UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

THISSIANY BEATRIZ ALMEIDA

SELEÇÃO DE ATRIBUTOS USANDO A ABORDAGEM WRAPPER PARA CLASSIFICAÇÃO HIERÁRQUICA MULTIRRÓTULO

EXAME DE QUALIFICAÇÃO

PONTA GROSSA
2017

THISSIANY BEATRIZ ALMEIDA

SELEÇÃO DE ATRIBUTOS USANDO A ABORDAGEM WRAPPER PARA CLASSIFICAÇÃO HIERÁRQUICA MULTIRRÓTULO

Exame de Qualificação apresentado ao Programa de Pós-Graduação em Ciência da Computação da Universidade Tecnológica Federal do Paraná - Campus Ponta Grossa. Área de Concentração: Inteligência Artificial

PONTA GROSSA
2017

RESUMO

ALMEIDA, Thissiany Beatriz. *Seleção de atributos usando abordagem Wrapper para classificação hierárquica multirrótulo. 2017. 83*. Exame de Qualificação (Mestrado em Ciência da Computação) - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2017.

A seleção de atributos é uma das técnicas que podem ser utilizadas para a redução de dimensionalidade de base de dados. Tem como objetivo principal identificar os atributos relevantes aumentando assim o poder preditivo do classificador. No contexto de classificação hierárquica multirrótulo, onde as classes a serem preditas estão estruturadas de acordo com uma hierarquia, poucos trabalhos na literatura apresentam propostas de técnicas de seleção de atributos. Desse modo, neste trabalho propõe-se um novo método de seleção de atributos baseado na abordagem Wrapper para classificação hierárquica multirrótulo global. Para a realização dos experimentos são utilizados dados biológicos de 2 bases de dados da Ontologia Gênica, sendo que as classes das mesmas estão estruturadas em uma hierarquia no formato de um Grafo Acíclico Direcionado (DAG).

Palavras-chaves: Classificação Hierárquica Multirrótulo. Seleção de Atributos. Classificação Global. DAG. Wrapper.

ABSTRACT

ALMEIDA, Thissiany Beatriz. *Feature Selection using the Wrapper approach for hierarchical multi-label classification. 2017. 83*. Partial Exam (Master in Computer Science) - Federal University of Technology- Paraná. Ponta Grossa, 2017.

Feature Selection is one of the techniques that can be used for the reduction of database dimensionality. Its main objective is to identify the relevant features, thus increasing the predictive power of the classifier. In the context of Hierarchical Multi-label Classification, where classes to be predicted are structured according to a hierarchy, few works in the literature present proposals for attribute selection techniques. Thus, in this work we propose a new method of feature selection based on the Wrapper approach for Global Hierarchical Multi-label Classification. For the accomplishment of the experiments biological data of 2 databases of the Gene Ontology are used, being the classes of the same ones are structured in a hierarchy in the format of a Directed Acyclic Graph (DAG).

Key-words: Hierarchical Multi-label Classification. Feature Selection. Global Classification. DAG. Wrapper

LISTA DE EQUAÇÕES

Equação 1 – Medida Precisão Hierárquica 35
Equação 2 – Medida Revocação Hierárquica 35
Equação 3 – Medida F-Measure 35
Equação 4 – Medida H-Loss 36
Equação 5 – Medida da Contribuição de um Falso Positivo 37
Equação 6 – Medida da Contribuição de um Falso Negativo 37
Equação 7 – Medida RCon 37
Equação 8 – Medida da Contribuição de Falsos Positivos 37
Equação 9 – Medida da Contribuição de Falsos Negativos 37
Equação 10 – Medida da Precisão Hierárquica de cada classe 38
Equação 11 – Medida da Revocação Hierárquica de cada classe 38
Equação 12 – Medida da Micro Precisão Hierárquica 38
Equação 13 – Medida da Micro Revocação Hierárquica 38
Equação 14 – Medida da Macro Precisão Hierárquica 38
Equação 15 – Medida da Macro Revocação Hierárquica 38
Equação 16 – Medida da Precisão Hierárquica da curva PR 39
Equação 17 – Medida da Revocação Hierárquica da curva PR 39
Equação 18 – Medida AUPRC 39
Equação 19 – Medida InOrdinatio 50
Equação 20 – Cálculo do tamanho do população 66

LISTA DE FIGURAS

Figura 1 – Processo de classificação de dados 17
Figura 2 – Problema de classificação unirrótulo x Problema de Classificação multirrótulo 19
Figura 3 – Técnicas para classificação multirrótulo 20
Figura 4 – Processo utilizado pelo LP 21
Figura 5 – Processo utilizado pelo BR 23
Figura 6 – Processo utilizado pelo RAkEL 24
Figura 7 – Exemplo de árvore de decisão 25
Figura 8 – Estrutura de classes em formato hierárquico 28
Figura 9 – Classificação de gêneros musicais 29
Figura 10 – Classificação de gêneros textuais 29
Figura 11 – Hierarquia de classes de um problema hierarquico multirrótulo estruturado como uma árvore 30
Figura 12 – Predições usando uma subárvore 30
Figura 13 – Exemplo de PCT 32
Figura 14 – Modelo do hmAnt-Miner 33
Figura 15 – Modelo do MHC-CNN 34
Figura 16 – Técnica de seleção de atributos: Filtro 43
Figura 17 – Técnica de seleção de atributos: $Wrapper$ 44
Figura 18 – Funcionamento do Algoritmo Genético 47
Figura 19 – Etapas da Revisão Sistemática 50
Figura 20 – Áreas de aplicação x Quantidade de artigos 55
Figura 21 – Funcionamento do Método Proposto 64
Figura 22 – Funcionamento do Método proposto 65
Figura 23 – Hierarquia das classes 68
Figura 24 – População inicial do AG 70
Figura 25 – Descarte da população 70

