

Detalhes do Miniprojeto

Curso de Ciências da Computação – Inteligência Artificial – Prof. Francisco Madeiro

Alunos: Eduardo Braga, Henrique Franca, Isabela Medeiros, Julia Vilela, Rafael Angelim

Artigo escolhido

Diante do aumento do uso de plataformas de streaming como a Twitch e a necessidade de se criar um ambiente saudável em ambientes virtuais, o artigo “Applying machine learning to assess emotional reactions to video game content streamed on Spanish Twitch channels” (2024) publicado na “Computer Speech & Language” escrito por Noemí Merayo, Rosalía Cotelo, Rocío Carratalá-Sáez e Francisco J. Andújar, com fator de impacto de 3.1, foi escolhido por sua abordagem inovadora.

A pesquisa se destaca pela construção do primeiro corpus em espanhol com anotações manuais de sentimentos e emoções em mensagens reais da Twitch, identificando a polaridade das mensagens dos telespectadores. Trazendo consigo uma acurácia relativamente boa ao utilizar técnicas em inteligência artificial.

Descrição da base de dados

Apesar de existir um interesse na análise de sentimentos/emoções nos ambientes digitais, poucas bases de dados criadas para isso apresentam abordagem para interações espontâneas e dinâmicas como as encontradas em transmissões ao vivo, dificultando o desenvolvimento de modelos eficazes para essa tarefa.

Diante disso, um corpus composto por comentários extraídos da plataforma de streaming “Twitch” foi criado. Tendo a base composta por 2216 mensagens coletadas durante sessões reais de streaming, de jogos diferentes, em espanhol, conferindo um caráter espontâneo e realista. Tendo em cada amostra da base três variáveis principais: o texto do comentário (variável independente) e duas variáveis-alvo: polaridade e emoção.

A variável de polaridade classifica os comentários em três categorias: positiva, negativa e neutra, enquanto a variável de emoção abrange seis classes distintas, sendo essas: aprovação, desaprovação, tristeza, raiva, empolgação e neutra. Assim, essa estrutura permite a formulação de um problema de classificação multiclasse com rótulo único, fazendo com que cada mensagem pertença a exatamente uma categoria de polaridade e uma de emoção.

A base está disponível publicamente no repositório GitHub sob a licença Creative Commons Attribution-nonCommercial-ShareAlike 4.0 International (CC BY-NC-SA). Adicionalmente, a base foi utilizada na avaliação de diferentes modelos de classificação, incluindo algoritmos tradicionais como Support Vector Machines (SVM) e Random Forest (RF), bem como modelos baseados em Deep Learning, como o BERT. Demonstrando que os experimentos realizados viabilizam o corpus para o treinamento de classificadores voltados a análise de sentimentos e detecção de emoções em textos curtos e informais.

Percentual de dados utilizados para treinamento e teste

O artigo fornece duas tabelas de dados, uma com 2216 comentários e a outra com 4601. Assim, pelo tamanho das tabelas, foi possível usar todos os dados fornecidos nos treinamentos.

Algoritmo(s) considerado(s) / implementado(s)

- Random Forest (RF).
- Support Vector Machine (SVM).

Figuras de mérito utilizadas para avaliação de desempenho

- Acurácia: Utilizada para avaliar tanto a detecção de **polaridade** (positivo, negativo, neutro) quanto de emoções (Approval, Hype, Disapproval, Sadness, Anger, Neutral).

- Recall: Entre as instâncias que realmente pertencem a uma classe, quantas foram corretamente identificadas pelo modelo.

- Precisão: Entre as instâncias que o modelo classificou como pertencentes a uma classe, quantas realmente pertencem a ela.

- F1-score: Média harmônica entre precisão e revocação.

- Matriz de Confusão: Tabela que mostra o desempenho do modelo em termos de verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos, para cada classe.

Implementação

1. Tratamento dos dados

Os dados utilizados são extraídos da “CorpusTwitchVideojuegos.xlsx”, composta por mensagens de canais espanhóis da Twitch com suas respectivas polaridades. Mas, apenas as colunas relevantes (TEXTO e Polaridad) são mantidas, e mensagens com valores ausentes são descartadas.

