



# Bozok Journal of Engineering and Architecture

Araştırma Makalesi/Research Article

## Transformer (Dönüştürücü) Mimarisi Kullanarak Zaman Serilerinde Anomali Tespiti

Osman Safa ÇİFÇİ

Yozgat Bozok Üniversitesi, Mühendislik-Mimarlık Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Yozgat,

### MAKALE BİLGİSİ

#### Makale Tarihleri:

Geliş tarihi  
31.01.2023  
Kabul tarihi  
07.04.2023  
Yayın tarihi  
21.06.2023

#### Anahtar Kelimeler:

Zaman serileri  
Anomali tespiti  
Dönüştürücü modeli

### ÖZET

Çok değişkenli zaman verilerinde anomali tespiti pek çok uygulama için önem arz etmektedir. Ancak, yüksek doğrulukla ve hızlı bir şekilde anomali tespiti tam olarak çözülebilmemiş bir problem değildir. Pek çok zaman serisi verilerindeki oynaklığı da problemi güçlendirmektedir. Son yıllarda anomali tespiti için istatistiksel yaklaşımların yanısıra derin öğrenme tabanlı yöntemler ortaya çıkmıştır. Bunların içinde 2017 yılında ortaya çıkan ve ilk olarak dil çeviri uygulamalarında kullanılan dönüştürücü (transformer) tabanlı modeller önerilmiştir. Dönüştürücü tabanlı modeller paralel olarak işlem yaparak eğitim sürecini kısaltıp çeşitli uygulamalarda yüksek performans sergilemektedir. 2022 yılında çıkan TranAD isimli model hem temel modellere göre 100 kata kadar daha hızlı eğitim sürecine sahiptir hem de performans olarak pek çok modelden daha iyidir. Bu çalışmada ilgili model daha da hızlı eğitim yapacak şekilde düzenlenmiş ve PSM (Pooled Server Metrics) isimli çok değişkenli zaman serisi verisinde uygulanmıştır. Ortaya çıkan model, Intel(R) i5-5200 işlemci ve 8GB RAM'e sahip dizüstü bilgisayarda yirmi saniyenin altında eğitilip, kendisinden daha fazla veri kullanan ve daha çok kaynak gerektiren modellere benzer performans sergilemiştir. Elde edilen model hesaplama maliyeti açısından getirdiği avantajla çeşitli zaman serisi verilerinde, özellikle de sanayi 4.0 uygulamalarında, kendine yer bulabilir.

## Time Series Anomaly Detection Using Transformer Networks

### ARTICLE INFO

#### Article history:

Received  
31.01.2023  
Accepted  
07.04.2023  
Published  
21.06.2023

#### Keywords:

Time series  
Anomaly detection  
Transformer model

### ABSTRACT

Anomaly detection in multivariate time series data is important for many applications. However, high accuracy and rapid detection is not a fully resolved problem. Data volatility in many time series data sets also reinforces the problem. In recent years, deep learning-based methods have emerged for anomaly detection in addition to statistical approaches. Among these, transformer-based models, which were initially introduced in 2017 and first used in language translation applications, are proposed. Transformer-based models operate in parallel, shortening the training process and exhibiting high performance in various applications. A model named TranAD, which was released in 2022, has a training process up to 100 times faster than the baseline models and is better than many anomaly detection models in terms of performance. In this study, the constructed model was arranged to train even faster and was applied to the multivariate time series data called PSM (Pooled Server Metrics). The resulting model was trained in under twenty seconds on a laptop having Intel(R) i5-5200 processor with an 8GB RAM and performed similar to models that used more data and required more resources. With the advantages it brings in terms of computational cost, the resulting model can find its place in various time series data, especially in industry 4.0 applications.

## 1. GİRİŞ

Son zamanlarda, zaman serisi metriklerine dayalı hesaplama açısından maliyetli anomali tespiti literatürde aktif bir konu olmuştur. Zaman serisi verileri, belirli bir zaman aralığında toplanan bir dizi veri noktasıdır ve veri noktaları, Bilişim Teknolojileri sistemlerindeki varlıkların (cihazlar ve uygulamalar) sağlığını ve performansını gösteren ölçümlerdir. Zaman serisi verilerine dayalı doğru ve zamanında uyarılarla, sistem operatörleri gerekli tepkileri verebilir ve ciddi sonuçları önceden önleyebilir. Web uygulamalarının temel performans göstergeleri [1], sunucu makinelerinin takibi (CPU yükü, ağ kullanımı ve bellek kullanımı) [2], enerji santrali sistemleri [3] ve atık su arıtma tesisi [4] gibi çok çeşitli sektörlerde zaman serisi verilerinde anomali tespiti

ORCID ID: Osman Safa ÇİFÇİ: 0000-0003-4431-0783

\*Sorumlu yazar(lar)/Corresponding author(s): Yozgat Bozok Üniversitesi, Mühendislik-Mimarlık Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Yozgat, Türkiye

Tel: +90 354 242 10 01

Fax:

E-mail: o.safa.cifci@bozok.edu.tr

Bu makaleye atıfta bulunmak için/To cite this article: O. S. Çifçi, "Transformer (Dönüştürücü) Mimarisi Kullanarak Zaman Serilerinde Anomali Tespiti", Bozok Journal of Engineering and Architecture, vol. 2, no. 1, pp. 1-7, 2023

uygulamıştır. Sanayi 4.0 uygulamalarındaki sensör verisi büyüklüğünü de göz önünde bulundurduğumuzda anomali tespitinin önemi de gün geçtikçe artmaktadır [5], [6].