LISTA DE QUADROS

Quadro 1 – Exemplo de uma base de dados multirrótulo 18
Quadro 2 – Conjunto de dados obtidos ao utilizar o método BR nos dados da Quadro 1 22
Quadro 3 – Conjunto de dados transformados usando o método RAKEL 24
Quadro 4 – Definição do protocolo da revisão sistemática 51
Quadro 5 – Bases de dados utilizadas 52
Quadro 6 – Tipos de contribuição científica x Artigos selecionados 56
Quadro 7 – Áreas de aplicação x Base de dados utilizadas 57
Quadro 8 – Cronograma de desenvolvimento do trabalho 75

LISTA DE TABELAS

Tabela 1 – Quantidade de resultados por base 52
Tabela 2 – Ranking de classificação dos trabalhos após avaliação..... 54
Tabela 3 – Técnicas e abordagens de seleção de atributos x Quantidade de trabalhos... 55
Tabela 4 – Trabalhos de seleção de atributos para classificação hierárquica 62
Tabela 5 – Passos para Redução do subconjunto de atributos utilizando a abordagem *Wrapper* 66
Tabela 6 – Passos para o funcionamento da fase de teste do algoritmo..... 67
Tabela 7 – Base de dados 68
Tabela 8 – Base de treinamento..... 69
Tabela 9 – Base de teste..... 69
Tabela 10 – Parâmetros de configuração do AG 69
Tabela 11 – Base de teste reduzida 71
Tabela 12 – Características da base de dados GO 72
Tabela 13 – Parâmetros do Algoritmo Genético 73
Tabela 14 – Resultado obtido na base Cellcycle 73
Tabela 15 – Resultado obtido na base Church 73

LISTA DE ABREVIATURAS E SIGLAS

ABT	Adaptive Binary Tree
AD	Árvore de Decisão
AFIS	*Automatic Fingerprint Identification System*
AG	Algoritmo Genético
AIRS	*Artificial Immune Recognition System*
AM	Aprendizagem de Máquina
ANN	*Artificial Neural Network*
AUPRC	*Area Under the Precision Recall Curve*
AVG-NN	*Average Group Nearest Neighbor*
B&B	*Branch and Bound procedure*
BDE	*Binary Differencial Evolution*
BR	*Binary Relevance*
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CART	*Classification and Regression Tree*
CD	Classificação de Dados
CFS	*Consistency-based Feature Selection*
cSFS	*Cost-Sensitive Feature Selection*
DAG	Grafo Acíclico Direcionado
ECA	*Emotion Component Analysis*
FM	*F-Measure*
FSHC	*Feature Selected Hierarchical Classifier*
GO	*Gene Ontology*
H-Loss	*Hierarchical Loss Function*
hmAnt-Miner	*Hierarchical Multi-label Classification Anti-Miner*
HMC	*Hierarchical Multi-label Classification*
IMBHN	*Inductive Model based on Bipartite Heterogeneous Networks*
JCR	*Journal Citation Reports*
kNN	*k-Nearest Neighbor*

kNNC	*k-Nearest Neighbor Classifier*
LDA	*Linear Discriminant Analysis*
LDC	*Linear Classifier*
LP	*Label Powerset*
MAP	*Maximum a Posteriori*
MD	Mineração de Dados
MHCAIS	*Multi-label Hierarchical Classification with an Artificial Immune System*
MHC-CNN	*Multi-label Hierarchical Classification using a Competitive Neural Network*
MLP	*Multi-Layer Perceptron*
MSRD	*Maximize the Sum of Relevance and Distance*
NEURC	*Neural Network Classifier*
OAO-SVM	*One-Against-One Support Vector Machine*
OOA-SVM	*One-Against-All Support Vector Machine*
PCT	*Predictive Clustering Trees*
PMI	*Pointwise Mutual Information*
Prec	Precisão
PSO	*Particle Swarm Otimization*
PTHS	*Parallel opTimization and Hierarchical Selection*
QDC	*Quadratic Classification*
RAkEL	*Random k-Labelsets*
RAkELd	*Random k-Labelsets disjoint*
RBF	*Radial Basis Function*
Rev	Revocação
RF	*Random Forest*
R-LDA	*Robust Linear Discriminant Analysis*
RMSE	*Root Mean Squared Error*
SIA	Sistemas Imunológicos Artificiais
SJR	*Scientific Journal Rank*
SSAP	*Sequencial Structural Aligment Program*
SVM	*Support Vector Machine*
SVM-BDT	*Support Vector Machine Binary Decision Tree*
SVM-RFE	*Support Vector Machine Recursive Feature Elimination*

