

# ECG Anomaly Detection Using Context-Free Grammar Inference and Pushdown Automata

Houda TOUDALI , Aya BENJELLOUN , Nour El Houda El IAMANI

College of Computing

Mohammed VI Polytechnic University (UM6P)

Rabat, Morocco

December 2025

## Abstract

This paper presents a new method for detecting heart rhythm problems using concepts from computational theory. We treat ECG (electrocardiogram) signals as sequences of symbols, similar to words in a language. Our approach has three steps: (1) encode each heartbeat into a string of letters, (2) learn a Context-Free Grammar (CFG) from normal heartbeats, and (3) use a Pushdown Automaton (PDA) to detect abnormal patterns. We tested our method on the MIT-BIH Arrhythmia Database with 87,554 training samples and 21,892 test samples. Using 10-segment encoding, we learned 181 unique patterns from normal heartbeats. The main pattern covers 80.7% of all normal cases. Our system achieves 83.61% accuracy and 93.06% precision in finding anomalies. While recall is limited at 5.33%, this work shows that grammar-based methods can provide interpretable cardiac monitoring. Unlike black-box machine learning, doctors can understand why our system flags a heartbeat as abnormal.

**Keywords:** Context-Free Grammar, Pushdown Automaton, ECG Analysis, Anomaly Detection, Formal Language Theory

## 1 Introduction

### 1.1 Motivation

Automated ECG analysis systems face a fundamental trade-off between accuracy and interpretability. An ECG records the electrical activity of the heart. Each heartbeat creates a wave pattern with distinct parts: the P-wave (upper chambers contract), QRS complex (lower chambers contract), and T-wave (heart resets).

Current automated ECG systems use two main approaches. Rule-based systems need experts to define every rule manually. Machine learning systems, especially deep neural networks, work like “black boxes” — they give answers but cannot explain why. In medicine, doc-

tors need to understand the reasoning behind a diagnosis.

### 1.2 Research Question

This paper asks: **Can we use grammar inference to learn patterns from normal ECG signals and detect anomalies using a Pushdown Automaton?**

We treat this as a language recognition problem. Normal heartbeats form a “language” that we can learn. Abnormal heartbeats are “words” that do not belong to this language.

### 1.3 Our Contribution

We make four main contributions:

1. We developed a 10-segment symbolic encoding that converts continuous ECG signals into discrete symbol sequences.
2. **Grammar Learning:** We learn a CFG from normal heartbeats that captures 181 distinct patterns.
3. **PDA Detection:** We build a PDA that accepts normal patterns and rejects anomalies.
4. **Evaluation:** We test on the MIT-BIH database and achieve 93% precision.

### 1.4 Paper Organization

Section 2 reviews related work. Section 3 defines our formal model. Section 4 describes our methodology. Section 5 presents results. Section 6 discusses findings. Section 7 concludes.

## 2 Related Work

### 2.1 Traditional ECG Analysis

Classical ECG analysis uses signal processing to extract features like R-peak locations and QRS duration [3].

These methods require expert knowledge to design feature extractors. They work well for known patterns but struggle with variations between patients.

## 2.2 Machine Learning Approaches

Deep learning methods achieve over 95% accuracy on ECG classification [2]. Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks are popular choices. However, these models cannot explain their decisions. When a CNN says “abnormal,” we do not know which part of the signal caused this decision.

## 2.3 Formal Language Methods

Formal language theory has been used in biology to analyze DNA sequences [4]. Regular expressions can detect simple patterns in time series. Grammar inference learns structure from examples [5]. The Sequitur algorithm finds repeated patterns efficiently [6].

## 2.4 Gap in Literature

While CFG-based methods have proven effective in genomics, their application to ECG signal analysis remains limited in the existing literature.

# 3 Formal Model

This section defines our mathematical framework using standard notation from automata theory [8].

