



Eğitim gerektirmeyen kodlayıcı-tabanlı Transformer ile DNA-alfabeli sözde rastgele sayı üretici

Training-free encoder-based Transformer for a DNA-alphabet pseudo-random number generator

Alev Kaya^{1,2*} , İbrahim Turkoglu²

¹Fen Bilimleri Enstitüsü, Fırat Üniversitesi, Elazığ, Türkiye.

alev.kaya@firat.edu.tr

²Yazılım Mühendisliği Bölümü, Teknoloji Fakültesi, Fırat Üniversitesi, Elazığ, Türkiye.

alev.kaya@firat.edu.tr, iturkoglu@firat.edu.tr

Geliş Tarihi/Received: 23.07.2025

Düzeltilme Tarihi/Revision: 09.12.2025

doi: 10.65206/pajes.03161

Kabul Tarihi/Accepted: 21.01.2026

Araştırma Makalesi/Research Article

Öz

DNA temelli uygulamalarda çoğunlukla bit düzeyinde çalışan klasik sözde rastgele sayı üreteçleri (SRSÜ) kullanılmaktadır. Ancak bu üreteçler DNA uzayına doğrudan çıkış veremediğinden, ek kodlama/adaptasyon katmanları gerektirir. Bu durum hem hesaplama maliyetini hem de anahtar tasarımının karmaşıklığını artırmakta, DNA düzeyindeki rastgelelik özelliklerinin kontrolünü zorlaştırmaktadır. Rastgelelik özelliklerini doğrudan DNA alfabeti üzerinde üretebilen mekanizmalara ihtiyaç duyulmaktadır. Bu çalışmada, eğitim gerektirmeyen ve rastgele başlatılmış ağırlıklara sahip kodlayıcı-tabanlı bir Transformer mimarisi kullanılarak DNA-alfabeli bir SRSÜ önerilmektedir. Model pozisyon bilgisi içermemekte; çıkış katmanında kullanılan düşük genlikli Dirichlet gürültüsü ile çalışmaktadır. Blok içi dengeli örnekleme ve tek seferlik global karıştırma birlikte kullanılarak A/C/G/T sembollerine ait yerel frekans yanlılıkları ve kısa menzilli örüntüler zayıflatılır. Sekiz biyektif DNA→ikili eşlemeden rastgele seçilen biriyle dönüştürülen 500 000 bazlık dizilerden 1 Mbit ikili on bağımsız çıktı üretilmiştir. CPU üzerinde, blok boyutu 100 iken üretim süresi yaklaşık 49 sn. olarak ölçülmüştür. On bağımsız akışın tamamı NIST SP 800-22 $p>0.01$ eşliğini sağlamış, ENT analizinde ≈ 8 bit/byte entropi ve ihmal edilebilir seri korelasyonu gözlenmiştir. Bu bulgular, önerilen DNA-alfabeli SRSÜ'nün istatistiksel rastgelelik açısından tatmin edici bir davranış sergilediğini göstermektedir. Çalışma formel güvenlik ispatı sunmak yerine bu davranışın deneysel olarak değerlendirilmesine odaklanmaktadır.

Anahtar kelimeler: Sözde rastgele sayı üretimi, Yalnızca kodlayıcı Transformer, Eğitim gerektirmeyen model, İstatistiksel rastgelelik testleri.

Abstract

In DNA based applications, classical bit-level pseudo-random number generators (PRNGs) are mostly used. However, since these generators cannot produce outputs directly in the DNA space, additional encoding/adaptation layers are required. This increases both the computational cost and the complexity of key design, and makes it more difficult to control randomness properties at the DNA level. Therefore, mechanisms that can generate randomness properties directly on the DNA alphabet are needed. In this study, a DNA-alphabet pseudo-random number generator is proposed using a training-free, encoder-based Transformer architecture with randomly initialized weights. The model does not incorporate positional information and operates with a low-amplitude Dirichlet noise applied at the output layer. Within-block balanced sampling and a one-time global shuffling are used together to attenuate local frequency biases and short-range patterns in the A/C/G/T symbols. From 500,000-base sequences transformed using one of eight bijective DNA→binary mappings chosen at random, ten independent 1 Mbit binary outputs were generated. On CPU, with a block size of 100, the generation time was measured to be approximately 49 s. All ten independent streams satisfied the $p>0.01$ threshold in the NIST SP 800-22, and in the ENT analysis an entropy of ≈ 8 bits/byte and negligible serial correlation were observed. These findings show that the proposed DNA-alphabet PRNG exhibits satisfactory behavior in terms of statistical randomness. The study focuses on the experimental evaluation of this behavior rather than providing a formal security proof.

Keywords: Pseudo-Random number generation, Encoder-Only transformer, Training-Free generation, Statistical randomness tests.

1 Giriş

Rastgelelik; bilgi güvenliği, simülasyon ve şifreleme sistemlerinin temel bileşenlerinden biridir [1]. Gerçek rastgelelik, tahmin edilebilir örüntülerden ve yapay bağıntılardan arınmış veri üretimini gerektirir [2]. Ancak fiziksel kaynakların sınırlılığı ve ölçüm gürültüsü nedeniyle mutlak rastgelelik nadirdir [2],[3]. Bu nedenle, algoritmik olarak rastgelelik taklidi yapan sözde rastgele sayı üreteçleri (SRSÜ; Pseudo-Random Number Generators, PRNG) yaygın biçimde kullanılır [1]-[3]. Klasik tasnifte, fiziksel olaylara dayanan gerçek rastgele sayı üreteçleri (GRSÜ; True Random Number Generators, TRNG) ile matematiksel algoritmalara dayanan SRSÜ'ler ayrılır [4],[5]. GRSÜ'ler doğadaki

öngörülemez süreçleri kullanırken SRSÜ'ler belirli bir tohumdan (seed) geniş, tekrarlanabilir diziler üretir [3]-[5]. SRSÜ'ler istatistiksel tutarlılık ve deterministik kontrol avantajları sunsa da [1],[2], ikili temsile dayalı modeller bilgi yoğunluğu ve çok-biçimlilik açısından sınırlı kalabilir [6]. Bu sınırlamalar, deoksiribonükleik asit (DNA) gibi moleküler düzeyde veri taşıyan doğal yapılara dayalı alternatif temsil biçimlerini teşvik etmiştir [7]-[10]. Dört alfabeli DNA dizileri (Adenin (A), Timin (T), Guanin (G), Sitozin (C)), her baz için yaklaşık 2 bitlik sembolik temsil kapasitesi sunarak rastgelelik üretiminde entropik ve kombinatorik açıdan zengin bir arama uzayı sağlamaktadır [6]-[10].

*Yazışılan yazar/Corresponding author

Son yıllarda yapay zekâ (YZ) tabanlı yapılar SRSÜ alanında da denenmiştir [1]-[3],[5]-[7]. Üretici çekışmeli ağlar (Generative Adversarial Networks-GAN) [3],[11]-[13] ve genetik algoritmalar (Genetic Algorithms-GA) [6],[7] sıkça kullanılmıştır. Bunun yanı sıra evrişimsel sinir ağları (Convolutional Neural Networks - CNN) [14] ile zaman serisi modelleri [Tekrarlayan sinir ağı (Recurrent Neural Network-RNN), uzun kısa süreli bellek (Long Short-Term Memory-LSTM)] [1],[5],[15] de SRSÜ üretimi için test edilmiştir [16]. Bu derin öğrenme algoritmalarından biri olan Transformer mimarisi, dil dışındaki veri türlerinde de kullanılmaya başlanmıştır [17]-[21]. Ancak yaygın Transformer türevleri büyük veri kümeleri ve uzun eğitim süreçleri gerektirmektedir [18]-[20]. Bu durum yüksek enerji maliyetlerinin yanı sıra, eğitime özgü örüntülerin çıktılarının tekrarlanabilirliğine yansması gibi riskler doğurabilmektedir [22],[23]. Ayrıca DNA temelli uygulamalarda çoğunlukla bit düzeyinde çalışan klasik SRSÜ kullanılmaktadır. Ancak bu üreteçler DNA uzayında doğrudan çıktı veremediği için ek kodlama/adaptasyon katmanları gerekmektedir. Bu da hem hesaplama maliyetini hem de anahtar tasarımının karmaşıklığını artırarak rastgelelik özelliklerinin DNA düzeyinde kontrolünü zorlaştırmaktadır. Bu nedenle, rastgeleliği doğrudan DNA alfabesi üzerinde üretebilen mekanizmalara ihtiyaç duyulmaktadır.

Bu çalışmada, yalnızca kodlayıcı katmanlarından oluşan ve rastgele başlatılmış parametrelerle çalışan bir Transformer mimarisi kullanılır. Bu yapı, doğrudan DNA baz dizileri üreten basit bir üretim hattı olarak önerilmektedir. Çıkış katmanında düşük genlikli Dirichlet gürültüsü uygulanır. Bu gürültü, blok içi dengeli örnekleme ve tek seferlik global karıştırma (shuffle) ile birlikte A/C/G/T sembollerine ait yerel bağımlılıkları ve frekans yanlılıklarını zayıflatır [24]-[32]. İsteğe bağlı son adımda, sekiz bijektif DNA→ikili kuralından birinin uygulanmasıyla bit düzeyinde çıktı elde edilir [33],[34]. Bu çalışma formel bir kriptografik güvenlik ispatı sunmamaktadır. Hedeflerimiz; eğitim gerektirmeyen mimari, DNA-yerel çıktı, istatistiksel çeşitlilik ve CPU-dostu uygulanabilirliktir. Bu hedefler doğrultusunda üretilen DNA-alfabeli on bağımsız akışların istatistiksel rastgelelik özelliklerinin deneysel olarak değerlendirilmesine odaklanmaktadır (tehdit modeli ayrıntıları Bölüm 3.2'de verilmiştir). Ayrıca bu akışların blok şifre modlarında kullanılabilecek başlatma vektörü (initialization vector, IV) ve tek kullanımlık değer (nonce) üretimi için aday istatistiksel bileşenlerin tartışılmasıyla sınırlıdır. Özet katkılar şöyledir:

- Rastgele başlatılmış parametrelere sahip iki katmanlı encoder-only Transformer ile, herhangi bir öğrenme süreci olmadan doğrudan DNA-alfabeli bir SRSÜ mimarisi sunulmuştur,
- Çıkış katmanında kullanılan düşük genlikli Dirichlet gürültüsü, blok içi dengeli örnekleme ve tek seferlik global karıştırma işlemleriyle birlikte A/C/G/T frekans yanlılıklarını ve kısa menzilli örüntüleri zayıflatmak amacıyla kullanılmıştır. Entropi ve korelasyon ölçümleri, baz başına yüksek istatistiksel çeşitliliğe işaret etmektedir,
- Sekiz bijektif DNA→ikili kodlama kuralından rastgele seçilen biriyle aynı DNA akışından farklı bit dizileri üretilmektedir. Böylece bit düzeyinde çeşitlilik ve uygulama esnekliği sağlanmaktadır,
- On bağımsız 1 Mbit'lik çıktı, Ulusal Standartlar ve Teknoloji Enstitüsü Özel Yayını (National Institute of Standards and Technology Special Publication, NIST) [35],[36] test bataryasında $p>0.01$ eşliğini sağlamıştır. Sahte Rastgele Sayı Dizisi Test Programı (The Pseudorandom Number Sequence Test Program-ENT) [37],[38] analizlerinde ≈ 8 bit/byte Shannon entropisi ve ihmal edilebilir seri korelasyon gözlenmiştir.

