Machine Learning Models for Decoding Eye State from EEG Data

Keane J. Haesle
Stats 220 Final Project
Stanford University
khaesle@stanford.edu

Luke D. Duthie
Stats 220 Final Project
Stanford University
lduthie@stanford.edu

1 Introduction

Rösler and Suendermann (2013) recorded EEG data using a 14-channel Emotiv EPOC headset over a short session with one volunteer, labeling each frame as eyes open or closed [5]. They found that even simple classifiers could distinguish the two states with high accuracy, with their best model achieved a 2.7% error rate. Building on their dataset, our project compares five modeling approaches after significant preprocessing of the inherently noisy EEG data. We use a Generalized Linear Model as a transparent baseline, and Naive Bayes as a simple probabilistic benchmark, which performed the worst overall. Our unsupervised Hidden Markov Model (HMM) improved upon the static models by capturing temporal patterns in the signal. Our supervised HMM, which incorporates the eye-state labels into its state transitions, performed even better, achieving the second highest accuracy. Our best results came from a unidirectional LSTM, which effectively handled the temporal dependencies and noise in the EEG data, highlighting the value of sequence-aware models for predicting real world behavior. You can find our code for each model here HMM, LSTM, Naive Bayes, and GLM.

2 Background and Related Work

2.1 Physiological and Technical Background

EEG is a noninvasive method for recording neural activity, offering high temporal resolution but low spatial precision. EEG can be used to decode attention states during a focus activity [3] and during driving [4], and it can also be used to decode speech with non-trivial accuracy [2]. In decoding eye states (eye open versus eye closed), there exist hallmarks of each state, such as an increase in alpha wave (8 - 13 Hz) power in eye closed state, as discovered by Berger in the 1930's [1], and an increase in beta wave (13 - 30 Hz) power. Roesler and Suendermann's research [5] attempted to decode eye state from EEG data, and we created models in order to replicate their findings.

2.2 Dataset

The EEG Eye State dataset, developed by Rösler and Suendermann [5], is the foundation of this study. Recorded using a 14-channel Emotiv EEG Neuroheadset, the dataset captures a 117-second continuous measurement at a 128 Hz sampling rate, yielding 14,980 instances. The data was collected in a quiet room with a single proband, who was videotaped and instructed to change eye states (open or closed) at free will, ensuring approximately equal cumulative durations for both states with varying interval lengths, as illustrated in Fig. 1. After removing four instances due to transmission errors, the final corpus comprises 14,976 instances, with 55.12% of the data labeled as eyes open and 44.88% labeled as eyes closed, indicating a slight label imbalance. Each instance has 14 voltage amplitude values, ranging from 3924.10 to 4833.85. The instances reflect eye state information, as well as extensive artifacts and noise.

2.3 Rosler and Suendermann's work

Rösler and Suendermann work [5] introduced the EEG Eye State dataset and established an initial benchmark for eye state classification. They evaluated 42 machine learning algorithms on the 14-channel data, with the KStar algorithm achieving the highest performance, reporting a 2.7 % error rate (97.3% accuracy). Their methodology likely relied on static feature extraction, such as power spectral density (PSD), applied to individual epochs.

3 Preprocessing

To prepare the EEG data for modeling, we applied a sequence of standard preprocessing steps to clean the signal.

3.1 Extreme Value Removal

In our data, we found 4 instances out of 14,980 that had extremely large values, orders of magnitude smaller and larger than the maximum and minimum values of the dataset without the values, such as 642,564 compared to an average of 4.000. We removed these extreme values so as that our dataset contained 14, 976 instances, and all of these were within the interval [1000, 4000]

3.2 Common Average Referencing (CAR)

To reduce reference bias across electrodes, we applied Common Average Referencing. For each time point, we subtracted the mean across all channels from each individual channel value. This step helps eliminate global noise that affects all electrodes uniformly. The resulting signal is centered per time point,

3.3 Bandpass Filtering (1–40 Hz)

EEG contains information across many frequencies, but for cognitive state classification, the 1–40 Hz range is the most relevant. We applied a bandpass filter to retain this range while removing irrelevant slow drifts and high-frequency noise.

