Agentic Feature Selection via LLM for Epileptic Seizure Detection

Aizierjiang Aiersilan ✉ **Xiaodong Qu**
The George Washington University
alexandera@gwu.edu

Abstract

Automated epileptic seizure detection from electroencephalography (EEG) signals is a clinically important task in which feature selection is typically performed using purely statistical criteria. We investigate whether a small instruction-tuned large language model (LLM) can guide iterative feature selection for binary seizure detection on the Epileptic Seizure Recognition dataset (11,500 samples, 178 features). The LLM agent (Qwen2.5-1.5B-Instruct) receives five complementary statistical summaries and selects a feature subset through multi-round reasoning. The agent achieves 96.5% accuracy and 0.911 F1 with 40 features, compared to 97.9% accuracy and 0.946 F1 for the best full-feature baseline (SVM-RBF on 178 features). Critically, 39 of the agent's 40 features coincide with the top-39 mutual-information features, and a deterministic Top-39 MI filter, evaluated by the same Random Forest classifier, attains the same 96.5% accuracy and 0.911 F1. We therefore present this work as an empirical baseline: at the 1.5B-parameter scale, the LLM behaves close to a univariate MI ranker. We situate the result against the recent LLM-based feature selection literature and enumerate the ablations and multi-dataset extensions required to determine whether larger or domain-specialized LLMs add value beyond statistical filtering.

Code: github.com/Ezharjan/AgenticFS4EEG.

1 Introduction

Epilepsy affects approximately 50 million people worldwide and is characterized by recurrent, unprovoked seizures (Shoeb, 2009). Electroencephalography (EEG) is the primary non-invasive diagnostic modality, but manual review of long-term recordings is slow and subject to inter-rater variability. Automated detectors based on classical machine learning such as random forests (Breiman, 2001), support vector machines (Cortes and Vapnik, 1995), and gradient boosting (Friedman, 2001) achieve strong performance on benchmark EEG tasks (Shimanto, 2017), but operating on the full feature space retains redundant or noisy dimensions, limiting interpretability and efficiency. Feature selection is therefore a critical preprocessing step (Guyon and Elisseeff, 2003), and is traditionally performed using statistical criteria such as mutual information (MI) (Kraskov et al., 2004), ANOVA F-scores, or wrapper-based search, none of which carry semantic understanding of the features or the domain. Recent LLM advances open the possibility of incorporating natural-language reasoning into such pipelines (Zhao et al., 2025; Wang et al., 2026).

We propose an *agentic LLM-guided feature selection* framework for binary epileptic seizure detection. The LLM agent receives five complementary statistical measures for each feature and produces a feature subset through five rounds of iterative reasoning. We evaluate the agent against three full-feature baselines and against a purely MI-driven filter baseline. Our contributions are: (1) an agentic framework coupling LLM reasoning with five complementary statistics for iterative EEG feature selection (Algorithm 1, Appendix D); (2) an evaluation showing that compressing the feature space from 178 to 40 features (a 77.5% reduction) via LLM mediation incurs only a 1.4 percentage-point accuracy gap relative to the best full-feature baseline; and (3) a controlled comparison against a deterministic Top-39 MI filter, evaluated with the same Random Forest classifier on the same split, showing that at the 1.5B-parameter scale the LLM agent's selection coincides with the MI ranking on 39 of 40 features and produces statistically indistinguishable downstream metrics. We present this as an empirical baseline rather than as evidence that LLMs add value beyond a univariate statistical filter on this benchmark, and enumerate the ablations and extensions required to revisit the question in Section 8.

Proceedings of the 25th Workshop on Biomedical Language Processing (BioNLP 2026), pages 64–74

2 Related Work

2.1 EEG-Based Seizure Detection

The Bonn University EEG database (Andrzejak et al., 2001) and its derivative Epileptic Seizure Recognition dataset (Shimanto, 2017) are primary benchmarks for seizure detection. Classical approaches extract time-, frequency-, and wavelet-domain features and apply supervised classifiers (Hastie, 2009), with random forests (Breiman, 2001) and nonlinear-kernel SVMs (Cortes and Vapnik, 1995) consistently achieving top performance. More recently, CNNs and transformers have been applied to raw EEG signals (Yang and Modesitt, 2023; Konkar and Qu, 2026), but typically require larger datasets and offer less interpretability than feature-based methods. Our work retains the feature-based paradigm and focuses on *how* features are selected rather than how they are extracted.

2.2 Feature Selection for EEG

Feature selection methods for EEG span filter, wrapper, and embedded approaches (Guyon and Elisseeff, 2003). NeuroFeat (Choudhury et al., 2026) adaptively selects features based on signal characteristics. Presacan et al. (Presacan et al., 2025) review explainable AI methods for EEG, and Lai et al. (Lai et al., 2025) note that feature engineering remains essential for small-to-medium datasets where deep models tend to overfit. These works rely on statistical or model-intrinsic criteria; we instead delegate selection to an LLM that can jointly interpret multiple statistical summaries.

2.3 LLM Agents for Scientific Data Analysis

EEGAgent (Zhao et al., 2025) and NeuroWeaver (Wang et al., 2026) are recent end-to-end LLM-driven frameworks for EEG analysis, and Murungi et al. (Murungi et al., 2023) survey LLM-assisted EEG interpretation. To our knowledge, neither EEGAgent nor NeuroWeaver reports detection metrics on the specific Epileptic Seizure Recognition benchmark used here, so direct numerical comparison is not currently possible. Our work differs by focusing on feature selection rather than automating the whole pipeline.