SUMÁRIO

1 INTRODUÇÃO **12**
1.1 DESCRIÇÃO DO PROBLEMA E MOTIVAÇÃO 13
1.2 OBJETIVOS 15
1.2.1 Objetivo Geral 15
1.2.2 Objetivos Específicos 15
1.3 ORGANIZAÇÃO DO TRABALHO 15
2 CLASSIFICAÇÃO DE DADOS **17**
2.1 CONCEITOS FUNDAMENTAIS SOBRE CLASSIFICAÇÃO 17
2.2 CLASSIFICAÇÃO MULTIRRÓTULO 18
2.2.1 Métodos de Classificação Multirrótulo: Transformação do problema 21
2.2.1.1 *Label Powerset* 21
2.2.1.2 *Binary Relevance* 22
2.2.1.3 *Random k-labelsets* 23
2.2.2 Métodos de Classificação Multirrótulo: Adaptação de algoritmos 24
2.2.2.1 Árvore de decisão 25
2.2.2.2 *Multi-Label k-Nearest Neighbor* 26
2.2.2.3 AdaBoost 27
2.2.2.4 Métodos Probabilísticos 27
2.3 CLASSIFICAÇÃO HIERÁRQUICA 28
2.4 CLASSIFICAÇÃO HIERÁRQUICA MULTIRRÓTULO 29
2.4.1 Métodos de Classificação Hierárquica Multirrótulo 31
2.4.1.1 Clus-HMC 31
2.4.2 Multi-Label Hierarchical Classification with an Artificial Immune System 32
2.4.2.1 Hierarchical Multi-Label Classification Anti-Miner 33
2.4.2.2 Multi-label Hierarchical Classification using a Competitive Neural Network 33
2.4.3 Medidas de avaliação hierárquica 34
2.4.3.1 Medidas baseadas nas relações de ancestralidade e descendência 34
2.4.3.1.1 Precisão e revocação hierárquica 35
2.4.3.2 Hierarchical Loss Function 36
2.4.3.3 Medidas baseadas em distância 36
2.4.3.4 Medidas baseadas na Curva de Precisão e Revocação 39
2.5 CONSIDERAÇÕES FINAIS DO CAPíTULO 40
3 SELEÇÃO DE ATRIBUTOS **41**
3.1 CONCEITOS FUNDAMENTAIS DE SELEÇÃO DE ATRIBUTOS 41
3.2 ABORDAGENS DE SELEÇÃO DE ATRIBUTOS 42
3.2.1 Abordagem de Seleção de Atributos do tipo Filtro 42
3.2.2 Abordagem de Seleção de atributos do tipo $Wrapper$ 43
3.2.2.1 Estratégias de busca 44
3.2.2.1.1 Busca exaustiva 45
3.2.2.1.2 Seleção para frente 45
3.2.2.1.3 Retroalimentação 45
3.2.2.1.4 Buscas heurísticas 46
3.3 CONSIDERAÇÕES FINAIS DO CAPíTULO 48
4 REVISÃO SISTEMÁTICA DA LITERATURA **49**
4.1 MÉTODO DE REVISÃO SISTEMÁTICA 49
4.2 APLICAÇÃO DO MÉTODO DE REVISÃO SISTEMÁTICA 51

4.2.1 Estabelecimento da intenção de pesquisa 51
4.2.2 Definição das palavras-chave e base de dados 51
4.2.3 Pesquisa nas bases de dados 52
4.2.4 Procedimentos de filtragem 53
4.2.5 Classificação e ordenação dos artigos 53
4.2.6 Leitura e análise dos artigos em formato integral 53
4.2.6.1 Quais são as abordagens e técnicas de seleção de atributos utilizadas em classificação hierárquica? 53
4.2.6.2 Qual foi a contribuição científica do trabalho? 54
4.2.6.3 Quais foram as áreas e as bases de dados em que a seleção de atributos foi aplicada? 55
4.2.6.4 Quais foram os classificadores e as medidas de avaliação utilizados? 56
4.2.6.5 Os resultados obtidos pelas técnicas propostas foram superiores as técnicas existentes? 60
4.3 OUTROS TRABALHOS 62
4.4 CONSIDERAÇÕES FINAIS DO CAPíTULO 63
5 METODOLOGIA 64
5.1 DESCRIÇÃO DO ALGORITMO 64
5.1.1 Estratégica de busca 65
5.1.2 Avaliação do subconjunto 67
5.2 SIMULAÇÃO DO ALGORITMO 67
5.2.1 Passos para redução do subconjunto de atributos utilizando a abordagem $Wrapper$ e Classificação Hierárquica Multirrótulo 69
5.3 CONSIDERAÇÕES FINAIS DO CAPíTULO 71
6 EXPERIMENTOS E RESULTADOS 72
6.1 BASE DE DADOS 72
6.2 EXPERIMENTOS INICIAIS 72
6.3 CONSIDERAÇÕES FINAIS 74
7 CRONOGRAMA 75
7.1 ATIVIDADES A SEREM REALIZADAS 75
7.2 CONSIDERAÇÕES FINAIS DO TRABALHO 76
REFERÊNCIAS 83

1 INTRODUÇÃO

O termo mineração de dados é associado a busca de conhecimento útil em grandes bases de dados. Uma de suas aplicações é permitir com que um número significativo de atributos ou mesmo registros presentes na base de dados sejam removidos tornando o processo de aprendizagem mais eficiente e rápido.

Uma das tarefas exploradas na MD é a classificação de dados. Esse processo tem como objetivo atribuir uma ou mais classes para um novo exemplo a partir de suas características (atributos). Por exemplo, em um diagnóstico médico a classe pode indicar se uma pessoa está ou não com uma determinada doença.

Os problemas de CD podem ser agrupados em: Classificação Plana (Tradicional) ou Classificação Hierárquica. Os dois grupos são diferenciados pelo relacionamento de dependência entre as classes. Na classificação plana não se tem uma relação hierárquica entre as classes. Uma instância do conjunto de treinamento está relacionada a uma determinada classe. Esse tipo de classificação pode ser dividida em unirrótulo ou multirrótulo (BORGES, 2012).