3.4 Epoching

After cleaning, the continuous EEG signal was segmented into fixed-length epochs of 0.5s to 2.0s. For each epoch, we computed the mean eye state label across all time points, rounding to the nearest binary value. Each epoch was thus assigned a single label (0 for eyes open, 1 for eyes closed).

3.5 Feature Extraction: Power Spectral Density (PSD)

To convert each time-domain epoch into frequency-domain features, we computed the Power Spectral Density (PSD) using Welch's method. This yielded a 2D matrix per epoch: frequency $\times$ channel. We only kept the frequencies below 40 since they fit within the band pass window. This gave us data with shape 20×14, for 20 frequency bins and 14 channels.

4 Models

4.1 GLM

Before implementing more complex models, we used a GLM and a Naive Bayes models as a baseline. We began with a generalized linear model (GLM) – specifically, logistic regression – to classify EEG data into binary eye states Logistic regression is widely used for binary classification tasks and provides a valuable baseline due to its simplicity.

4.1.1 Assumptions

The model makes several assumptions, including linearity and independence of features. Specifically, it assumes that each input features contributes independently to the log-odds, and that the relation-

ship between the features and the eye state is linear after a binary classification. Based on these assumptions, we use the model:

$$P(y = 1 \mid \mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x} + b) \tag{1}$$

Where w is the weights, b is the bias, x is the flattened set of power by channel, y is the attention state, and σ is the sigmoid link function. To compute the liklihood, we use the log-liklihood function using a bernoulli distribution.

$$\log L(\mathbf{w}, b) = \sum_{i=1}^{N} \left[y_i \log \sigma(\mathbf{w}^\top \mathbf{x}_i + b) + (1 - y_i) \log \left(1 - \sigma(\mathbf{w}^\top \mathbf{x}_i + b)\right) \right] \tag{2}$$

4.1.2 Results

Using this model, we obtained 54.9% accuracy. The model could be poor because of the assumptions we made when implementing the model. Our data likely violates key assumptions of logistic regression — namely, linearity and, independence.

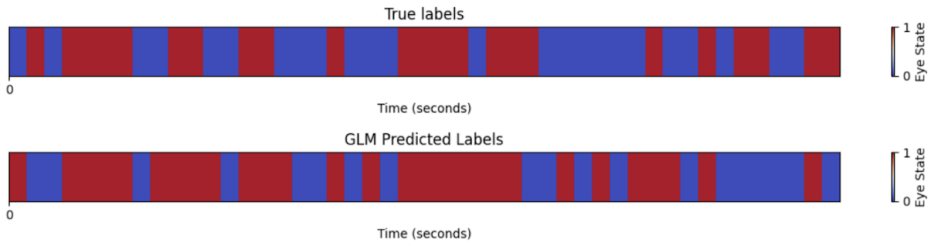

Figure 1: **GLM Sample Test Data** Singular test fold from evaluation of model (Blue and red correspond to closed and open).

EEG data exhibits nonlinear dynamics: the relationship between spectral power and cognitive state is not simply additive or proportional, but depends on complex interactions between frequency bands and time. Additionally, the features are correlated, both across channels (spatial correlation due to neural connectivity) and across time (temporal dependencies within brain states). These correlations violate the assumption that features independently contribute to the log-odds.

While the GLM provides a useful baseline, its limiting assumptions restricts its ability to capture the complicated temporal and spectral patterns of EEG signals. These limitations motivate the use of models that can handle temporal dependencies and nonlinear dynamics, such as HMM's or LSTMs .

4.2 Naive Bayes

We also implemented a Naive Bayes classifier as a second baseline model. Naive Bayes is a probabilistic classifier based on Bayes' Theorem, with the simplifying assumption that features are conditionally independent given the class label. Despite its simplicity, Naive Bayes can perform surprisingly well on high-dimensional data, making it a popular choice for early-stage model evaluation.