2.4 LLM-Based Feature Selection

LLM-Select (Jeong et al., 2024) queries an LLM with feature names and short descriptions to rank features zero-shot. CAAFE (Hollmann et al., 2023) uses an LLM to iteratively generate and evaluate new feature transformations with downstream classifier feedback. LLM-FS-Agent (Bal-Ghaoui and Sabri, 2025) adopts a deliberative multi-role LLM architecture in which several agents debate each feature's inclusion. We share the paradigm of delegating selection to an LLM but differ in three ways: (i) we operate on raw amplitude time-step features (X1–X178) without semantically meaningful names, so zero-shot name-based ranking is not applicable and the agent must reason over numeric statistics; (ii) unlike CAAFE we do not synthesize features but only select from a fixed set, giving a stricter test of the LLM's marginal contribution beyond a univariate ranker; and (iii) in contrast to LLM-FS-Agent's multi-agent debate, we use a single 1.5B-parameter instruction-tuned model with a structured statistical prompt. Our findings (Section 6) thus provide an empirical baseline complementary to the broadly positive results reported by larger or more elaborate LLM-FS systems.

3 Dataset

We use the Epileptic Seizure Recognition dataset (Shimanto, 2017), a publicly available benchmark derived from the Bonn University EEG database (Andrzejak et al., 2001). It contains 11,500 EEG segments, each represented by 178 time-domain amplitude features (sampled at 173.61 Hz) and a class label $y \in \{1, 2, 3, 4, 5\}$. The five classes encode recording conditions: y=1 denotes ictal seizure activity; y=2 and y=3 correspond to seizure-free recordings from the epileptogenic zone and hippocampal formation, respectively; and y=4 and y=5 correspond to surface EEG from healthy subjects. Each class contains exactly 2,300 samples.

Following the standard experimental protocol (Shimanto, 2017), the task is converted to binary classification: class 1 (seizure, n = 2,300) vs. classes 2–5 (non-seizure, n = 9,200), yielding a moderate 4:1 class imbalance ratio.

4 Methodology

The proposed pipeline comprises five stages: preprocessing, baseline training, LLM-guided feature selection, evaluation, and visualization. Figure 2 illustrates the overall architecture, and Figure 1 details the iterative reasoning loop at the core of the agent.

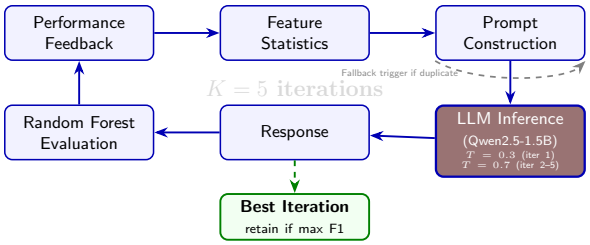

Figure 1: LLM Agent iterative reasoning loop. At each iteration, statistical summaries are compiled into a structured prompt, the LLM proposes a feature subset, a Random Forest evaluates the subset, and performance feedback informs the next round. The iteration with the highest F1-score is retained. A diversity fallback (dashed self-loop) activates if any proposal duplicates a prior selection.

4.1 Preprocessing

The 178 amplitude features (X1–X178) form the feature matrix $\mathbf{X} \in \mathbb{R}^{11500\times178}$. Labels are binarized such that $y{=}1$ maps to the positive (seizure) class and $y \in \{2,3,4,5\}$ to the negative (non-seizure) class. All features are standardized to zero mean and unit variance, with statistics computed on the training set and applied to the test set. A stratified 80/20 split preserves class proportions across training ($n{=}9{,}200$) and test ($n{=}2{,}300$) partitions.

4.2 Baseline Models

Three classifiers are trained on the full 178-dimensional standardized feature set as performance upper bounds, all implemented with scikit-learn (Pedregosa et al., 2011): a Random Forest (RF) ensemble of 200 trees with depth 20 and $\sqrt{d}$ features per split (Breiman, 2001); an SVM with RBF kernel ($C{=}10$, $\gamma{=}$scale) (Cortes and Vapnik, 1995); and a Gradient Boosting (GB) ensemble of 200 shallow trees (depth 5, learning rate 0.1) (Friedman, 2001).

4.3 LLM-Guided Feature Selection Agent

The agent is the methodological focus of this work. It is an iterative reasoning loop that combines information-theoretic feature profiling with LLM decision making to produce a compact subset for downstream classification; Algorithm 1 formalizes the procedure.

4.3.1 Feature Statistics Computation

For each of the 178 features x_j, five complementary statistics are computed on the training set. Mutual information (MI) is estimated with the k-nearest-neighbor method of Kraskov et al. (Kraskov et al., 2004) and captures nonlinear feature-label dependency. Pearson correlation captures the linear component of the same association, allowing the agent to distinguish purely linear from nonlinear discriminative structure. Variance is included to confirm the absence of degenerate (near-constant) features after standardization. Random Forest importance, the mean decrease in impurity from a preliminary RF (Breiman, 2001), complements the filter statistics by reflecting utility within an ensemble classifier. The ANOVA F-score (between-class over within-class variance) signals strong univariate class separation. These five measures span information-theoretic (MI), linear-statistical (correlation, F-score, variance), and model-based (RF importance) views, so the LLM has complementary evidence to cross-reference rather than a single ranking to imitate. Features are presented to the LLM sorted by MI in descending order.