Na classificação hierárquica, as classes estão dispostas em uma hierarquia. Os problemas de classificação hierárquica podem ser categorizados conforme o tipo de hierarquia (Árvore ou Grafo Acíclico Direcionado), caminho percorrido na hierarquia (única classe ou multi-classe) e a profundidade das rotulações (rotulação parcial ou até um nó folha). Com esse tipo de classificação, busca-se obter uma maior capacidade preditiva, pois admite-se que quanto maior a profundidade percorrida no processo de classificação, maior o conhecimento adquirido (PAES; PLASTINO; FREITAS,2013; FACELI *et al.*, 2011).

A popularidade crescente da classificação hierárquica multirrótulo pode ser explicada em razão da sua aplicabilidade em vários problemas relevantes, tais como: categorização de texto (ZHOU; XIAO; WU,2011; GOPAL; YANG, 2013), bioinformática (ALVES,2010; ROMAO, 2012; BORGES, 2012;CERRI *et al.*,2016), processamento de imagens(BARUTCUOGLU; SCHAPIRE; TROYANSKAYA, 2006; DIMITROVSKI *et al.*,2012;HUANG; BOOM; FISHER, 2015), entre outros.

O desenvolvimento de classificadores hierárquicos pode fazer uso de duas abordagens principais: Classificador Local e Classificador Global. A principal diferença entre essas abordagens é a maneira de como o modelo de classificação é criado. O primeiro avalia cada nó do conjunto de dados hierárquico independentemente. Já o segundo leva em consideração toda a hierarquia de classes do conjunto ao predizer uma instância.

A grande maioria dos trabalhos de classificação hierárquica para estruturas do tipo DAG utiliza a abordagem de classificação local. Isso ocorre porque classificadores dessa abordagem são mais simples de serem construídos e podem fazer uso de algoritmos usados na classificação plana. Já o desenvolvimento de classificadores globais é mais complexo, pois a estrutura hierárquica deve ser respeitada.

Um dos problemas enfrentados por pesquisadores da área de MD é que as bases possuem um volume grande de dados e com isso necessitam de técnicas para redução de dimensionalidade dessas bases, afim de remover atributos irrelevantes e/ou irredundantes, sendo a seleção de atributos uma das técnicas mais explorada (GUYON; ELISSEEFF, 2006; CHANDRASHEKAR; SAHIN, 2014).

Dada a importância de se reduzir o espaço de dados e atributos, pesquisas na área de seleção de atributos foram iniciadas na década de 80 nas áreas de estatística e reconhecimento de padrão (BEN-BASSAT, 1982; FOROUTAN; SKLANSKY, 1987), e só posteriormente passaram a ser tratadas na área de Aprendizagem de Máquina (JOHN *et al.*, 1994; AHA; BANKERT, 1996; BLUM; LANGLEY, 1997).

Na AM a utilização da seleção de atributos busca reduzir o tempo de execução do classificador, aumentar a sua capacidade preditiva e obter uma representação mais compacta do conceito a ser aprendido. Sendo assim, esse trabalho busca aplicar a seleção de atributos em base de dados hierárquicas multirrótulos e estruturadas em formato de DAG.

Este trabalho apresenta um método de seleção de atributos utilizando a abordagem *Wrapper*, que combina o algoritmo genético (estratégia de busca) com um classificador hierárquico multirrótulo, na busca de um conjunto de atributos que promova o melhor desempenho do classificador, para o domínio de aplicação escolhido.

1.1 DESCRIÇÃO DO PROBLEMA E MOTIVAÇÃO

Um dos problemas enfrentados na MD é a complexidade apresentada pelo grande volume de dados (atributos) e que muitos desses não estão relacionados com o processo de classificação (PAPPA, 2002). Atributos redundantes prejudicam a performance do algoritmo de aprendizagem tanto na velocidade (devido à dimensionalidade dos dados) quanto na taxa de acerto (devido à presença de informações redundantes que podem confundir o algoritmo, ao invés de auxiliá-lo na busca de um modelo correto para o conhecimento) (WITTEN *et al.*, 2016).

Dentre as técnicas criadas com o intuito de reduzir a dimensionalidade das bases de dados, pode-se citar a seleção de atributos. Nesse contexto, a seleção de atributos é uma técnica muito explorada na área de mineração de dados, pode ser vista como um processo de busca onde o algoritmo usado deve encontrar o menor subconjunto de atributos com a melhor classificação em conformidade com custos computacionais razoáveis (GUYON; ELISSEEFF, 2006; SANTOS, 2007; PAPPA, 2002).

Quando a tarefa alvo da seleção de atributos é a classificação, a seleção de atributos normalmente busca minimizar a taxa de erro do classificador, a complexidade do conhecimento gerado por ele, e o número de atributos selecionados para compor a “nova” base.

A aplicação de algoritmos de seleção de atributos em classificação hierárquica é uma

área recente de pesquisa e possui alguns trabalhos publicados como em (WEI *et al.*,2017; PERALTA *et al.*,2017; DONG; ZHAO; JIN,2017; BARALDI *et al.*,2016; CAO *et al.*,2016; ROSSI *et al.*,2014; ZHU; LIU,2014; PAN; ZHU; XIA,2013; XU; YANG; WANG,2015; SECKER *et al.*,2010; CHANG *et al.*,2012; PAES; PLASTINO; FREITAS,2013; FREEMAN; KULIC; BASIR,2013), sendo estes descritos no capítulo 4. Estes trabalhos mostram que a seleção de atributos pode trazer resultados superiores as métricas do processo de classificação quando comparado com o processo de classificação com as bases originais. Além do fato de ser uma área nova e despertar o interesse de pesquisadores de diversas áreas, como a bioinformática, categorização de textos, processamento de imagens e diagnósticos médicos, por exemplo.

