

Classificação de Pokémons da Primeira Geração Utilizando Descritores de Imagens

Alexandre Borges Baccarini Júnior, Leonardo Naime Lima, Mateus Santos Fernandes

¹Curso de Bacharelado em Ciência da Computação – UTFPR-CM

1. Introdução

Este trabalho propõe o desenvolvimento de um sistema inteligente para a classificação automática de imagens dos Pokémons da primeira geração: Bulbasaur, Charmander e Squirtle. O sistema foi construído utilizando extração de características, combinando um descritor de textura (LBP) e uma rede neural convolucional (VGG16). Foram empregados e otimizados três classificadores distintos: *k-Nearest Neighbors (k-NN)*, *Support Vector Machine (SVM)* e *Árvores de Decisão*, utilizando a biblioteca *scikit-learn*. O objetivo principal foi avaliar e comparar o desempenho destas abordagens na tarefa de classificação proposta.

2. Base de Dados

A base de dados foi composta por imagens obtidas de uma coleção disponível na plataforma *Kaggle*. O conjunto de dados contém um total de 337 imagens, divididas em 216 para treino e 121 para teste. As imagens foram organizadas em três classes correspondentes aos Pokémons alvo. Para a extração de características, todas as imagens foram pré-processadas conforme a necessidade de cada descritor.

3. Extração de Características

Foi adotada uma abordagem de combinação de características, na qual o vetor final de cada imagem é a concatenação de características extraídas pelos métodos, visando capturar tanto informações de texturas quanto semânticas.

3.1. Descritor Local Binary Pattern (LBP)

O LBP é um descritor de textura que analisa a vizinhança de cada pixel. As imagens foram convertidas para escala de cinza e o LBP foi aplicado com os seguintes parâmetros:

- Número de vizinhos: $P = 8$
- Raio: $R = 2$
- Método: `nri_uniform` (uniforme e invariante à rotação)

O resultado para cada imagem foi um histograma de 59 posições, representando a distribuição dos padrões de textura.

3.2. Deep Features com VGG16

Utilizou-se a rede neural convolucional VGG16, pré-treinada na base ImageNet, para extrair características profundas. As imagens (em formato RGB e redimensionadas para 224×224) foram processadas pela rede, e as ativações da última camada de *pooling* convolucional foram extraídas. Para reduzir a dimensionalidade deste vetor, foi aplicado o método *Principal Component Analysis (PCA)*, gerando um vetor de 50 componentes principais por imagem.

3.3. Pipeline de Pré-processamento e Otimização de Hiperparâmetros

O vetor de características combinado (LBP + VGG16), com 109 dimensões, passou por um pipeline de pré-processamento que consistiu em:

- Normalização dos dados com `StandardScaler`.
- Balanceamento com a técnica `SMOTE` para corrigir desequilíbrio entre as classes majoritárias.

A otimização dos hiperparâmetros foi uma etapa crucial para maximizar o desempenho de cada classificador. Para isso, foi utilizada a validação cruzada com 5 vias ($k = 5$), com o objetivo de encontrar os parâmetros que maximizassem a métrica `f1_score` média. O processo variou conforme o modelo:

- **k-NN e Árvore de Decisão:** Foi empregado o método `GridSearchCV`, que realiza uma busca exaustiva por todas as combinações de hiperparâmetros definidas em um espaço de busca pré-determinado.
- **SVM:** Para este modelo, foi utilizado o `RandomizedSearchCV`, que testa um número fixo de combinações amostradas aleatoriamente a partir de distribuições de parâmetros.

O melhor conjunto de hiperparâmetros encontrado por este processo para cada modelo é detalhado na Seção 4.

4. Classificadores Utilizados

Os parâmetros aqui descritos são os melhores encontrados após o processo de otimização.

4.1. k-Nearest Neighbors (k-NN)

Configurado com os seguintes hiperparâmetros:

- Número de vizinhos (k): 7
- Peso dos vizinhos: `distance` (vizinhos mais próximos têm maior influência)
- Métrica de distância: `cosine`
- PCA adicional para manter 85% da variância.