4.3.2 Prompt Architecture

At each iteration k, the agent constructs a structured prompt with four components. A system message establishes the LLM's role as a feature selection expert and constrains outputs to a JSON list of feature names. A statistical summary table lists, for the top-40 features by MI, each feature's MI score, Pearson correlation, RF importance, and F-score. For iterations $k \geq 2$, the prompt is augmented with performance feedback (the prior iteration's accuracy, F1, AUC; the five most and five least important features by RF importance; and the full previous feature list), and with a mutation instruction directing the model to drop at least five low-importance features and replace them with high-MI or high-F-score alternatives, returning exactly 30 features as a JSON list. This structure mimics a human analyst's workflow: inspect, then refine based on downstream performance. Exact prompt templates are in Appendix D.

4.3.3 Temperature Schedule and Diversity Mechanisms

Exploration is controlled via a two-phase temperature schedule. Iteration 1 uses $T{=}0.3$ to produce a stable, high-confidence baseline selection, and iterations 2–5 use $T{=}0.7$ to encourage exploration. A diversity safeguard ensures no two iterations produce identical subsets: an exact match triggers a deterministic mutation that drops the 5 lowest-MI features and adds the highest-MI features not yet selected. The LLM's output is also clamped to the

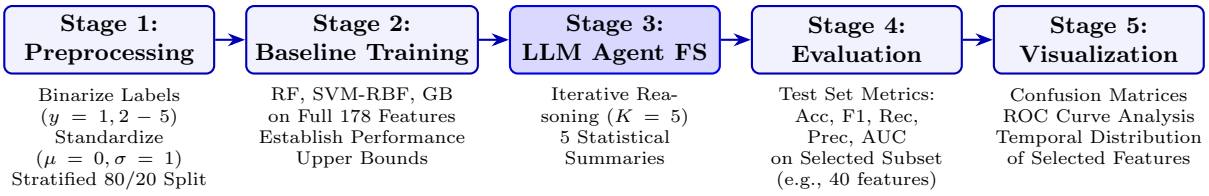

Figure 2: Five-stage experimental pipeline. The LLM Agent Feature Selection stage (emphasized) is the methodological focus of this work.

loose size range $[20, 80]$ (small subsets are padded with high-MI features; large ones are truncated).

Note on requested vs. returned size. The prompt requests *exactly* 30 features, but the post-processor only enforces $[20, 80]$ and otherwise preserves the model's choice. Because Qwen2.5-1.5B-Instruct (Qwen Team, 2025) does not strictly satisfy length constraints at this scale, the returned subsets ranged from 20 to 40 features across iterations (Table 2); we deliberately did not pad or truncate to 30, since doing so would overwrite the model behavior we wished to study. The discrepancy between the prompted 30 and the retained 40 (iteration 1) is therefore a faithfully reported instruction-following limitation, discussed further in Section 7.

4.3.4 Iterative Refinement Loop

The agent executes $K{=}5$ iterations (Algorithm 1). At each iteration the LLM receives the statistics (and, from $k{\geq}2$, the prior iteration's feedback), proposes a subset, which a Random Forest (200 trees, depth 20) trains and scores on accuracy, F1, and AUC; the highest-F1 iteration is retained. If the LLM fails to return a valid response, a statistical fallback selects the top-k features from one of the five precomputed statistics, cycling through MI, F-score, RF importance, correlation magnitude, and variance across successive fallbacks.

Algorithm 1 Agentic LLM-Guided Feature Selection

Require: Matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, labels $\mathbf{y}$, LLM $\mathcal{M}$, iterations $K = 5$
Ensure: Best feature subset S^*

Compute stats $\mathbf{s}_j = (\text{MI}_j, \rho_j, \sigma_j^2, \text{RF}_j, F_j)$ and rank by MI
$S^* \leftarrow \varnothing, F_1^* \leftarrow 0, \mathcal{H} \leftarrow \varnothing$ ▷ Init best set, score, and history
for $k = 1$ **to** K **do**
  $T_k \leftarrow (k = 1)?0.3 : 0.7$ ▷ Temperature schedule
  $p_k \leftarrow \textsc{BuildPrompt}(\mathbf{s}, \mathcal{H}, \text{fb}_{k-1})$
  $r_k \leftarrow \mathcal{M}.\text{gen}(p_k, T_k)$ ▷ LLM sampling
  $S_k \leftarrow \textsc{ParseAndClamp}(r_k, 20, 80)$ ▷ Enforce size constraints
  if $S_k \in \mathcal{H}$ **then** $S_k \leftarrow \textsc{MutateFallback}(S_k, \mathbf{s})$ ▷ Handle stagnation
  $\mathcal{H} \leftarrow \mathcal{H} \cup \{S_k\}$
  $(\text{acc}_k, \text{F1}_k, \text{auc}_k, \mathcal{I}_k) \leftarrow \textsc{Evaluate}(S_k, \mathbf{X}, \mathbf{y})$
  if $\text{F1}_k > F_1^*$ **then**
    $S^* \leftarrow S_k, F_1^* \leftarrow \text{F1}_k$ ▷ Update global optimum
  $\text{fb}_k \leftarrow (\text{acc}_k, \text{F1}_k, \text{auc}_k, \mathcal{I}_k)$ ▷ Feedback for next iteration
return S^*

4.3.5 Evaluation of Selected Features

The final feature subset trains a Random Forest classifier (200 trees, depth 20, identical to the full-feature RF baseline) on the training set and is evaluated on the held-out test set; the LLM agent uses RF as its internal evaluator, so final reported metrics are computed on unseen data.