Como exemplo de problemas em que se pode aplicar a técnica de seleção de atributos tem-se a classificação de proteínas. As funções exercidas por uma proteína no meio celular são organizadas hierarquicamente, podendo estas estruturas serem no formato de uma árvore ou um DAG (CLARE, 2003; PEREIRA; NIEVOLA, 2016). A principal diferença entre as estruturas hierárquicas é que na árvore cada nó de classe, exceto o nó raiz, tem apenas um pai, enquanto que no DAG cada nó de classe pode ter um ou mais nós pais.

Outro exemplo são os problemas de categorização de textos. Classes de textos correlatos são comumente agrupadas em tópicos, os quais, por sua vez, também podem ser agrupados em temas principais (SUN; LIM, 2001). Assim, um texto que trata de uma partida de futebol pode ser classificado na seção de esportes,que por sua vez pode também incluir tópicos como basquete, futebol, tênis, entre outros ligados ao tema. Desse modo, o modelo induzido deve ser capaz de classificar o texto mencionado tanto na categoria futebol quando em sua supercategoria esporte. Essa abordagem também é utilizada na classificação de gêneros musicais, de fonemas, de objetos 3D, de animais e de imagens (SILLA JR.; FREITAS, 2011).

No que refere-se a problemas de classificação hierárquica multirrótulo, não tem-se uma quantidade de pesquisas em que foram aplicadas técnicas de seleção de atributos; isso pode ser observado no Capítulo 4 que aborda os trabalhos correlatos. Com relação aos problemas de classificação de proteínas, não conhece-se técnicas de seleção de atributos que levem em consideração a estrutura hierárquica do tipo DAG, requisito esse fundamental para o tipo de base de dados trabalhada, sendo esta a base de dados da Ontologia Gênica (Gene Ontology -GO) [1].

Tem-se nesse trabalho como objeto de estudo então o desenvolvimento de uma técnica de seleção de atributos utilizando a abordagem *Wrapper* para os Problemas de Classificação Hierárquica Multirrótulo, com o intuito de medir o ganho computacional de economia de recursos (memória, tempo, entre outros).

Algoritmos genéticos foram escolhidos por serem um método de busca robusto, capaz de explorar grandes espaços de atributos. Além disso, ao contrário da maioria dos algoritmos de busca tradicionais, eles identificam e exploram interações não lineares entre os atributos, e realizam uma busca global (GOLDBERG, 1989).

[1] http://www.geneontology.org/page/go-database

1.2 OBJETIVOS

Esta Seção apresenta o objetivo geral e os objetivos específicos deste trabalho. Na Subseção 1.2.1 está estabelecido o objetivo geral e por fim, na Subseção 1.2.2, os objetivos específicos.

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é desenvolver um método de seleção de atributos usando a abordagem *Wrapper* para classificação hierárquica multirrótulo global.

1.2.2 Objetivos Específicos

Como objetivos específicos desse trabalho pode-se citar:

- Compreender o funcionamento da classificação hierárquica multirrótulo;
- Identificar as abordagens de classificação nas bases de dados hierárquicas (nó, nível, global, local);
- Identificar na literatura os algoritmos de seleção de atributos existentes;
- Desenvolver um algoritmo para reestruturação das bases de dados geradas pelo método desenvolvido;
- Realizar experimentos com os subconjuntos de dados gerados pelos algoritmos de seleção de atributos;
- Avaliar os resultados obtidos por meio de métricas e testes estatísticos.

1.3 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado em sete capítulos, do seguinte modo:

- **Capítulo 1:** O capítulo introdutório contextualiza este trabalho, apresentando o problema a ser estudado, os objetivos e a motivação para sua realização.
- **Capítulo 2:** Este capítulo aborda os principais conceitos e técnicas referentes a classificação de dados, que proporcionam o embasamento teórico bem como a contextualização em que esse trabalho se insere.

- **Capítulo 3:** O capítulo apresenta os conceitos, abordagens e principais técnicas de seleção de atributos. Além disso, também estão descritas as principais estratégias de busca utilizadas em conjunto com as técnicas de seleção de atributos.

- **Capítulo 4:** Neste capítulo é apresentada a revisão sistemática realizada na literatura, permitindo assim com que fossem identificados os trabalhos correlatos que serviram de referência para o desenvolvimento deste trabalho.

- **Capítulo 5:** O capítulo contém a metodologia utilizada, como o detalhamento do método criado. Apresenta também a simulação da execução do método proposto com uma base fictícia a fim de melhor entendimento.

- **Capítulo 6:** Este capítulo mostra os experimentos realizados e os resultados preliminares obtidos.

- **Capítulo 7:** Esse capítulo apresenta o cronograma de atividades para o desenvolvimento do projeto que está em andamento.

