Uso de Autoencoders e Clustering para Análise de Jogos de Dota 2

Gustavo F. Ceccon[1]

[1] Universidade Estadual Paulista "Júlio de Mesquita Filho"
Caixa Postal 13506-692 – (19) 3526-9000 – Rio Claro – SP – Brasil

gustavo.ceccon@unesp.br

Abstract. *This article explores the application of autoencoder techniques and clustering methods to identify similar professional matches in Dota 2, one of the main games in the MOBA (Multiplayer Online Battle Arena) genre. Using public match data, the paper proposes an approach based on dimensionality reduction and unsupervised clustering to analyze patterns in hero compositions, economic performance and objectives achieved. The method employs Deep Embedded Clustering (DEC) to simultaneously integrate latent representation learning and clustering, demonstrating the relevance of these techniques for eSports analysis and presenting an overview of the most effective methodologies for clustering similar matches.*

Resumo. *Este artigo explora a aplicação de técnicas de autoencoders e métodos de clustering para identificar semelhanças entre partidas profissionais de Dota 2, um dos principais jogos do gênero MOBA (Multiplayer Online Battle Arena). Utilizando dados públicos de partidas, o trabalho propõe uma abordagem baseada em redução de dimensionalidade e agrupamento não supervisionado para analisar padrões em composições de heróis, desempenho econômico e objetivos alcançados. O método emprega Deep Embedded Clustering (DEC) para integrar simultaneamente a aprendizagem de representações latentes e a formação de clusters, demonstrando a relevância dessas técnicas para análise de eSports e apresentando um panorama das metodologias mais eficazes para agrupamento de partidas similares.*

1. Introdução

A análise de dados em eSports tem experimentado crescimento devido à riqueza e complexidade dos dados gerados em jogos como Dota 2. O cenário competitivo envolve não apenas jogadores e equipes, mas também analistas, comentaristas e plataformas de estatísticas que buscam prever resultados, identificar padrões estratégicos e fornecer ideias táticas para melhorar o desempenho [Drachen and Schubert 2016]. Empresas bilionárias patrocinam não só os eventos, mas na área de inteligência artificial, da área de tecnologia como Nvidia, desde 2011 quando o jogo estava surgindo. Outras grandes empresas envolvem alimentícia, como a Monster Energy e Red Bull, móveis como Secretlab e aposta como 1Bet e GG.Bet, além do patrocino dos jogadores para a premiação. Milhões de espectadores e premiações milionárias, tem impulsionado a necessidade de métodos analíticos sofisticados para compreender a dinâmica complexa desses jogos [Costa et al. 2023].

Estudos recentes demonstram que a aplicação de aprendizado de máquina em Dota 2 tem sido utilizada para diversas tarefas, incluindo classificação de papéis de jogadores [Eggert et al. 2015], predição de eventos críticos como mortes de heróis [Katona et al. 2019], análise de composições de equipes [Cadman 2024] e detecção de encontros táticos [Schubert et al. 2016]. No entanto, existe uma lacuna na literatura quanto à análise de similaridade entre partidas inteiras, especialmente utilizando técnicas avançadas de autoencoders e clustering profundo, que podem capturar relações não-lineares complexas nos dados de jogos.

O problema central deste trabalho é identificar partidas similares entre si, considerando múltiplas variáveis categóricas e numéricas, com o objetivo de apoiar análises táticas e históricas. Dota 2, como um jogo MOBA complexo, apresenta características únicas que tornam essa análise desafiadora: mais de 120 heróis únicos, milhares de combinações de itens, estratégias emergentes e meta-jogos em constante evolução [**?**]. A identificação de padrões de similaridade entre partidas pode revelar tendências estratégicas, auxiliar na preparação de equipes e fornecer planos valiosos para análise pós-jogo.

A solução proposta envolve a extração de características relevantes das partidas, pré-processamento dos dados, aplicação de técnicas de redução de dimensionalidade usando autoencoders e agrupamento não supervisionado, seguido de análise supervisionada para validação dos clusters formados. O foco está em jogos profissionais, especialmente partidas de campeonatos como The International, utilizando dados compreendendo o período de 2021 a 2024, capturando assim diferentes versões do jogo e evoluções do meta.