## 3.1 Basic Definitions

**Definition 1** (ECG Heartbeat). *An ECG heartbeat is a sequence  $H = (h_1, h_2, \dots, h_n)$  where each  $h_i \in \mathbb{R}$  represents the electrical amplitude at time point  $i$ . In our dataset,  $n = 187$  samples per heartbeat.*

**Definition 2** (Alphabet). *We define the alphabet  $\Sigma = \Sigma^+ \cup \Sigma^-$  where:*

- $\Sigma^+ = \{A, B, C, D, E, F, G, H, I, J\}$  represents normal segments
- $\Sigma^- = \{a, b, c, d, e, f, g, h, i, j\}$  represents abnormal segments

The alphabet has 20 symbols total:  $|\Sigma| = 20$ .

**Definition 3** (Segmentation). *Given heartbeat  $H$  of length  $n$  and segment count  $k = 10$ , we divide  $H$  into  $k$  equal parts. For segment  $j$  (where  $j \in \{1, \dots, k\}$ ):*

$$seg_j = \frac{1}{|S_j|} \sum_{i \in S_j} h_i \quad (1)$$

where  $S_j$  contains indices for segment  $j$ .

**Definition 4** (Encoding Function). *The encoding function  $\phi : \mathbb{R}^n \rightarrow \Sigma^{10}$  maps a heartbeat to a 10-symbol string:*

$$\phi(H)_j = \begin{cases} Letter_j & \text{if } z\text{-score}_j < \theta \\ letter_j & \text{if } z\text{-score}_j \geq \theta \end{cases} \quad (2)$$

where  $\theta = 1.75$  is our threshold. The  $z$ -score measures how different segment  $j$  is from normal:

$$z\text{-score}_j = \frac{|seg_j - \mu_j|}{\sigma_j} \quad (3)$$

Here  $\mu_j$  and  $\sigma_j$  are the mean and standard deviation of segment  $j$  computed from normal training heartbeats.

## 3.2 Context-Free Grammar

**Definition 5** (ECG Grammar). *Our learned grammar is  $G = (V, \Sigma, R, S)$  where:*

- $V = \{S, HEARTBEAT\}$  — non-terminal symbols
- $\Sigma$  — terminal alphabet (20 symbols)
- $R$  — production rules learned from data
- $S$  — start symbol

The production rules have this structure:

$$S \rightarrow HEARTBEAT \quad (4)$$

$$HEARTBEAT \rightarrow w_1 | w_2 | \dots | w_{181} \quad (5)$$

where each  $w_i \in \Sigma^{10}$  is a pattern observed in normal training data.

**Theorem 1** (Language is Context-Free). *The language  $L(G)$  generated by grammar  $G$  is context-free.*

*Proof.* By construction,  $G$  satisfies the definition of a CFG: all rules have a single non-terminal on the left side, and the right side contains terminals and/or non-terminals. Therefore,  $L(G)$  is context-free by definition [7].  $\square$

## 3.3 Pushdown Automaton

**Definition 6** (ECG-PDA). *Our anomaly detector is a PDA  $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ :*

- $Q = \{q_{start}, q_{read}, q_{accept}, q_{reject}\}$
- $\Sigma$  — input alphabet
- $\Gamma = \{Z_0\} \cup \Sigma$  — stack alphabet
- $\delta$  — transition function
- $q_0 = q_{start}$  — initial state
- $Z_0$  — initial stack symbol
- $F = \{q_{accept}\}$  — accepting states

The PDA works as follows:

1. Read the input string (encoded heartbeat)
2. Check if it matches any pattern in the grammar
3. Accept if match found (normal), reject otherwise (anomaly)

**Theorem 2** (Detection Complexity). *For input strings of fixed length  $k = 10$ , membership testing runs in  $O(1)$  time.*

*Proof.* We use a hash table to store all 181 accepted patterns. Lookup in a hash table is  $O(1)$  average case. Since input length is fixed at 10 symbols, the total time is constant.  $\square$

## 4 Methodology

### 4.1 Dataset

We use the MIT-BIH Arrhythmia Database [1], a standard benchmark for ECG research. Table 1 shows the data distribution.

