ISCTE-IUL
Licenciatura em Ciência de Dados

Previsão do desempenho de estudantes ao jogar

Trabalho realizado no âmbito da Unidade Curricular Introdução a Processamento de Big Data do 2º ano em 2022/2023 da Licenciatura em Ciência de Dados

André Plancha, 105289
Andre_Plancha@iscte-iul.pt
Allan Kardec Rodrigues, 103380
aksrs@iscte-iul.pt

7 Abril 2023
Versão 1.0.0

Introdução

Jo Wilder and the Capitol Case é um jogo educacional sobre a história de Wisconsin, direcionado a crianças com idades entre os 8 e os 12 anos. O jogo de aventura e investigação segue a hsitória de Jo Wilder, que descobre histórias sobre artefactos da história do estado, investigando objetos, e encontrando pessoas. O jogo pode ser jogado no site oficial do PBD Wisconsin Education[1]. Esta competição está a ser promovida pela plataforma kaggle[2].

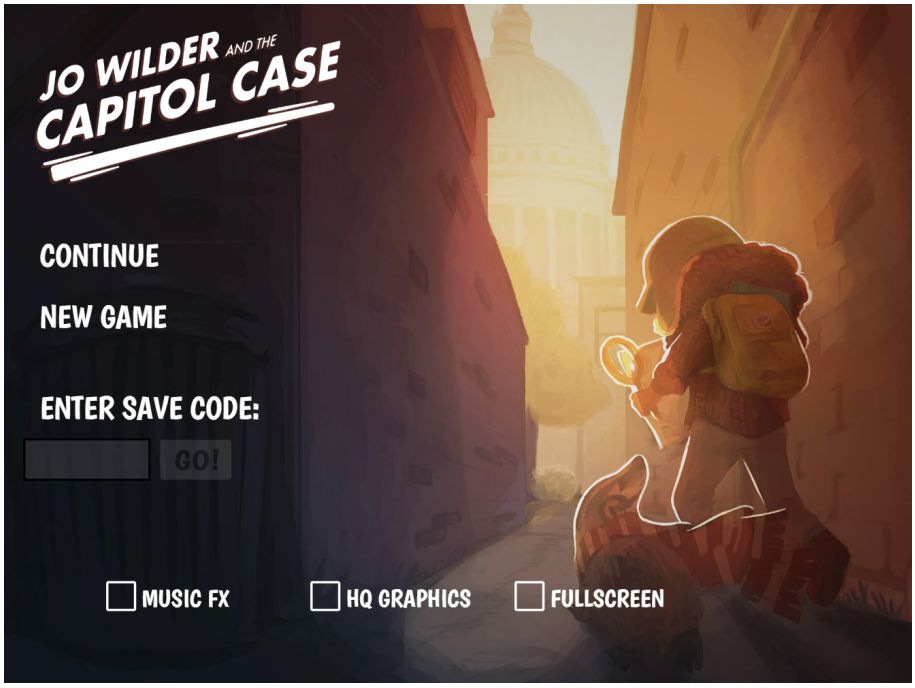

Figura 1: Imagem do menu principal

Este projeto tem como objetivo prever como os jogadores vão responder às 18 questões que o jogo presenta, baseado na sua atividade durante o tal, implementando uma solução computacional para estudo e análise de dados de grande dimensão. Para isso, vamos usar Apache Spark™e *PySpark* para processar os dados e a biblioteca *MLlib* para a criação de um modelo de aprendizagem supervisionada.

Análise exploratória dos dados

A competição disponibiliza um conjunto de ficheiros de dados, nos quais 2 são de interesse: `train.csv` e `train_labels.csv`, elas tendo as variáveis de interesse e as respetivas respostas, respetivamente. `train.csv` contém as seguintes colunas[3], entre outras:

- `session_id`: Identificador da sessão do evento.
- `index`: Índice do evento na sessão.
- `elapsed_time`: Tempo decorrido desde o início da sessão em ms.
- `event_name`: Tipo do evento.
- `name`: Específicos do tipo de evento.

[1] `pbswisconsineducation.org/jowilder/play-the-game/`
[2] `https://www.kaggle.com/competitions/predict-student-performance-from-game-play`
[3] Mais informações sobre cada coluna está no site da competição, e no EDA em anexo

- `hover_duration`: O tempo de permanência do cursor sobre um objeto, em ms, se aplicável.
- `text`: O diálogo/monólogo que o jogador viu no o evento, se aplicável
- `fqid`: O identificador único do evento.
- `fullscreen`: Se o jogador está em modo de ecrã inteiro.
- `hq`: Se o jogo está em alta qualidade.
- `music`: Se a música do jogo está ligada.
- `level_group`: O grupo de níveis a que o evento pertence.

; e `train_labels.csv` contém as seguintes colunas:

- `session_id`: Identificador da sessão do evento, em conjunto da pergunta que se pretende responder.
- `correct`: Se a pergunta está correta ou não.

Perante as colunas, como há uma incompatibilidade entre perguntas e respostas, nós decidimos criar tabela com cada sessão, características de tal (recursos) e as suas respostas, de forma usar classificação para prever as respostas.

