# Elaboração de experimento que realiza classificação por eixo IRMA

## Resumo

Sub-bases são construídas a partir das bases criadas no Experiments04. Para cada base ARFF de Experiments04 são criadas mais quatro sub-bases, uma para cada eixo do código IRMA. Uma avaliação é realizada sobre quais técnicas de extração de características fornecem características mais relevantes e para quais eixos. Também é avaliado o desempenho da classificação multirrótulo por meio das abordagens de transformação do problema e adaptação do algoritmo.

## Abstract

Sub-bases were constructed from the bases created in Experiments04. For each ARFF base from Experiments04, more four sub-bases were created, one for each axe of the IRMA code. An evaluating is performed about what extracting features techniques provides features more relevant and for what axis. The performance from the multi-label classification was evaluated too through from problem transform and algorithm adapting approach.

## Fase 0 - Construção 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.

### Algoritmo

1. Inicio o registro de logs do experimento chamado **Experiments05-Exe[yyyymmddhhmm].log**;
2. Abro o arquivo com os parâmetros de entrada para as fases seguintes;
3. Executo cada uma das fases enviando os parâmetros de entrada necessários;
4. Finalizo o registro de logs do experimento.

### Entradas

1. Arquivo **Conf.ini**.

### Saídas

1. Arquivo de log do experimento **Experiments05-Exe[yyyymmddhhmm].log**.

## Fase 1 - Construção das bases ARFF

Building.java

### Descrição

Para construção das bases ARFF serão reutilizadas as bases ARFF criadas na execução **Exe2013129180** do **Experiments04**. São elas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ord** | **EHD** | **LBP** | **SIFT** | **Gabor** |
|  | Ehd-Sub0.arff | Lbp-Sub0.arff | Sift-Sub0.arff | Gabor-Sub0.arff |
|  | Ehd-Sub1.arff | Lbp-Sub1.arff | Sift-Sub1.arff | Gabor-Sub1.arff |
|  | Ehd-Sub2.arff | Lbp-Sub2.arff | Sift-Sub2.arff | Gabor-Sub2.arff |
|  | Ehd-Sub3.arff | Lbp-Sub3.arff | Sift-Sub3.arff | Gabor-Sub3.arff |
|  | Ehd-Sub4.arff | Lbp-Sub4.arff | Sift-Sub4.arff | Gabor-Sub4.arff |
|  | Ehd-Sub5.arff | Lbp-Sub5.arff | Sift-Sub5.arff | Gabor-Sub5.arff |
|  | Ehd-Sub6.arff | Lbp-Sub6.arff | Sift-Sub6.arff | Gabor-Sub6.arff |
|  | Ehd-Sub7.arff | Lbp-Sub7.arff | Sift-Sub7.arff | Gabor-Sub7.arff |
|  | Ehd-Sub8.arff | Lbp-Sub8.arff | Sift-Sub8.arff | Gabor-Sub8.arff |
|  | Ehd-Sub9.arff | Lbp-Sub9.arff | Sift-Sub9.arff | Gabor-Sub9.arff |
| **Total** | **10 bases** | **10 bases** | **10 bases** | **10 bases** |

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.