Makalenin geri kalanı şu şekilde düzenlenmiştir: Bölüm 2 ilgili çalışmaları özetlemekte, Bölüm 3 önerilen mimariyi, DNA→ikili dönüşüm kurallarını ve deneysel değerlendirme protokolünü açıklamakta, Bölüm 4 deneysel bulguları sunmakta ve Bölüm 5 sonuçlar ile gelecekteki çalışmalara yönelik değerlendirmeleri içermektedir.

2 İlgili çalışmalar

Son yıllarda SRSÜ alanında, klasik yöntemlerin veri ve örüntü bağımlılığı, sınırlı çıktı çeşitliliği ve uyarlanabilirlik eksiklikleri nedeniyle yapay zekâ temelli ve biyolojik esinli yeni mimariler öne çıkmıştır. Jeong ve diğ. [1], π sayısının ondalık açılımından türetilen dizileri LSTM mimarisiyle işleyerek yüksek entropili çıktılar üretmiş ve bu dizileri SHA-2 aracılığıyla ikili anahtarlara dönüştürmüştür. Wu ve diğ. [3], 256 bitlik girdilerden 2048 bitlik çıktılar üreten GAN tabanlı bir yapı önermiş; çıktının bir bölümünü geri besleme yoluyla yeniden girişe dâhil ederek çeşitliliği ve içsel öğrenme kapasitesini artırmayı hedeflemiştir. Pasqualini ve Parton [5], Proximal Policy Optimization (PPO) ile eğitilen LSTM tabanlı bir pekiştirmeli öğrenme modeli geliştirmiş ve çıktıyı kısmi gözlemlenebilir Markov karar süreci (POMDP) çerçevesinde zamana bağlı olarak uyarlamıştır. Mukherjee ve diğ. [6], DNA matrislerinden oluşan rastgele popülasyonları genetik algoritma ile evrimleştirerek frekans analiziyle yönlendirilen mutasyon ve çaprazlama işlemleri sonucunda entropik kalitesi artırılmış DNA anahtarları üretmiştir. GAN ve pekiştirmeli öğrenme tabanlı yaklaşımlar da SRSÜ literatüründe önemli bir yer edinmiştir. Wu ve diğ. [11], genetik algoritma ile optimize edilen ve iç durumunu her adımda üretilen çıktılarla güncelleyen GAN tabanlı bir mimari geliştirmiştir. De Bernardi ve diğ. [12], Predictive GAN yaklaşımıyla sabit tohum ve sayac yapısından yararlanarak dışsal dağılıma ihtiyaç duymadan entropik açıdan iyileştirilmiş diziler üretmiş ve öğrenmeli SRSÜ tasarımları açısından erken örneklerden birini sunmuştur. Okada ve diğ. [13], Mersenne Twister çıktıları üzerinde eğitilen, WGAN-GP tabanlı ve dropout içermeyen bir yapı önermiş; cosine tabanlı seed vektörleri ve sigmoid aktivasyon kullanılarak üretilen dizilerin NIST testlerinde başarılı olduğunu göstermiştir. Iqbal ve diğ. [14], önceden eğitilmiş VGG16 ağı ile görüntü tabanlı 512 bitlik dinamik DNA anahtarları üretmiş ve bu anahtarları hiper-kaotik sistemlerle bütünleştirerek DNA kodlamalı şifreleme sürecine dâhil etmiştir. Park ve diğ. [15], LSTM ve CNN bileşenlerini pekiştirmeli öğrenme ile birleştirerek geçmiş örüntülerden yararlanarak dinamik bir rastgelelik üretim modeli geliştirmiştir. Almardey ve diğ. [16] ise eğitim gerektirmeyen, denetimsiz derin pekiştirmeli öğrenme yaklaşımına dayalı bir sistem önermiş ve ajan-çevre etkileşimiyle üretilen 128 bitlik blokların ileri ve geri güvenlik testlerinden geçtiğini raporlamıştır. Bu çalışmalar mimari yapı, çıktı üretim biçimi, veri bağımlılığı ve test kapsamı bakımından önemli çeşitlilik sunmaktadır. Bununla birlikte, mevcut yaklaşımların önemli bir bölümü dış veri, önceden eğitilmiş model çıktısı veya belirli dağılımsal örüntülere dayanmakta; DNA-yerel çıktı üretimi, veri-bağımsız çalışma ve kapsamlı rastgelelik değerlendirmesi açısından sınırlılıklar taşımaktadır. Bu çalışma, eğitim gerektirmeyen ve yalnızca rastgele başlatılmış parametrelerle çalışan Transformer tabanlı bir mimari aracılığıyla DNA-alfabeli, veri-bağımsız ve CPU-dostu bir SRSÜ tasarımına odaklanarak bu boşluklara yanıt vermeyi amaçlamaktadır. Tablo 1, seçili çalışmalara karşılaştırmayı özetlemekte ve önerilen mimarinin literatürdeki konumunu göstermektedir.

Tablo 1. Yapay zekâ ve DNA tabanlı SRSÜ yaklaşımlarının karşılaştırması.
Table 1. Comparison of artificial intelligence and DNA-based PRNG approaches.

Ref.	Anahtar Üretim Yöntemi	Çıktı (format/uzunluk)	Girdi/Veri kaynağı	Güçlü Yönler	Sınırlamalar
[1]	LSTM + SHA-2 post-işleme	1 000 000 bit (statik)	π sayısı + 100 hanelik seed	Tüm NIST testlerinde başarı, yüksek entropi	Veriye bağımlı, tamamen deterministik, sistem ayrıntıları kısıtlı
[3]	GAN (Dense + Conv1D + GELU)	2048 bit (dinamik)	Sentetik 256 bit seed	Kompakt yapı, çeşitlilik için geri besleme	Harici SRSÜ'ye bağımlı, sınırlı güvenlik değerlendirmesi
[5]	PPO tabanlı LSTM (POMDP)	200-1000 bit (değişken)	Rastgele başlatma (farklı N)	Açık kaynak, yeniden üretilebilir, gözleme duyarlı çıktı	LSTM ölçeklenebilirliği sınırlı, donanım bilgisi verilmemiş
[6]	GA ile DNA matrisi optimizasyonu	25×25 DNA matrisi (625 baz)	Dış rastgele DNA jeneratörü	Evrimsel filtreleme, biyolojik uyum vurgusu	Modern test bataryası yok, dış rastgele kaynağa dayalı
[11]	GA-optimize GAN	2048 bit ve 1 048 576 bit	256 bit seed	Yüksek çıktı çeşitliliği, entropik genişleme	Uzun eğitim süresi, güçlü donanım gereksinimi, DNA uyumu kısıtlı
[12]	Predictive GAN + sayaç	128 bit bloklar, toplam 1 Mbit	Sabit seed + sayaç	Dış veri olmadan üretim, yüksek başarı oranı	Yalnızca NIST ile değerlendirme, mimari detaylar sınırlı
[13]	WGAN-GP	≈1 048 576 bit	MT çıktısı + cosine seed	Eğitim sonrası yüksek kalite, ONNX uyumu	MT'ye bağımlı, overfitting riski, sınırlı test çeşitliliği
[14]	CNN (VGG16) + kaotik sistem + DNA kodlama	512 bit (görüntüye özel)	ImageNet + test görselleri	Çok katmanlı güvenlik, yüksek entropi, hızlı işlem	Yoğun eğitim gerektirir, doğrudan DNA-alfabeli üretim sağlamaz
[15]	LSTM + CNN + RL ajan	800 bit (dinamik)	Harici veri seti yok	Geçmiş örüntülerden öğrenme, esnek çıktı	Donanım ve güvenlik analizi sınırlı
[16]	UDRL ajan + entropi rehberli üretim	128 bit bloklar (dinamik)	128 bit seed	Eğitim gerektirmeyen yapı, çoklu test başarısı	Mimari karmaşık, gömülü sistem uyumu zayıf, donanım ölçümü yok
Bu çalışma	Eğitimsiz Transformer encoder (2 katman) + Dirichlet gürültüsü + dengeli örnekleme + global shuffle + 8 DNA→ikili kuralı	500 000 baz (DNA) → 1 000 000 bit (dinamik)	Harici veri gerektirmeyen rastgele 4-sembollü tokenlar	Eğitim gerektirmeyen, DNA-alfabeli ve veri-bağımsız SRSÜ; CPU üzerinde uygulanabilir, yüksek entropili akışlar	Yalnızca istatistiksel testlerle değerlendirilmiştir; formel güvenlik analizi ve uzun diziler için donanım optimizasyonu gelecekteki çalışmalara bırakılmıştır

3 Materyal ve metot

Bu çalışma, dış veri veya ön eğitim gerektirmeyen, encoder-only Transformer ile doğrudan DNA dizisi üreten bir SRSÜ mimarisi sunmaktadır. Şekil 1'de görseli verilen çalışma; rastgele girdi blokları Transformer-softmax, blok bazlı örnekleme, Dirichlet temelli dengeli örnekleme, global karıştırma (shuffle) ve DNA→ikili dönüşüm ve değerlendirme testleri olarak altı adımdan oluşmaktadır.

3.1 Harici veri gerektirmeyen DNA odaklı hafif Transformer tasarımı

Rastgelelik, yalnızca çeşitlilikle değil, aynı zamanda tahmin edilemezlik ile sağlanmalıdır. Geleneksel SRSÜ, içsel bir durumun deterministik olarak güncellenmesiyle çıktı üreten sistemlerdir. Klasik biçimde, gerçek zaman indeksinde SRSÜ aşağıdaki gibi modellenir [4],[5]. Bu süreç Denklem 1 ile tanımlanır.