Zaman serisi girdi verileri iki kategoriye ayrılabilir: yalnızca tek bir zamana bağlı metrik için tek değişkenli ve birden fazla zamana bağlı metrik için çok değişkenli [7]. Çok değişkenli zaman serisi verilerinde anomali tespiti daha zor bir problemdir ve bu makalemizde odaklanacağımız veri çeşididir. Birden fazla metrik arasında korelasyon veya bağımlılıklar mevcut olsun veya olmasın, tespit yöntemleri her bir metrik üzerinde bağımsız olarak analiz yapılabilir [8] veya tüm metrikleri bir kerede dikkate alabilir [2]. Bununla birlikte, sistemler zaman içinde geliştikçe, metriklerin sayısı önemli ölçüde artabilir [9]. Bu durumda ise literatürde önerildiği gibi her değişken için ayrı bir model eğitmek ve saklamak pratik değildir [10]. Ayrıca, pratikte, sistem operatörlerinin sistem durumlarını metrik düzeyde gözlemlemesi bireysel parçaların gözlemlenmesine nazaran çok daha fazla çaba gerektirmektedir [11].

Gelecekte, hedef sistemler daha karmaşık hale gelecek ve büyük miktarda veri üretmeye devam edecektir. Bu da birkaç nedenden dolayı eğitim sürecinde denetimli öğrenmenin kullanılmasının imkansızlığına yol açacaktır [12]. İlk olarak, alan uzmanlarının anomali verileri etiketlemesi yoğun emek gerektirmektedir. İkincisi, anomaliler nadiren meydana gelir ve verilerin geri kalanına normal veri noktaları hakimdir. Bu nedenle, normal verilerin anomali verilerden çok daha fazla olduğu dengesiz bir veri kümesine neden olur ve sınıflandırıcılar anomalileri kaçırmaya eğilimli hale getirir [13]. Yukarıda belirtilen nedenlerden dolayı, denetimsiz öğrenmeye dayalı anomali tespiti daha çok tercih edilmektedir ve son yıllarda araştırmalara konu olmuştur.

Hayati bir gerçek dünya sorunu olarak, denetimsiz zaman serisi anomali tespiti geniş çapta araştırılmıştır. Anomali belirleme kriterine göre kategorize edilen paradigmlar, kabaca yoğunluk tahmini, kümeleme tabanlı, yeniden yapılandırma tabanlı ve otoregresyon tabanlı yöntemleri içerir. Yoğunluk tahmin yöntemlerine gelince, klasik yöntemler yerel aykırı değer faktörü (LOF) [14] ve bağlantı aykırı değer faktörü (COF) [15] aykırı değer belirleme için sırasıyla yerel yoğunluğu ve yerel bağlantıyı hesaplar. DAGMM [16] ve MPPACD [17], temsillerin yoğunluğunu tahmin etmek için Gauss Karışım Modelini entegre eder. Kümelemeye dayalı yöntemlerde, anomali puanı her zaman küme merkezine uzaklık olarak belirlenir. SVDD [18] ve Deep SVDD [19], temsilleri normal verilerden kompakt bir kümeye toplar. THOC [20], ara katmanlardan gelen çok ölçekli zamansal özellikleri hiyerarşik bir kümeleme mekanizmasıyla birleştirir ve çok katmanlı mesafelerle anomalileri tespit eder. ITAD kümelemeyi ayrıştırılmış tensörler üzerinde yürütür [21]. Rekonstrüksiyon tabanlı modeller, anomalileri rekonstrüksiyon hatasıyla tespit etmeye çalışır. Park ve arkadaşları, geçici modelleme için LSTM omurgasını ve yeniden yapılandırma için Variational AutoEncoder'ı (VAE) kullanan LSTM-VAE modelini sunmuştur [22]. Su ve diğerleri tarafından önerilen OmniAnomaly, LSTM-VAE modelini normalleştirici bir akışla daha da genişletir ve algılama için yeniden yapılandırma olasılıklarını kullanır [2]. Li ve arkadaşları, aynı anda birden fazla seri arasındaki karşılıklı ve iç bağımlılığı modellemek için omurgayı hiyerarşik bir VAE ile kurmuştur [23]. Çekişmeli üretici ağlar (GAN) [24] yeniden yapılandırmaya dayalı anomali tespiti için de kullanılmıştır ve çekişmeli bir düzenleme işlevi görür [25]. Otomatik regresyon tabanlı modeller, tahmin hatası ile anomalileri tespit eder. VAR (vektör otoregresyon) modeli ARIMA (otoregresif fark alma, hareketli ortalama) modelini genişletir ve gecikmeye bağlı kovaryansa dayalı olarak geleceği tahmin eder [26]. Otoregresif model ayrıca LSTM modeli ile değiştirilebilir [27].