2 CLASSIFICAÇÃO DE DADOS

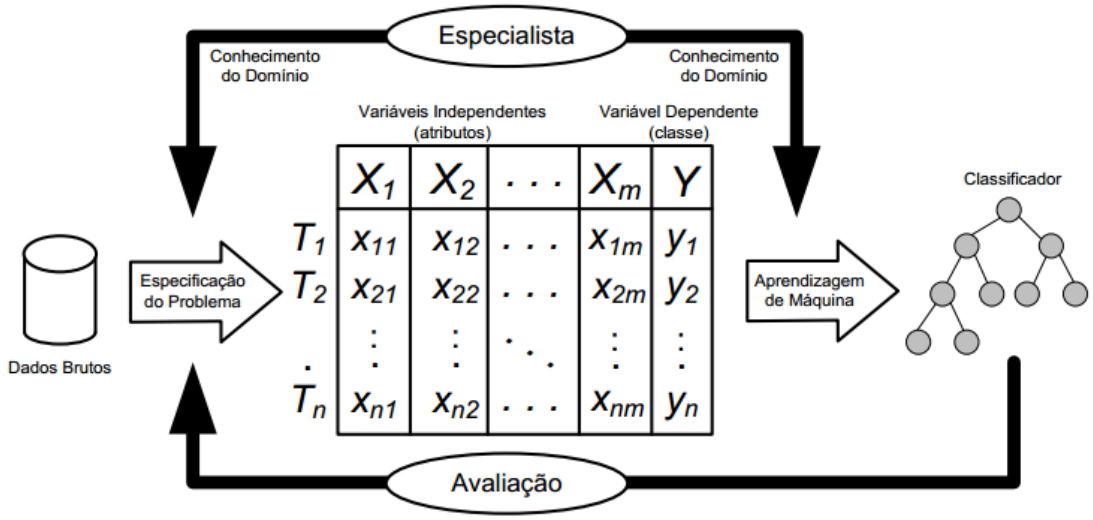
Este Capítulo apresenta os conceitos básicos de classificação de dados. A Seção 2.1 descreve os conceitos fundamentais de classificação. A Seção 2.2 descreve os conceitos relacionados a classificação multirrótulo. A Seção 2.3 aborda sobre classificação hierárquica. A Seção 2.4 aborda os conceitos, as técnicas e as medidas de avaliação referentes a classificação hierárquica multirrótulo. Por fim, a Seção 2.5 apresenta as considerações finais do Capítulo.

2.1 CONCEITOS FUNDAMENTAIS SOBRE CLASSIFICAÇÃO

A Aprendizagem de Máquina tem como objetivo desenvolver técnicas computacionais que possibilitem que o computador tome decisões baseadas em experiências acumuladas. Dessa maneira, a AM é a área que trata a questão de como construir programas que melhorem seu desempenho automaticamente com a experiência (MONARD; BARANAUSKAS, 2002; MITCHELL, 1997).

A classificação é um dos problemas mais importantes da AM e da mineração de dados (FREITAS; CARVALHO, 2007). Esse processo faz parte de um tipo de aprendizagem denominada de aprendizagem supervisionada, em que são desenvolvidos algoritmos que realizam induções de classificadores a partir de exemplos previamente classificados. A Figura 1 ilustra esse processo de classificação.

Figura 1 – Processo de classificação de dados



Fonte: Adaptado de Rezende (2003)

Na aprendizagem supervisionada, tem-se um classificador utilizando exemplos que possuem a informação da sua saída esperada. Esse classificador é obtido por meio de um algoritmo de indução (indutor), que tem como objetivo fazer com que o classificador seja capaz de classificar corretamente novos exemplos (CERRI, 2010).

Inicialmente os dados brutos devem ser preparados em um conjunto de exemplos para que possam ser processados. Um conjunto de exemplos é composto por valores de atributos, que são características do exemplo, e pelo atributo classe.

Na Figura 1 é mostrado o formato padrão de um conjunto de exemplos T com m exemplos e n atributos. A linha i refere-se ao i-ésimo exemplo onde $i = 1, 2, 3, ..., n$ e a entrada $x_i j$ refere-se ao valor do j-ésimo atributo X_j do exemplo i, onde $j = 1, 2, 3, ..., m$ (BORGES, 2012).

Após o processamento dos dados, esse conjunto de exemplos será submetido à entrada do algoritmo de indução para que seja feito o treinamento do classificador. O objetivo do treinamento é encontrar uma função que mapeie cada exemplo T_i com a sua classe y_i correspondente. Segundo TAN *et al.*, o processo de classificação consiste do aprendizado de uma função objetivo f que mapeia cada conjunto de atributos T_i em uma das classes predefinidas y_i (TAN *et al.*, 2006).

Depois de terminada a fase de treinamento tem-se um classificador que deve ser capaz de predizer corretamente o rótulo de novos exemplos, que ainda não foram rotulados (REZENDE, 2003).

2.2 CLASSIFICAÇÃO MULTIRRÓTULO

Existe uma grande quantidade de problemas em que alguns exemplos dos dados podem pertencer a mais de uma classe (rótulo) simultaneamente. Esse grupo de problemas é conhecido como classificação multirrótulo (TROHIDIS *et al.*, 2008).

Para uma descrição formal, seja $L = \lambda_j : j = 1, ..., M$ um conjunto finito de classes (rótulos) em uma tarefa de aprendizado multirrótulo e $D = \{\{\bar{x}_i | L_i\}, i = 1, ...N\}$ o conjunto de exemplos de treinamento multirrótulo, onde $\bar{x}_i$ é o vetor de características (atributos) e o $L_i \subseteq Y$ conjunto de classes da i-ésima instância. No Quadro 1 são apresentados quatro exemplos rotulados com um ou mais rótulos.