Top-k MI Filter baseline. To isolate the LLM's marginal contribution, we evaluate a deterministic Top-k MI Filter baseline that selects the k features with highest training-set MI with the label, with $k{=}39$ chosen to closely match the agent's 40-feature budget. For full comparability, this MI-filter subset is evaluated using *exactly the same Random Forest classifier* (200 trees, depth 20, seed 42) used to score the agent's selection; only the input feature columns differ. Any gap between the two is therefore attributable to feature choice alone.

5 Experimental Setup

We use Qwen2.5-1.5B-Instruct (Qwen Team, 2025) as the LLM backbone via a hosted inference endpoint with a 500-token output limit. For each method we report accuracy, precision, recall (sensitivity), F1, and AUC; confusion matrices are pro-

vided for all methods. Experiments ran on an NVIDIA RTX 5060 GPU with Python 3.x and scikit-learn (Pedregosa et al., 2011), with random seed 42 for reproducibility.

Choice of LLM backbone. We deliberately use a small (1.5B), general-purpose instruction-tuned model rather than a larger or biomedical LLM. First, small open-weight models are realistic for clinical edge environments where data residency and latency constraints discourage routing EEG statistics to large hosted models. Second, our goal is not peak accuracy but characterizing what an off-the-shelf lightweight LLM contributes *above* a purely statistical filter; a small contribution at this scale is itself an informative empirical baseline for the LLM-FS literature (Section 2.4). Third, available medical LLMs at comparable scale are predominantly clinical-text models tuned for narrative question answering rather than reasoning over numeric tables of unnamed time-step features, so their advantage here is not obvious in advance. Scaling and domain-specialized backbones are listed as future work in Section 8.

6 Results

6.1 Overall Performance Comparison

Table 1 presents the classification performance of all methods on the held-out test set (2,300 samples).

Table 1: Binary seizure detection performance on the test set. Full-feature baselines use all 178 features; the Top-39 MI Filter uses 39 features and the LLM agent uses 40 features.

Method	Acc ↑	Prec ↑	Rec ↑	F1 ↑	AUC ↑
Random Forest	0.971	0.960	0.891	0.925	0.996
SVM-RBF	**0.979**	0.956	**0.937**	**0.946**	0.996
Grad. Boosting	0.974	**0.967**	0.902	0.934	**0.996**
Top-39 MI Filter (39 ft.)	0.965	0.922	0.900	0.911	0.993
LLM Agent (40 ft.)	0.965	0.924	0.898	0.911	0.993

All methods exceed 96.4% accuracy and 0.992 AUC, confirming the dataset's high discriminability. SVM-RBF achieves the highest accuracy (97.9%), F1 (0.946), and recall (93.7%, the highest among all methods), missing the fewest seizure events. Gradient Boosting achieves the highest precision (96.7%) and AUC (0.996), indicating low false-positive rate at the cost of slightly lower recall. Random Forest balances precision (96.0%) and recall (89.1%), with competitive AUC (0.996). The LLM Agent isolates 40 features, dropping roughly three-quarters of the input space, and still preserves strong predictive power with only a minor metric drop.

The Top-39 MI Filter, evaluated with the identical Random Forest classifier, attains comparable results from 39 features selected by mutual information alone. The LLM's chosen set is a strict superset of these 39 features, adding only X61; the overlap is therefore 39/40 (97.5%). Across Table 1's metrics the two methods differ by at most 0.002 on precision and recall and are identical to three decimals on accuracy, F1, and AUC. The LLM's marginal contribution beyond a univariate MI ranking at this scale is thus not statistically meaningful on this benchmark; we treat this as the central empirical finding and discuss its implications in Section 7.

Clinically, the LLM agent's 89.8% recall is below the full-feature baselines, reflecting the cost of dimensionality reduction; acceptability is deployment-specific (screening tolerates lower recall, ICU monitoring demands maximal sensitivity, see Section 7.4).

6.2 Confusion Matrix Analysis

Figure 3 presents the confusion matrices for all methods.

SVM-RBF has 29 false negatives (FN) and 20 false positives (FP), the best sensitivity. The LLM agent has 47 FN and 34 FP, a modest increase but a 10.2% false-negative rate on the seizure class. The Top-39 MI Filter has 46 FN and 35 FP, comparable to the agent. Gradient Boosting has the fewest FP (14) but 45 FN, consistent with its high-precision profile.

6.3 ROC Curve Analysis

Figure 4 shows the receiver operating characteristic (ROC) curves for all methods.

All five ROC curves are nearly indistinguishable, hugging the upper-left corner. AUC ranges from 0.993 (Top-39 MI Filter) to 0.996 (Gradient Boosting), a 0.004 spread, so the reduced-feature methods preserve nearly all ranking quality of the full-feature models.

6.4 Feature Selection Analysis

The LLM agent selected 40 features from the 178 available. Figure 5 shows the RF feature importances across all 178 features, providing context for the agent's selection.

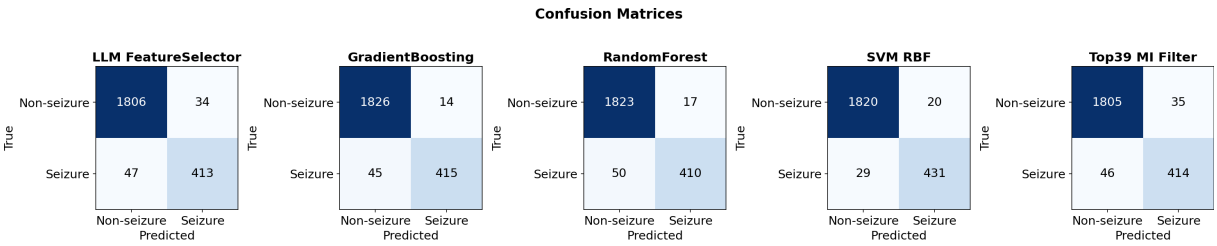

Figure 3: Confusion matrices on the test set for all five methods.