4.2. Support Vector Machine (SVM)

A SVM foi otimizada com busca aleatória, resultando nos seguintes hiperparâmetros:

- Kernel: `linear`
- Parâmetro de regularização (C): ≈ 0.0072
- Gamma (γ): ≈ 0.0022
- Peso de classes: `None`

4.3. Árvore de Decisão

Utilizou-se (*ensemble*) de 10 árvores de decisão através do `BaggingClassifier`, com os seguintes parâmetros para as árvores:

- Critério de divisão: `entropy`
- Profundidade máxima: 10
- Mínimo de amostras para divisão: 10
- Mínimo de amostras por folha: 2
- Máximo de features para divisão: `sqrt`

5. Resultados e Análise

Os modelos foram avaliados no conjunto de teste. As métricas de desempenho (Acurácia, Precisão, Recall e F1-Score) e as matrizes de confusão são apresentadas a seguir.

5.1. Desempenho dos Classificadores

Na validação cruzada, a Árvore de Decisão obteve a melhor Acurácia (0.8678), seguida pelo SVM (0.8099) e pelo k-NN (0.7355). As tabelas abaixo detalham o desempenho no conjunto de teste.

Tabela 1. Métricas do k-NN no conjunto de teste (Acurácia: 73.55%)

Classe	Precisão	Recall	F1-score	Suporte
Bulbasaur	1.00	0.73	0.85	41
Charmander	0.59	0.86	0.70	35
Squirtle	0.72	0.64	0.68	45
Média Ponderada	0.78	0.74	0.74	121

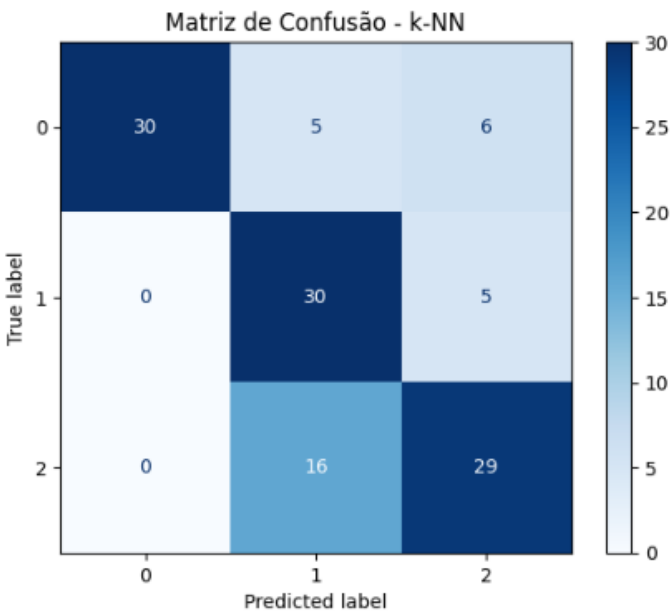


Figura 1. Matriz de confusão - k-NN

Tabela 2. Métricas do SVM no conjunto de teste (Acurácia: 80.99%)

Classe	Precisão	Recall	F1-score	Suporte
Bulbasaur	0.89	0.83	0.86	41
Charmander	0.74	0.83	0.78	35
Squirtle	0.80	0.78	0.79	45
Média Ponderada	0.81	0.81	0.81	121