LSTM'ler gibi yinelenen modellerin yavaş ve hesaplama açısından pahalı olduğu bilinmektedir [28]. Son üç yıl içinde önerilen yöntemler, daha doğru tahminler için bir girdi olarak bir zaman serisi penceresine sahip derin sinir ağlarını kullanır [29]. Bununla birlikte, girdiler daha fazla veri yoğun hale geldikçe, küçük sabit boyutlu pencere girdileri, modele verilen sınırlı yerel bağlam bilgisi nedeniyle bu tür modellerin algılama performansını sınırlar [28]. Bu nedenle hızlı ve minimum genel giderle üst düzey performansları yakalayabilen bir modele ihtiyaç vardır. İlk defa 2017 yılında ortaya atılan dönüştürücü (transformer) modelleri [30] doğal dil işleme [31], ses işleme [32] ve bilgisayarlı görü [33] gibi sıralı veri işlemede ciddi performans göstermiştir ve kısa sürede herkesin ilgisini çeken ChatGPT modeli de dönüştürücü tabanlı bir modeldir. Zaman serisi analizi için, öz dikkat mekanizmasının avantajından yararlanarak, güvenilir uzun vadeli zamansal bağımlılıkları keşfetmek için transformer modelleri kullanılmaktadır [34].

Bu çalışmada, çok değişkenli zaman serisi içeren bir veride anomali tespiti için modelden bağımsız meta öğrenmeyi (MAML) [35] ve çekişmeli bir eğitim süreci [24] kullanan dönüştürücü tabanlı bir model olan ve 2022 yılında sunulan TranAD [36] tabanlı bir model baz alındı. Mimarisi, eğitim ve test için hızlı olmasını sağlarken, büyük girdi dizileriyle kararlılığını korumaktadır. Basit dönüştürücü tabanlı kodlayıcı-kod çözücü ağlar, sapma çok küçükse, yani normal verilere nispeten yakınsa anomalileri kaçırmaya eğilimindedir. Çekişmeli eğitim süreci, bu yeniden yapılandırma hatalarını hafifletilebilmektedir. TranAD modeli tabanlı çalışmamızda ilk öncelik olarak hızlı bir eğitim sürecini hedefledik. Bunun için verideki eğitim verilerinin sadece %20'sini kullandık, temel bileşenler analizi [37] ile girdilerimizi ciddi oranda düşürdük ve hiperparametre ayarlaması kullandık. Sonuçta, PSM (Pooled Server Metrics) isimli eBay şirketinin sunucularından çekilen veri

(https://github.com/eBay/RANSynCoders/tree/main/data) 5. Nesil I5 işlemci ve 8 GB RAM'e sahip bir bilgisayar ile yirmi saniyenin altında eğitimimizi tamamladık ve çok daha uzun eğitim sürelerine sahip diğer modellere benzer sonuç elde ettik.

## 2. MATERYAL VE METOT

### 2.1. Problem Tanımı ve Verilerin Ön İşlemden Geçirilmesi

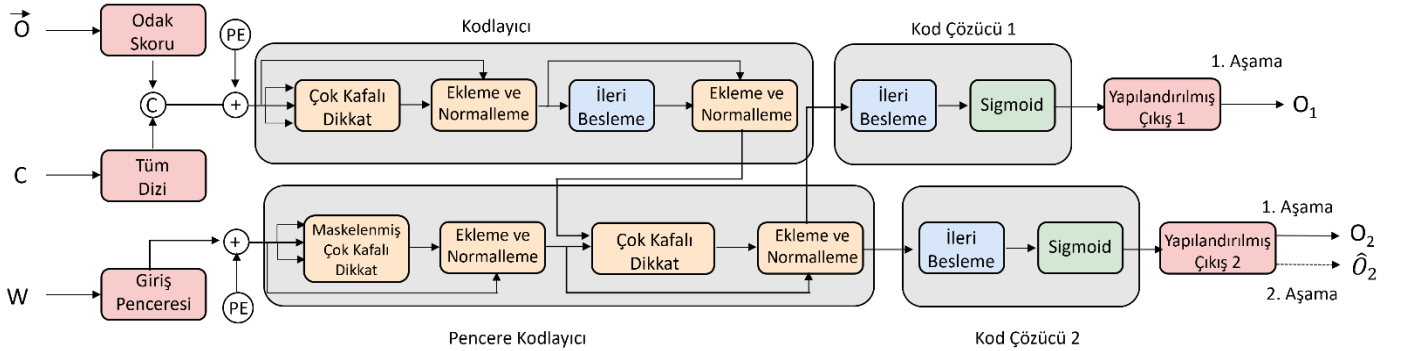
Çok değişkenli  $T$  büyüklüğündeki bir zaman serisi aşağıdaki gibi gösterilebilir.

$$T = \{x_1, x_2, \dots, x_T\} \quad (1)$$

Burada  $x_t$ ,  $t$  anında alınan veriyi göstermektedir.  $x_t \in \mathbb{R}^m, \forall t$ . Burada  $m$  verinin boyutunu göstermektedir. Problem tanımı olarak, bir  $T$  zaman serisi eğitim girdisi olarak verildiğinde eğitim serisiyle aynı modaliteye sahip önceden sistemin görmediği herhangi bir  $\hat{T}$  test zaman serisi için,  $\mathcal{Y} = \{y_1, \dots, y_T\}$  tahminini elde etmemiz gerekir. Burada  $y_t \in \{0, 1\}$  test verinin  $t$ . zaman damgasındaki veri noktasının anomali olup olmadığını belirtmek için kullanılır. 1 sayısı anomali bir veri noktasını belirtmektedir ve 0 sayısı da normal bir veri noktasını belirtir.

Modelimizi daha sağlam hale getirmek için birkaç ön işlemden geçirdik. Bunlardan birincisi 25 boyutlu verimizi PCA yöntemi ile 4 boyuta indirgedik. Ardından, eğitim kümesindeki verileri her bir kolondaki minimum ve maksimum değerlerini gözönünde bulundurarak normalleştirdik ve son olarak hem eğitim hem de test için zaman serisi pencerelerine dönüştürdük. Zaman serisi penceresi  $W$  ile gösterilmiştir. Ayrıca bir  $T$  dizisinin geçerli zaman damgasına  $t$  anına kadar olan zaman dilimini de göz önünde bulundurur ve bunu da  $C_t$  olarak gösteririz.

Veriler ön işlemden geçtikten sonra, her giriş penceresi  $W_t$  için anomali etiketi olan  $y_t$  değerini doğrudan tahmin etmek yerine, önce bu pencere için bir anomali skoru  $s_t$  tahmin edilir. Geçmiş giriş pencereleri için anomali puanlarını kullanarak bir eşik değeri  $D$  hesaplanır ve eşik değerini geçen giriş penceresi anomali olarak etiketlenir. Matematiksel olarak,  $y_t = 1(s_t \geq D)$  olarak ifade edilir.



Şekil 1. Orjinal TranAD modeli [36]. Burada C harfi birleştirmeyi (concatenate) ve artı işareti toplamay göstermektedir.

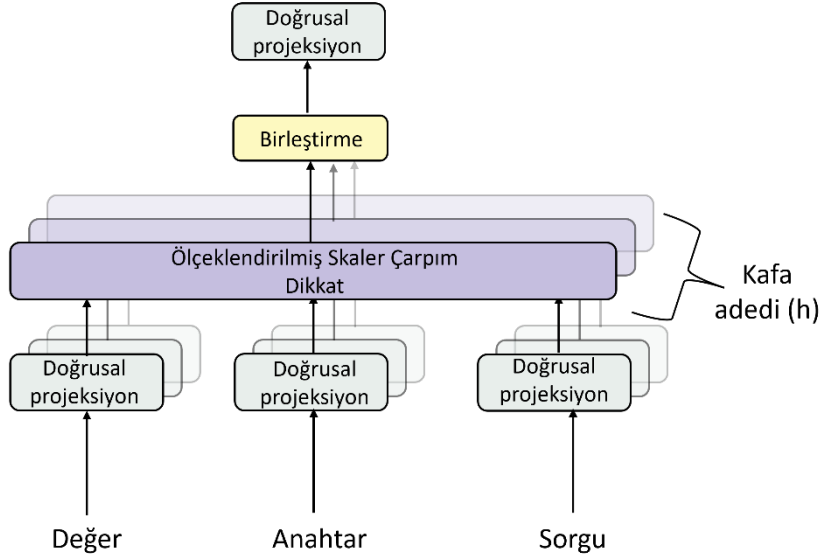
### 2.2. Transformer Modeli

Tıpkı diğer kodlayıcı-kod çözücü modellerinde olduğu gibi, bir dönüştürücü modelinde bir giriş dizisi, birkaç dikkat tabanlı dönüşümden geçer. Şekil 1, TranAD'de kullanılan sinir ağının mimarisini göstermektedir. Kodlayıcı, geçerli zaman damgasına ( $C$ ) kadar tüm diziyi bir odak puanıyla kodlar. Pencere kodlayıcı bunu giriş penceresinin  $W$  kodlanmış bir temsili oluşturmak için kullanır ve bu daha sonra yeniden yapılandırmasını oluşturmak için iki kod çözücüye iletir.

Orjinal transformer makalesinde olduğu gibi [30]  $W$  veya  $C$  gibi çok değişkenli bir dizi, önce  $m$  modalitesiyle bir matris formuna dönüştürülür. Üç matrisin  $Q$  (sorgu),  $K$  (anahtar) ve  $V$  (değer) ölçekli nokta çarpımını dikkat matrisi olarak tanımlarız:

$$Dikkat(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{m}}\right)V \quad (2)$$

$Q$ ,  $K$  ve  $V$  girdi matrisleri için,  $Q_i$ ,  $K_i$ ,  $V_i$  değerlerini elde etmek için önce  $h$  (kafa sayısı) ileri beslemeli katmanlardan geçirek çok başlı öz dikkat uygularız ve ardından ölçekli skaler çarpım (dot product) uygulanır. Çok başlı öz dikkat, modelin farklı konumlardaki farklı temsil alt uzaylarından gelen bilgilere ortaklaşa katılmasına olanak tanır ve bunları paralel şekilde yürütebilir. Şekil 2 sorgu, anahtar ve değer girdileri ile çok başlı dikkat mekanizmasını göstermektedir.



