al. (2025). Methods for integrating large language models into requirements management in Agile methodologies. International Workshop on Information Technologies: Theoretical and Applied Problems.

Zhu, L. (2026). Prompt engineering strategies for automated software requirements classification. 2026 7th International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE).