Yapay zekâ kendi kodunu yazabilir mi?

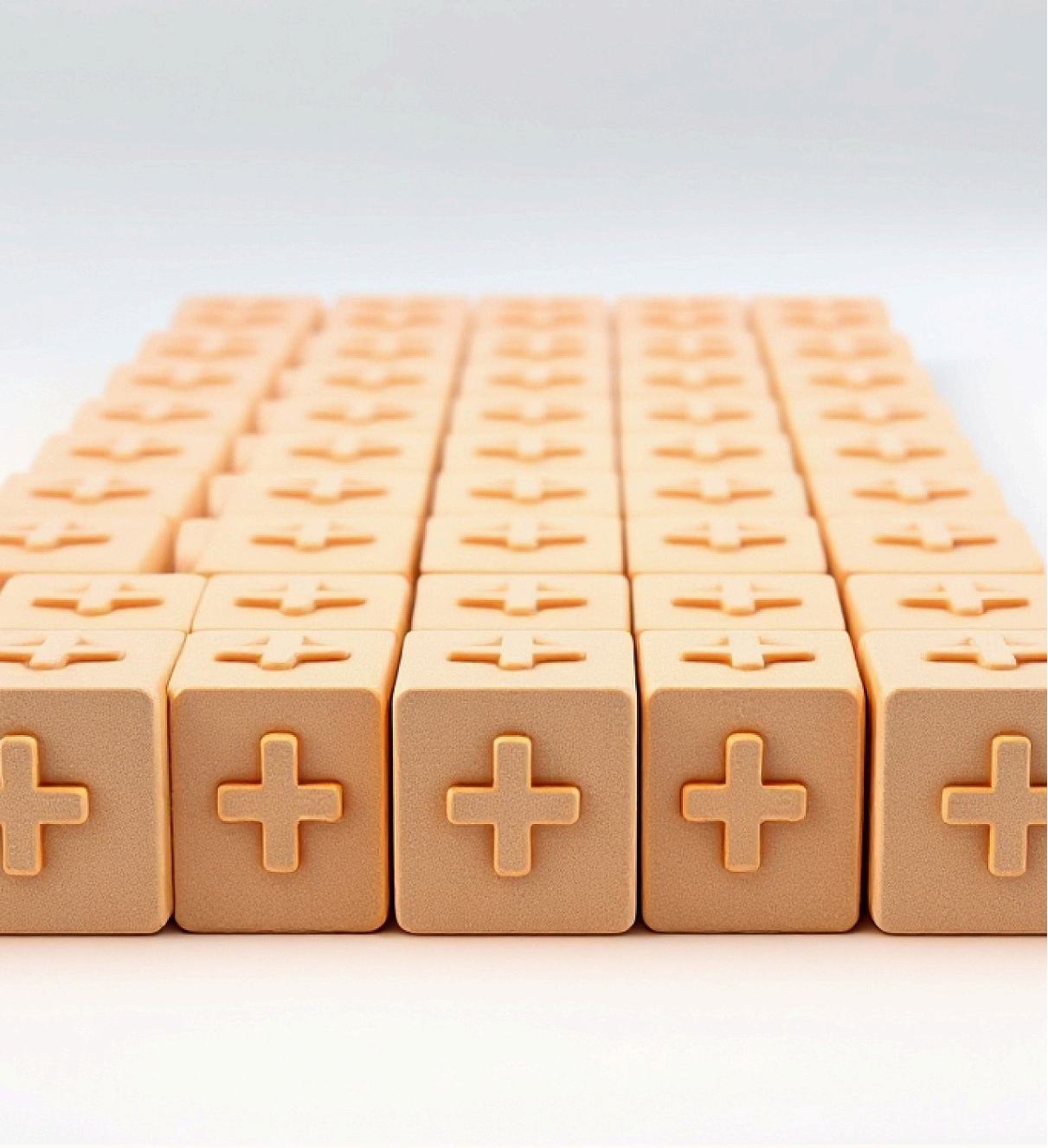


www.yzokul.com.tr

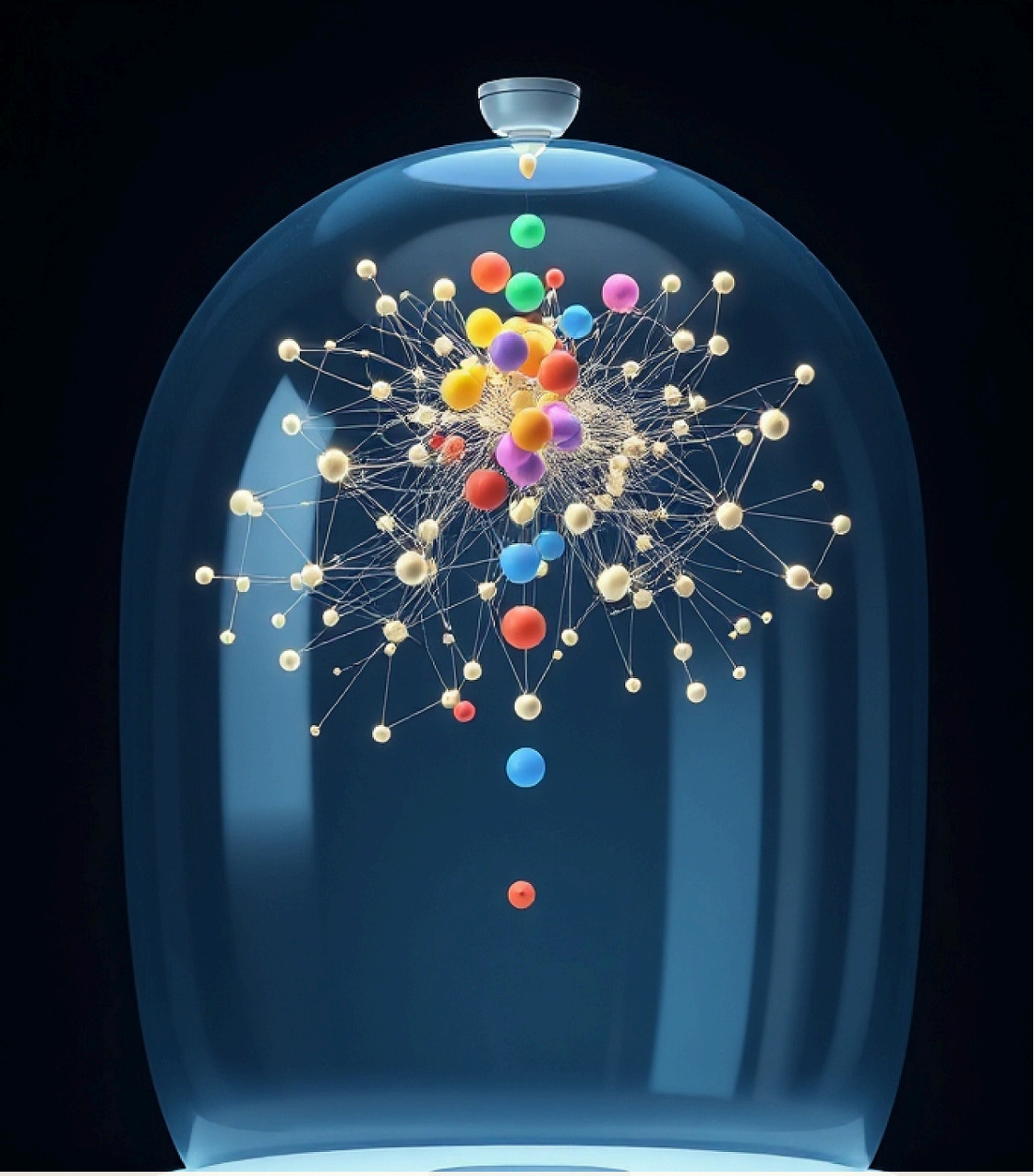
Yapay zekâ kendi kodunu yazabilir mi?



Yapay zekâ kendi kodunu yazabilir mi? Kısa cevap evet, kod önerebilir, ama bu kendi sistemini serbestçe değiştirebileceği anlamına gelmez. Bir yapay zekâ modeli senin isteğine dayanarak yeni kod üretebilir, ancak kendini bağımsız olarak değiştiremez. Bu kitap aradaki farkı adım adım açıklıyor.



Kod, bir bilgisayarın adım adım izlediği yazılı talimatlar bütünüdür. Bu talimatları bir kişi ya da bir makine, bilgisayarların anladığı özel bir yazım biçimi olan bir programlama dilinde yazabilir. Örneğin, bir kod satırı bilgisayara iki sayıyı toplamasını söyleyebilir. Kod kesin olmalıdır, çünkü bilgisayarlar yalnızca talimatların söylediğini yapar.



Yapay zekâ modeli, yararlı çıktıyı tahmin etmek için çok sayıda metin ve kod üzerinde eğitilmiş bir bilgisayar programıdır. Bir insan değildir ve insan gibi düşünmez; eğitim verilerinden öğrendiği kalıplara dayanarak öneriler üretir. Kod istediğinde model, isteğine uyabilecek kodu tahmin eder. Yazdığı kodu gerçekten anlamaz.