2. Fundamentação Teórica

2.1. Análise de eSports e Jogos MOBA

A análise de dados em eSports, particularmente em jogos MOBA, representa um campo emergente que combina elementos de ciência da computação, estatística e estudos de jogos [Drachen and Schubert 2016]. MOBAs como Dota 2 geram grandes volumes de dados telemétricos durante cada partida, incluindo posições de jogadores, uso de habilidades, economia do jogo, e interações complexas entre elementos do jogo [Kamal et al. 2025]. Esta riqueza de dados oferece oportunidades únicas para aplicação de técnicas de aprendizado de máquina e mineração de dados.

Empresas como SAP e a própria desenvolvedora Valve já fazem estatísticas e análises de jogos de jogos em tempo real. A OpenAI, explorou a área de agentes (robôs) [OpenAI 2017], e testaram contra jogadores profissionais no maior torneio, The International [OpenAI 2018]. Dota plus é uma ferramenta que pode ser adquirida no jogo e mostra estatísticas de decisões e vantagens para o jogo. Durante o jogo os analistas conseguem ver a predição de vitória e derrota, ferramenta disponível já implementada dentro do jogo.

Trabalhos recentes em análises de eSports têm focado em diferentes aspectos: [Schubert et al. 2016] desenvolveram métodos para detecção automática de encontros em Dota 2, demonstrando como eventos de combate podem ser identificados e analisados; [Costa et al. 2023] realizaram um mapeamento sistemático da literatura sobre inteligência

artificial em jogos MOBA, identificando tendências e lacunas de pesquisa; e [Ijäs 2021] explorou análises espaciais em jogos competitivos, mostrando como a análise de posicionamento pode revelar padrões estratégicos.

2.2. Autoencoders: Redução de Dimensionalidade e Extração de Características

Autoencoders são uma classe de redes neurais artificiais projetadas para aprendizagem de representações não supervisionada, introduzidas inicialmente por [Rumelhart et al. 1986] como parte do desenvolvimento de redes neurais e [LeCun et al. 1988] para a retropropagação. Matematicamente, um autoencoder consiste em duas funções principais: um encoder f_θ que mapeia dados de entrada $\mathbf{x} \in \mathbb{R}^d$ para uma representação latente $\mathbf{h} \in \mathbb{R}^k$ (onde tipicamente $k < d$), e um decoder g_ϕ que reconstrói os dados originais a partir da representação latente.

A função de encoder é definida como:

$$\mathbf{h} = f_\theta(\mathbf{x}) = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{1}$$

onde $\mathbf{W} \in \mathbb{R}^{k \times d}$ é a matriz de pesos, $\mathbf{b} \in \mathbb{R}^k$ é o vetor de bias, e σ é uma função de ativação não-linear. O decoder é definido analogamente como:

$$\hat{\mathbf{x}} = g_\phi(\mathbf{h}) = \sigma'(\mathbf{W}'\mathbf{h} + \mathbf{b}') \tag{2}$$

O objetivo do treinamento é minimizar a função de perda de reconstrução:

$$\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \tag{3}$$

Diferentes variações de autoencoders foram desenvolvidas para aplicações específicas, incluindo Autoencoders Variacionais (VAE) para modelagem probabilística, (SAE) Autoencoders Esparsos para aprendizagem de características esparsas, e (CAE) Autoencoders de Remoção de Ruído para robustez a ruído [Baldi 2012, Le 2015].

2.3. Clustering: Algoritmos Clássicos e Aprendizagem Profunda

O agrupamento (clustering) é uma tarefa fundamental em aprendizado não supervisionado que busca particionar dados em grupos homogêneos. O algoritmo k-médias, proposto por [MacQueen 1967] e formalizado por [Lloyd 1982], permanece como um dos métodos mais utilizados. O k-médias minimiza a soma dos quadrados intra-grupo:

$$J = \sum_{i=1}^{n} \sum_{j=1}^{k} w_{ij} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 \tag{4}$$

onde w_{ij} é um indicador binário de atribuição do ponto i ao grupo j, e $\boldsymbol{\mu}_j$ é o centroide do grupo j.

Métodos baseados em densidade, como HDBSCAN [Campello et al. 2013], oferecem vantagens para dados com grupos de formas irregulares e densidades variáveis, sendo especialmente úteis quando o número de grupos não é conhecido a priori. O

HDBSCAN constrói uma hierarquia de grupos baseada em estimativas de densidade, permitindo a identificação automática do número apropriado de grupos.

