Games classifier

Team name

Members: Julia Cygan, Borys Adamiak, Patryk Flama
Supervisor: Marek Adamczyk

UWr

11 lutego 2025

https://github.com/PatrykFlama/GameClassifier

Your task

Goal and motivation

We want to be able to automatically assign tags to games, based on their description.

Solution to such problem has real-world applications, such as game grouping/filtering or finding simmilar games or trends analysis.

In aspect of ML project we want to make a small comparison of different models and data processing.

Your task

Goal and motivation

We want to be able to automatically assign tags to games, based on their description.

Solution to such problem has real-world applications, such as game grouping/filtering or finding simmilar games or trends analysis.

In aspect of ML project we want to make a small comparison of different models and data processing.

Info about the data

Steam has its own official API, from which we want to download all of the data (and since Steam is the largest library it will allow for a lot of diverse, high quality, data).

Currently there are above 100'000 games, which does create a large dataset.

Methods

Data processing

- Bag of Words - binary vector records if word appears in text (input representation) ✓
- TF-IDF - term frequency * inverse document frequency (input representation) ✓
- multi label binary vector (output representation) ✓

Methods

Data processing

- Bag of Words - binary vector records if word appears in text (input representation) ✓
- TF-IDF - term frequency * inverse document frequency (input representation) ✓
- multi label binary vector (output representation) ✓
- Hashing input to low-dimensional vector ✓

Methods

Models

- Baseline
  - KNN ✓
  - Logistic Regression ✓
  - Decision Trees + Random Forest ✓
  - Naive Bayes ✓

Methods

Models

- Baseline
  - KNN ✓
  - Logistic Regression ✓
  - Decision Trees + Random Forest ✓
  - Naive Bayes ✓
  - Simple perceptron-based neural network ✓
  - Support Vector Machine ✓

Methods

Models

- Baseline
  - KNN ✓
  - Logistic Regression ✓
  - Decision Trees + Random Forest ✓
  - Naive Bayes ✓
  - Simple perceptron-based neural network ✓
  - Support Vector Machine ✓

Evaluation

- Recall *TP/(TP+FN)* - we prefer to have more FN than to have an TP ✓
- F1-score *(2 * precision * recall) / (precision + recall)* - nice name, but also it combines precision with recall thus both TP and FN are equally expensive ✓

Methods

Models

- Baseline
  - KNN ✓
  - Logistic Regression ✓
  - Decision Trees + Random Forest ✓
  - Naive Bayes ✓
  - Simple perceptron-based neural network ✓
  - Support Vector Machine ✓

Evaluation

- Recall *TP/(TP+FN)* - we prefer to have more FN than to have an TP ✓
- F1-score *(2 * precision * recall) / (precision + recall)* - nice name, but also it combines precision with recall thus both TP and FN are equally expensive ✓
- Hammming loss
- Intersection over union score
- Exact match