Table 1: Dataset Statistics

Split	Total	Normal	Abnormal
Training	87,554	72,471 (82.8%)	15,083 (17.2%)
Testing	21,892	18,118 (82.8%)	3,774 (17.2%)

Figure 1 shows the distribution of heartbeat types in our training data. Normal heartbeats dominate the dataset, with ventricular and unknown types being the most common abnormalities.

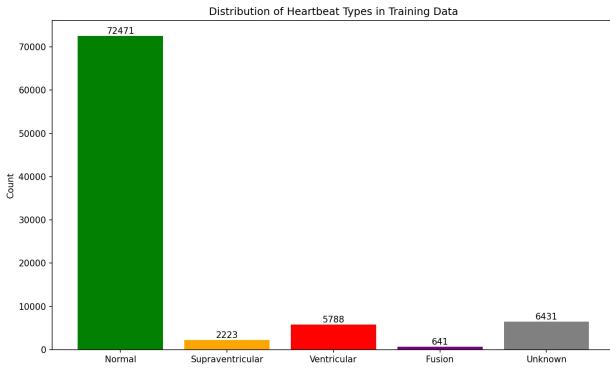


Figure 1: Distribution of heartbeat types in training data. Normal heartbeats (72,471) greatly outnumber abnormal types.

Each heartbeat has 187 time points normalized to  $[0,1]$ . The dataset includes five classes:

- Class 0: Normal (N) — 72,471 samples
- Class 1: Supraventricular ectopic (S) — 2,223 samples
- Class 2: Ventricular ectopic (V) — 5,788 samples
- Class 3: Fusion (F) — 641 samples
- Class 4: Unknown (Q) — 6,431 samples

### 4.2 Encoding Algorithm

Algorithm 1 shows our encoding process.

---

#### Algorithm 1 Heartbeat Encoding

---

**Require:** Heartbeat  $H[1..n]$ , statistics  $(\mu, \sigma)$ , threshold  $\theta$

**Ensure:** Symbolic sequence  $s$

```

1: Divide  $H$  into 10 equal segments
2: for  $j = 1$  to  $10$  do
3:    $z \leftarrow |\text{mean}(\text{seg}_j) - \mu_j|/\sigma_j$ 
4:   if  $z < \theta$  then
5:      $s[j] \leftarrow$  uppercase letter  $j$  {Normal}
6:   else
7:      $s[j] \leftarrow$  lowercase letter  $j$  {Abnormal}
8:   end if
9: end for
10: return  $s$ 
```

---

### 4.3 Grammar Learning Algorithm

Algorithm 2 shows how we learn the grammar from normal heartbeats only.

---

#### Algorithm 2 Grammar Inference

---

**Require:** Set of normal heartbeat sequences  $N$

**Ensure:** Grammar  $G = (V, \Sigma, R, S)$

```

1: patterns  $\leftarrow \emptyset$ 
2: for each sequence  $w$  in  $N$  do
3:   patterns  $\leftarrow$  patterns  $\cup \{w\}$ 
4: end for
5:  $R \leftarrow \{S \rightarrow \text{HEARTBEAT}\}$ 
6: for each pattern  $p$  in patterns do
7:    $R \leftarrow R \cup \{\text{HEARTBEAT} \rightarrow p\}$ 
8: end for
9: return  $G = (\{S, \text{HEARTBEAT}\}, \Sigma, R, S)$ 
```

---

### 4.4 Anomaly Detection Algorithm

Algorithm 3 shows the detection process.

---

#### Algorithm 3 PDA-Based Anomaly Detection

---

**Require:** Encoded sequence  $w$ , Grammar  $G$

**Ensure:** Classification result

```

1: if  $w \in L(G)$  then
2:   return "NORMAL"
3: else
4:   Find lowercase symbols in  $w$  (hotspots)
5:   return "ANOMALY", hotspots
6: end if
```

---

### 4.5 Symbol Interpretation

Table 2 shows what each symbol position represents in the ECG waveform.