4.2.1 Assumptions

The model assumes conditional independence of features and a gaussian distribution for each feature. Given these assumptions, the model estimates the posterior probability of a class label y given input $\mathbf{x} = [x_1, \ldots, x_n]$ as:

$$P(y \mid \mathbf{x}) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y) \tag{3}$$

In our implementation, we assumed that each EEG feature was normally distributed given the eye state, using a Gaussian Naive Bayes formulation. The model is trained by estimating the mean and covariance of each feature within each class, and then using those statistics to compute the likelihood of new samples.

Given input features $\mathbf{x} = [x_1, x_2, \ldots, x_n]$, the Naive Bayes classifier computes the posterior probability of label $y \in 0, 1$ as:

$$P(y \mid \mathbf{x}) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(\mathbf{x})} \tag{4}$$

Since $P(\mathbf{x})$ is constant across classes, classification is done by comparing:

$$\hat{y} = \arg\max_{y \in 0,1} P(y) \prod_{i=1}^{n} P(x_i \mid y) \tag{5}$$

Log-Likelihood Function (for Gaussian Naive Bayes) To avoid multiplying many probabilities, we work with the log-likelihood:

$$\log P(y \mid \mathbf{x}) = \log P(y) + \sum_{i=1}^{n} \log P(x_i \mid y) \tag{6}$$

Assuming that each feature x_i is normally distributed within each class, we have:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_{iy}^2}} \exp\left(-\frac{(x_i - \mu_{iy})^2}{2\sigma_{iy}^2}\right) \tag{7}$$

Plugging this into the log-likelihood:

$$\log P(y \mid \mathbf{x}) = \log P(y) - \sum_{i=1}^{n} \left[\frac{(x_i - \mu_{iy})^2}{2\sigma_{iy}^2} + \log\sqrt{2\pi\sigma_{iy}^2}\right] \tag{8}$$

Where μ_{iy} = mean of feature x_i in class y, σ_{iy}^2 = variance of feature x_i in class y, and $P(y)$ = prior probability of class y.

4.2.2 Results

Using this approach, we achieved a classification accuracy of 58.7%. While slightly better than the GLM, this model still suffered from a number of violated assumptions. Most notably, the EEG features were not conditionally independent — both spatial and temporal correlations existed between electrodes. For example, neural connectivity means that electrodes can be related, and the assumption that each channel has its own independent distribution is very reductive. Additionally, the persistence of the state of eyes open and eyes closed means that there exists a clear temporal dependence between one brain state to another. A model that assumes time- dependence between time - bins will likely perform better because of this.

Despite these limitations, Naive Bayes served as a baseline. However, like the GLM, it cannot model temporal sequences or adapt to nonstationary signal patterns. This further motivates the need for time-aware and nonlinear models such as HMMs and LSTMs, which can leverage temporal structure in the data to improve performance.

4.3 HMM

After evaluating the GLM and Naive Bayes, we decided to move toward a model that could capture the temporal dynamics inherent in EEG data. Both earlier models treated each time point independently, ignoring the fact that eye state transitions—like opening or closing the eyes—tend to happen over

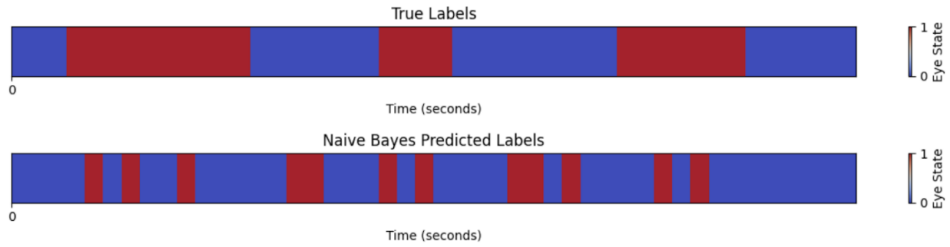

Figure 2: **Naive Bayes Sample Test Data** Singular test fold from evaluation of model (blue and red correspond to closed and open).