A integração de aprendizagem profunda com agrupamento tem levado ao desenvolvimento de métodos como Agrupamento Profundo Incorporado (DEC), que realiza simultaneamente a aprendizagem de representações e o agrupamento [Xie et al. 2016]. Esta abordagem supera limitações dos métodos tradicionais em dados de alta dimensionalidade, como é comum em aplicações de análise de jogos.

3. Metodologia

3.1. Conjunto de Dados e Pré-processamento

O conjunto de dados utilizado neste estudo compreende partidas profissionais de Dota 2 obtidas através da API pública do OpenDota (`https://docs.opendota.com`) e bases de dados disponíveis no Kaggle, abrangendo o período de 2013 a 2024. A porção selecionada foi de 2021 a 2024, onde a discrepância de mapa e mudanças é menor e o conjunto está mais refinado. O jogo cresceu significativamente em complexidade, com a adição de novos heróis, itens e mecânicas de jogo. Porém o mais impactante são as mudanças de mapa e heróis, que afetam diretamente a estratégia e o meta do jogo.

As variáveis extraídas para análise incluem:

- **Composição de heróis**: Heróis escolhidos e banidos por cada equipe
- **Características de heróis**: Categoria, atributos principais, papel
- **Métricas econômicas**: Ouro Por Minuto (GPM), Experiência Por Minuto (XPM), valor líquido final
- **Desempenho individual**: Proporção Abates/Mortes/Assistências (KDA), dano causado, dano em estrutura, cura realizada
- **Controle de visão**: Uso de sentinelas observadoras, sentinelas sensitivas
- **Objetivos estratégicos**: Torres destruídas, abates do Roshan
- **Duração** : Tempo total de partida

O pré-processamento dos dados seguiu várias etapas críticas. Primeiro, foi realizada limpeza para remoção de partidas incompletas ou com dados inconsistentes. Em seguida, variáveis categóricas como heróis e itens foram convertidas utilizando codificação binária (one-hot), resultando em vetores esparsos de alta dimensionalidade. Para variáveis numéricas, aplicou-se normalização z-escore para garantir que diferentes escalas não introduzissem viés no processo de agrupamento.

Uma contribuição metodológica importante foi o desenvolvimento de um sistema de codificação para capturar interações entre heróis, reconhecendo que certas combinações têm sinergias específicas que não são capturadas por análise individual. Esta codificação preserva informações sobre seleções completas, incluindo ordem de escolhas e banimentos, que são estrategicamente relevantes em jogos profissionais.

3.2. Arquitetura do Autoencoder

A arquitetura do autoencoder foi projetada especificamente para lidar com a natureza esparsa e de alta dimensionalidade dos dados de Dota 2. O codificador consiste em múltiplas camadas densamente conectadas com ativação ReLU, progressivamente reduzindo a dimensionalidade:

- Camada de entrada: dimensão original dos dados ($\sim$2000 características)
- Camada oculta 1: 1024 neurônios + Abandono (0.2)
- Camada oculta 2: 512 neurônios + Abandono (0.2)
- Camada latente: 128 neurônios (representação comprimida)

O decodificador espelha esta arquitetura, reconstruindo gradualmente os dados originais. A função de perda combina erro de reconstrução quadrático com regularização L2 para prevenir overfitting:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{reconstruction}} + \lambda \|\theta\|^2 \tag{5}$$

onde λ é o coeficiente de regularização ajustado através de validação cruzada.

3.3. Agrupamento Profundo Incorporado (DEC)

O método DEC foi escolhido por sua capacidade de integrar o processo de agrupamento ao treinamento do autoencoder, otimizando simultaneamente a qualidade da representação latente e a coesão dos grupos [Xie et al. 2016]. O algoritmo DEC opera em duas fases principais:

Fase 1 - Pré-treinamento: O autoencoder é treinado independentemente para aprender uma boa representação dos dados, minimizando apenas a função de reconstrução. Esta fase é crucial para inicializar o espaço latente com características significativas.

