# ML-MedImage

Framework para Avaliação de Anotação Automática de Imagens Médicas Bidimensionais

## Resumo

O framework ML-MedImage dispõe um ambiente para avaliação de classificadores multirrótulos para a tarefa de anotar automaticamente imagens bidimensionais da área médica. Os rótulos são atribuídos de acordo com o código IRMA.

ML-MedImage foi utilizado na avaliação experimental em (Referência). Neste framework, dez subconjuntos são construído a partir de um conjunto com mais de 12.000 imagens médicas de Raio-X da região do tórax. As técnicas EHD, Gabor, LBP e SIFT são utilizadas para caracterizar as amostras dos subconjuntos formados (Experiments04).

A partir desses subconjuntos, os desempenhos de diversos classificadores multirrótulos são avaliados na tarefa de anotar imagens. Dez iterações são realizadas para essa avaliação. Em cada iteração, um subconjunto é usado para treino e os nove demais para teste. Os classificadores utilizados são BRkNN, ClassifierChain(RandomForest), LabelPowerset(kNN) e MLkNN (Experiments06).

Além dessa abordagem, também é avaliada uma abordagem alternativa, em que a classificação é realizada por eixo IRMA ao invés de atribuir os rótulos para todos os eixos em uma só etapa como na primeira abordagem (Experiments07).

A avaliação fornece resultados para diversas medidas em cada iteração. Esses resultados são então agrupados por medidas em arquivos individuais para que se possa obter médias e desvios-padrões de cada iteração (Tabulating 01 e 02). As medidas consideradas na avaliação experimental realizada foram Average Precision, Hamming Loss e Micro F.

## Fase 0 – Configuração do ambiente de avaliação

Main.java

### Descrição

Recebe as entradas para a execução das fases seguintes e registra em um arquivo os logs do experimento.

## Fase 1 – Preparação dos dados

Preparing.java

### Descrição

Obter NS subconjuntos de NI imagens, sem reposição, do conjunto de imagens IRMA CP que não contém a extensão TIF.

## Fase 2 – Construções das bases no formato ARFF

Building.java

### Descrição

Para cada subconjunto, construir quatro bases ARFF. Uma para cada técnica de extração de características.

## Fase 3 – Avaliação dos classificadores para a anotação direta dos eixos

Evaluating.java

### Descrição

Em cada iteração, as bases ARFF são usadas da seguinte maneira:

1. Cada base de cada técnica deverá ser o conjunto de treinamento usado para criar o modelo do classificador;
2. As demais bases da respectiva técnica serão os conjuntos de testes usados para avaliação do modelo criado.

Para a avaliação dos classificadores são consideradas as médias das medidas obtidas na avaliação dos nove conjuntos de testes em cada iteração e para cada técnica.

## Fase 4 – Organização dos resultados da avaliação da anotação direta

Tabulating.java

### Descrição

Agrupa por medida os resultados das iterações da classificação direta dos eixos do código IRMA.

## Fase 5 – Separação dos eixos do código IRMA

BuildingAxes.java

### Descrição

De cada uma dessas 40 bases serão obtidas mais quatro sub-bases, uma para cada eixo IRMA (**T**echnical, **D**irectional, **A**natomical, **B**iological), totalizando 160 sub-bases (Experiments05).

## Fase 6 – Avaliação dos classificadores para a anotação eixo-a-eixo

EvaluatingAxes.java

### Descrição

Nessa fase, serão reutilizadas as bases serializadas em Experiments05. Em Experiments05, das 40 bases constituídas a partir das técnicas EHD, LBP, SIFT e Gabor, foram criadas outras 160 sub-bases. Em cada sub-base criada, somente os rótulos de um único eixo foi mantido além das características.

## Fase 7 – Organização dos resultados da avaliação da anotação eixo-a-eixo

TabulingAxes.java

### Descrição

Agrupa por medida os resultados das iterações da classificação eixo-a-eixo do código IRMA.