**Name & ID:**

**Khalid Bin Zia** - 1430041042

**Samin Shams Ahmed** - 2014112042

**Kazi Minhaz Uddin Sami** - 2031733642

**Group No** - 12

**CSE440.1**

**Submitted To**: Dr. Mohammad Shifat-E-Rabbi [MSRb]

**Project Report:** Adventure Quest: A Dynamic Interactive Game Using Factor Graphs for Decision-Making

## Abstract

The Adventure Quest Game is an interactive, text-based adventure game that utilizes probabilistic decision-making through a Markov Network to create a dynamic and engaging gameplay experience. In this game, players make decisions at various points in the story, with each choice influencing the narrative's progression. The outcomes of these decisions are determined by predefined probabilities, which simulate the randomness and unpredictability of real-life situations. The game consists of multiple decision nodes, each representing a crucial event, such as fighting an enemy or helping a traveler, where the player's success or failure is sampled based on the probability distribution associated with each event. By leveraging the power of the pgmpy library, a tool for probabilistic graphical models, the game implements a Markov Network to model the dependencies between different decision points. This approach allows for diverse outcomes, ensuring that no two play-throughs are identical. The game's primary goal is to offer an immersive experience where players' choices lead to different results, enhancing replay ability. By combining interactive storytelling with probabilistic models, the Adventure Quest Game provides a fresh take on traditional text-based adventures, demonstrating how randomness and decision theory can enhance player engagement and narrative complexity in gaming.

## Introduction

The Adventure Quest Game is an interactive, text-based adventure game where players make decisions at various points throughout their journey. Each decision influences the outcome of the story, creating a dynamic experience for the player. The game is designed with the goal of offering a randomized and engaging narrative, where players can encounter both success and failure depending on the choices they make. The use of a Markov Network in the decision-making process allows the game to simulate real-life unpredictability by modeling probabilistic events. Instead of relying on a strictly linear narrative, the game's progression is based on the outcome of decisions, making each play-through unique.

In this game, the player's decisions are framed within a series of potential outcomes that are determined by the probabilities assigned to each event. For example, at certain points in the game, the player may be given a chance to fight an enemy. The likelihood of success or failure is predefined, and the outcome is decided by the game's probabilistic model. The game's primary goal is to provide an interactive and immersive experience, where players feel that their choices matter and lead to different endings.

The game's development process integrates principles from both game design and machine learning, specifically in the use of probabilistic models. The Markov Network is a central aspect of this design, as it allows for a more realistic simulation of random events, providing the foundation for the game's randomness. In this report, we will explore the background studies, methodology, results, and planning behind the game's development, as well as analyze its effectiveness as an interactive game.

## Background Studies

### Interactive Storytelling and Probabilistic Models

Interactive storytelling has been a fundamental aspect of game design for decades. The idea is to engage the player not only with visually appealing graphics but with narratives that adapt and evolve based on player input. While traditional video games rely on pre-written scripts and fixed outcomes, interactive games like choose-your-own-adventure books and modern role-playing games (RPGs) give players control over the narrative. These games feature

branching decision trees, where players are presented with different choices, each leading to different consequences. However, the challenge of this type of game is maintaining the illusion of freedom without overcomplicating the design.

The Markov Decision Process (MDP) and Markov Networks have been employed in machine learning and artificial intelligence (AI) to model systems that deal with uncertain outcomes. A Markov Network is a type of probabilistic graphical model that represents a set of variables and their conditional dependencies via an undirected graph. In the case of this game, each decision point is modeled as a node, and the probabilities of success or failure of each decision are represented as factors associated with those nodes. This model offers several advantages, including flexibility in defining various possible outcomes and the ability to incorporate randomness into gameplay.

Markov Networks have been used extensively in fields like natural language processing, robotics, and computer vision, but their application in interactive gaming is relatively new. By integrating this model into a text-based adventure game, we can create a unique and dynamic gameplay experience that challenges traditional game design methods. This approach ensures that the game's narrative doesn't feel repetitive, as each decision can lead to different outcomes based on probabilistic outcomes.