Fase 2 - Refinamento conjunto: O agrupamento é integrado ao processo de treinamento através de uma distribuição alvo auxiliar. Para cada ponto de dado $\mathbf{x}_i$ na representação latente, calcula-se a probabilidade de pertencer ao grupo j usando uma distribuição t-Student:

$$q_{ij} = \frac{(1 + \|\mathbf{z}_i - \boldsymbol{\mu}_j\|^2/\alpha)^{-(\alpha+1)/2}}{\sum_{j'}(1 + \|\mathbf{z}_i - \boldsymbol{\mu}_{j'}\|^2/\alpha)^{-(\alpha+1)/2}} \tag{6}$$

onde $\mathbf{z}_i = f_\theta(\mathbf{x}_i)$ é a representação latente, $\boldsymbol{\mu}_j$ são os centroides dos grupos, e α são os graus de liberdade (configurado para $\alpha = 1$).

A distribuição alvo P é computada para enfatizar predições de alta confiança:

$$p_{ij} = \frac{q_{ij}^2/f_j}{\sum_{j'}(q_{ij'}^2/f_{j'})} \tag{7}$$

onde $f_j = \sum_i q_{ij}$ são as frequências dos grupos. O objetivo de agrupamento é minimizar a divergência KL entre Q e P:

$$\mathcal{L}_{\text{KL}} = \text{KL}(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{8}$$

4. Experimentos e Resultados

4.1. Hipóteses de Avaliação

4.2. Métricas de Avaliação

A avaliação da qualidade do agrupamento foi realizada usando múltiplas métricas complementares:

- **Índice de Silhueta**: Mede a coesão intra-grupo versus separação inter-grupo
- **Índice de Calinski-Harabasz**: Avalia a razão entre dispersão inter-grupo e intra-grupo
- **Índice de Davies-Bouldin**: Quantifica a similaridade média entre grupos
- **Índice Rand Ajustado (ARI)**: Compara grupos descobertos com verdade fundamental quando disponível

Para validação, utilizou-se uma abordagem de separação temporal, onde partidas mais recentes foram reservadas para teste, simulando um cenário realístico de predição. Adicionalmente, análise qualitativa foi conduzida por especialistas em Dota 2 para avaliar a coerência estratégica dos grupos formados.

4.3. Validação dos Resultados

5. Conclusão e Trabalhos Futuros

5.1. Dimensionalidade

5.2. Resultados Obtidos

5.3. Trabalhos Futuros

Referências

Baldi, P. (2012). Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pages 37–49. JMLR Workshop and Conference Proceedings.

Cadman, A. (2024). Studying the effects of team composition in role-based competitive video games. Master's thesis, Lancaster University. Department of Computer Science.

Campello, R. J., Moulavi, D., and Sander, J. (2013). Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer.

Costa, L. M., Drachen, A., Souza, F. C., and Schön, D. (2023). Artificial intelligence in MOBA games: A multivocal literature mapping. *IEEE Transactions on Games*, 15(3):392–405.

Drachen, A. and Schubert, M. (2016). Esports analytics through encounter detection. In *MIT Sloan Sports Analytics Conference*.

Eggert, C., Herrlich, M., Smeddinck, J., and Malaka, R. (2015). Classification of player roles in the team-based multi-player game Dota 2. In *Entertainment Computing–ICEC 2015*, pages 112–125. Springer.

Ijäs, T. T. (2021). Spatial analytics in competitive gaming and e-sports. Master's thesis, University of Helsinki. Department of Geosciences and Geography.

Kamal, A. A., Mansor, M. A., Truna, L., and Daud, S. M. (2025). Machine learning applications in multiplayer online battle arena esports: A systematic review. *Pertanika Journal of Science & Technology*, 33(2):461–482.

Katona, A., Spick, R., Hodge, V. J., Demediuk, S., Block, F., Drachen, A., and Walker, J. A. (2019). Time to die: Death prediction in Dota 2 using deep learning. *IEEE Transactions on Games*, 12(3):273–284.

Le, Q. V. (2015). A tutorial on deep learning part 2: Autoencoders, convolutional neural networks and recurrent neural networks. Technical report, Google Brain.

LeCun, Y., Touresky, D., and Hinton, G. (1988). A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, pages 21–28.

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

OpenAI (2017). Dota 2. Acessado em: dezembro 2024.

OpenAI (2018). Openai five. Acessado em: dezembro 2024.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Schubert, M., Drachen, A., and Mahlmann, T. (2016). Esports analytics through encounter detection. In *Proceedings of the MIT Sloan Sports Analytics Conference*.

Xie, J., Girshick, R., and Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 478–487. PMLR.