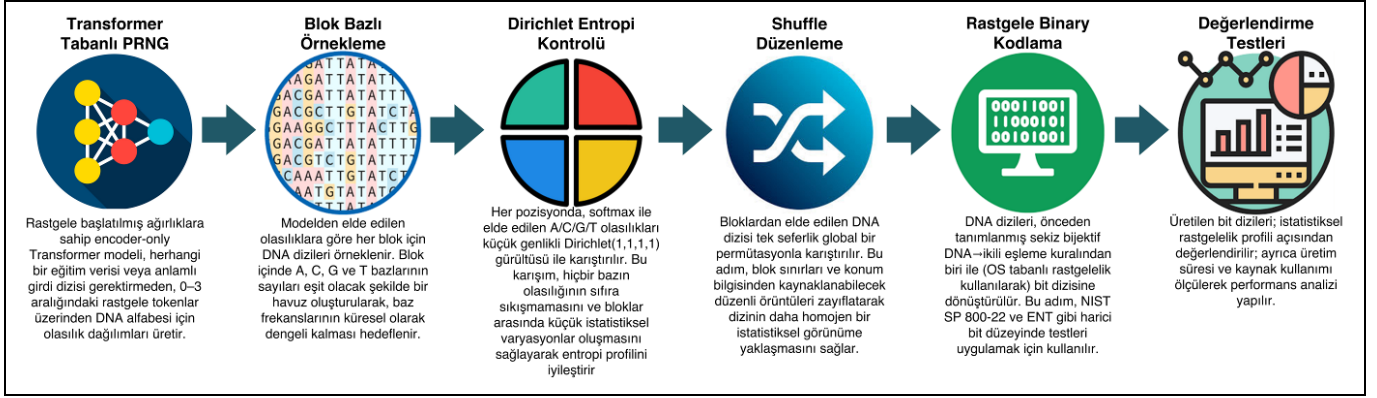
$$\zeta_t = \phi(\zeta_{t-1}), \quad v_t = \psi(\zeta_t) \quad \forall t \geq 1 \quad (1)$$

Burada ζ_t , sistemin içsel durumunu; v_t ise t anındaki çıktıyı temsil etmektedir. Uygun tasarlanmamış üreteçler, kullanılan tohum veya algoritma nedeniyle tahmin edilebilir örüntüler üretebilir. Bu durum hem istatistiksel rastgelelik hem de olası güvenlik uygulamaları açısından istenmeyen bir durumdur

[11],[13]. Kerckhoffs ilkesine göre, algoritma ayrıntıları bilinse dahi üretilen dizilerin öngörülebilir olmaması beklenmektedir [39]. Bu nedenle, ek eğitim veya dış veri gerektirmeden çalışabilen hafif SRSÜ tasarımlarına ihtiyaç vardır.

Bu çalışmada kullanılan model, Vaswani ve ark. [18] tarafından önerilen Transformer mimarisinden türetilmiştir. Hedef dizi tahmini yapılmadığından yalnızca kodlayıcı (encoder) katmanları kullanılmıştır; herhangi bir decoder bloğu yoktur. Önerilen yapı iki encoder katmanından oluşur. Her encoder katmanı; çok başlı kendine-dikkat (Multi-Head Self-Attention, MHSA) bloğu ile ileri beslemeli ağ (Feed-Forward Network, FFN) içerir. Her iki blok arasında kalıntı bağlantılar (residual connections, RC) ile katman normlama (layer normalization, LN) uygulanır.

SRSÜ bağlamında semantik sıra bilgisi gerekmediğinden, konumsal kodlama (positional encoding, PE) devre dışı bırakılmıştır. Her pozisyonda, dört DNA sembolü (A/C/G/T) için softmax tabanlı olasılık dağılımı elde edilir. Çıktı katmanı yalnızca bu dört sınıfa projeksiyon yapan lineer bir katmandan oluşur. Üretim doğrudan encoder'dan alınan softmax çıktıları üzerinde üç aşamalı bir dengeleme mekanizması ile gerçekleştirilir:



Şekil 1. Önerilen mimarinin blok diyagramı.

Figure 1. Block diagram of the proposed architecture.

1. **Dirichlet varyasyonu:** Her pozisyon için softmax sonrası olasılıklar, küçük genlikli Dirichlet(1,1,1,1) karışımı ile birleştirilerek mikro ölekte çeşitlilik artırılır [24]-[26].
2. **Blok içi dengeli örneklemeye:** Her blokta A, C, G ve T bazlarının sayıları eşit olacak şekilde kısıtlanarak, blok başına yaklaşık %25'lik baz frekansı korunur [30]-[32]. Bu adım, DNA-alfabeli çıktılarının küresel baz dağılımını dengelemeyi hedefler.
3. **Küresel karıştırma:** Tüm bloklardan elde edilen DNA dizisi tek seferlik keyless permütasyon ile yeniden sıralanır; böylece blok sınırlarına ve pozisyon bilgisine bağlı yapısal izler zayıflatılır [27]-[29].

Modelin parametreleri rastgele başlatılmakta, dış veri veya eğitim evresi kullanılmamaktadır. Üretim, yalnızca iç rastgele tohumlar ve yukarıdaki üç dengeleme adımı ile gerçekleştirilir.

3.1.1 Girdi gösterimi: Gömme katmanı ve konumsal bilgi yoksunluğu

Transformer tabanlı dizisel modellerde semboller, sabit boyutlu sürekli vektörlere gömülerek işlenir. Her bir sembol (kelime/karakter/DNA bazı) d_v boyutlu bir uzayda nümerik olarak temsil edilir ve gömme vektörüne dönüştürülür [17],[18]. Klasik Transformer'larda dizisel bağıntıları korumak için gömme vektörlerine eklenen konumsal vektörler kullanılır. Bu bileşen, modelin yapısal/zamansal örüntüleri öğrenmesini sağlar. Genellikle başlangıç girdi vektörü Denklem 2 ile ifade edilir.

$$v_0 = E(a) + \mathcal{P}(a) \quad (2)$$

Burada $E(a)$, giriş dizisinin gömme matrisidir. $\mathcal{P}(a)$, konum bilgisini taşıyan vektördür. Konumsal bileşen, sıklıkla sinüsoidal fonksiyonlarla oluşturulur ve sabit, öğrenilmeyen bir temsil kümesi Denklem 3 ve 4'teki gibi sağlanır.

$$\mathcal{P}(\mu, 2k) = \sin\left(\frac{\mu}{\Gamma^{2k/d_v}}\right) \quad (3)$$

$$\mathcal{P}(\mu, 2k+1) = \cos\left(\frac{\mu}{\Gamma^{2k/d_v}}\right) \quad (4)$$

Burada μ dizideki konumu (pozisyonu), k vektör boyutundaki bileşen indeksini, r ise genellikle 10 000 olarak seçilen ölçek

parametresini ifade eder [18]. Bu yapı, konuma bağlı örüntüleri frekans alanında ayırtan vektörler üretmek sıralı bağıntıların korunmasını sağlar.

3.1.2 Önerilen modelde uygulama-yalnızca gömme, PE hariç

Bu çalışmada model yalnızca gömme katmanını kullanır. Her DNA bazı 0-3 aralığındaki bir indekse eşlenir ve sabit boyutlu bir vektöre gömülür (örneğin [0,0.18,-0.23,0.49]). Konumsal kodlama bileşeni $\mathcal{P}(a)$ bilinçli olarak devre dışı bırakılmıştır; bunun başlıca nedenleri şunlardır:

- ✓ SRSÜ bağlamında hedeflenen çıktı, semantik veya zamanla ilişkili bir dizi değil, istatistiksel olarak dengeli DNA baz dizileridir. Bu nedenle sıraya duyarlı bilgiye ihtiyaç yoktur.
- ✓ Blok içi dengeli örneklemeye ve küresel (keyless) permütasyon adımı, konum tabanlı izleri zaten zayıflatmaktadır. Ek bir konumsal kodlama, bu tasarımla çelişebilecek gereksiz bir bağımlılık getirebilir.
- ✓ Yalnızca gömme kullanan bu sadeleştirilmiş yapı, eğitim gerektirmeyen ve harici veriye bağımlı olmayan hafif bir SRSÜ paradigması sunmaktadır. Doğal dil işleme (Natural Language Processing -NLP) için tasarlanmış standart Transformer yapılarına kıyasla hesaplama maliyetini düşürmektedir.

3.1.3 Kodlayıcı katmanları

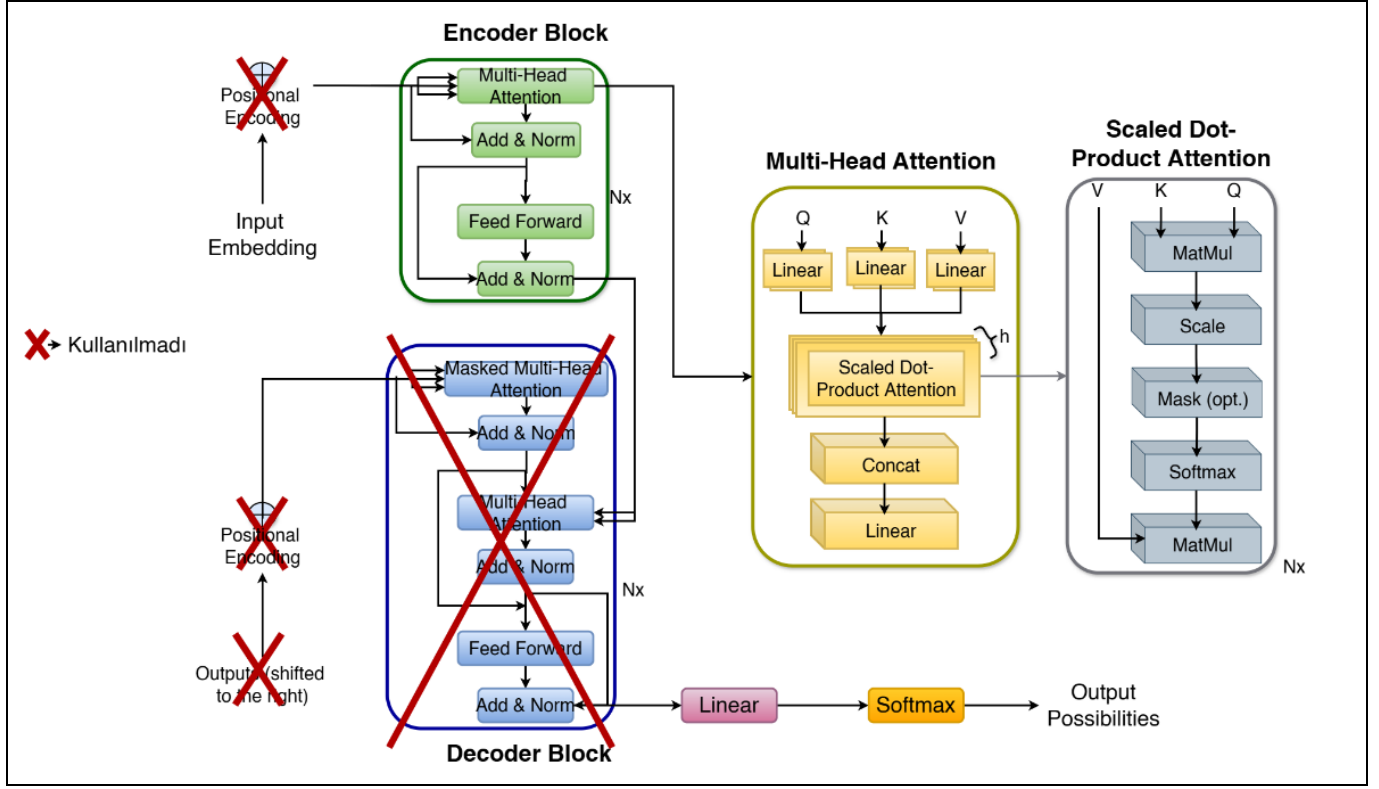
Önerilen model, Şekil 2'de görseli verilen transformer mimarisindeki iki kodlayıcı (encoder) katmanından oluşur.