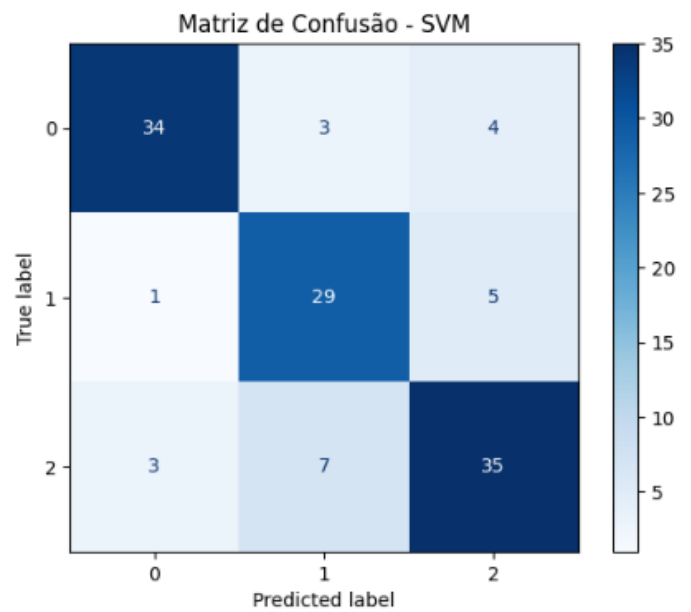


Figura 2. Matriz de confusão - SVM

Tabela 3. Métricas da Árvore de Decisão no teste (Acurácia: 86.78%)

Classe	Precisão	Recall	F1-score	Suporte
Bulbasaur	0.86	0.93	0.89	41
Charmander	0.93	0.77	0.84	35
Squirtle	0.83	0.89	0.86	45
Média Ponderada	0.87	0.87	0.87	121

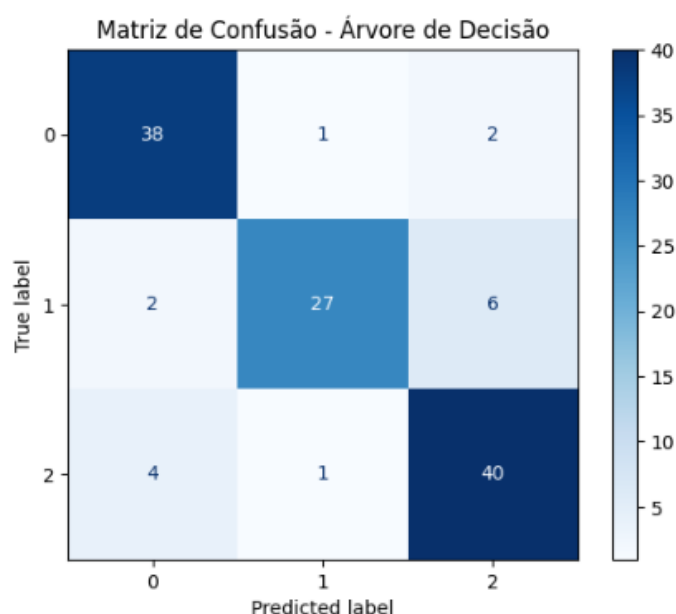


Figura 3. Matriz de confusão - Árvore de Decisão

5.2. Considerações

Pontos Fortes:

- O classificador de Árvore de Decisão apresentou o melhor desempenho geral, com acurácia de 86.78% e F1-score ponderado de 0.87. O uso de um comitê de árvores com profundidade controlada (max_depth=10) se mostrou a abordagem mais eficaz.
- Este modelo demonstrou alta precisão para a classe Charmander (0.93) e alto recall para a classe Bulbasaur (0.93), indicando uma boa capacidade de generalização para classes distintas.
- A fusão de características LBP e VGG16 foi bem-sucedida, fornecendo um vetor de características rico o suficiente para permitir altas taxas de acerto.

Pontos Fracos:

- O classificador k-NN obteve o desempenho mais baixo (acurácia de 73.55%), com notável dificuldade na precisão para a classe Charmander (0.59), classificando incorretamente muitos exemplos como sendo desta classe.
- Mesmo no melhor modelo, a classe Charmander apresentou o menor recall (0.77), sugerindo que ainda é a classe mais difícil de ser completamente identificada.

- A extração das características com a VGG16 é um processo computacionalmente intensivo, representando um gargalo em termos de tempo de execução.

6. Conclusão

O sistema desenvolvido foi capaz de classificar imagens de Pokémons da primeira geração com resultados excelentes, destacando-se o modelo de Árvore de Decisão, que atingiu 86.78% de acurácia. A combinação de descritores de textura e profundos provou ser eficaz, e a otimização de hiperparâmetros foi fundamental para o bom desempenho do ensemble de árvores. O desempenho inferior do k-NN reforça que, para este espaço de características, as fronteiras de decisão complexas criadas pelas árvores foram mais adequadas que a simples proximidade de vizinhos.