sustained periods rather than instantaneously. To address this, we turned to a Gaussian Hidden Markov Model (HMM). We chose the Gaussian variant over the more traditional Poisson HMM because it makes posterior inference more straightforward and aligns naturally with the continuous, multivariate nature of EEG signals. The HMM framework is well-suited for this task because it assumes that observed data are generated from a sequence of latent states, which matches the idea that one's eye state will produce a certain EEG signal, and that the transition between states will produce a certain EEG signal which may indicate a persistent state, creating temporal dynamics that the HMM can use to predict current states. The hidden states not only emit different EEG patterns but also influence what states are likely to follow, allowing the model to capture temporal dependencies that the GLM and Naive Bayes couldn't do this. We implemented both an unsupervised HMM and a supervised HMM. Both use Gaussian emissions to model the continuous EEG features, but they differ in how they treat the label information.

4.3.1 Assumptions

Our model makes several assumptions. First, we assume a first-order Markov chain, so each hidden state depends only on its immediate predecessor. Second, we treat the transition probabilities as stationary throughout the session, even though blink rate and attention may drift over time. Third, we posit that once in a given state, the 14-channel EEG observations are conditionally independent of past data except through that state, and that they follow a Gaussian distribution around a state-specific mean. Finally, we fix the number of hidden states in advance, which may not perfectly match the true complexity of eye-state dynamics. These assumptions simplify inference and let us isolate the value of temporal structure, but they also mean the model can blur slower drifts or violate stationarity if the participant's behavior changes. The model consists of the following components: initial state distribution $P(s_1 = k) = \pi_k$; transition model $P(s_t = j \mid s_{t-1} = i) = A_{ij}$, where A is the state transition matrix; and emission model $P(\mathbf{x}_t \mid s_t = k) = \mathcal{N}(\mathbf{x}_t; \mu_k, \Sigma_k)$, where each hidden state emits observations from a multivariate Gaussian with mean μ_k and covariance Σ_k.

The joint likelihood for the unsupervised model:

$$P(\mathbf{x}_{1:T}, s_{1:T}) = P(s_1) \prod_{t=2}^{T} P(s_t \mid s_{t-1}) \prod_{t=1}^{T} P(\mathbf{x}_t \mid s_t) \tag{9}$$

The joint likelihood for the supervised model:

$$P(\mathbf{x}_{1:T}, y_{1:T}) = P(y_1) \prod_{t=2}^{T} P(y_t \mid y_{t-1}) \prod_{t=1}^{T} P(\mathbf{x}_t \mid y_t) \tag{10}$$

4.3.2 Results

The unsupervised HMM achieved an accuracy of 60.9%, clearly outperforming both Naive Bayes and the GLM. We found this to be quite impressive given that it is after all completely unsupervised. Its edge came from leveraging temporal structure, something the earlier models completely ignored. By modeling how eye states evolve over time, the HMM was better able to smooth over noisy frame-by-frame fluctuations that misled static classifiers.

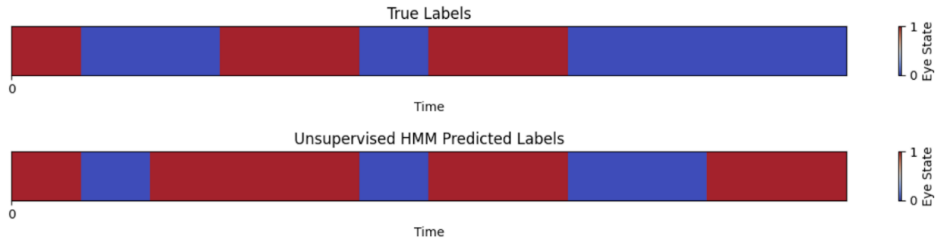

Figure 3: **Unsupervised HMM Sample Test Data** Singular test fold from evaluation of model (Blue and red correspond to closed and open).

To improve on this, we implemented a supervised HMM, which used the true eye-state labels to guide its transition and emission probabilities. With 1-second time bins, it reached 64.4% accuracy, outperforming the unsupervised version and showing that label-informed temporal structure makes a meaningful difference.