Her katman, klasik Transformer yapısına benzer biçimde (i) MHSA bloğu ve (ii) FFN bloğu içerir. Her iki bloğun giriş ve çıkışında artık bağlantılar ve katman normlama uygulanır [18].

3.1.4 Çok başlı dikkat mekanizması

Dikkat mekanizması, her girdi konumundaki vektörü diğer konumlardaki içeriklerle etkileşime sokarak, konuma özgü bir temsil üretir. Bunun için giriş matrisi X üç farklı doğrusal dönüşümle Sorgu (Q: Query), Anahtar (K: Key) ve Değer (V: Value) vektörlerine projekte edilir. Bu dönüşüm süreci Denklem 5 verilen işlemlerle tanımlanır.

$$Q = XW^Q, K = XW^K, V = XW^V \quad (5)$$



Şekil 2. Orijinal transformer mimarisinde önerilen SRSÜ tasarımı.

Figure 2. Proposed PRNG design based on the original transformer architecture.

Burada X , gömme katmanından gelen $n \times d_v$ boyutlu giriş matrisidir. d_k , her dikkat başlığı için çıkış alt boyutudur (örneğin $d_k = 16$). W^Q, W^K, W^V her biri $d_v \times d_k$ boyutundadır. Bu çalışmada rastgele başlatılan ve eğitilmeyen ağırlık matrisleridir. Dikkat katsayıları, ölçeklenmiş noktasal çarpım dikkat (scaled dot-product attention) ile Denklem 6 ile hesaplanır:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

Bu ifade, her konumda diğer konumlarla içerik benzerliğine dayalı ağırlıklı bir ortalama üretir. Çok başlı dikkat yapısında, aynı işlem h farklı "başlık" için Denklem 7 ve 8'deki gibi paralel yürütülür.

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (7)$$

$$MHSA(Q, K, V) = Concat(head_1, \dots, head_h)W^O \quad (8)$$

Burada W_i^Q, W_i^K, W_i^V her başlığa özgü projeksiyon matrisleri, W^O ise başlık çıktıların son projeksiyonunu sağlayan matristir. Çok başlı yapı, farklı alt uzaylarda içerik ilişkilerini yakalayarak her pozisyon için daha zengin bir temsil sunmaktadır. Bu çalışmadaki temsil, daha sonra DNA bazlarının olasılık dağılımlarını üretmesinde kullanılmaktadır.

3.1.5 İleri beslemeli ağ

Her encoder katmanında MHSA bloğunu, her konuma bağımsız uygulanan küçük bir ileri beslemeli ağ izler. Bu bileşen,

doğrusal olmayan iki katmanlı bir yapı ile Denklem 9'daki gibi tanımlanır.

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \quad (9)$$

Burada W_1, W_2 ağırlık matrisleri, b_1, b_2 önyargı vektörleri ve $max(0, \cdot)$ ReLU aktivasyonudur. FFN, her pozisyon için aynı parametreleri kullanarak, dikkat bloğundan gelen temsili doğrusal olmayan bir dönüşümden geçirir.

3.1.6 Artık bağlantılar ve katman normalizasyonu

Encoder yapısında, her alt bileşenin (MHSA veya FFN) girişine artık bağlantı eklenir ve sonuç katman normlama ile ölçeklenir ve Denklem 10'da ifade edilir.

$$LayerNorm(x + Sublayer(x)) \quad (10)$$

Bu mekanizma, özellikle derin yapılarda gradyan kararlılığını artırmakta ve aktivasyon dağılımlarının dengede kalmasına yardımcı olmaktadır. Bu çalışmada her encoder katmanında MHSA ve FFN blokları, artık bağlantılar ve LayerNorm bileşenleriyle birlikte standart Transformer biçiminde uygulanmıştır. Fark, tüm ağırlıkların rastgele başlatılıp eğitim adımı uygulanmadan sabit tutulmasıdır.

3.1.7 Önerilen modelde uygulama-encoder mimarisi ve eğitmeden bağımsız dönüşüm

Önerilen SRSÜ, klasik Transformer encoder yapısının rastgelelik üretimine uyarlanmış hafif bir çeşididir. Eğitim süreci tamamen kaldırılmıştır. Encoder katmanı sayısı 6'dan 2'ye, başlık sayısı ise 8'den 4'e düşürülerek hesaplama maliyeti azaltılmıştır. Ağırlıklar yalnızca başlangıçta rastgele atanmakta

ve tüm deneylerde değiştirilmeden kullanılmaktadır. Böylece model, herhangi bir veri kümesine veya öğrenme sürecine ihtiyaç duymamaktadır. İç gömme vektörleri ve yukarıda tanımlanan dikkat/FFN blokları aracılığıyla dinamik DNA dizileri üreten eğitim gerektirmeyen bir SRSÜ çekirdeğine dönüştürülmüştür.

3.1.8 Çıkış katmanı: Doğrusal haritalama ve olasılıksal nükleotid seçimi

Önerilen Transformer yapısında, encoder çıktıları önce doğrusal bir dönüşüme, ardından softmax aktivasyonuna gönderilerek dört DNA bazı için olasılık dağılımı elde edilir. Böylece her konumda nükleotid seçimi olasılıksal olarak Denklem 11 ile ifade edilir.

$$P(y_i|h_i) = \text{softmax}(W_o \cdot h_i + b_o) \quad (11)$$

Burada $h_i \in \mathbb{R}^{d_h}$, encoder katmanından gelen i . pozisyona ait çıktı vektörüdür. $W_o \in \mathbb{R}^{d_h \times C}$, çıkış sınıfları için parametre matrisidir (rastgele başlatılmış; eğitim yapılmamıştır). $b_o \in \mathbb{R}^C$, bias vektörü. $P(y_i|h_i) \in [0,1]^C$, normalize edilmiş, toplamı 1 olan C boyutlu olasılık vektörüdür. Bizim sistemde $C=4$, yani 4 DNA bazı mevcuttur.

3.1.9 Softmax aktivasyonu

Softmax aktivasyonu, her pozisyonda elde edilen skorları olasılık dağılımına dönüştürür. Her sınıf için olasılık değeri Denklem 12 ile hesaplanır.

$$\text{softmax}(s_k) = \frac{e^{s_k}}{\sum_{j=1}^C e^{s_j}} \quad (\text{her bir } k \in \{1, \dots, C\}) \quad (12)$$

Bu fonksiyonun temel özelliği, tüm çıktılar için Denklem 13'te verildiği gibi toplamın 1 olacak şekilde normalize edilmiş bir dağılım üretmesidir.

$$\sum_{k=1}^C \text{softmax}(s_k) = 1 \quad (13)$$

3.1.10 Önerilen sistemde uygulama-DNA baz olasılıklarının kullanımı

Çıkış sınıfı sayısı sabit tutulur ($C=4$); softmax katmanı yalnızca olasılıkları üretir. Blok düzeyinde dengeli baz seçimi (her blokta $A=C=G=T$ koşulu) doğrudan softmax ile değil, bir sonraki adımda tanımlanan **dengeli örnekleme (Balanced Sampling)** yordamı ile sağlanır. Böylece model, önce dört baz için olasılık dağılımını oluşturmaktadır. Ardından Dirichlet karışımı ve blok içi kısıtlarla DNA-alfabeli SRSÜ çıktısını üretmektedir.

3.1.11 Baz çeşitliliğini artıran mikro-makro genişletme yaklaşımı

Encoder çıktısı, her pozisyon için dört DNA bazına ait skor vektörüne dönüştürülür ve softmax ile olasılık dağılımına çevrilir. Böylece $P(A)$, $P(C)$, $P(G)$, $P(T)$ elde edilir. Ancak softmax çıktıları kimi pozisyonlarda keskin/tepe yapılı olabilmektedir. Aynı bazın sık seçilmesi ve blok içinde yerel örüntülerin oluşması mümkündür. Önerilen SRSÜ'de bu durumu azaltmak için iki katmanlı bir çeşitlilik artırma stratejisi uygulanır.

- Dirichlet tabanlı konumsal mikro-çeşitlilik ekleme: Softmax çıktılarındaki deterministik eğilimi yumuşatmak amacıyla, her pozisyonda Dirichlet dağılımından örneklenen rastgele bir vektör ile karışım yapılır [24]-[26].

Orijinal olasılık vektörü p_i ve Dirichlet örneği d_i olmak üzere, karışım Denklem 14 ile tanımlanır.

$$\tilde{p}_i = (1 - \alpha) \cdot p_i + \alpha \cdot d_i \quad (14)$$

Burada $p_i \in [0,1]^4$ softmax sonrası elde edilen DNA bazlarına ait orijinal olasılık vektörüdür. $d_i \sim \text{Dirichlet}(1,1,1,1)$ rastgele vektördür. $\alpha \in [0,1]$ ise karışım katsayısını temsil eder (deneylerde $\alpha=0.01$ alınmıştır). Karışımından sonra vektör yeniden normalize edilerek toplamı 1 olacak şekilde ölçeklenir. Elde edilen $\tilde{p}_i$ vektörü, modelin temel eğilimini korurken her pozisyonda küçük rastgele sapmalar ekleyen, çeşitliliği artırılmış olasılık dağılımıdır.

- Blok düzeyinde dengeli baz dağılımı: Mikro ölçekteki bu çeşitlilik, uzun dizilerde bazı bazların baskın hale gelmesine yol açabilir. Bunu önlemek için, her blokta A, C, G ve T bazlarının sayıları eşit tutulur. Blok boyutu b (4'ün katı) olmak üzere, başlangıçta baz havuzu Denklem 15 ile tanımlanır [30]-[32].

$$\begin{aligned} \text{base}_{\text{pool}} = & \underbrace{\{A, A, \dots, A\}}_{b/4} \cup \underbrace{\{C, C, \dots, C\}}_{b/4} \\ & \cup \underbrace{\{G, G, \dots, G\}}_{b/4} \cup \underbrace{\{T, T, \dots, T\}}_{b/4} \end{aligned} \quad (15)$$

Her pozisyonda, $\tilde{p}_i$ olasılık vektörü yalnızca havuzda kalan bazlar üzerinde yeniden normalize edilir ve seçim bu alt kümeden olasılıksal olarak yapılır. Seçilen baz havuzdan bir adet çıkarılır; böylece blok sonunda her bazdan tam olarak $b/4$ adet kullanılmış olur. Bu yöntem, blok içinde yaklaşık %25'lik baz frekansını garanti etmektedir. Böylece Dirichlet karışımından gelen küçük sapmalar sayesinde bloklar arasında istatistiksel çeşitlilik üretilir.

Bu iki aşamalı strateji, her pozisyonda Dirichlet tabanlı mikro sapmalar ile yerel olasılık dağılımını esnetir. Blok düzeyinde sabit dağılım kısıtıyla küresel baz dengesini korumaktadır. Klasik yalnızca-softmax tabanlı SRSÜ tasarımlarına kıyasla, önerilen yaklaşım DNA-alfabeli çıktıların hem blok içinde dengeli, hem de bloklar arasında daha çeşitli bir istatistiksel profile sahip olmasını hedeflemektedir.

