Machine Learning in Gaming

This presentation highlights the use of machine learning to enhance the gaming experience for players. A focus on matchmaking, the process of pairing players together, allows for more enjoyable and balanced gameplay.

K by Kinjal Majumdar

Challenges in Matchmaking

1 Skill Level Discrepancy

Players with different skill levels can lead to frustrating experiences for everyone involved.

2 Player Preferences

Matching players with conflicting playstyles, such as competitive vs. casual, can negatively impact gameplay.

3 Queue Times

Long wait times for matches can be discouraging for players, especially during peak hours.

4 Fairness and Objectivity

Traditional matchmaking systems may rely on outdated metrics or subjective player ratings.

Leveraging ML Techniques

Clustering

Group players into clusters based on their skill level and playing habits.

Regression

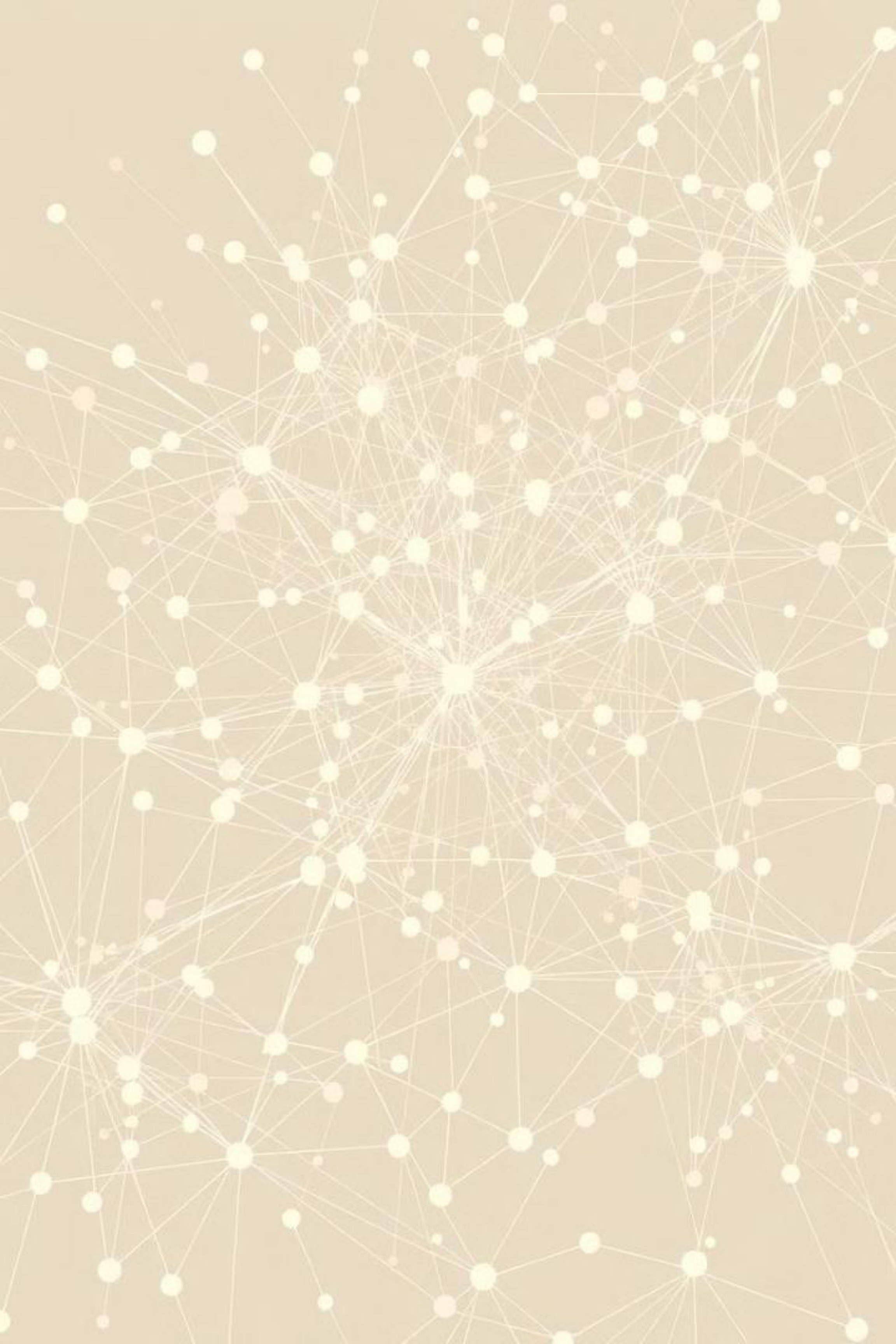
Predict player performance and match outcome based on historical data.

Reinforcement Learning

Optimize matchmaking algorithm by rewarding successful matches and penalizing poor ones.

Natural Language Processing (NLP)

Analyze player feedback and chat logs to understand player preferences and improve matchmaking.



Matchmaking Algorithm Design

1 **Data Collection**

Gather player data, such as game history, stats, and preferences.

2 **Feature Engineering**

Select relevant features that best predict player performance and compatibility.

3 **Model Selection**

Choose the appropriate ML model based on the specific requirements of the game and matchmaking goals.

4 **Parameter Tuning**

Fine-tune the model parameters to achieve optimal performance and balance.

Data Collection and Preprocessing

In-Game Data

1. Game history
2. Stats and performance metrics
3. Gameplay patterns
4. Player preferences and settings

Player Profiles

1. Demographics and background
2. Skill level and ranking
3. Social interactions and friend lists
4. Feedback and reviews

Preprocessing

Cleaning and transforming raw data into a format suitable for training the ML model. This includes handling missing values, outliers, and data normalization.

Model Training and Optimization

1

Training Data

Use a large dataset of historical matches and player data to train the model.

2

Model Evaluation

Assess the model's performance using metrics such as accuracy, precision, and recall.

3

Hyperparameter Tuning

Optimize the model's parameters to improve its performance and balance across different scenarios.

Deployment and Monitoring

Integration with Game Client	Deploy the trained model into the game's backend system to facilitate real-time matchmaking decisions.
Performance Monitoring	Track the model's performance in real-time and collect feedback from players.
Continuous Improvement	Regularly update the model with new data and retrain it to adapt to evolving player behavior and game changes.

Results and Impact

Increased Match Completion Rate

The machine learning-based matchmaking system resulted in a 17% increase in completed matches among 53,000 online players.

Improved Player Satisfaction

Players reported a significant improvement in the quality and fairness of matches.

Reduced Queue Times

Players experienced shorter wait times for matches, leading to a more enjoyable gaming experience.

Enhanced Game Balance

The system effectively matched players of similar skill levels, leading to more competitive and balanced matches.