Şekil 2. Orjinal dönüştürücü modelinde önerilen çok kafalı dikkat mekanizması [30].

GAN modellerinin, bir girdinin anomali olup olmadığı gibi karakteristik görevlerde iyi performans sergilediği literatürde gösterildiğinden, bu model de zaman açısından verimli bir GAN tarzı çekişmeli eğitim yönteminden yararlanmaktadır. Model iki dönüştürücü tabanlı kodlayıcı ve iki kod çözücünden oluşmaktadır (Şekil 1). Model çıkarımı kısmı ise iki aşamada gerçekleştirilmektedir. Önce  $W$  ve  $C$  çiftini bir girdi ve (başlangıçta sıfır matrisi olan) bir odak puanı  $F$  olarak alınır. Bunlar konum kodlamasından geçip, çok başlı dikkat katmanlarına girer (girdi dizileri içindeki zamansal eğilimleri yakalamak için girdi zaman serisi pencerelerini ve tüm diziyi kullanarak dikkat ağırlıkları üretir) ve buradan sonra toplama ve normalizasyon kısmı yapılır. Pencere kodlayıcı kısmı da kodlayıcı ile benzer işlemleri yürütür fakat verileri sonraki konumlarda maskeleyerek için pencere kodlayıcıdaki öz dikkat değiştirilir. Bu, paralel eğitime izin vermek için tüm  $W$  ve  $C$  verileri bir kerede verildiğinden, eğitim sırasında kod çözücünün gelecekteki zaman damgası değerleri için veri noktalarına bakmasını önlemek için yapılır. İki kodlayıcı da ardından sigmoid aktivasyon fonksiyonuna sahip ileri beslemeli yapay sinir ağına girer. Sigmoid kullanmamızın nedeni 0 ile 1 arasındaki değerlere sahip çıkış üretebilmektir.

### 2.3. Çekişmeli Eğitim

Yukarıda kısaca bahsedildiği üzere eğitim süreci iki fazdan oluşmaktadır. Birincisinde girdiyi tekrar üretmeye çalışırken (odak değeri 0), ikinci fazda model odak girdiyi inşa etmeye çalışmaktadır.

#### 1. Aşama - Girişin Yeniden Yapılandırılması

Transformer modeli, her girdi zaman serisi penceresinin yeniden oluşturulmasını tahmin etmemizi sağlar. Bunu, her zaman damgasında bir kodlayıcı-kod çözücü ağı gibi davranarak yapar. Bununla birlikte, geleneksel kodlayıcı-kod çözücü modelleri genellikle kısa vadeli eğilimleri yakalayamaz ve sapmalar çok küçükse anomalileri gözden kaçırma eğilimi gösterir [38]. Bu zorluğun üstesinden gelmek için, TranAD modeli yeniden oluşturulmuş pencereyi iki aşamada tahmin eden bir çıkarım stili geliştirmiştir. İlk aşamada, model giriş penceresinin yaklaşık bir yeniden yapılandırmasını oluşturmayı amaçlar. Daha önce bahsedilen odak puanı olarak adlandırılan bu çıkarımdan sapma, Transformer kodlayıcı içindeki dikkat ağına sapmaların yüksek olduğu alt dizilere odaklanarak geçici eğilimleri çıkarmasını kolaylaştırır. Böylece, ikinci fazın çıktısı, birinci fazdan üretilen sapmalara göre şartlandırılmıştır.  $O_1$  ve  $O_2$  çıktıları bu aşamada üretilir.

#### 2. Aşama - Odaklanmış Girdinin Yeniden Yapılandırılması

İkinci aşamada, birinci kod çözücü için yeniden oluşturma kaybını odak puanı olarak kullanılmaktadır. Bu aşamada  $\hat{O}_2$  çıktıları elde edilmektedir. Bu iki aşamalı eğitim süreci sayesinde, hem kısa süreli ortalamadan çok fazla sapmayan anomalilerin tespiti kolaylaşacak hem de model daha stabil hale gelecektir.

Diğer rakip eğitim çerçevelerinde olduğu gibi, kritik zorluklardan biri eğitim istikrarını korumaktır. Bunun üstesinden gelmek için, iki ayrı kod çözücünün (Şekil 1'deki Kod Çözücüler 1 ve 2) çıktılarını kullanan bir çekişmeli eğitim prosedürü tasarlanmıştır. Başlangıçta, her iki kod çözücü de giriş zaman serisi penceresini bağımsız olarak yeniden oluşturmayı amaçlar. Eğitim sırasındaki kayıp fonksiyonları, literatürdeki gibi belirlenmiştir [22]. İkinci aşamadaki kaybı tanımlamak gerekirse, ikinci kod çözücü  $\|\hat{O}_2 - W\|_2$  farkını maksimize ederek giriş penceresi ile birinci aşamadaki (odak puanlarını kullanarak) birinci kod çözücü tarafından oluşturulan aday rekonstrüksiyonu arasında ayırım yapmayı amaçlar. Öte yandan, birinci kod çözücü, girişi mükemmel bir şekilde yeniden yapılandırarak (yani,  $O_1 = W$ ) dejenere bir odak puanı (sıfır vektörü) oluşturmayı amaçlayarak ikinci kod çözücüye kandırmayı amaçlar. Bu aşamada ikinci kod çözücü, aşama 1'deki girişi eşleştirmeyi amaçladığı  $O_2$  ile aynı çıktıyı üretmeye iter. Bu nedenle, birinci kod çözücünün amacı, bu kendi kendini şartlandıran çıkışın yeniden oluşturma hatasını en aza indirmekken, ikincinin amacı aynı şeyi en üst düzeye çıkarmaktır.