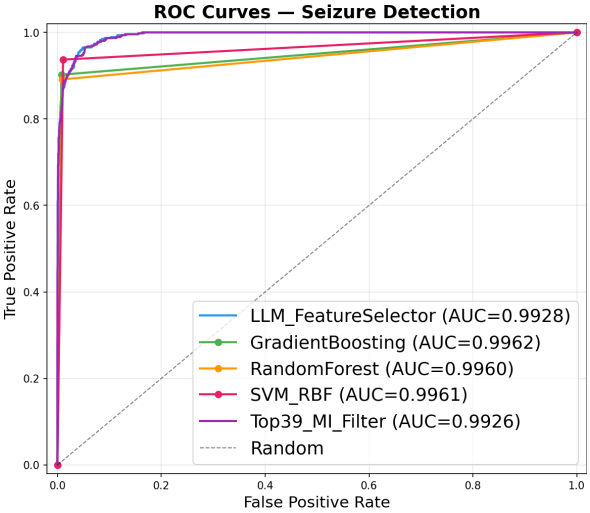

Figure 4: All methods' ROC curves achieve AUC $>$ 0.992.

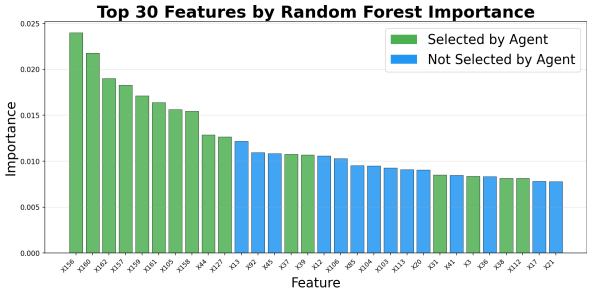

Figure 5: Random Forest feature importances (mean decrease in impurity) for all 178 features.

The five highest-MI features in the LLM's selection are X44, X172, X2, X160, and X39, all with MI above 0.199. The 40 selected features span the full EEG segment (X2 to X174), indicating the LLM captured signal dynamics across the whole recording window rather than concentrating on one region. Figure 6 visualizes this spread.

As noted earlier, the agent's chosen 40-feature subset perfectly encompasses the Top-39 MI Filter selections, supplementing them with only X61. This near-total alignment is explored structurally in Section 7.

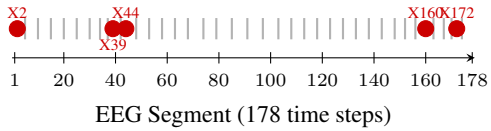

Figure 6: Temporal distribution of the 40 LLM-selected features across the 178-step EEG segment. Red circles mark the five highest-MI features; gray ticks mark the remaining selected features. The selection spans the full recording window.

6.5 Agent Iteration Behavior

Figure 7 and Table 2 present the full performance profile and subset sizes of the LLM agent across the 5 iterations.

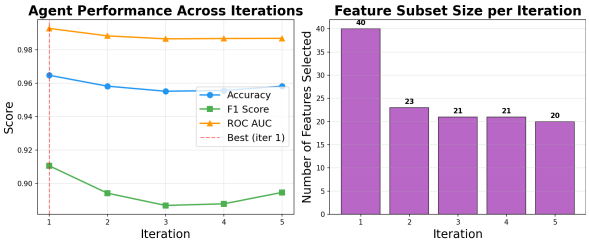

Figure 7: Full performance profile of the LLM agent across 5 iterations.

Table 2: Per-iteration agent outputs. All evaluation metrics are listed for each iteration.

Iter.	T	#Feat. ↓	Acc ↑	F1 ↑	AUC ↑
1	0.3	40	**0.965**	**0.911**	**0.993**
2	0.7	23	0.958	0.894	0.988
3	0.7	21	0.955	0.887	0.987
4	0.7	21	0.956	0.888	0.987
5	0.7	**20**	0.958	0.895	0.987

The temperature schedule and feature-level feedback produced genuinely different subsets across iterations, with no diversity fallback triggered. The best selection was iteration 1 (T=0.3, F1 0.911, 40 features); higher-temperature iterations (T=0.7) selected 20–23 features and underperformed by an F1 range of 0.024 (0.887–0.911). Both regimes

deviated from the prompted 30-feature target, indicating loose length-constraint adherence at this scale. Iteration 5 reached F1 0.895 with only 20 features, slightly above iterations 3–4 (21 features), so subset composition affects outcomes beyond raw count.

6.6 Performance Comparison Overview

Figure 8 compares all methods across the five metrics.

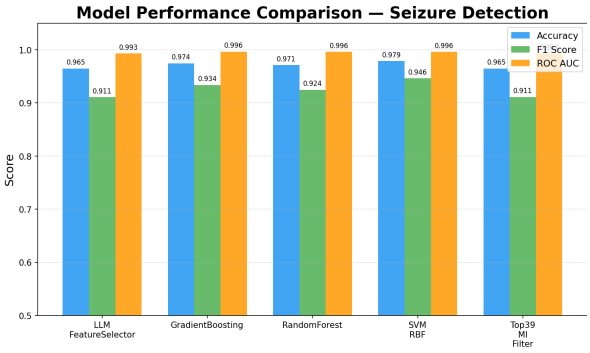

Figure 8: Bar chart comparing Accuracy, F1, and AUC across all five methods.