Quadro 1 – Exemplo de uma base de dados multirrótulo

Exemplos	**Rótulos**
1	$\{\lambda_1, \lambda_4\}$
2	$\{\lambda_3, \lambda_4\}$
3	$\{\lambda_1\}$
4	$\{\lambda_2, \lambda_3, \lambda_4\}$

Fonte: Tsoumakas et. al (2010)

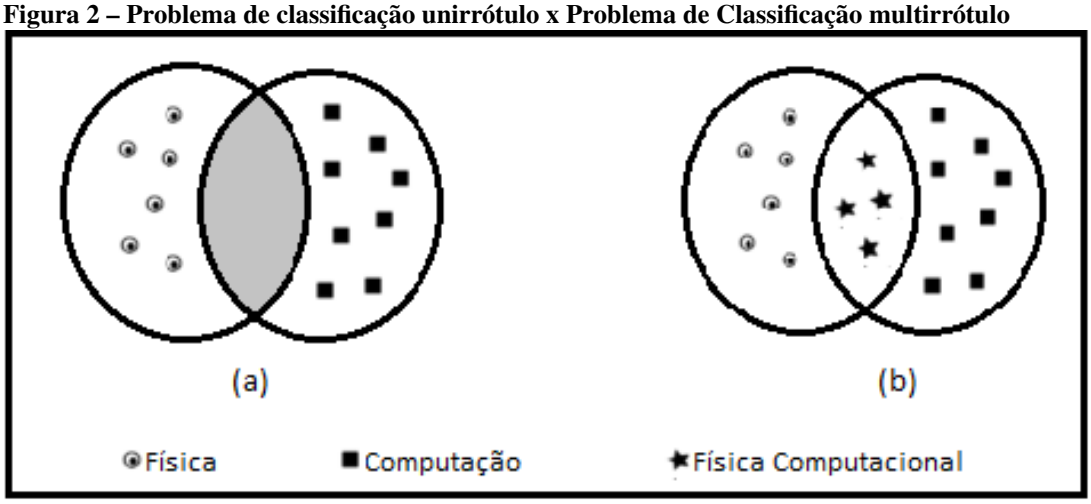
Em um problema de classificação de textos, por exemplo, cada documento pode pertencer simultaneamente a mais de uma classe (ou tópico). Um documento, por exemplo, pode ser classificado como pertencente à área de Ciência da Computação e Física. Um artigo de jornal pode ser classificado como artes, cinema e religião (NAM *et al.*,2014; NANCULEF; FLAOU-

NAS; CRISTIANINI,2014).

Na área de diagnósticos médicos, um paciente pode sofrer de diabetes e câncer de próstata ao mesmo tempo (TSOUMAKAS; KATAKIS, 2006). A área de classificação de textos é a que tem maior aplicação de técnicas de classificação multirrótulo (GONÇALVES; QUARESMA, 2003; LAUSER; HOTHO, 2003; LUO; ZINCIR-HEYWOOD, 2005). Entretanto, muitos trabalhos podem ser encontrados nas áreas de Bioinformática (CLARE; KING, 2001; ELISSEEFF; WESTON, 2002; ZHANG; ZHOU, 2007) e classificação de imagens (BOUTELL *et al.*, 2004; SHEN *et al.*, 2004).

A Figura 2a ilustra um problema de classificação no qual um documento pode pertencer ou à classe "Física" ou à classe "Ciência da Computação", mas nunca às duas classes ao mesmo tempo. Já na Figura 2b é ilustrado um exemplo de classificação multirrótulo, em que os documentos pertencentes simultaneamente às classes "Física" e "Ciência da Computação" são classificados na classe "Física Computacional".

Figura 2 – Problema de classificação unirrótulo x Problema de Classificação multirrótulo



Fonte: Adaptado de Cerri (2010)

Diferentes técnicas têm sido propostas na literatura para tratar problemas de classificação multirrótulo. Em algumas dessas técnicas, classificadores simples-rótulo podem ser combinados para tratar problemas de classificação multirrótulo.

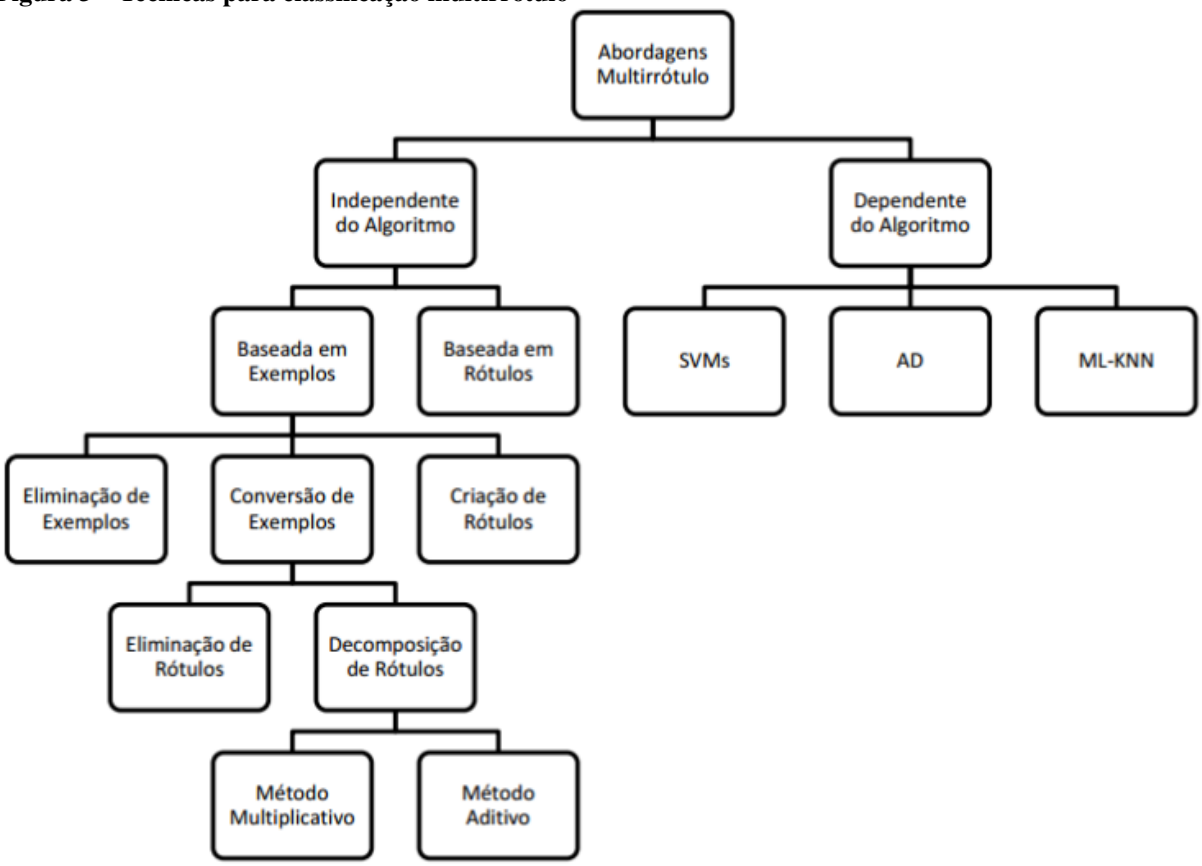
Outras técnicas modificam classificadores simples-rótulo, através de adaptações em seus mecanismos internos, para permitir que sejam utilizados em problemas multirrótulo. Ainda, novos algoritmos podem ser desenvolvidos especificamente para tratar problemas de classificação multirrótulo (FREITAS; CARVALHO, 2007).