Son olarak, eğitim döngümüz, sinir ağlarının hızlı adaptasyonu için birkaç adımlık (few-shot) bir öğrenme modeli olan modelden bağımsız meta öğrenmeyi (MAML) kullanır [35]. Bu, modelin sınırlı verilerle girdi eğitimi zaman serisindeki geçici eğilimleri

öğrenmesine yardımcı olur. Özellikle daha hızlı bir eğitim için elimizdeki verilerin sadece %20'sini kullandığımız için modelden bağımsız meta öğrenme kullanmak lehimize olacaktır.

## 2.4. Çıkarım ve Anomali Tespiti

Önceden görülmemiş bir veri için  $((\hat{W}, \hat{C}))$  anomali skoru aşağıdaki gibi ifade edilir:

$$s = \frac{1}{2} \|O_1 - \hat{W}\|_2 + \frac{1}{2} \|\hat{O}_2 - \hat{W}\|_2 \quad (3)$$

Test süresindeki çıkarım yine iki aşamada çalışır ve dolayısıyla tek bir yeniden yapılandırma çifti elde ederiz ( $O_1$  ve  $\hat{O}_2$ ). Test zamanında, sadece mevcut zaman damgasına kadar olan veriler dikkate alınır ve bu nedenle bu işlem sıralı olarak çalışır. Bir zaman damgasında her boyut için anomali puanlarına ( $s_i$ ) sahip olduğumuzda, elde edilen bu puan belirlenen eşikten büyükse, zaman damgası anomali olarak etiketlenir. Önceki çalışmalarda yaygın olarak kullanıldığı gibi, adil karşılaştırma için ve eşik otomatik ve dinamik olarak seçmek için "Peak Over Threshold" (POT) yöntemi kullanılmıştır [39].

## 3. BULGULAR VE TARTIŞMA

Modelimiz PyTorch kütüphanesi kullanılarak eğitildi. Model, 0.0005 başlangıç öğrenme oranı ile adım planlayıcı kullanarak eğitmek için Adam iyileştiricisini kullanmaktadır. Ayrıca, modelimizde seçtiğimiz pencere boyutu 10, dönüştürücü enkoderlerindeki katman sayısı 1, kodlayıcıların ileri beslemeli birimindeki katman sayısı 2, kodlayıcı katmanlarındaki gizli birim adedi 64 ve kodlayıcılardaki seyreltme (dropout) oranı da 0.6 olarak seçildi. Kafa sayısı da verinin boyutu olan 4 olarak belirlendi. Eğitim verilerinin %20'si doğrulama için ayrıldı ve eğitime katılmadı. Son olarak epoch sayısı olarak 3 belirlendi.

Orjinal TranAD modelinde kullanılmayan PSM verisi orjinalde 25 boyutlu 132,841 adet veriye ve test kümesinde 87,841 adet veriye sahip olan bir veridir. Bu veride anomali oranı %11.07 olarak verilmiştir. Çalışmanın özgün yönünü oluşturan önemli noktalarında birisinin hızlı bir eğitim süreci olmasından dolayı deneyler Windows 10 işletim sistemi çalıştıran, Intel(R) i5-5200 işlemci ve 8GB RAM'e sahip bir bilgisayar kullanılarak çalıştırılmıştır. Ayrıca herhangi bir GPU desteği kullanılmamıştır. PSM verisinin eğitimi için verilerin sadece yüzde 20'si kullanılmıştır ve tüm eğitim süreci bu eski bilgisayarda 20 saniyenin altında sürmüştür.

Elde edilen veriler, literatürde kullanılan modeller ile karşılaştırılmıştır. Karşılaştırma için 3 ayrı metrik kullandık. Bunlardan birincisi kesinlik, yani gerçek pozitiflerin gerçek pozitifler ve yanlış pozitiflere oranı, ikincisi duyarlılık, yani gerçek pozitiflerin gerçek pozitifler artı yanlış negatiflere oranıdır. Son metrik ise, hem kesinliğin hem de duyarlılığın etkisi olduğu ve genellikle daha geçerli bir metrik olarak kabul edilen F1 skorudur. Modelimizin performans değerleri ve diğer modeller ile kıyası Tablo 1'de verilmiştir.