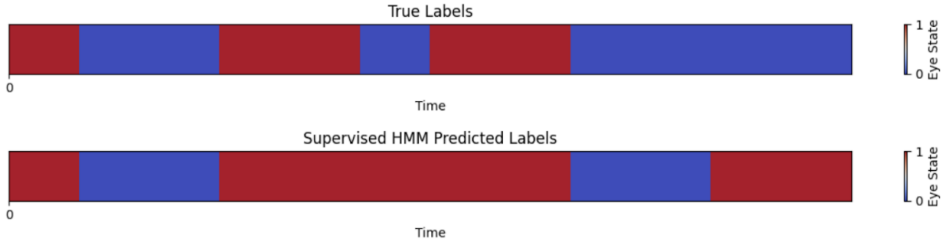

Figure 4: **Supervised 1s Time Bins HMM Sample Test Data** Singular test fold from evaluation of model (Blue and red correspond to closed and open).

We then experimented with 2-second time bins, based on the idea that longer windows could reduce noise further and provide a more stable input to the model. In theory, this should help the HMM better distinguish between states by smoothing over short-term signal spikes. However, the performance dropped slightly to 63.0%. While that's only a small decrease, it highlights a subtle but important tradeoff: the longer bins may have averaged away useful transitions or introduced ambiguity by mixing eye-open and eye-closed signals in a single observation.

So although 2-second binning had potential to improve robustness, in practice it blurred key signal dynamics, slightly hurting performance. The small drop in accuracy emphasizes that even minor shifts in temporal resolution can affect results, and more smoothing isn't always better when it comes to EEG.

4.4 LSTM

We chose to explore an LSTM model next due to its conceptual alignment with KStar, the top-performing classifier in the original study. KStar, a non-parametric, instance-based learner, excels at adapting to local patterns and capturing complex, nonlinear relationships in EEG data. Given that our best static models (logistic regression, Naive Bayes) and HMM-based models underperformed compared to KStar, we selected the LSTM as a modern, scalable alternative to capture flexible temporal patterns. Unlike KStar, which relies on instance similarity without explicit temporal modeling, LSTMs leverage a learnable architecture to model sequential dependencies, making them well-suited for the time-varying nature of EEG signals.

Unlike HMMs, which assume a first-order Markov process where each hidden state depends solely on the immediately preceding state, LSTMs can model dependencies over variable-length time spans without this constraint. In EEG data, particularly during eye state transitions, brain activity often reflects influences from several seconds prior, violating the Markov property. For instance, neural patterns during eye blinks or closures may depend on a history spanning hundreds of samples (at 128 Hz sampling rate). LSTMs, with their gated memory cells, selectively retain and update information across long sequences, enabling them to capture these delayed, non-stationary patterns in EEG signals