Para exemplificar, considerando apenas as bases EHD, tem-se a seguinte configuração:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ord** | **EHD** | **Eixo T** | **Eixo D** | **Eixo B** | **Eixo A** |
|  | Ehd-Sub0.arff | Ehd-Sub0-T.arff | Ehd-Sub0-D.arff | Ehd-Sub0-B.arff | Ehd-Sub0-A.arff |
|  | Ehd-Sub1.arff | Ehd-Sub1-T.arff | Ehd-Sub1-D.arff | Ehd-Sub1-B.arff | Ehd-Sub1-A.arff |
|  | Ehd-Sub2.arff | Ehd-Sub2-T.arff | Ehd-Sub2-D.arff | Ehd-Sub2-B.arff | Ehd-Sub2-A.arff |
|  | Ehd-Sub3.arff | Ehd-Sub3-T.arff | Ehd-Sub3-D.arff | Ehd-Sub3-B.arff | Ehd-Sub3-A.arff |
|  | Ehd-Sub4.arff | Ehd-Sub4-T.arff | Ehd-Sub4-D.arff | Ehd-Sub4-B.arff | Ehd-Sub4-A.arff |
|  | Ehd-Sub5.arff | Ehd-Sub5-T.arff | Ehd-Sub5-D.arff | Ehd-Sub5-B.arff | Ehd-Sub5-A.arff |
|  | Ehd-Sub6.arff | Ehd-Sub6-T.arff | Ehd-Sub6-D.arff | Ehd-Sub6-B.arff | Ehd-Sub6-A.arff |
|  | Ehd-Sub7.arff | Ehd-Sub7-T.arff | Ehd-Sub7-D.arff | Ehd-Sub7-B.arff | Ehd-Sub7-A.arff |
|  | Ehd-Sub8.arff | Ehd-Sub8-T.arff | Ehd-Sub8-D.arff | Ehd-Sub8-B.arff | Ehd-Sub8-A.arff |
|  | Ehd-Sub9.arff | Ehd-Sub9-T.arff | Ehd-Sub9-D.arff | Ehd-Sub9-B.arff | Ehd-Sub9-A.arff |
| **Total** | **10 bases** | **10 bases** | **10 bases** | **10 bases** | **10 bases** |

### Algoritmo

1. Crio um vetor de STRING chamado técnicas com os valores {Ehd, Lbp, Sift, Gabor}
2. Crio um vetor de CHAR chamado eixos com os valores {T, D, A, B};
3. Para cada elemento com o nome técnica do vetor técnicas, executo as operações:
   1. Para cada elemento com nome eixo do vetor eixos, executo as operações:
      1. Para cada iteração de i, com i valendo de 0 a 9, faça:
         1. Instancio uma base multirrótulo a partir do arquivo ARFF técnica-Sub[i];
         2. Desse objeto, removo os atributos rótulos que não iniciam com a letra eixo;
         3. Serializo as amostras desse objeto no arquivo **Experiments05-Exe[yyyymmddhhmm]-técnica-Sub[i]-eixo.bsi**.
         4. Serializo os rótulos nesse objeto no arquivo **Experiments05-Exe[yyyymmddhhmm]-técnica-Sub[i]-eixo.labels**.

### Entradas

1. Valores do vetor técnicas;
2. Valores do vetor eixos

### Saídas

1. (técnicas.lenght \* eixos.lenght \* 10) bases ARFF

## Fase 3 - Avaliação dos classificadores

Evaluating.java

### Descrição

As bases ARFF construídas deverão ser usadas da seguinte maneira:

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

Para a avaliação dos classificadores serão consideradas as médias das medidas obtidas na avaliação dos nove conjuntos de testes para cada eixo usando cada técnica.

Os classificadores previstos para serem avaliados são:

1. Multilabel kNN (MLkNN);
2. Binary Relevance kNN (BRkNN);
3. ClassifierChain;

### Algoritmo

1. Verifica se há um classificador a ser avaliado. Caso contrário, encerro fase;
2. Instancio o conjunto de treino;
3. Instancio o classificador;
4. Construo um modelo para o classificador a partir do conjunto de treinamento;
5. Instancio um avaliador;
6. Instancio o conjunto de teste;
7. Instancio uma lista de medidas;
8. Instancio uma avaliação e atribuo a ela o resultado do classificador avaliado;
9. Armazeno em arquivo o resultado da avaliação para liberar RAM;
10. Repito 9 vezes o algoritmo a partir do passo 5.

### Entradas

1. Classificadores que serão avaliados.

### Saídas

1. (9execuções \* 4eixos \* 4tecnicas \* 3classificadores = 432) resultados de avaliação.