**Tablo 1.** Modelimizin diğer temel yöntemlerle performans karşılaştırması

Model	Kesinlik (Precision)	Duyarlılık (Sensitivity)	F1 Skoru
OCSVM	62.75	80.89	70.67
Isolation Forest	76.09	92.45	83.48
LOF [14]	57.89	90.49	70.61
Deep-SVDD [19]	95.41	86.49	90.73
DAGMM [16]	93.49	70.03	80.08
MMPCAD [17]	76.26	78.35	77.29
VAR	90.71	83.82	87.13
LSTM [40]	76.93	89.64	82.80
CL-MPPCA	56.02	99.93	71.80
ITAD [21]	72.80	64.02	68.13
LSTM-VAE [22]	73.62	89.92	80.96
BeatGAN [25]	90.30	93.84	92.04
OmniAnomaly [2]	88.39	74.46	80.83
InterFusion [23]	83.61	83.45	83.52
THOC [20]	88.14	90.99	89.54
<b>Bu Çalışma</b>	<b>65.62</b>	<b>92.56</b>	<b>76.79</b>

Burada birkaç noktayı belirtmenin önemli olduğuna inanıyoruz. Birincisi, bu çalışmanın özgün yönü hızlı bir eğitim süresine sahip olmasıdır. Öyle ki orjinal TranAD modeli diğer bazı modellerin %1'i kadar bir sürede eğitimi tamamlayabilmiştir. Bu çalışmada biz de PCA ile verilerin boyutunu azaltmak ve eğitim verilerinin sadece %20'sini kullanarak 20 saniyede 25 boyutlu ve 87,841 adet veri noktasına sahip bir test kümesinde eğitimi çok daha uzun süren modellerin performansına yaklaşmıştır. Orjinal TranAD modeli diğer verilerinde boyut azaltma ve eğitim kümesini azaltmaya gitmeden pek çok modelden daha üstün performans sağlamıştır. Bu çalışmadaki kompakt model diğer verilerde çalıştırılmamıştır fakat performanstan çok fazla kayıp vermeden iyi değerlerin elde edilebileceğini düşünmekteyiz. İkincisi, Tablo 1'de gösterilen diğer modellerde eğitim kümesinin tamamı

kullanıldığı için, eğitim kümesinin sadece %20'sinin kullanılması durumunda modelimizin daha da öne çıkacağını öngörmekteyiz fakat bu çalışmada ilgili deneyler yapılmamıştır.

#### 4. SONUÇLAR

Sonuç olarak, uzun bir süredir pek çok kişinin üzerinde çalıştığı ve finanstan bilgi teknolojilerine, sanayiden çevre sağlığına kadar çeşitli sektörleri kapsayan çok değişkenli zaman serilerinde anomali tespiti yapabilen dönüştürücü tabanlı ve çekişmeli eğitim süreci izleyen bir model daha hızlı olabilmesi için modifiye edildi. Intel(R) i5-5200 işlemci ve 8GB RAM'e sahip bir dizüstü bilgisayarda 20 saniyenin altında süren eğitim süreci ile PSM (Pooled Server Metrics) isimli çok değişkenli zaman serisi verisinde çalıştırılıp bu verileri kullanan literatürdeki diğer modeller ile karşılaştırıldı. Modelimiz eldeki eğitim verilerinin sadece %20'lik bir kısmını kullanarak %66 kesinlik, %93 duyarlılık ve %77 F1 skoruna ulaşmıştır. İlgili modelin basit uyarlamalar ile çok değişkenli zaman serisi verilerinde, özellikle de yoğun sensör verisi toplayan Sanayi 4.0 için önemli uygulamalarda kullanılabilirliği öngörülmektedir.

#### YAZAR KATKILARI

Osman Safa ÇİFÇİ: Verilerin işlenmesi, modelin oluşturulması, sonuçların alınması ve makalenin hazırlanması.

#### ÇIKAR ÇATIŞMASI

Herhangi bir çıkar çatışması bulunmamaktadır.

#### ETİK

Bu makalenin yayımlanmasında herhangi bir etik sorun bulunmamaktadır.