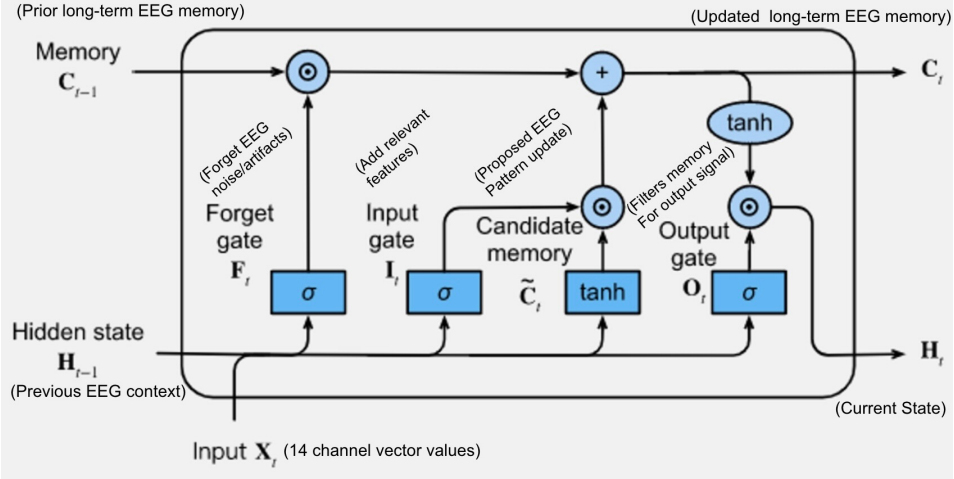

Figure 5: **LSTM cell for EEG classification.** At each time step t, the 14-channel input X_t is first passed through the *forget gate* F_t to discard noise, then through the *input gate* I_t and candidate memory $\tilde{C}_t$ to update the long-term cell state C_t, and finally through the *output gate* O_t to produce the short-term hidden state H_t, which drives the eye-state prediction.

(Adapted from `http://d2l.ai/chapter_recurrent-modern/lstm.html` *[6]).*

more effectively than HMMs. The following equations describe the update mechanism within an LSTM cell:

$$f_t = \sigma(W_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_f) \tag{11}$$

$$i_t = \sigma(W_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_i) \tag{12}$$

$$\tilde{C}_t = \tanh(W_C[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_C) \tag{13}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{14}$$

$$o_t = \sigma(W_o[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_o) \tag{15}$$

$$\mathbf{h}_t = o_t \odot \tanh(C_t) \tag{16}$$

Where $\mathbf{x}_t$ is the input EEG feature vector at time t, $\mathbf{h}_{t-1}$ is the previous hidden state, C_{t-1} is the previous cell state, f_t, i_t, and o_t are the forget, input, and output gates, respectively, $\tilde{C}_t$ is the candidate memory content, W_* and b_* represent learnable weights and biases, $\sigma(\cdot)$ is the sigmoid activation, $\tanh(\cdot)$ is the hyperbolic tangent, and $\odot$ denotes element-wise multiplication.

4.4.1 Assumptions

Let us now acknowledge the assumptions implicit in our LSTM approach. We assume that our single-subject dataset – though relatively small – is nevertheless large enough for the network to learn meaningful temporal features without overfitting. We also rely on uniform time bins (e.g. 1 s windows) so that each step aligns consistently across sequences, even though true EEG rhythms can drift. The model presumes that the training and test distributions share the same non-stationary dynamics, so that patterns the LSTM learns will generalize rather than simply memorize session-specific noise. Finally, we count on the gating mechanisms to filter out transient artefacts, accepting that very abrupt or rare bursts may still confuse the network. These assumptions trade off interpretability and theoretical guarantees for the practical flexibility needed to model EEG's complex, long-range dependencies.

4.4.2 Results

In conclusion, our LSTM model achieved the highest accuracy among our models at 70.6%, surpassing logistic regression, Naive Bayes, and both unsupervised and supervised HMMs (1s and 2s time bins). The LSTM's ability to learn long-range dependencies, adapt to EEG's non-stationary dynamics, and mitigate noise without rigid state transition assumptions drove its success. Similar to KStar's

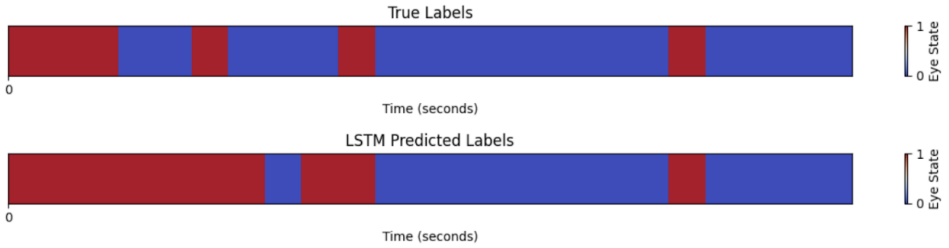

Figure 6: **LSTM Sample Test Data** Singular test fold from evaluation of model (Blue and red correspond to closed and open).

flexibility in handling irregular, nonlinear patterns, the LSTM's capacity to model complex temporal sequences proved critical for decoding eye state transitions from EEG data.