Stopwords em espanhol foram obtidas via NLTK, além do uso do stemmer “SnowballStemmer” e o “TweetTokenizer” para tokenização especializada em mensagens curtas.

O texto das mensagens foi pré-processado pela função preprocess(), que executa:

- Conversão para minúsculas e remoção de acentuação.
- Normalização de risadas e interjeições frequentes.

- Remoção de menções, hashtags, link e palavras irrelevantes como “streamlabs” ou “nightbot” (bot de mensagens da Twitch).

- Eliminação de tokens com números e pontuação.

- Tokenização, remoção de stopword e aplicação de stemming.

Após o pré-processamento, os textos foram transformados em vetores numéricos utilizando a técnica TF-IDF com a classe TfidfVectorizer.

Os dados vetorizados foram divididos em conjunto de treinamento (70%) e teste (30%) com o “train_test_split”, garantindo aleatoriedade.

2.Treinamento dos Modelos

A validação cruzada foi feita com estratégia Stratified K-Fold (n=10) para preservar a proporção entre classes nas divisões.

a) Random Forest

- Foi aplicado Grid Search com os seguintes hiperparâmetros:

 - n_estimators: [100, 300, 500]

 - max_features: ['sqrt', 'log2', None]

- O modelo com melhor desempenho nos folds é selecionado como final, e reavaliado no conjunto de teste.

b) SVM (Support Vector Machine)

- Também foi utilizado Grid Search com os parâmetros:

 - C: [1, 10, 50, 100, 150, 500, 1000]

 - Kernel: ['linear', 'rbf', 'poly']

3. Avaliação

A avaliação do modelos foi feita com base nos dado de teste, utilizando as seguintes figuras de mérito da biblioteca scikit-learn:

- classification_report(): para apresentar métricas de precisão, revocação (recall), f1-score e acurácia geral por classe.

- f1_score(average=None): para análise comparativa entre modelos por classe.

- confusion_matrix(): para gerar as matrizes de confusão.

- Visualizações gráficas foram geradas com as bibliotecas matplotlib e seaborn, incluindo:

 - Gráfico de barras comparando os F1-Scores de cada modelo por classe (Indeterminado, Negativo, Positivo);

- Matrizes de confusão separadas para Random Forest e SVM.

Modificação realizada na técnica considerada e/ou avaliação de aspectos não considerados no artigo

Foram incorporadas as seguintes extensões metodológicas:

- Validação cruzada estratificada: assim como no artigo original, o nosso projeto utiliza validação cruzada estratificada com 10 folds para garantir a robustez da avaliação dos modelos e a manutenção da proporção entre classes. No entanto, os valores dos hiperparâmetros encontrados foram diferentes, refletindo as diferenças nos vetores de entrada e possíveis variações de pré-processamento e divisão de dados.
- Otimização por Grid Search: para obter o melhor desempenho dos classificadores, foram aplicadas buscas em grade (Grid Search) para ajuste dos principais hiperparâmetros dos algoritmos Random Forest (número de árvores, quantidade de features) e SVM (C, kernel).
- SMOTE (Synthetic Minority Over-sampling Technique): diferentemente do artigo original, foi empregada a técnica SMOTE para lidar com o desbalanceamento entre classes no conjunto de treinamento. Essa abordagem foi essencial para evitar o viés do modelo em favor da classe majoritária e melhorar o desempenho, especialmente em métricas como recall e f1-score das classes menos representadas.

-Contribuições de cada integrante da equipe

-Treinamento e implementação dos modelos: Isabela Medeiros e Henrique Franca

-Pesquisa, filtragem e comparação dos dados fornecidos: Eduardo Costa, Julia Vilela, Rafael Angelim

MERAYO, Noemí; COTELO, Rosalía; CARRATALÁ-SÁEZ, Rocío; ANDÚJAR, Francisco J.
Applying machine learning to assess emotional reactions to video game content streamed on Spanish Twitch channels. Received 21 December 2023, revised 6 March 2024, Accepted 20 April 2024, Available online 25 April 2024, Version of Record 3 May 2024.