All methods are tightly clustered on AUC and accuracy; the LLM agent and Top-39 MI Filter perform nearly identically, both slightly below the full-feature baselines, with the gap most visible on precision and recall.

7 Discussion

7.1 Accuracy–Interpretability Trade-off

The LLM-driven pipeline yields competitive detection metrics with a fraction of the features, but this must be qualified: the same compact subset is achievable here by a statistical MI filter, so the interpretability gain reported is best attributed to dimensionality reduction rather than to the LLM. Within that caveat, a small feature set is practically useful in settings where model decisions must be auditable, since a clinician can inspect a concise list of waveform amplitudes against known electrophysiological signatures, whereas auditing all 178 features is impractical at run time. The marginal accuracy drop may be acceptable in screening or outpatient contexts, but is harder to justify in intensive-care monitoring where false-negative costs dominate (Section 7.4).

7.2 LLM Reasoning Capabilities and Limitations

Qwen2.5-1.5B-Instruct (Qwen Team, 2025) reliably parses statistical tables, ranks features by MI and F-score, and emits well-formatted JSON, showing a small LLM can serve as a functional component given a clear, structured prompt. However, four behavioral observations constrain its practical value at this scale (each consolidated in the Limitations of Section 8). First, instruction adherence is weak: the model returned 40 features at low temperature and 20–23 at higher temperatures despite an explicit 30-feature request. Second, the near-complete overlap (39/40, 97.5%) with the top-39 MI features is consistent with the model disproportionately weighting MI, which is presented both as a tabulated value and as the row sort key. This interpretation is only suggested by the overlap; without the no-MI or shuffled-MI ablations in Section 8 we cannot rule out that the LLM rediscovered the MI ranking from the other three columns. Third, iteration did not improve the initial selection (best F1 stays at iteration 1) but did not degrade it either, with F1 spanning only 0.024; we call this robust but non-improving. Fourth, the model produced no chain-of-thought rationale (Wei et al., 2022), because the prompt restricts the output to a JSON list, which is a prompt-design choice rather than a backbone limitation.

7.3 Comparing Statistical Feature Selection

The Top-39 MI Filter, evaluated with the identical Random Forest and split, matched the agent's headline metrics to three decimals on accuracy, F1, and AUC, with at most a 0.002 difference on precision and recall. The equivalence derives directly from the model's selections (39 of 40 features are the top-39 MI features). Because the prompt foregrounds MI both positionally and informationally, the subset can plausibly be reproduced by a copy-the-top-ranked-rows heuristic. Temperature variation does not control for this confound, since temperature affects stochasticity rather than which input cues the model attends to. The cleanest controls (deterministic decoding $T{=}0$, a no-MI or shuffled-MI prompt, and per-iteration quantification of deviation from the rolling top-k MI list) are enumerated as required follow-ups in Section 8. The equivalence still has value as an empirical upper bound on purely MI-driven filtration at a 40-feature budget, and the agentic framework retains structural

advantages, since it can readily incorporate natural-language instructions (e.g., "prefer early temporal features") and be paired with more capable LLMs without architectural changes.

7.4 Comparing Deep Learning Baselines

Recent end-to-end deep models trained on the raw EEG of this benchmark report 98–99%+ accuracy; for example, Guhdar et al. (Guhdar et al., 2025) report a 1D-CNN with multi-head attention surpassing 99%. Our best classical baseline (SVM-RBF, 97.9%) and our LLM-guided 40-feature subset (96.5%) sit below this deep-learning state of the art. We do not claim parity. Instead, the contribution is a small, explicitly enumerated, human-auditable feature subset that exposes which time-step amplitudes drive the decision. End-to-end CNN/transformer detectors offer higher accuracy and recall but operate on hundreds of millions of opaque weights over the raw waveform, which complicates the post-hoc audit our pipeline targets. This trade-off is unfavorable in high-acuity settings (e.g., ICU monitoring); we revisit this in the limitations.

7.5 Claim, Caveat, and Path Forward

Interpretable rationales are arguably the strongest motivation for an LLM in a feature-selection loop, and we acknowledge that this paper's interpretability claim rests on subset size rather than on model-generated narrative. The current prompt (Appendix D) restricts the output to a JSON list, which is a prompt-design choice, not a backbone limitation. A revised prompt requiring per-feature justifications, or an explicit chain-of-thought trace (Wei et al., 2022) preceding the JSON list, would yield qualitative material amenable to neurologist inspection. We have not run that experiment here, and we list eliciting and evaluating such rationales as a primary direction for future work in Section 8.

8 Conclusion

We presented an agentic feature-selection framework for EEG seizure detection in which a small instruction-tuned LLM (Qwen2.5-1.5B-Instruct) iteratively reduces a 178-dimensional input over five statistical measures. On this benchmark the agent attained 96.5% accuracy and 0.911 F1 with 40 features (a 77.5% reduction), versus 97.9% accuracy and 0.946 F1 for the strongest full-feature baseline (SVM-RBF). The headline finding is a negative one: 39 of the agent's 40 features are exactly the top-39 MI features, and a deterministic Top-39 MI filter evaluated by the same Random Forest classifier matches the agent's downstream metrics. At the 1.5B-parameter scale, with MI presented as both a tabulated value and the row sort key, the model behaves close to a copy-the-top-ranked-rows heuristic. We frame this as an empirical baseline that future, more capable LLM-FS systems must beat to justify their added complexity.