5 Conclusion

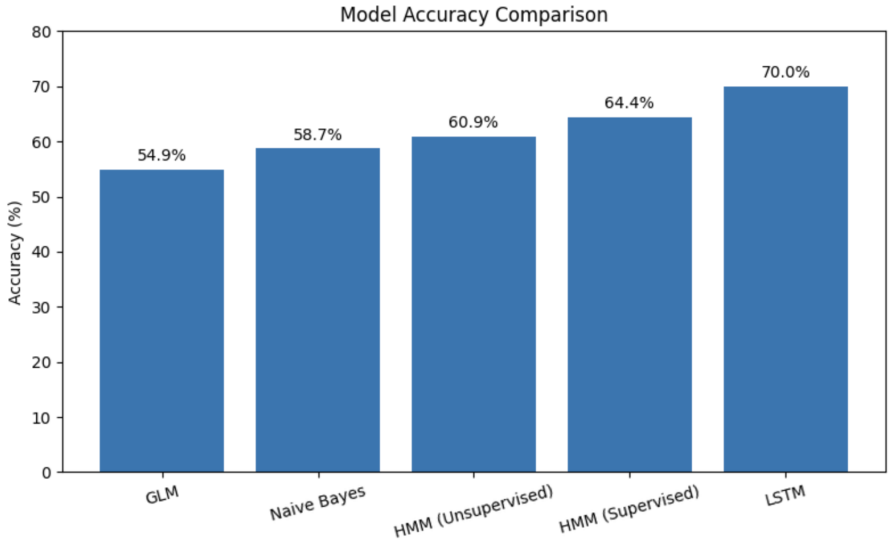

Figure 7: **Compiled Results**

Our experiments compared five models of increasing complexity to decode eye state from EEG data. The GLM and Naive Bayes baselines performed the worst, limited by assumptions of independence and static structure. The unsupervised HMM improved substantially by modeling temporal sequences, while the supervised HMM further boosted performance by aligning state transitions with labeled data. Ultimately, the LSTM achieved the highest accuracy at 70.6%, thanks to its ability to learn flexible, long-range dependencies in noisy EEG signals. These results emphasize the importance of temporal modeling in capturing the dynamic patterns embedded in neural recordings.

Future work could extend this analysis to more complex EEG datasets involving richer behavioral or cognitive tasks beyond simple eye state detection. Additionally, the limitations of EEG – particularly its susceptibility to artifacts and low spatial resolution – suggest the value in exploring complementary neuroimaging modalities such as fNIRS, MEG, or even non-invasive fMRI protocols where available. Hybrid approaches combining multiple physiological signals (e.g. EEG and eye tracking) may also enhance robustness.

References

[1] Hans Berger. Über das elektroenzephalogramm des menschen. sechste mitteilung. *Archiv für Psychiatrie und Nervenkrankheiten*, 99:555–574, 1933.

[2] Alexandre Défossez, Charlotte Caucheteux, Jérôme Rapin, et al. Decoding speech perception from non-invasive brain recordings. *Nature Machine Intelligence*, 5:1097–1107, 2023.

[3] Pradeep Kaushik, Andrew Moye, Mark van Vugt, et al. Decoding the cognitive states of attention and distraction in a real-life setting using eeg. *Scientific Reports*, 12(1):20649, 2022.

[4] Shitao Lai, Li Yao, and Yu Wang. Sadnet: Sustained attention decoding in a driving task by self-attention convolutional neural network. *Brain-Apparatus Communication: A Journal of Bacomics*, 3(1), 2024.

[5] Oliver Roesler and David Suendermann. A first step towards eye state prediction using eeg. In *Proceedings of the Artificial Intelligence in Human Language and Speech (AIHLS)*, Istanbul, Turkey, 2013.

[6] Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. Recurrent neural networks (rnn)—lstm. `https://d2l.ai/chapter_recurrent-modern/lstm.html`, 2023. Accessed: 2025-06-09.