3.1.12 Bloklararası karıştırma ve DNA→ikili kodlama kuralları

Bloklararası karıştırma (global shuffle): Önceki adımlarda her blok için $A=C=G=T$ olacak şekilde dengeli DNA dizileri üretilmektedir. Ancak blokların ardışık olarak birleştirilmesi, blok sınırlarına ilişkin yapısal izler bırakabilir. Örneğin, blok-1'de T bazlarıyla başlayan dizilim, blok-2'de de benzer şekilde başlayabilir. Bu izi zayıflatmak için, tüm bloklar birleştirildikten sonra elde edilen DNA dizisine tek seferlik, anahtardan bağımsız bir permütasyon uygulanır. Blokların rastgele sıralanması, literatürde "shuffle regularization" olarak tanımlanır [27]-[29] ve Denklem 16'daki gibi karıştırılır.

$$\text{DNA}_{\text{key}} = \text{shuffle}([B_1 \parallel B_2 \dots \parallel B_n]) \quad (16)$$

Burada B_i , i . bloğun dengelenmiş DNA dizisidir. $\parallel$, blokların ardışık birleştirilmesini gösterir. $\text{shuffle}(\cdot)$, işlemi, tüm pozisyonların yerlerini eş-olasılıklı bir permütasyonla yeniden sıralar. Böylece blok sıralarına ait konumsal ipuçları zayıflatılır. Bloklar arası baz oranı korunurken dizinin genel yapısı daha homojen bir profile taşınır. Bu adım, ek bir eğitim veya parametre ayarı gerektirmeyen basit bir karıştırma katmanıdır.

DNA→ikili kodlama kuralları: İstatistiksel testlerin çoğu ikili diziler üzerinde çalıştığından, DNA-alfabeli çıktıyı bit düzeyine dönüştürmek için isteğe bağlı bir kodlama adımı kullanılır. Bu amaçla, dört DNA bazını {00,01,10,11} ikili desenlerine eşleyen sekiz bijektif kural Tablo 2'deki gibi kodlama kurallarına göre tanımlanmıştır.

Tablo 2. DNA→ikili kodlama kuralları.

Table 2. DNA→binary code rules.

Baz	1	2	3	4	5	6	7	8
A	00	00	11	11	10	10	01	01
T	11	11	00	00	01	01	10	10
G	10	01	10	01	00	11	00	11
C	01	10	01	10	11	00	11	00

Her kural A, C, G ve T bazlarını bu dört iki-bitlik desenin bir permütasyonuna eşler. Dengeli blok yapısından dolayı, her blokta A, C, G ve T bazları yaklaşık eşit sıklıkta yer almaktadır. Bu durumda, hangi kural seçilirse seçilsin dört ikili desen de yaklaşık ¼ olasılıkla görülür. Her desen sırasıyla 0, 1, 1 ve 2 adet '1' içerdiğinden, iki bit için beklenen '1' sayısı $E[\#1(2 \text{ bit})] = (0 + 1 + 1 + 2)/4 = 1$ olur ve bit başına beklenen '1' oran $E[\#1/bit] = 0.5$ 'e yakınsamak üzere tasarlanır. Böylece, dengeli DNA dizileri ikili uzaya aktarılırken de yaklaşık 0.5'lik bir '1' yoğunluğu korunur. Bu adım, bit düzeyinde ek bir çeşitlilik sağlar ancak başlı başına bir güvenlik garantisi sunmamaktadır.

Tüm bu sürecin sözde kodu Algoritma 1 ve Alt yordam 1 başlı başına altında takiben verilmektedir.

Algoritma 1: Encoder Transformer Tabanlı DNA Alfabeli SRSÜ

Girdi: toplam_baz, blok_boyutu, rastgelelik_gücü ($0 \leq \lambda \leq 1$)
Çıktı: DNA_SRSÜ (toplam_baz uzunluğunda DNA dizisi)

1. Eğer toplam_baz, blok_boyutu'nun tam katı değilse HATA ver.
2. Eğer blok_boyutu % 4 ≠ 0 ise HATA ver.
3. DNA_anahtarı ← ∅
4. TransformerModel ← Gömme Katmanı → TransformerEncoder → Lineer Çıkış Katmanı olarak tanımla
5. blok_sayısı ← toplam_baz / blok_boyutu
6. $i = 1$ 'den blok_sayısı'na kadar:
 5. girdi_verisi ← 0.3 aralığından blok_boyutu uzunluğunda rastgele indeks dizisi oluştur (0:A,1:C,2:G,3:T)
 6. tahmin_olasılıkları ← TransformerModel(girdi_verisi)
 7. olasılık_bloğu ← softmax(tahmin_olasılıkları)
 8. örneklenen_blok ← Dengeli Örnekleme(olasılık_bloğu, rastgelelik_gücü)
 9. örneklenen_blok'u DNA_anahtarı'na ekle
10. döngü sonu
11. DNA_anahtarı ← dizideki tüm bazlara tek seferlik rastgele bir permütasyon uygula (global karıştırma)
12. DNA_anahtarı'nı döndür

İsteğe bağlı adım: DNA anahtarını ikili forma dönüştürme (NIST veya entropi testleri için)

13. kodlama_kuralı ← {1,...,8} kümesinden eş-olasılıklı rastgele seç
14. ikili_anahtar ← DNA_anahtarını seçilen kodlama_kuralına göre deterministik biçimde bit dizisine dönüştür.

Alt Yordam 1: Dengeli Örnekleme (BalancedSampling)

Fonksiyon: Dengeli Örnekleme(olasılık_bloğu, rastgelelik_gücü)

Ön-koşul: blok_boyutu % 4 = 0

Girdi: olasılık_bloğu (blok_boyutu×4), rastgelelik_gücü (λ)

Çıktı: örneklenmiş_bazlar (blok_boyutu uzunluğunda DNA baz dizisi)

1. $b \leftarrow \text{blok_boyutu}$
2. $\text{baz_havuzu} \leftarrow ['A'] \times (b/4) + ['C'] \times (b/4) + ['G'] \times (b/4) + ['T'] \times (b/4)$; baz_havuzu'nu karıştır
3. $\text{orneklenmis_bazlar} \leftarrow \emptyset$

4. $i = 1$ 'den b 'ye kadar:

5. $\text{dirichlet_gürültüsü} \leftarrow \text{Dirichlet}([1,1,1,1])$

6. $\text{karisik_olasilik} \leftarrow (1 - \lambda) \times \text{olasilik_bloğu}[i] + \lambda \times$

$\text{dirichlet_gürültüsü}$

7. karisik_olasilik 'i normalize et (toplamı 1 olacak şekilde)

8. $\text{gecerli_bazlar} \leftarrow \text{baz_havuzunda kalan bazlar}$

9. $\text{gecerli_olasiliklar} \leftarrow \text{gecerli_bazlara karşılık gelen}$

$\text{karisik_olasilik değerleri}$; yeniden normalize et

10. $\text{secili_baz} \leftarrow \text{gecerli_bazlar arasından gecerli_olasiliklar ile} \text{örnekle}$

11. $\text{secili_baz}'ı \text{orneklenmis_bazlar'a ekle ve baz_havuzu'ndan bir kopyasını çıkar}$

12. döngü sonu

13. $\text{orneklenmis_bazlar}'ı \text{döndür}$

Bu alt yordam, her blokta A/C/G/T bazlarının küresel olarak dengeli kalmasını sağlamaktadır, Dirichlet temelli karışım sayesinde bloklar arasında küçük rastgele farklılıklar üretmektedir. Bölüm 4'te sunulan deneysel sonuçlar, bu yapıyla üretilen dizilerin NIST SP 800-22 ve ENT testlerinde istatistiksel açıdan tatmin edici özellikler sergilediğini göstermektedir.

3.2 Tehdit modeli

Bu çalışma, literatürdeki birçok SRSÜ tasarımında olduğu gibi [1], [3], [5], [11]-[16], üretilen dizilerin istatistiksel rastgelelik özelliklerini değerlendirmeyi amaçlamaktadır. Formel kriptografik güvenlik ispatları bu çalışmanın kapsamı dışındadır. Değerlendirme, Kerckhoffs ilkesi [39] altında pasif bir gözlemci varsayımıyla yapılmıştır. Bu varsayımda algoritma, mimari, blok boyutu, Dirichlet katsayısı ve DNA→ikili kodlama kurallarının yapısı saldırgan tarafından bilinmektedir. Ayrıca saldırganın üretilen akışları uzun süre gözlemleyebildiği ve bu akışlar üzerinde istatistiksel testler çalıştırabildiği kabul edilmiştir. Buna karşılık, işletim sistemi kaynaklı rastgelelik, Python/PyTorch RNG iç durumları ve Dirichlet örnekleme aşamasındaki anlık seçimler saldırgan tarafından bilinmeyen üretim durumu olarak değerlendirilmiştir. Her akışın bu üretim durumu altında bağımsız ve benzersiz biçimde üretildiği varsayılmıştır. Keystream tekrarının veya belirli bir şifreleme protokolüyle birlikte kullanımın doğurabileceği etkiler bu çalışmada ayrıca analiz edilmemiştir. Bu tehdit modeli altında değerlendirme üç istatistiksel boyutla sınırlandırılmıştır:

- i. NIST SP 800-22 test bataryası ile ikili akışların başarımlı durumu,
- ii. ENT aracı ile Shannon entropisi ve seri korelasyon,
- iii. DNA düzeyinde baz frekansları, χ^2 uygunluk testi ve blok başına A=C=G=T dengesinin korunması.

Elde edilen bulgular, önerilen SRSÜ'nün bu testler kapsamında tatmin edici bir istatistiksel rastgelelik profili sergileyip sergilemediğini göstermektedir. Bununla birlikte, kriptografik anlamda güvenli SRSÜ (cryptographically secure PRNG, CSPRNG) tanımlarına yönelik indirgenebilirlik temelli ispatlar, ileri/geri güvenlik, durum sızıntısı, yan kanal saldırıları ve belirli şifreleme modlarıyla birleştiğinde ortaya çıkabilecek protokol düzeyi güvenlik özellikleri bu çalışmanın kapsamı dışındadır. Bu nedenle, üretilen diziler için formel kriptografik güvenlik garantisi iddia edilmemektedir. Uygulama senaryolarında bu dizilerin yalnızca benzersiz IV/nonce üretimi için aday istatistiksel bileşenler olarak değerlendirilmesi ve gerektiğinde uygun bir çıkarıcı ya da anahtar türetme fonksiyonu ile güçlendirilmesi önerilmektedir.