Limitations. We list the following limitations so the contribution is not overstated. (L1) MI is presented both as a numeric column and as the row sort key, and we did not run an ablation withholding or shuffling MI, so the agent's behavior cannot be cleanly separated from a univariate MI ranker. (L2) All results are on one saturated benchmark where multiple methods already exceed 96% accuracy, so generalization to multi-channel, longer, or noisier recordings is unsupported here. (L3) Selected features are evaluated only with Random Forest, leaving cross-classifier transfer (notably to SVM-RBF) untested. (L4) The JSON-only output precludes per-feature rationales, so the interpretability claim rests on subset size rather than model narrative. (L5) The model did not respect the prompted "exactly 30 features" constraint (returning 20–40 across iterations). (L6) Recent end-to-end deep models report 98–99%+ accuracy on the same dataset (Guhdar et al., 2025); the recall drop is clinically non-trivial. (L7) Larger LLMs and domain-specialized medical LLMs were not evaluated. (L8) Headline numbers use a single fixed seed, with no variance reported across seeds.

Future Work. The most informative follow-ups are: (1) an MI ablation that removes or shuffles the MI column and uses deterministic decoding ($T{=}0$); (2) per-iteration quantification of how often the LLM's selection deviates from the rolling top-k MI list; (3) cross-classifier transfer of the agent's 40-feature subset to SVM-RBF and Gradient Boosting, with the same reported for the Top-k MI baseline; (4) multi-dataset generalization on CHB-MIT (Guttag, 2010), Siena (Detti, 2020), and EPILEPSIAE (Ihle et al., 2012); (5) scaling to 7B–70B LLMs and to domain-specialized medical LLMs to test for genuine cross-statistic synthesis; (6) prompts that elicit per-feature rationales or full chain-of-thought (Wei et al., 2022) for clinically auditable justifications; (7) recall-targeted refinement that feeds false-negative patterns back into the next iteration; and (8) multi-seed reporting with confidence intervals.

References

Ralph G Andrzejak, Klaus Lehnertz, Florian Mormann, Christoph Rieke, Peter David, and Christian E Elger. 2001. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E*, 64(6):061907.

Mohamed Bal-Ghaoui and Fayssal Sabri. 2025. Llm-fs-agent: a deliberative role-based large language model architecture for transparent feature selection. *arXiv preprint arXiv:2510.05935*.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Nitin Choudhury, Daisy Das, Deepjyoti Deka, Rajdeep Ghosh, Nabamita Deb, and Ebrahim Ghaderpour. 2026. Neurofeat: An adaptive neurological eeg feature engineering approach for improved classification of major depressive disorder. *Biomedical Signal Processing and Control*, 113:109031.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

Paolo Detti. 2020. Siena Scalp EEG Database. *PhysioNet*. Version 1.0.0.

Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Mohammed Guhdar, Ramadhan J Mstafa, and Abdulhakeem O Mohammed. 2025. Hybrid deep learning model for epileptic seizure classification by using 1d-cnn with multi-head attention mechanism. *arXiv preprint arXiv:2501.10342*.

John Guttag. 2010. CHB-MIT Scalp EEG Database. *PhysioNet*. Version 1.0.0.

Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

Trevor Hastie. 2009. The elements of statistical learning: data mining, inference, and prediction.

Noah Hollmann, Samuel Müller, and Frank Hutter. 2023. Large language models for automated data science: Introducing caafe for context-aware automated feature engineering. *Advances in Neural Information Processing Systems*, 36:44753–44775.

Matthias Ihle, Hinnerk Feldwisch-Drentrup, César A Teixeira, Adrien Witon, Björn Schelter, Jens Timmer, and Andreas Schulze-Bonhage. 2012. Epilepsiae–a european epilepsy database. *Computer methods and programs in biomedicine*, 106(3):127–138.

Daniel P Jeong, Zachary C Lipton, and Pradeep Ravikumar. 2024. Llm-select: Feature selection with large language models. *arXiv preprint arXiv:2407.02694*.

Aniket Konkar and Xiaodong Qu. 2026. A review of transformer-based and hybrid deep learning approaches for eeg analysis. In *International Conference on Human-Computer Interaction*, pages 391–404. Springer.

Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. 2004. Estimating mutual information. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 69(6):066138.

Junhong Lai, Jiyu Wei, Lin Yao, and Yueming Wang. 2025. A simple review of eeg foundation models: Datasets, advancements and future perspectives. *arXiv preprint arXiv:2504.20069*.

Nathan Koome Murungi, Michael Vinh Pham, Xufeng Caesar Dai, and Xiaodong Qu. 2023. Empowering computer science students in electroencephalography (eeg) analysis: A review of machine learning algorithms for eeg datasets. In *The 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and 1 others. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Oriana Presacan, Jaya Ojha, Anis Yazidi, Eric Monteiro, and Pedro G Lind. 2025. A comprehensive review of explainable ai in deep learning algorithms for eeg analysis. *ACM Transactions on Computing for Healthcare*.

Qwen Team. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Harun Shimanto. 2017. Epileptic seizure recognition. https://www.kaggle.com/datasets/harunshimanto/epileptic-seizure-recognition. Accessed: 2026-02-18.

Ali Hossam Shoeb. 2009. *Application of machine learning to epileptic seizure onset detection and treatment*. Ph.D. thesis, Massachusetts Institute of Technology.