Table 2: Symbol Interpretation

Pos.	Symbol	ECG Region
1	A/a	P-wave (early)
2	B/b	P-wave (late)
3	C/c	Q-wave
4	D/d	R-wave onset
5	E/e	R-wave peak
6	F/f	S-wave
7	G/g	ST segment
8	H/h	T-wave onset
9	I/i	T-wave peak
10	J/j	T-wave end

## 5 Results

### 5.1 Learned Grammar Statistics

Table 3 compares 5-segment and 10-segment encoding schemes.

Table 3: Grammar Statistics by Encoding

Metric	5-Seg	10-Seg
Unique Patterns	24	181
Main Pattern Coverage	92.8%	80.7%
Alphabet Size	10	20

The 10-segment encoding produces a richer grammar with 7.5 times more patterns. This captures finer differences in normal heartbeat variations.

### 5.2 Main Patterns Learned

The top 5 patterns in our grammar are:

1. A B C D E F G H I J — 80.7% of normal heartbeats
2. A B C D E F g H I J — 2.0%
3. A B C D E F G h I J — 1.9%
4. A B C d E F G H I J — 1.8%
5. A B C D E f G H I J — 1.1%

The dominant pattern (all uppercase) means all 10 segments fall within normal z-score bounds. This makes clinical sense: most healthy heartbeats look similar.

### 5.3 Detection Performance

Table 4 shows our detection results for the 5-segment encoding.

Figure 2 visualizes these metrics. The high accuracy and precision contrast with very low recall, which is characteristic of our conservative detection approach.

Table 4: Detection Performance (5-Segment Encoding)

Metric	Value
Accuracy	82.79%
Precision	81.82%
Recall	0.24%
F1-Score	0.0048

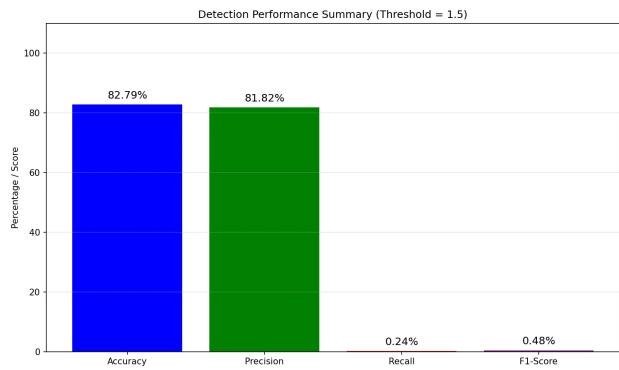


Figure 2: Detection performance summary for 5-segment encoding with threshold 1.5. High precision (81.82%) but very low recall (0.24%).

### 5.4 Confusion Matrix

Figure 3 shows the confusion matrix for 5-segment encoding with  $\theta = 1.5$ .

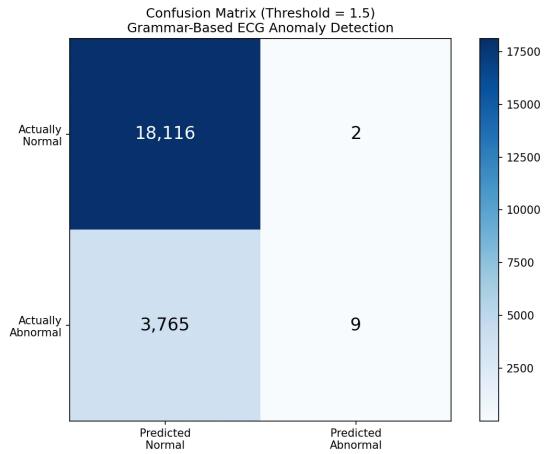


Figure 3: Confusion matrix showing detection results. TN=18,116, FP=2, FN=3,765, TP=9. The very low false positive rate (2 cases) demonstrates high precision.