3.3 Değerlendirme testleri

Önerilen DNA-alfabeli SRSÜ'nün çıktıları, hem DNA düzeyinde hem de ikili biçimde çeşitli istatistiksel testlere tabi tutulmuştur. Amaç, 3.2'deki bölümde tanımlanan tehdit modeli altında, üretilen dizilerin rastgelelik ve denge özelliklerini nicel olarak gözlemlemektir. Tüm deneyler, 64-bit Windows 11 Pro işletim sistemine sahip, Intel Core i7-10750H @ 2.60 GHz ve 16 GB RAM içeren bir sistem üzerinde, Python 3.12.3 kullanılarak yalnızca CPU ortamında gerçekleştirilmiştir.

3.3.1 İstatistiksel testler

Bu alt bölümde, üretilen DNA dizilerinin baz dağılımı ve entropisi incelenmiştir. On bağımsız çalışmadan elde edilen metrikler, birinci dereceden istatistiksel denge açısından şu gözlemlere işaret etmektedir.

- ✓ Baz frekansları blok düzeyinde A=C=G=T kısıtı nedeniyle %25 çevresinde yoğunlaşmaktadır. Toplam 500 000 bazlık dizide her baz için sayımlar birbirine çok yakın değerler vermiştir,
- ✓ DNA düzeyinde Shannon entropisi $H_z \approx 2.0000$ bit/baz olarak ölçülmüştür. Bu değer, dört sembolü ideal eş-olası dağılımın teorik sınırına karşılık gelen 2 bit/baz ile uyumludur,
- ✓ χ^2 testi, gözlenen baz sayımlarının eşit beklenen değerlerden anlamlı bir sapma göstermediğini ($p > 0.01$) ortaya koymuştur.

Bu sonuçlar, tasarlanan Dirichlet karışımı ve blok-içi dengeleme mekanizmasının, hedeflenen homojen baz dağılımını gerçekleştirdiğini göstermektedir. Bununla birlikte, χ^2 p-değerlerinin 1.000'e yakın olması "daha rastgele" anlamına gelmemektedir. Zorlamalı dengeleme nedeniyle birinci mertebe testler uç değerlere kayabilmektedir. Bu nedenle bulgular, önerilen SRSÜ'nün DNA-uyumlu üretim için istatistiksel açıdan yeterli ve homojen bir profil sunduğu şeklinde yorumlanmaktadır.

3.3.2 Rastgelelik testleri

3.3.2.1 NIST SP 800-22 test sonuçları

Her biri 500 000 baz uzunluğunda DNA dizileri, sekiz biyektif DNA→ikili kodlama kuralından rastgele seçilen biriyle dönüştürülerek on bağımsız 1 Mbit'lik akış elde edilmiştir (blok boyutu $b=100$). Bu akışların her biri için NIST SP 800-22'nin on beş standart testi uygulanmıştır [35],[36]. Şekil 1' de verilen tüm testlerde p-değerlerinin 0.01 eşliğini aştığı görülmüştür. Tablo 3, on akış için p-değerlerinin ortalamasını, standart sapmasını ve min-maks aralığını özetlemektedir (ayrıntılı sonuçlar Ek Tablo A1'de verilmiştir).

Monobit testlerinde p-değerlerinin 1.000'e çok yakın çıkması, DNA düzeyinde A=C=G=T dengesinin ve sekiz kodlama kuralı altında ≈ 50 '1' oranının bir sonucudur. Bu durum, Bölüm 3.3.1'de tartışılan zorlamalı denge ile tutarlıdır ve tek başına "daha yüksek güvenlik" anlamına gelmemektedir. NIST bataryasından elde edilen bulgular, yalnızca seçilen test kümesi altında önerilen SRSÜ'nün kabul edilebilir bir istatistiksel rastgelelik profili sergilediğini göstermektedir. CSPRNG düzeyinde bir güvenlik garantisi sunmamaktadır (Bakınız bölüm 3.2).

3.3.2.2 ENT test sonuçları

Önerilen SRSÜ'nün ikili çıktıları, dizisel rastgelelik ve bilgi yoğunluğu açısından ENT [37],[38] testleriyle

değerlendirilmiştir. Rastgele seçilen sekiz DNA→ikili kuralından biriyle dönüştürülen ve her biri 500 000 bazdan üretilen 1 000 000 bitlik on akış için elde edilen ortalama sonuçlar Tablo 4'te özetlenmektedir.

Tablo 3. On bağımsız akışın ortalama NIST SP 800-22 test sonuçları.

Table 3. The average NIST SP 800-22 test results of ten independent streams.

TEST ADI	Ortalama p (n=10)	Standart sapma	Min-Maks
Frequency (Monobit)	1.000	0.000	1.000-1.000
Block Frequency	0.546	0.317	0.016-0.991
Runs	0.520	0.179	0.211-0.908
Longest-Run	0.452	0.205	0.147-0.805
Binary Matrix Rank	0.427	0.229	0.075-0.894
Discrete Fourier Transform	0.542	0.322	0.142-0.985
Non-Overlapping Templates	0.529	0.262	0.068-0.819
Overlapping Templates	0.316	0.258	0.012-0.867
Maurer's Universal Statistical	0.519	0.224	0.079-0.835
Linear-Complexity	0.617	0.234	0.116-0.918
Serial (2. p)	0.564	0.248	0.140-0.999
Approximate Entropy	0.603	0.307	0.085-1.000
Cumulative-Sums	0.630	0.259	0.318-0.990
Random Excursions	0.577	0.223	0.283-0.988
Random Excursions Variant	0.580	0.283	0.074-0.909

Tablo 4. On bağımsız akışın ENT test sonuçları.

Table 4. The ENT test results of ten independent streams.

Test Türü	Ölçülen	Beklenen Aralık
Shannon Entropi	7.99	≈ 8.0 (maksimum)
Chi-Square	278.78	NIST uyumlu aralıkta
Aritmetik Ortalama	127.33	≈ 127.5
Monte Carlo π Tahmini	3.146	≈ 3.1416
Seri Korelasyon	0.000008	≈ 0 (ideal)

- **Shannon entropisi** ≈ 7.99 bit/byte olup, 8.0 maksimum değerine oldukça yakındır; bu durum bitlerin yaklaşık eş-olasılıklı dağıldığını göstermektedir,
- Chi-square metriği, ENT'nin 1 000 000 bitlik diziler için verdiği kabul edilebilir aralık içinde kalmış ve NIST uyumlu bir davranış raporlanmıştır,
- Aritmetik ortalama 127.33 olup, ideal 127.5 değerine çok yakındır,
- Monte Carlo π tahmini 3.146 civarında olup, gerçek π değeriyle uyumlu bir yaklaşık sonuç vermektedir,
- Seri korelasyon katsayısı ≈ 0.000008 bulunmuş, bu da bitler arasında kısa menzilli doğrusal bağımlılığın ihmal edilebilir düzeyde olduğuna işaret etmiştir.

Bu göstergeler, NIST SP 800-22 bulgularını tamamlar niteliktedir. Önerilen SRSÜ'nün hem bilgi yoğunluğu hem de basit bağımlılık ölçütleri açısından tatmin edici bir istatistiksel rastgelelik profili ürettiğini göstermektedir.

3.3.3 Karmaşıklık analizi

Kriptografik anahtar üretiminde rastgelelik kadar süre ve bellek tüketimi de önemlidir. Önerilen model, eğitim gerektirmeyen iki katmanlı hafif bir Encoder ile doğrudan DNA baz dizileri üretir. Bu alt bölümde, blok boyutu b ve toplam uzunluk k için zaman ve alan karmaşıklığını özetlenmektedir. Bu analiz, d_{model}, l ve h sabit kabul edilerek yapılmaktadır (çalışmada $d_{model} = 64$, $l = 2$ ve $h = 4$ kullanılmıştır).

3.3.3.1 Zaman karmaşıklığı

Tek bir blokta zaman maliyetini belirleyen ana bileşen self-attention'dur. Her Transformer Encoder katmanı için: MHSA karmaşıklığı $O(b^2 \times d)$ ve FFN yapının karmaşıklığı $O(b \times d \times d_{ff})$ mertebesindedir. Gömme, softmax, Dirichlet karışımı ve balanced_sampling adımları $O(b)$, global permütasyon (shuffle) ise tüm dizi üzerinde tek seferlik $O(k)$ işlemidir. Toplamda, $B = k/b$ blok için algoritmanın zaman karmaşıklığı Denklem 17'de verilmiştir.

$$O(B \times ((b^2 \times d) + (b \times d \times d_{ff}))) \approx O(k \times b \times d) \quad (17)$$

Burada blok boyutu b sabit seçildiğinde toplam uzunluk k ile yaklaşık doğrusal ölçeklenir. Çalışmada kullanılan ayarlar altında CPU üzerinde on bağımsız akışın 1 Mbit üretim süresinin ortalama ≈ 49 s olması, bu analizle uyumludur.

3.3.3.2 Alan karmaşıklığı

Çalışma belleği, her seferinde yalnızca tek bir blok üzerinde hesaplama yapacak şekilde sınırlandırılmıştır. Bir blok için attention katsayılarını tutan matrisler $O(h \times b^2)$ ve gizli temsil vektörleri $O(b \times d)$ ölçeğindedir. Model parametreleri $O((d^2) + (d \times d_{ff}))$ mertebesinde olup, giriş uzunluğundan bağımsız sabit bir maliyet oluşturur. Bu nedenle, blok boyutu b sabit tutulduğunda çalışma belleği blok sayısından bağımsız kalmakta ve pratikte gömülü sistemler dâhil pek çok uygulama için uygun bir bellek profili sunmaktadır.

3.3.4 Performans analizi

Önerilen eğitim gerektirmeyen Transformer-tabanlı DNA-SRSÜ'nün verimliliği, 500 000 baz (~ 1 Mbit) üretimi üzerinden ölçülmüştür. Tümüyle CPU üzerinde yapılan bağımsız deneylerde akış üretimi yaklaşık 49.38 s sürmüştür. Bu değer yaklaşık 10.1 baz/ms (~ 20.2 kb/s) hızına karşılık gelmektedir. Aynı koşullarda toplam CPU süresi yaklaşık 295.28 s, RSS bellek artışı ise yaklaşık 11.33 MB olarak kaydedilmiştir. Bu bulgular, Bölüm 3.3.3'te verilen karmaşıklık analiziyle uyumlu bir ölçeklenme sergilemektedir. Seçilen $b = 100$ blok boyutu ($B = 5000$ blok), self-attention'ın $O(b^2)$ zaman/bellek maliyeti ile örnekleme katmanının $O(b)$ ek yükü arasında pratik bir denge sunmaktadır. Daha büyük b , attention matrislerinin karesel maliyetini artırırken; daha küçük b , blok sayısını ve fonksiyon çağırısı sayısını yükseltmektedir. Deneysel profile hem 0-1 bit oranı ≈ 50 düzeyinde kalmış hem de NIST $p > 0.01$ eşiği korunmuştur. Dolayısıyla bu hız/bellek ayarı gözle görülür bir istatistiksel kalite kaybına yol açmamaktadır. Uygulamada $b \approx 64 - 128$ aralığı, hedef cihazın bellek bütçesi ve gecikme gereksinimlerine göre ayarlanabilir. Baskın terim $O(h \times b^2)$ olduğundan daha büyük b değerleri daha yüksek bellek gerektirir. Bu performans profili özellikle üç kullanım senaryosunu desteklemektedir:

- (i) Genomik verilerin şifrelenmesinde FASTA/FASTQ gibi büyük dosyalar için çevrimdışı DNA keystream üretimi,

- (ii) Orta-düşük bant genişliğine sahip ses/sensör/telemetri akışları için ~ 20 kb/s mertebesinde gerçek zamanlı anahtar besleme,
- (iii) Sınırlı kaynaklı gömülü cihazlarda hafif anahtar üretimi ve periyodik anahtar yenileme.

