Hackathon Report — Hangman using Reinforcement Learning and HMM

TEAM NO : 01

TEAM MEMBERS: SMRITHI AS – PES1UG23AM306
TANUU SHREE M – PES1UG23AM336
SURABHI M – PES1UG23AM325
GAHNAVI B – PES1UG23AM900

Key Observations

- The most challenging aspect was designing an efficient reward system that balancespartial word progress and incorrect guesses.
- Integrating HMM with RL required careful synchronization — HMM predictions needed tocomplement the RL agent's exploration.
- Context-aware letter prediction (via HMM) improved convergence speed compared to apurely random RL agent.
- Corpus quality impacted generalization — short, frequent words were easier to learn.
- Reward shaping and epsilon decay tuning were key to improved stability and accuracy.

Strategies

Hidden Markov Model (HMM) Design

- States represented hidden letters in the target word.
- Observations represented guessed letters and correctness.
- Transition probabilities modeled positional dependencies.
- Emission probabilities mapped guessed letters to potential matches.
- HMM provided a probabilistic prior over next letter predictions.

Reinforcement Learning (RL) Design

- State: word pattern, guessed letters, remaining attempts, normalized word length.
- Actions: 26 English letters.
- Reward Function:
  +10 for correct letter
  -5 for incorrect letter
+50 for correct word

-10 for loss
• These choices promoted efficient learning, balancing exploration and goal success.

Exploration vs. Exploitation

• Managed using an epsilon-greedy policy.
• ε decayed from 1.0 → 0.05 as learning progressed.
• Early exploration captured general letter patterns; later exploitation focused on learnedstrategies.
• Random exploration persisted to avoid local optima.
• Experience Replay stabilized training by reusing diverse past experiences.

Future Improvements

• Integrate Transformer-based word prediction (e.g., BERT).
• Curriculum learning: gradually increase word complexity.
• Reward shaping with entropy regularization.
• Interactive UI for human-vs-agent gameplay.
• Ensemble models combining RL, HMM, and statistical heuristics.

Conclusion

Combining probabilistic reasoning (HMM) with goal-driven Reinforcement Learning enabled the agent to learn efficient letter prediction strategies in Hangman. The hybrid model demonstrated measurable improvement in accuracy and success rate, proving the potential of merging structured probability models with adaptive learning agents.

GITHUB LINK: https://github.com/Smrithi2005/ML-Hackathon