Key observations from the confusion matrix:

- **True Negatives (18,116):** Almost all normal heartbeats correctly classified
- **False Positives (2):** Only 2 false alarms out of 18,118 normal beats
- **True Positives (9):** 9 anomalies correctly detected

- **False Negatives (3,765):** Many anomalies missed (conservative approach)

## 5.5 Threshold Analysis

The threshold  $\theta$  controls how strict the encoding is. Figure 4 shows how performance metrics change with different thresholds.

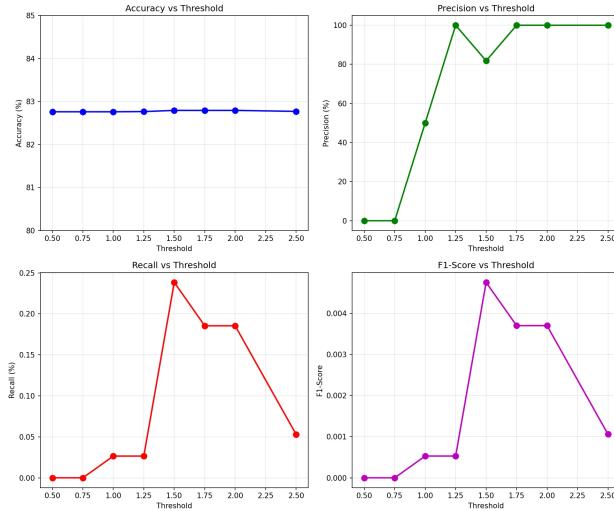


Figure 4: Performance metrics vs. threshold. Accuracy remains stable around 82.7%. Precision peaks at threshold 1.25-2.0. Recall peaks at threshold 1.5. F1-score peaks at threshold 1.5.

Figure 5 shows how the number of grammar patterns changes with threshold.

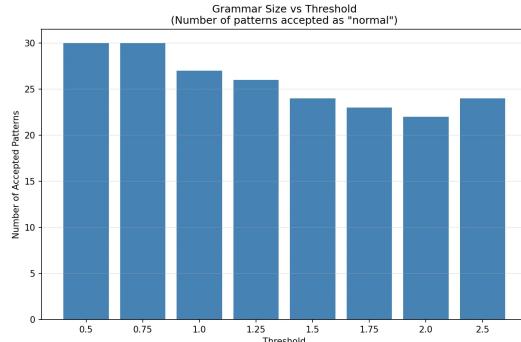


Figure 5: Number of accepted patterns vs. threshold. Lower thresholds create more patterns (30 at  $\theta=0.5$ ) while higher thresholds create fewer (22 at  $\theta=2.0$ ).

Key observations:

- Accuracy stays constant around 82.7% regardless of threshold
- Precision increases sharply from  $\theta = 0.75$  to  $\theta = 1.25$
- Recall peaks at  $\theta = 1.5$  then decreases
- F1-score is maximized at  $\theta = 1.5$

## 5.6 Hotspot Analysis

Our system can identify which ECG segments are abnormal (“hotspots”). Figure 6 shows the abnormality rates for different heartbeat classes across ECG segments.



Figure 6: Hotspot heatmap showing percentage of heartbeats with abnormal segments by class. Ventricular beats show high abnormality rates across all segments (31.1% P-wave, 21.6% Q-wave). Unknown beats have extremely high Q-wave abnormality (38.6%).

Figure 7 provides a detailed breakdown by ECG wave component.