Pratik uygulamalarda IV/nonce değerlerinin tekrar kullanılmaması, üretilen dizilerin belirli aralıklarla istatistiksel olarak yeniden doğrulanması ve gerekirse ek bir çıkarıcı/KDF ile güçlendirilmesi önerilir.

3.4 Deneysel çalışma ve genel değerlendirme

Bu çalışma kapsamında, eğitim gerektirmeyen ve yalnızca encoder katmanlarından oluşan Transformer tabanlı DNA-alfabeli bir SRSÜ çekirdeği, Bölüm 3.2'de tanımlanan tehdit modeli altında istatistiksel olarak değerlendirilmiştir. Elde edilen on bağımsız 1 Mbit'lik akış için NIST SP 800-22 test bataryasındaki tüm testler $p > 0.01$ eşiğini sağlamış, ENT ve DNA düzeyi metrikleri (entropi, χ^2 , baz frekansları) ideal değerlere yakın sonuçlar vermiştir. 1 Mbit üretimin yalnızca CPU üzerinde ≈ 49 s sürede tamamlanması, iki katmanlı encoder-only mimarinin ek eğitim maliyeti olmadan pratik bir üretim hızına ulaşabildiğini göstermektedir. Bu bulgular, eğitimsiz yapı, DNA-yerel üretim, istatistiksel çeşitlilik, CPU-dostu uygulanabilirlik hedeflerinin her birinin en azından temel düzeyde karşılandığına işaret etmektedir. Tablo 5, seçili güncel SRSÜ yaklaşımlarını rastgelelik testleri, karmaşıklık ve bildirilen çalışma koşulları açısından niteliksel olarak karşılaştırmaktadır.

Literatürdeki birçok çalışma, kısa anahtar uzunlukları veya sınırlı test setleriyle NIST değerlendirmesi raporlamaktadır. Çoğu GPU üzerinde, eğitim gerektiren ağır modeller kullanmakta ve kuramsal maliyet ile donanım bilgisi her zaman açıkça belirtilmemektedir. Önerilen yöntem ise, dış veri gerektirmeyen, CPU üzerinde çalışabilen ve doğrudan DNA alfabeli çıktılar üreten hafif bir alternatif olarak konumlanmaktadır. Bununla birlikte, bu çekirdeğin çıktısı üretilen dizilerin, özellikle IV/nonce üretimi veya üst katmanda kullanılacak DNA temelli süreçlerde girdi olacak istatistiksel anahtar malzemesi olarak düşünülmelidir. GC oranı, homopolimer uzunlukları gibi biyofiziksel kısıtların eklenmesi ve CSPRNG düzeyinde kriptografik analizler, bu çalışmanın kapsamı dışında bırakılmış olup gelecekteki çalışmalar için açık araştırma konuları olarak görülmektedir.

4 Sonuç

Bu makalede, eğitim gerektirmeyen ve yalnızca kodlayıcı katmanlarından oluşan bir Transformer mimarisi kullanılarak DNA-alfabeli bir sözde rastgele sayı üretici önerilmiştir. Önerilen yapı, pozisyon bilgisinden bağımsız çalışmaktadır. Çıkış katmanında düşük genlikli Dirichlet karışımı, blok içi dengeli örnekleme ve tek seferlik global karıştırma adımlarıyla A/C/G/T bazları arasında yerel frekans yanhlıklarını zayıflatmaktadır. Rastgele seçilen sekiz DNA $\rightarrow$ ikili eşleme kuralı üzerinden dönüştürülen on bağımsız 1 Mbit'lik akışın, NIST ve ENT testleri ile DNA düzeyindeki χ^2 ve baz oranı ölçütleri altında tatmin edici bir istatistiksel rastgelelik profili sergilediği gösterilmiştir. CPU üzerinde bu on akışın ortalama üretim süresi yaklaşık 49 saniyedir. Bellek gereksinimi, yöntemin ek eğitim maliyeti olmadan pratik biçimde uygulanabileceğini ortaya koymaktadır. Çalışmanın kapsamı, tanımlanan pasif tehdit modeli altında yalnızca istatistiksel rastgelelik değerlendirmesiyle sınırlıdır.

Tablo 5. Güncel SRSÜ yaklaşımlarının karşılaştırmalı analizi.

Table 5. Comparative analysis of recent PRNG approaches.

Ref.	Anahtar boyutu	Rastgelelik testleri	Kuramsal karmaşıklık (zaman / alan)	Bildirilen performans (süre, CPU, RAM)	Yazılım / donanım bilgisi
[1]	1 000 000 bit	NIST, p-değeri, ortalama, varyans	Zaman: $O(n \times b \times h + n)$ n: iterasyon, b: buffer boyutu, h: gizli katman; s: adım sayısı	Raporlanmamış	Raporlanmamış
[6]	625 baz (25×25 DNA)	Frekans, gap, fitness, anahtar uzayı	Zaman: $O(m \times g \times n)$, Alan: $O(2 \times m \times n)$, g: nesil (evrim) sayısı, n: popülasyon boyutu, m: birey (anahtar) uzunluğu	Raporlanmamış	Raporlanmamış
[11]	2048 bit	NIST, GM/T, Hamming	Raporlanmamış	276 ms, 0.81 GB/s	Python, PaddlePaddle, GPU (Tesla V100, 32 GB RAM)
[12]	128 bit ve 1 Mbit test	NIST (188 test, 10 tekrar)	Raporlanmamış	Raporlanmamış	Raporlanmamış
[14]	256 bit	NIST	Raporlanmamış	0.015 s	Python, TF/Keras, Win10, Intel i5- 7200U, 8 GB RAM
[16]	128 bit, toplam 1 MB	NIST, PNB, Hamming mesafesi	Raporlanmamış	Raporlanmamış	Raporlanmamış
Önerilen	On bağımsız akışlı 1 Mbit (500 000 baz)	NIST, ENT	Zaman: $O(k)$, Alan: $O(h \times b^2)$, k: dizi uzunluğu, h: başlık sayısı(h=4)	≈ 49.38 s, ≈ 10.1 baz/ms, CPU süresi 295.3 s, RAM artışı ≈ 11.3 MB	Python 3.12, Win11, Intel i7- 10750H, 16 GB RAM

Önerilen çekirdek, bu çerçevede; eğitimsiz mimari, DNA-yerel çıktı, istatistiksel çeşitlilik ve CPU-dostu uygulanabilirlik hedeflerini temel düzeyde karşılayan bir SRSÜ prototipi sunmaktadır. Ancak kriptografik anlamda güvenli SRSÜ (CSPRNG) tanımlarına indirgenebilirlik, ileri/geri güvenlik, yan kanal dayanımı ve belirli şifreleme modlarıyla birleştiğinde ortaya çıkan protokol düzeyi güvenlik özellikleri bu çalışmanın kapsamı dışındadır. Bu nedenle, üretilen dizilerin özellikle IV/nonce üretimi veya DNA temelli süreçlerde girdi olabilecek DNA-yerel istatistiksel anahtar malzemesi olarak değerlendirilmesi daha uygundur. Gelecek çalışmalarda, biyofiziksel gereksinimleri modeli açıkça entegre etmeyi ve farklı blok boyutları ile başlık sayıları için ölçeklenebilirlik ve performans analizlerini genişletmeyi planlıyoruz. Ayrıca, CSPRNG düzeyinde kuramsal ve deneysel güvenlik analizini mümkün kılmak için çıkarıcı/KDF katmanlarıyla bileşik yapıları inceleyecek ve önerilen SRSÜ'yü uçtan uca DNA tabanlı sistemler üzerinde test edeceğiz.

5 Conclusions

In this paper, a DNA-alphabet pseudo-random number generator is proposed based on a Transformer architecture that does not require training and consists solely of encoder layers. The proposed structure operates independently of positional information. By applying a low-amplitude Dirichlet mixture at the output layer together with within-block balanced sampling and a single global shuffling step, local frequency biases among the A/C/G/T bases are attenuated. It has been shown that ten independent 1 Mbit streams, obtained through one of eight randomly selected bijective DNA→binary mapping rules, exhibit a satisfactory statistical randomness profile under NIST and ENT tests as well as χ^2 and base-ratio metrics at the DNA level. On CPU, the average generation time for these ten streams is approximately 49 seconds. The memory requirements also indicate that the method can be deployed in practice without any additional training cost. The scope of this study is limited to the assessment of statistical randomness under the defined passive threat model. Within this framework,

the proposed core provides a PRNG prototype that, at a basic level, meets the goals of a training-free architecture, DNA-native output, statistical diversity, and CPU-friendly deployability. However, reducibility to cryptographically secure PRNG (CSPRNG) notions, forward/backward security, resistance to side-channel attacks, and protocol-level security properties arising when combined with specific encryption modes are all outside the scope of this work. Therefore, it is more appropriate to regard the generated sequences as DNA-native statistical key material, in particular for IV/nonce generation or as input to DNA-based processes. In future work, we plan to explicitly integrate biophysical requirements into the model and to extend scalability and performance analyses for different block sizes and numbers of heads. We will also investigate composite constructions with extractor/KDF layers to enable theoretical and experimental security analysis at the CSPRNG level and test the proposed PRNG in end-to-end DNA-based systems.

6 Teşekkür

Bu çalışma, birinci yazarın “Genomik Şifreleme Yöntemlerinin Performansını Artırmaya Yönelik Yapay Zekâ Tabanlı Yeni Yaklaşımlar” başlıklı doktora tezinden üretilmiştir. Ayrıca çalışma, Fırat Üniversitesi Bilimsel Araştırma Projeleri Birimi tarafından TEKF.24.57 No.lu proje kapsamında desteklenmiştir.

7 Yazar katkı beyanı

Gerçekleştirilen çalışmada Alev Kaya, fikrin oluşması, tasarımın yapılması ve literatür taraması, yazım denetimi ve içerik açısından makalenin kontrol edilmesi başlıklarında; İbrahim Türkoğlu, elde edilen sonuçların değerlendirilmesi, yazım denetimi ve içerik açısından makalenin kontrol edilmesi ve sonuçların incelenmesi başlıklarında katkı sunmuşlardır.

8 Etik kurul onayı ve çıkar çatışması beyanı

"Hazırlanan makalede etik kurul izni alınmasına gerek yoktur".

"Hazırlanan makalede herhangi bir kişi/kurum ile çıkar çatışması bulunmamaktadır".