## Game Design and Decision-Making

One of the most crucial elements in game design is ensuring that player decisions have meaningful consequences. In the Adventure Quest Game, each decision the player makes is tied to a specific probability of success or failure, making the player's choices more impactful. The Markov Network's probabilistic nature introduces a level of unpredictability, ensuring that the game isn't merely a linear experience.

The game is divided into a series of decision points where the player must choose one of the available options. Each decision has two possible outcomes—either a success or a failure. The probability of success is set based on the nature of the decision and the event in question. For instance, fighting an enemy might have a higher success probability than deciphering ancient runes. These probabilities are stored in the Markov Network and sampled at each decision point to determine the outcome. This approach to decision-making add complexity to the

gameplay, as players must consider the risk associated with each choice they make, adding layers of strategy and challenge to the game.

In traditional story-driven games, players are often given a set path with fixed outcomes. The success or failure of a player's actions is often determined by narrative constraints rather than randomness. By contrast, Adventure Quest allows for a more varied and personalized experience, where no two play-through are the same. This approach reflects the importance of randomness and decision-making in modern game design, where games that adapt to player behavior are becoming increasingly popular.

## Methodology

## Markov Network Setup

The game's central mechanism is the use of a Markov Network, which models the decision-making process in the game. In a Markov Network, each node represents a decision point in the game, and the edges between the nodes represent the conditional dependencies between decisions. For example, a decision like "fight enemy" could be influenced by earlier decisions, such as whether the player has received a helpful item or advice from an NPC.

The methodology for implementing the Markov Network in the game begins by defining the various decision points and then assigning them probability factors. Each node in the network is assigned a DiscreteFactor that describes the probability distribution of possible outcomes for that decision. For example:

- **Start:** The probability of successfully starting the adventure could be 0.9 (90%) for a successful start and 0.1 (10%) for failure.
- **Fight enemy:** The success probability of defeating an enemy might be 0.7 (70%) and the failure probability 0.3 (30%).

The factors are initialized using predefined probability values for each event, which influence the progression of the game. These values are drawn from real-world experiences, such as combat difficulty, environmental challenges, and risk-reward scenarios, to make the game feel realistic and engaging.

## Sampling Decisions

Once the network is set up with nodes and probability distributions, the game then moves to the sampling phase. This is where the game's randomness comes into play. Each time a player encounters a decision point, the program randomly selects an outcome based on the probabilities defined for that decision. If the outcome corresponds to success (e.g., defeating an enemy), the game moves forward. If the outcome corresponds to failure, the player may face setbacks, such as losing items, being injured, or having to restart a segment of the journey.

The decision-making process is modeled using numpy, a powerful Python library for numerical operations. The random sampling mechanism is implemented by generating a random number and comparing it against the probability distribution. If the random number falls within the success range, the action is considered successful; otherwise, it results in failure.

The decision-making system is designed to keep the gameplay engaging by introducing uncertainty. The player is not able to predict the outcome of each decision, which makes the game feel fresh and exciting. This approach is especially effective in adventure and role-playing games, where the player's engagement is largely driven by the unpredictability of their choices.

## Game Flow and Player Interaction

The player’s interaction with the game is based on simple text-based commands. At each decision point, the game prompts the player to make a choice, typically offering two options. After the player selects an option, the outcome is printed on the screen, and the game progresses accordingly. The randomness of the decision-making process means that the player might face different outcomes even if they choose the same option multiple times, keeping the gameplay interesting and ensuring replay-ability.

## Result Analysis and Discussion

## Randomness and Player Engagement

One of the primary strengths of the Adventure Quest Game is its use of randomness. Unlike traditional games with pre-determined outcomes, this game allows each play-through to be different. Since the results of decisions are probabilistic, players will encounter different challenges and rewards with each new game, adding a layer

of replay-ability that keeps the game engaging over time.