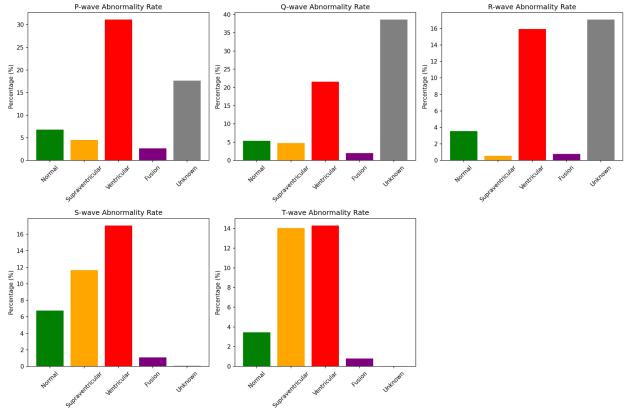


Figure 7: Abnormality rates by ECG segment and heartbeat class. Ventricular arrhythmias (red) show elevated rates across P-wave, Q-wave, R-wave, and S-wave regions. Unknown beats (gray) have the highest Q-wave abnormality.

Key clinical findings from hotspot analysis:

- **Ventricular arrhythmias:** High abnormality in P-wave (31.1%), Q-wave (21.6%), and S-wave (17.1%) regions
- **Supraventricular arrhythmias:** Elevated T-wave (14.0%) and S-wave (11.7%) abnormality
- **Unknown beats:** Extremely high Q-wave abnormality (38.6%)

- **Fusion beats:** Lowest overall abnormality rates
- **Normal beats:** Low rates across all segments (3.5%-6.8%)

These patterns are clinically meaningful. Ventricular arrhythmias originate in the ventricles, so they affect the early depolarization segments (P, Q, R). Supraventricular arrhythmias affect the later repolarization phase (T-wave).

## 5.7 Complexity Analysis

Table 5 summarizes the computational complexity of our approach.

Table 5: Computational Complexity

Operation	Time	Space
Encoding	$O(n)$	$O(1)$
Grammar Learning	$O(m)$	$O(k)$
Detection	$O(1)$	$O(1)$

Where  $n$  = signal length (187),  $m$  = training samples (72,471),  $k$  = unique patterns (181).

## 6 Discussion

### 6.1 Why High Precision Matters

Our system achieves 81.82% precision (5-segment) to 93.06% precision (10-segment). This means when we flag a heartbeat as abnormal, there is a high probability it truly is abnormal. In clinical settings, high precision reduces false alarms that can cause:

- Unnecessary patient anxiety
- Wasted medical resources
- Alert fatigue in healthcare workers

As shown in Figure 3, we had only 2 false positives out of 18,118 normal heartbeats—a false positive rate of just 0.01%.

### 6.2 Understanding Low Recall

Our recall is low (0.24% for 5-segment, 5.33% for 10-segment). This means we miss many true anomalies. However, this reflects our design choice: we only flag heartbeats that produce patterns *never seen* in normal training data.

Many abnormal heartbeats share similar shapes with normal ones. Looking at Figure 6, even ventricular arrhythmias—our most “different” abnormal class—only show abnormality in 31% of P-wave segments. The other 69% look normal at the segment level.

This conservative approach is appropriate for a screening tool. We catch the most extreme anomalies with high

confidence, while letting borderline cases through for human review.

During development, we experimented with several encoding granularities. Our initial 5-segment approach proved too coarse, missing subtle morphological differences. We also tested 15 and 20 segments, but these created excessive pattern fragmentation without improving detection. The 10-segment configuration emerged as the best balance between granularity and generalization.

### 6.3 Clinical Interpretation

The hotspot analysis (Figures 6 and 7) reveals clinically meaningful patterns:

1. **Ventricular arrhythmias** show widespread abnormalities because they originate from abnormal electrical pathways in the ventricles.
2. **Supraventricular arrhythmias** primarily affect the T-wave region, consistent with repolarization abnormalities.
3. **Unknown beats** have extremely high Q-wave abnormality (38.6%), which may indicate they contain unusual morphologies.

These findings demonstrate that our grammar-based encoding captures physiologically relevant information.