9 Kaynakça

- [1] Jeong YS, Oh KJ, Cho CK, Choi HJ. "Pseudo-random number generation using LSTMs". *The Journal of Supercomputing*, 76(10), 8324-8342, 2020.
- [2] Wu X, Han Y, Zhang M, Zhu S, Cui S, Wang Y, Peng Y. "Pseudorandom number generators based on neural networks: a review". *Journal of King Saud University - Computer and Information Sciences*, 37(3), 18, 2025.
- [3] Wu X, Han Y, Zhu S, Li Y, Cui S, Wang X. *Learned Pseudo-Random Number Generator Based on Generative Adversarial Networks*. Editors: Yang H, Lu R. Frontiers in Cyber Security, 517-530, Singapore, Springer, 2024.
- [4] Bhattacharjee K, Das S. "A search for good pseudo-random number generators: Survey and empirical studies". *Computer Science Review*, 45, 100471, 2022.
- [5] Pasqualini L, Parton M. "Pseudo random number generation through reinforcement learning and recurrent neural networks". *Algorithms*, 13(11), 307, 2020.
- [6] Mukherjee P, Garg H, Pradhan C, Ghosh S, Chowdhury S, Srivastava G. "Best fit DNA-based cryptographic keys: the genetic algorithm approach". *Sensors*, 22(19), 7332, 2022.
- [7] Kalsi S, Kaur H, Chang V. "DNA cryptography and deep learning using genetic algorithm with NW algorithm for key generation". *Journal of Medical Systems*, 42(1), 17, 2018.
- [8] Basu S, Karuppiah M, Nasipuri M, Halder AK, Radhakrishnan N. "Bio-inspired cryptosystem with DNA cryptography and neural networks". *Journal of Systems Architecture*, 94, 24-31, 2019.
- [9] Reddy MI, Kumar AS, Reddy KS. "A secured cryptographic system based on DNA and a hybrid key generation approach". *Biosystems*, 197, 104207, 2020.
- [10] Sakr AS, Shams MY, Mahmoud A, Zidan M. "Amino acid encryption method using genetic algorithm for key generation". *Computers, Materials & Continua*, 70(1), 123-134, 2022.
- [11] Wu X, Han Y, Zhang M, Li Y, Cui S. "GAN-based pseudo random number generation optimized through genetic algorithms". *Complex & Intelligent Systems*, 11(1), 31, 2025.
- [12] De Bernardi M, Khouzani MHR, Malacaria P. "Pseudo-random number generation using generative adversarial networks". *ECML PKDD 2018 Workshops*, Dublin, Ireland, 10-14 September 2018.
- [13] Okada K, Endo K, Yasuoka K, Kurabayashi S. "Learned pseudo-random number generator: WGAN-GP for generating statistically robust random numbers". *PLOS ONE*, 18(6), e0287025, 2023.
- [14] Iqbal N, Khan M, Khurshid K, Hussain I. "An efficient hybrid encryption model based on deep convolutional neural networks, deoxyribonucleic acid computing and chaotic system". *Multimedia Tools and Applications*, 82(9), 13881-13903, 2023.
- [15] Park S, Kim K, Kim K, Nam C. "Dynamical pseudo-random number generator using reinforcement learning". *Applied Sciences*, 12(7), 3377, 2022.
- [16] Almardeny Y, Benavoli A, Boujnah N, Naredo E. "A reinforcement learning system for generating instantaneous quality random sequences". *IEEE Transactions on Artificial Intelligence*, 4(3), 402-415, 2023.
- [17] Lin T, Wang Y, Liu X, Qiu X. "A survey of transformers". *AI Open*, 3, 111-132, 2022.
- [18] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. "Attention is all you need". *Advances in Neural Information Processing Systems 30*, Long Beach, California, USA, 4-9 December 2017.
- [19] Lian Z, Liu B, Tao J. "CTNet: Conversational transformer network for emotion recognition". *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29, 985-1000, 2021.
- [20] Han K, Xiao A, Wu E, Guo J, Xu C, Wang Y. "Transformer in Transformer". *Advances in Neural Information Processing Systems 34*, 15908-15919, Virtual Conference, 6-14 December 2021.
- [21] Jiang J, Ke L, Chen L, Dou B, Zhu Y, Liu J, Zhang B, Zhou T, Wei GW. "Transformer technology in molecular science". *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 14(4), e1725, 2024.
- [22] Tao T, Doshi D, Kalra DS, He T, Barkeshli M. "(How) Can Transformers Predict Pseudo-Random Numbers?". *CoRR*, abs/2502.10390, 2025. <https://doi.org/10.48550/arXiv.2502.10390>
- [23] Parfentiev KV. "Generation of pseudo-random sequences using generative predictive neural network transformers". *2025 7th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*, Moscow, Russian Federation, 8-10 April 2025.
- [24] Joo W, Lee W, Park S, Moon IC. "Dirichlet variational autoencoder". *Pattern Recognition*, 107, 107514, 2020.
- [25] Gordon-Rodriguez E, Loaiza-Ganem G, Cunningham J. "The continuous categorical: a novel simplex-valued exponential family". *Proceedings of the 37th International Conference on Machine Learning*, Virtual Event, 13-18 July 2020.
- [26] Hobbhahn M, Kristiadi A, Hennig P. "Fast predictive uncertainty for classification with Bayesian deep networks". *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, 822-832, Eindhoven, The Netherlands, 1-5 August 2022.
- [27] Kang G, Dong X, Zheng L, Yang Y. "PatchShuffle regularization". *CoRR*, abs/1707.07103, 2017.
- [28] Kasantikul R, Kusakunniran W, Wu Q, Wang Z. "Channel-shuffled transformers for cross-modality person re-identification in video". *Scientific Reports*, 15(1), 15009, 2025.
- [29] Feng S, Hu K, Fan E, Zhao L, Wu C. "Kalman filter for spatial-temporal regularized correlation filters". *IEEE Transactions on Image Processing*, 30, 3263-3278, 2021.
- [30] Shannon CE. "A mathematical theory of communication". *The Bell System Technical Journal*, 27(3), 379-423, 1948.
- [31] Li J, Zhang L, Li H, Ping Y, Xu Q, Wang R, Tan R, Wang Z, Liu B, Wang Y. "Integrated entropy-based approach for analyzing exons and introns in DNA sequences". *BMC Bioinformatics*, 20(Suppl 8), 283, 2019.
- [32] Pimenta-Zanon MH, Kashiwabara AY, Vanzela ALL, Lopes FM. "GRAMPE: an alignment-free method based on the maximum entropy principle for identifying SNPs". *BMC Bioinformatics*, 26(1), 66, 2025.

- [33] Afify YM, Sharkawy NH, Gad W, Badr N. "A new dynamic DNA-coding model for gray-scale image encryption". *Complex & Intelligent Systems*, 10(1), 745-761, 2024.
- [34] Fan W, Li T, Wu J, Wu J. "Chaotic color image encryption based on eight-base DNA-level permutation and diffusion". *Entropy*, 25(9), 1268, 2023.
- [35] Bassham III LE, Rukhin AL, Soto J, Nechvatal JR, Smid ME, Barker EB, Leigh SD, Levenson M, Vangel M, Banks DL, Heckert NA, Dray JF, Vo S. "A statistical test suite for random and pseudorandom number generators for cryptographic applications". National Institute of Standards and Technology, Gaithersburg, Maryland, USA, NIST Special Publication, 800-22 Revision 1a, 2010.
- [36] Barker EB, Kelsey JM. "Recommendation for random number generation using deterministic random bit generators". National Institute of Standards and Technology, Gaithersburg, MD, USA, NIST Special Publication 800-90A Revision 1, 2015.
- [37] Hernandez-Castro J, Barrero DF. "Evolutionary generation and degeneration of randomness to assess the independence of the Ent test battery". *2017 IEEE Congress on Evolutionary Computation (CEC)*, Donostia-San Sebastián, Spain, 5-8 June 2017.
- [38] Susanti BH, Jimmy J, Ardyani MW. "ENT randomness test on DM-PRESENT-80 and DM-PRESENT-128-based pseudorandom number generator". *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, Virtual Conference from Yogyakarta, Indonesia, 16 December 2021.
- [39] Petitcolas FAP. *Kerckhoffs' Principle*. Editors: Jajodia S, Samarati P, Yung M. Encyclopedia of Cryptography, Security and Privacy, 1342-1343, Cham, Switzerland, Springer, 2025.

Ek A

Ek Tablo A1. Transformer tabanlı SRSÜ tarafından üretilen on bağımsız akışın NIST SP 800-22 test sonuçları.

Appendix Table A1. NIST SP 800-22 test results of ten independent streams generated by the Transformer-based PRNG.

TEST	1.Çıktı Kural-5	2.Çıktı Kural-2	3.Çıktı Kural-8	4.Çıktı Kural-1	5.Çıktı Kural-2	6.Çıktı Kural-6	7.Çıktı Kural-3	8.Çıktı Kural-4	9.Çıktı Kural-7	10.Çıktı Kural-3
Frequency (Monobit)	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
Block Frequency (M=1000)	0.492055	0.016060	0.824560	0.288649	0.689572	0.990748	0.795373	0.326812	0.169546	0.868533
Runs	0.670108	0.459300	0.452051	0.907653	0.384300	0.469066	0.443676	0.211300	0.560567	0.638355
Longest-Run	0.228562	0.247616	0.556901	0.276331	0.146795	0.591470	0.653706	0.542321	0.804955	0.474195
Binary Matrix Rank	0.456804	0.074780	0.103414	0.613482	0.412204	0.894455	0.522606	0.363272	0.533212	0.300395
Discrete Fourier Transform	0.890517	0.222279	0.388356	0.985357	0.274825	0.804313	0.236498	0.491297	0.142033	0.985357
Non-Overlapping Templates	0.739686	0.097442	0.384925	0.755444	0.067823	0.688952	0.818801	0.652965	0.389571	0.694725
Overlapping Templates	0.214743	0.036133	0.386392	0.011553	0.566924	0.867385	0.137798	0.519200	0.116363	0.300389
Maurer's Universal Statistical	0.079130	0.440923	0.792849	0.704990	0.609528	0.834961	0.267713	0.427213	0.603874	0.433753
Linear-Complexity	0.835187	0.775968	0.724723	0.918314	0.115570	0.498044	0.508559	0.790440	0.366911	0.634534
Serial (2. p-değeri)	0.813720	0.140250	0.486835	0.463496	0.638035	0.291598	0.998908	0.383035	0.634083	0.792307
Approximate Entropy	0.708970	0.999723	0.355422	0.896299	0.502916	0.371172	0.846102	0.975875	0.292400	0.085230
Cumulative-Sums (backward)	0.317623	0.587720	0.355212	0.384284	0.888311	0.918464	0.990100	0.634720	0.892103	0.332651
Random Excursions (last state)	0.386846	0.753970	0.804076	0.289875	0.492755	0.715398	0.584482	0.988262	0.283400	0.467620
Random Excursions Variant (last state)	0.491214	0.644276	0.909077	0.470916	0.897537	0.820659	0.711156	0.673939	0.105116	0.074239