#### KAYNAKLAR

- [1] H. Xu *et al.*, "Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications," in *The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018*, 2018.
- [2] Y. Su, R. Liu, Y. Zhao, W. Sun, C. Niu, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.
- [3] C. Zhang *et al.*, "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 2019.
- [4] A. Sánchez-Fernández, F. J. Baldán, G. I. Sainz-Palmero, J. M. Benítez, and M. J. Fuente, "Fault detection based on time series modeling and multivariate statistical process control," *Chemom. Intell. Lab. Syst.*, 2018.
- [5] F. Zeng, M. Chen, C. Qian, Y. Wang, Y. Zhou, and W. Tang, "Multivariate time series anomaly detection with adversarial transformer architecture in the Internet of Things," *Futur. Gener. Comput. Syst.*, 2023.
- [6] M. Ma, L. Han, and C. Zhou, "BTAD: A binary transformer deep neural network model for anomaly detection in multivariate time series data," *Adv. Eng. Informatics*, 2023.
- [7] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A Review on Outlier/Anomaly Detection in Time Series Data," *ACM Computing Surveys*. 2021.
- [8] Y. Zhou, R. Qin, H. Xu, S. Sadiq, and Y. Yu, "A data quality control method for seafloor observatories: The application of observed time series data in the east China sea," *Sensors (Switzerland)*, 2018.
- [9] G. Li and J. J. Jung, "Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges," *Information Fusion*. 2023.
- [10] A. Garg, W. Zhang, J. Samarán, R. Savitha, and C. S. Foo, "An Evaluation of Anomaly Detection and Diagnosis in Multivariate Time Series," *IEEE Trans. Neural Networks Learn. Syst.*, 2022.
- [11] X. Wang, D. Pi, X. Zhang, H. Liu, and C. Guo, "Variational transformer-based anomaly detection approach for multivariate time series," *Meas. J. Int. Meas. Confed.*, 2022.
- [12] Z. Tian, M. Zhuo, L. Liu, J. Chen, and S. Zhou, "Anomaly detection using spatial and temporal information in multivariate time series," *Sci. Rep.*, 2023.
- [13] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep Learning for Anomaly Detection in Time-Series Data: Review, Analysis, and Guidelines," *IEEE Access*. 2021.
- [14] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," *SIGMOD Rec. (ACM Spec. Interes. Gr. Manag. Data)*, 2000.
- [15] J. Tang, Z. Chen, A. W. C. Fu, and D. W. Cheung, "Enhancing effectiveness of Outlier detections for low Density Patterns," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2002.

- [16] B. Zong *et al.*, “Deep autoencoding Gaussian mixture model for unsupervised anomaly detection,” in *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 2018.
- [17] T. Yairi, N. Takeishi, T. Oda, Y. Nakajima, N. Nishimura, and N. Takata, “A Data-Driven Health Monitoring Method for Satellite Housekeeping Data Based on Probabilistic Clustering and Dimensionality Reduction,” *IEEE Trans. Aerosp. Electron. Syst.*, 2017.
- [18] D. M. J. Tax and R. P. W. Duin, “Support Vector Data Description,” *Mach. Learn.*, 2004.
- [19] L. Ruff *et al.*, “Deep one-class classification,” in *35th International Conference on Machine Learning, ICML 2018*, 2018.
- [20] L. Shen, Z. Li, and J. T. Kwok, “Timeseries anomaly detection using temporal hierarchical one-class network,” in *Advances in Neural Information Processing Systems*, 2020.
- [21] Y. Shin *et al.*, “ITAD: Integrative Tensor-based Anomaly Detection System for Reducing False Positives of Satellite Systems,” in *International Conference on Information and Knowledge Management, Proceedings*, 2020.
- [22] D. Park, Y. Hoshi, and C. C. Kemp, “A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder,” *IEEE Robot. Autom. Lett.*, 2018.
- [23] Z. Li *et al.*, “Multivariate Time Series Anomaly Detection and Interpretation using Hierarchical Inter-Metric and Temporal Embedding,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2021.
- [24] I. J. Goodfellow *et al.*, “Generative adversarial nets,” in *Advances in Neural Information Processing Systems*, 2014.
- [25] B. Zhou, S. Liu, B. Hooi, X. Cheng, and J. Ye, “Beatgan: Anomalous rhythm detection using adversarially generated time series,” in *IJCAI International Joint Conference on Artificial Intelligence*, 2019.
- [26] E. Mahmoud, “Accuracy in forecasting: A survey,” *J. Forecast.*, 1984.
- [27] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, “Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2018.
- [28] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, “USAD: UnSupervised Anomaly Detection on Multivariate Time Series,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2020.
- [29] H. Zhao *et al.*, “Multivariate time-series anomaly detection via graph attention network,” in *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2020.
- [30] A. Vaswani *et al.*, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, 2017.
- [31] T. B. Brown *et al.*, “Language models are few-shot learners,” in *Advances in Neural Information Processing Systems*, 2020.
- [32] C. Z. A. Huang *et al.*, “Music transformer: Generating music with long-term structure,” in *7th International Conference on Learning Representations, ICLR 2019*, 2019.
- [33] Y. Wang, R. Huang, S. Song, Z. Huang, and G. Huang, “Not All Images are Worth 16x16 Words: Dynamic Vision Transformers with Adaptive Sequence Length,” in *NeurIPS*, 2021.
- [34] S. Li *et al.*, “Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting,” in *Advances in Neural Information Processing Systems*, 2019.
- [35] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in *34th International Conference on Machine Learning, ICML 2017*, 2017.
- [36] S. Tuli, G. Casale, and N. R. Jennings, “TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data,” *Proc. VLDB Endow.*, vol. 15, no. 6, pp. 1201–1214, Jun. 2022.
- [37] A. Abdulaal, Z. Liu, and T. Lancewicki, “Practical Approach to Asynchronous Multivariate Time Series Anomaly Detection and Localization,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2021.
- [38] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S. K. Ng, “MAD-GAN: Multivariate Anomaly Detection for Time Series Data with Generative Adversarial Networks,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019.
- [39] A. Siffer, P. A. Fouque, A. Termier, and C. Largouet, “Anomaly detection in streams with extreme value theory,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.
- [40] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, 1997.