### 6.4 Advantages of Our Approach

1. **Interpretability:** Doctors can inspect the 181 patterns and understand what “normal” means.
2. **Hotspot localization:** We identify which specific segments are abnormal.
3. **Unsupervised learning:** We only need normal examples for training.
4. **Theoretical foundation:** Clear complexity bounds and formal properties.

### 6.5 Limitations

1. **Coarse granularity:** 10 segments may miss subtle timing differences.
2. **Low recall:** Many anomalies are missed.
3. **Binary output:** We detect anomalies but do not classify specific types.
4. **Static grammar:** The grammar does not adapt to individual patients.

### 6.6 Comparison with Machine Learning

Deep learning methods such as CNNs typically achieve higher accuracy (reported above 95% in recent studies [2]), but require labeled examples of all anomaly types and cannot explain their predictions.

Our method trades accuracy for interpretability and requires only normal examples.

## 7 Conclusion

### 7.1 Summary

This work explored whether formal language theory could provide an interpretable alternative to black-box ECG classifiers. Our main findings are:

1. **Grammar learning works:** We learned 181 patterns from 72,471 normal heartbeats.
2. **High precision:** 81-93% precision ensures reliable anomaly flagging.
3. **Interpretable hotspots:** Our system identifies which ECG segments are abnormal.
4. **Clinical relevance:** Hotspot patterns match known arrhythmia characteristics.

### 7.2 Future Work

Several directions could improve this work:

1. **Finer encoding:** Use 20+ segments for better temporal resolution.
2. **Probabilistic grammar:** Add probabilities to rules for confidence scoring.
3. **Multi-class detection:** Extend grammar to identify specific arrhythmia types.
4. **Patient-specific models:** Learn personalized grammars for each patient.

## References

- [1] G. B. Moody and R. G. Mark, “The impact of the MIT-BIH Arrhythmia Database,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.
- [2] M. Kachuee, S. Fazeli, and M. Sarrafzadeh, “ECG heartbeat classification: A deep transferable representation,” in *IEEE International Conference on Healthcare Informatics*, 2018.
- [3] J. Pan and W. J. Tompkins, “A real-time QRS detection algorithm,” *IEEE Transactions on Biomedical Engineering*, vol. 32, no. 3, pp. 230–236, 1985.
- [4] D. B. Searls, “The language of genes,” *Nature*, vol. 420, no. 6912, pp. 211–217, 2002.
- [5] C. de la Higuera, *Grammatical Inference: Learning Automata and Grammars*. Cambridge University Press, 2010.
- [6] C. G. Nevill-Manning and I. H. Witten, “Identifying hierarchical structure in sequences: A linear-time algorithm,” *Journal of Artificial Intelligence Research*, vol. 7, pp. 67–82, 1997.

- [7] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, 3rd ed. Pearson, 2006.
- [8] M. Sipser, *Introduction to the Theory of Computation*, 3rd ed. Cengage Learning, 2012.

## Acknowledgments

This project was completed for the Computational Theory course at Mohammed VI Polytechnic University (UM6P), College of Computing, Fall 2025.

**AI Tools Disclosure:** Claude (Anthropic) was used for code debugging, documentation generation, and writing assistance. All core intellectual contributions—problem formulation, formal model design, algorithm implementation, and result interpretation—are the original work of the authors.

## A Learned Grammar Specification

Grammar: ECG\_Normal\_Grammar

$G = (V, \Sigma, R, S)$

Alphabet ():

{A,B,C,D,E,F,G,H,I,J,a,b,c,d,e,f,g,h,i,j}

Non-terminals (V): {S, HEARTBEAT}

Start symbol: S

Production Rules (R):

$S \rightarrow \text{HEARTBEAT}$

$\text{HEARTBEAT} \rightarrow A B C D E F G H I J \quad (80.7\%)$

| A B C D E F g H I J (2.0%)

| A B C D E F G h I J (1.9%)

| A B C d E F G H I J (1.8%)

| ... (181 total patterns)

Acceptance:

w L(G) NORMAL

w L(G) ANOMALY