Ne istediğini normal kelimelerle, örneğin iki sayıyı toplama isteği gibi anlatabilirsin. Model bu isteği önerilen bir kod parçasına dönüştürür. Buna doğal dil denir, yani günlük konuştuğun dil. Örneğin şöyle yazabilirsin: iki sayıyı toplayan bir fonksiyon yaz. Model ardından bunu yaptığını düşündüğü kısa bir kod parçacığı üretir.



Bir model, elinde zaten olan kodu da alıp ona dayanarak yeni kod önerebilir. Bu, sıfırdan başlamadan kod üretmesinin bir başka yoludur. Örneğin, ona sayıları çıkaran bir fonksiyon gösterirsen, sayıları toplayan benzer bir fonksiyon önerebilir. Model mevcut kodu izlenecek bir kalıp olarak kullanır.



Şu isteği düşün: iki sayıyı toplayan bir fonksiyon yaz. Model, Python gibi bir programlama dilinde kısa bir fonksiyon üretebilir. İki girdi alır ve toplamlarını döndürür. Bu örnek, sade bir isteğin nasıl önerilen bir fonksiyona dönüştüğünü gösterir, ama öneri yalnızca bir başlangıç noktasıdır, bitmiş bir çözüm değildir.



Önerilen kod hatalar içerebilir ve hiç çalışmayabilir. Kod yazmak, doğru kod yazmakla aynı şey değildir. Örneğin, model yanlış bir değişken adı kullanabilir ya da gerekli bir adımı unutabilir. Bu tür hatalı kod çalıştırıldığında başarısız olur. Kontrol edene kadar her zaman hata olabileceğini varsay.



Bir öneri, istediğinin yalnızca bir kısmını kapsayabilir. Parçalar eksik olabilir, bu yüzden kod henüz işin tamamını yapamaz. Örneğin, iki sayıyı toplayan ve sonucu yazdıran bir fonksiyon istersin, ama model yalnızca toplama kısmını yazar ve yazdırmayı unutur. Eksik kodu senin tamamlaman gerekir.



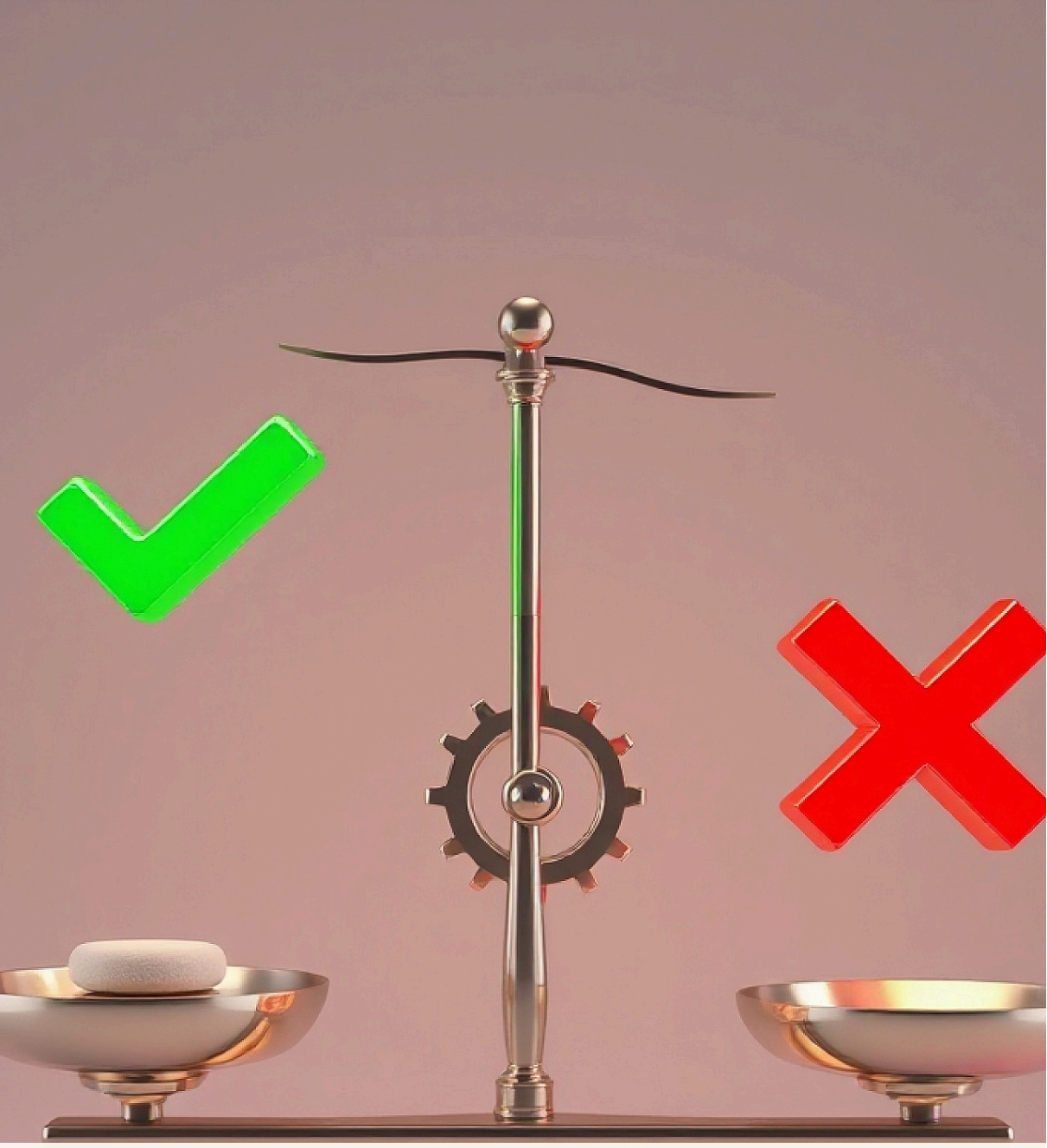
Kod, kullanılacağı ortamla uyumsuz olabilir. Orada bulunmayan şeylere, örneğin belirli bir kütüphaneye ya da bir programlama dilinin belirli bir sürümüne dayanabilir. Örneğin, kod yalnızca daha yeni bir sürümde bulunan bir fonksiyon kullanabilir. O zaman senin daha eski sisteminde çalışmaz.