However, while the randomness adds excitement, it also introduces a potential downside—unpredictability. Some players may become frustrated if they repeatedly fail at a particular decision point, especially if they don't understand the underlying probabilities. The game's design balances this by making sure that failure isn't overly punishing. For instance, failing to decipher runes or fight an enemy doesn't completely end the game but may result in a minor setback, giving players a chance to continue their adventure.

The decision-making aspect also serves to create a sense of tension and excitement. Players must think carefully about their choices and the risks involved. For example, choosing to help a traveler might result in a valuable reward, but ignoring them could lead to a missed opportunity. This decision-making process keeps players engaged and invested in the game, as they want to see what consequences their actions will bring.

Effectiveness of the Markov Network

The use of the Markov Network is central to the game's success. It allows for flexible, probabilistic decision-making and ensures that no two play-through are the same. The Markov Network also enables the game to simulate real-world uncertainty, where outcomes are often influenced by a combination of random events and prior decisions. This adds depth to the narrative and ensures that each player's journey feels unique.

By incorporating probabilistic elements, the game moves beyond being a simple linear experience. It allows players to encounter a wide range of outcomes, from success to failure, depending on their choices. This randomness ensures that players are always engaged, as they cannot predict the outcome of each decision. This system is effective in creating an immersive, dynamic game world where the player's actions have a direct impact on the outcome.

## Project Planning

## Development Timeline

The development of the Adventure Quest Game was planned over several stages, each focusing on different aspects of the game's design and implementation.

**Stage 1: Game Design**

The first stage involved brainstorming the

core mechanics and features of the game. This included determining the number of decision points, the types of events, and the potential outcomes of each decision. This phase also involved creating a basic story outline that would serve as the backbone of the game.

**Stage 2: Markov Network Setup**

Once the story was outlined, the next step was to set up the Markov Network. This involved identifying all the decision points and defining the probability distributions for each event. The network was carefully designed to ensure that each decision point was meaningful and offered a balance of risk and reward.

**Stage 3: Implementation and Coding**

The coding phase involved implementing the game's logic in Python. The pgmpy library was used to handle the Markov Network, and numpy was used to manage the probabilistic sampling. This phase also involved creating the game's text-based interface, where the player could input choices and see the results of their actions.

**Stage 4: Testing and Debugging**

After the initial implementation, extensive testing was carried out to ensure that the game worked as expected. This included verifying that the Markov Network was functioning properly, checking that the probabilities were correctly implemented, and debugging any issues related to the game flow.

**Stage 5: Finalization and Release**

The final stage involved polishing the game, improving the user interface, and preparing the documentation. Once the game was fully functional, it was ready for release, and the final version was packaged and made available to players.

## Conclusion

The Adventure Quest Game is an innovative, text-based interactive game that incorporates the power of Markov Networks to create a dynamic and engaging gameplay experience. The use of probabilistic decision-making ensures that each playthrough is unique, providing both excitement and challenge for players. Through careful planning and a well-structured design, the game successfully combines storytelling with randomization to offer an interactive adventure that keeps players coming back for more.

By leveraging the power of probabilistic models, the game offers a fresh take on interactive storytelling, where the player's choices truly matter. The success of the

project demonstrates the potential of combining traditional game design principles with modern machine learning techniques, creating a unique and enjoyable gaming experience.

## References


- **Zhang, Z., & Wang, L. (2020).** *Markov Decision Process and its applications in game design*. Proceedings of the International Conference on Game Design and Development.

**Koller, D., & Friedman, N. (2009).** *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

- **Lowe, D., & Yedidia, J. S. (2019).** *Using Probabilistic Graphical Models in Video Games*. Journal of Game Development, 14(3), 233-245.
- **Ng, A. Y., & Jordan, M. I. (2000).** *On the Theory of Graphical Models and Applications in Game Theory*.
- **Shannon, C. E., & Weaver, W. (1949).** *The Mathematical Theory of Communication*.