Para criar estes recursos, foi necessário de uma análise dos tipos de eventos disponíveis. Uma análise mais aprofundada encontra-se em anexo, mas para resumir, os eventos estão divididos em 3 categorias: Exploração, exposição e revisao. Exploração são os eventos onde a personagem se move, interage com objetos opcionais, tenta entrar em zonas ainda não acedidadas, etc... Exposição são os eventos onde a personagem interage com personagens, interage com objetivos principais, e é onde se encontra o que o utilizador aprende. Revisão são os eventos de quando o utilizador está perdido e portanto precisa de apoio de qual é o próximo passo.

A única excepção a estes eventos é o evento `checkpoint`, representando quando o utilizador começa a responder às perguntas (os eventos das respostas às perguntas não se encontra na tabela). Há 3 zonas onde o utilizador responde às perguntas, e na base de dados as 3 zonas são divididas pelos `checkpoints`, e cada zona está rotulada pelo seu `level_group`.

Análise e limpeza

Nós usámos como base de limpeza a nossa análise, e a análise de erros de um utilizador do kaggle[4]. Na Tabela 1 encontra-se os problemas encontrados e as soluções que nós usámos para os resolver.

Algumas das limpezas foram úteis na criação dos recursos, enquanto que outras apenas limparam a base de dados. Nós não fizemos (nem registámos) algumas das limpezas, pois não seria benéfico para a criação dos recursos necessários.

Algo importante a notar também é que os roteiros dos jogos são diferentes para cada utilizador, entre 4 roteiros diferentes: *dry*, *nohumor*, *nosnark*, e *normal*. Um exemplo pode ser encontrado na Figura 2. A diferença entre roteiros é mínima na opinião dos autores, mas decidimos usar esta informação na mesma.

Exploração

TODO

[4] `https://www.kaggle.com/code/abaojiang/eda-on-game-progress`

Tabela 1: Resolução dos problemas

Erros	Problemas	Solução	Notas
Tempo recorrido recuáva no tempo	Cálculo de tempos entre eventos pode ficar inválido	Transformar em diferenças entre os tempos e tornar zero quando negativo	Máximo desta coluna continua afetado
Saltos de `index`	O índice do evento não é sequencial	-	Suspeitamos que os eventos aconteceram na mesma, só não foram anotados
Sessões com menos ou mais de 3 checkpoints	As sessões são inválidas e dificultam análise	Foram removidas essas sessões	
Tempos de jogo demasiado elevados comparado com a maioria dos tempos	Dificulta o processo de aprendizagem	Tempos de jogo foram transformados e normalizados	

Recursos

As features que nós fizemos para cada sessão são entre as seguintes:

- O index máximo, que indica o número de eventos de cada sessão;
- Se a sessão esteve em tela cheia;
- Se a sessão estava em alta qualidade;
- Se a sessão tinha música ligada;
- Quantos objetivos opcionais a sessão interagiu com;
- Quantas salas inacessíveis a sessão tentou entrar em;
- Quantas vezes a sessão reviu o objetivo;
- Tempo médio de leitura do diálogo;
- Tempo médio por movimento do jogador;
- Tipo de roteiro.

Os promenores de como foram criadas encontram-se em anexo.

Modelação

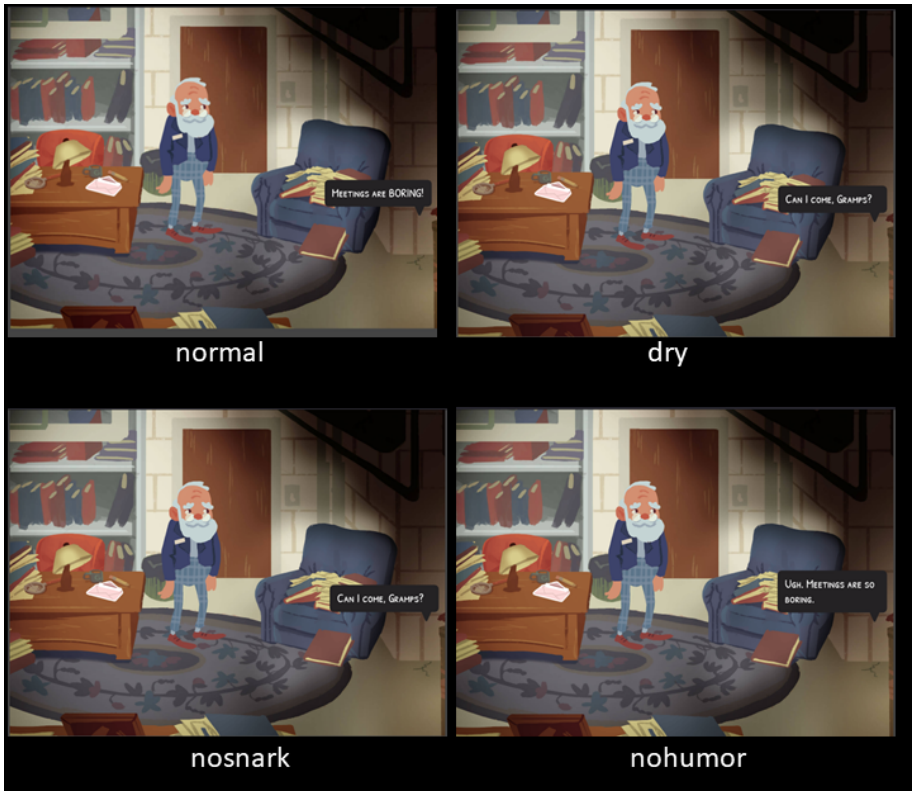

Figura 2: Diferenças entre roteiros