Guoan Wang, Shihao Yang, Jun-En Ding, Hao Zhu, and Feng Liu. 2026. Neuroweaver: An autonomous evolutionary agent for exploring the programmatic space of eeg analysis pipelines. *arXiv preprint arXiv:2602.13473*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Ruiqi Yang and Eric Modesitt. 2023. Vit2eeg: leveraging hybrid pretrained vision transformers for eeg data. *arXiv preprint arXiv:2308.00454*.

Sha Zhao, Mingyi Peng, Haiteng Jiang, Tao Li, Shijian Li, and Gang Pan. 2025. Eegagent: A unified framework for automated eeg analysis using large language models. *arXiv preprint arXiv:2511.09947*.

Appendix

A Experimental Details

All experiments were executed on an NVIDIA RTX 5060 GPU and an Intel Core processor. Software dependencies include Python 3.12, scikit-learn 1.4, and numpy. Classification endpoints were fully self-contained using a fixed random seed (42) to guarantee strict reproducibility across training/test splits, model initializations, and evaluation metrics. In training the baselines, standard hyperparameter selections were employed (e.g., standard ensemble counts for Gradient Boosting and Random Forest are set to 200, max depth set at 20 for RF and 5 for GB). The total compute time for agent iterations and baseline extractions took approximately several minutes per run, owing to the lightweight dataset properties.

B Complete Feature List

The LLM-guided agent selected a total subset of 40 features out of the 178 initial timeline points. The complete list of features elected by the best-performing iteration ($k = 1$) is given below.

- **Overlap with Top-39 MI Filter (39 features):** X1, X2, X3, X4, X14, X16, X31, X37, X38, X39, X44, X47, X52, X53, X67, X68, X77, X90, X91, X105, X111, X112, X124, X126, X127, X128, X129, X141, X156, X157, X158, X159, X160, X161, X162, X166, X167, X172, X174.
- **Unique to the LLM agent (1 feature):** X61.

C Per-Iteration Analysis

Over the 5 distinct interaction rounds, the agent successfully submitted distinct feature groupings, heavily influenced by temperature schedules and dynamic iteration histories.

- **Iteration 1 ($T = 0.3$):** 40 features. Reached an F1-score of 0.9107 and Accuracy of 96.48%. Validated as the top-performing configuration.
- **Iteration 2 ($T = 0.7$):** 23 features. Performance shifted down slightly to F1 of 0.8943.
- **Iteration 3 ($T = 0.7$):** 21 features. Showed further contraction prioritizing precision (Accuracy=95.52%, F1=0.8869).
- **Iteration 4 ($T = 0.7$):** 21 features. Stable subset retention (Accuracy=95.57%, F1=0.8879).
- **Iteration 5 ($T = 0.7$):** 20 features. Modest upward recovery reaching an F1 of 0.8947.

No deterministic fallback triggers were fired; each output strictly obeyed structural formatting requisites (i.e., valid JSON array), although the specific 30-feature length constraint was loosely interpreted by the model.

D LLM Prompt Templates

To ensure full reproducibility of the agentic feature selection process, we provide the exact text of the prompts supplied to the Large Language Model. The prompt architecture consists of a static system instruction, a dynamic statistical table, and an adaptive feedback mechanism that depends on the current iteration.

D.1 Initial Iteration Prompt

In the first iteration ($k = 1$), the agent is provided solely with the statistical summary of the features and is instructed to output an initial feature subset. The exact prompt template is as follows, where `{top_k}` is typically 30, and the tabular data is populated dynamically:

```
You are a feature selection agent for EEG seizure detection
(binary: seizure vs non-seizure).
Below are statistics for the top {top_k} features out of
178 total (X1-X178).

Feature    MI        Corr      RF_Imp    F_Score
--------------------------------------------------
[Dynamically generated statistical tabular data]

Select EXACTLY 30 features from X1-X178 for best seizure detection. Use the
    statistics above.
Output ONLY a JSON array of 30 feature names.
Example: ["X1","X2","X3",...,"X30"]
Your 30 features:
```

D.2 Refinement Iteration Prompt

For subsequent iterations ($k \geq 2$), the prompt is augmented with performance feedback from the preceding iteration's Random Forest evaluation. Notably, explicit mutation instructions are provided to encourage exploration and refinement based on empirical classification behavior:

```
You are a feature selection agent for EEG seizure detection
(binary: seizure vs non-seizure).
Below are statistics for the top {top_k} features out of
178 total (X1-X178).

Feature    MI        Corr      RF_Imp    F_Score
--------------------------------------------------
[Dynamically generated statistical tabular data]

Previous round: {num_prev} features, Accuracy={prev_acc:.4f}, F1={prev_f1:.4f}, AUC
    ={prev_auc:.4f}.
You previously selected: [{prev_feat_str}]. Top-5 most important features in that
    selection: {top5_str}
Bottom-5 least important features in that selection: {bot5_str}
You MUST select a DIFFERENT set of 30 features. Drop at least 5 features from your
    previous selection ideally the low-importance ones listed above and replace them
     with features NOT in your previous selection that have high mutual information
    or F-score from the table.

Select EXACTLY 30 features from X1-X178 for best seizure detection. Use the
    statistics above.
Output ONLY a JSON array of 30 feature names.
Example: ["X1","X2","X3",...,"X30"]
Your 30 features:
```

In instances where detailed importance feedback is unavailable, the mutation instruction gracefully defaults to a generic replacement directive: *"Try different features to improve F1. Add more features if too few, remove noisy ones if too many."* This structured prompt architecture reliably anchors the recursive reasoning loops described in Section 4.3.2.