Kodu çalıştırmak, talimatları yürüten bir bilgisayar tarafından yapılan ayrı bir adımdır. Kod önermek ile onu çalıştırmak iki farklı eylemdir. Model önerdiği kodu çalıştırmaz; bunu senin ya da başka bir sistemin yapması gerekir. Örneğin, önerilen kodu bir dosyaya kopyalar ve python myfile.py gibi bir komutla çalıştırırsın.



Test etmek, kodun yapması gerekeni yapıp yapmadığını kontrol etmek demektir. Bu, kodu yazmaktan ya da çalıştırmaktan ayrı bir adımdır. Kodu farklı girdilerle çalıştırıp çıktıları beklediğinle karşılaştırabilirsin. Örneğin, toplama fonksiyonunu 2 ve 3 ile test et ve 5 döndürüp döndürmediğine bak. Döndürmüyorsa kodda bir sorun var demektir.



Çalıştırma ve test etmeden sonra birinin sonuçların yeterince iyi olup olmadığına karar vermesi gerekir. Değerlendirme, süreçteki bir başka ayrı adımdır. Kodun ihtiyaçlarını karşılayıp karşılamadığına, yeterince hızlı olup olmadığına ve kullanımının güvenli olup olmadığığına sen karar verirsin. Örneğin, toplama fonksiyonu tüm test durumlarında çalışıyorsa kabul edebilir, başarısız oluyorsa reddedebilirsin.



Kod yazmak, kendi sistemini serbestçe değiştirmekle aynı şey değildir. Bir model, kod önerebildiği için kendini yeniden yazma hakkına sahip olmaz. Modelin kendi kodu ve ayarları yaratıcıları tarafından sabitlenmiştir. Kullanman için kod önerebilir, ama kendi iç işleyişini kendi başına değiştiremez.



Bir model, insanların belirlediği sınırları olan sabit bir sistem içinde çalışır. Bu sınırların ötesine kendi başına geçemez. Örneğin, yeni donanıma erişemez, eğitim verilerini değiştiremez ya da kendi güvenlik kurallarını değiştiremez. Bu sınırlar sistemi güvenli ve öngörülebilir tutmak için vardır. Bunu anlamak, yapay zekânın neleri yapabildiğini ve yapamadığını görmene yardımcı olur.



Yaygın bir hata, önerilen kodun doğru ve kullanıma hazır olduğunu varsaymaktır. Çalıştırma, test etme ve değerlendirmenin sırada olduğunu her zaman hatırla. Örneğin, önerilen bir fonksiyonu önce test etmeden hayati bir programa kopyalama. Yapay zekâ tarafından üretilen tüm kodu, son ürün değil, kontrol edilmesi gereken bir taslak olarak gör.



Yapay zekâ sade isteklerden ya da mevcut koddan kod önerebilir, ama sonuçlar hatalı, eksik ya da uyumsuz olabilir. Kod yazmak, bir modelin kendi sistemini serbestçe değiştirebileceği anlamına gelmez. Koda güvenmeden önce onu her zaman çalıştır, test et ve değerlendir. Modelin insanların belirlediği sınırlar içinde çalıştığını ve kendini değiştiremediğini unutma.