Essas técnicas, podem ser divididas em duas abordagens principais: abordagem independente de algoritmo e abordagem dependente de algoritmo. Na abordagem independente de algoritmo, também chamada de transformação do problema, problemas de classificação multirrótulo são tratados utilizando qualquer algoritmo de classificação tradicional. A ideia é transformar o problema original em um conjunto de problemas de classificação simples-rótulo. Essa transformação pode ser realizada baseando-se nos rótulos de classe ou nos exemplos.

Na transformação baseada em rótulos de classe, conhecida como *Binary Relevance*, são utilizados L classificadores, sendo L o número de classes que estão envolvidas no problema. Para cada classificador é então associado a uma classe e treinado para resolver um problema de classificação binária, na qual é considerada a classe à qual ele está associado contra todas as outras classes envolvidas.

Já na transformação baseada nos exemplos, o conjunto de classes associado a cada exemplo é definido de maneira a converter o problema multirrótulo original em um ou mais problemas simples-rótulo (FREITAS; CARVALHO, 2007).

Figura 3 – Técnicas para classificação multirrótulo



Fonte: Adaptado de Cerri (2010)

Por sua vez na abordagem dependente de algoritmo, como o próprio nome sugere, novos algoritmos são propostos para tratar problemas de classificação multirrótulo. Tais algoritmos podem ser desenvolvidos especificamente para classificação multirrótulo ou serem baseados em técnicas de classificação convencionais, como SVMs e árvores de decisão (CARVALHO; FREITAS, 2009).

2.2.1 Métodos de Classificação Multirrótulo: Transformação do problema

A abordagem de transformação do problema, também chamada de abordagem independente de algoritmo, consiste na conversão de uma tarefa multirrótulo em um conjunto de tarefas unirrótulo. A ideia é transformar o problema original em um conjunto de problemas de classificação unirrótulo. Assim, torna-se possível a utilização de qualquer algoritmo tradicional de classificação para tratar o problema multirrótulo (TSOUMAKAS; KATAKIS, 2006).

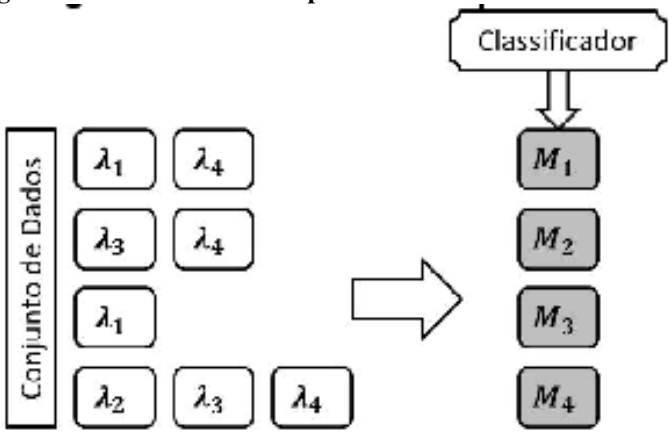
Na literatura há vários trabalhos propondo métodos de transformação simples que podem ser usados para converter um conjunto de dados multirrótulo em um conjunto de dados unirrótulo com o mesmo conjunto de rótulos, como por exemplo em (BOUTELL *et al.*, 2004; TSOUMAKAS; KATAKIS, 2006). Dentre os métodos utilizados podem ser citados: *Label Powerset* (TSOUMAKAS; KATAKIS; VLAHAVAS, 2010), *Binary Relevance* (TSOUMAKAS; VLAHAVAS, 2007), *RAndom k-LabELsets* (TSOUMAKAS; KATAKIS; VLAHAVAS, 2011), dentro outros.

2.2.1.1 *Label Powerset*

O método LP proposto em Tsoumakas et al. (2010) é um método simples de transformação do problema multirrótulo em um problema multiclasse. Ele considera cada conjunto de rótulos que existe em um conjunto de treinamento multirrótulo como uma das classes de uma nova tarefa de classificação de rótulo único (TSOUMAKAS; KATAKIS; VLAHAVAS, 2010).

Com o método, os multirrótulos são transformados em unirrótulos, utilizando por exemplo a concatenação dos múltiplos rótulos para a criação de um novo unirrótulo. Outras transformações podem ser utilizadas, como a criação de um índice único para cada combinação de multirrótulos no conjunto de treinamento. No Quadro 1 e na Figura 4 encontra-se ilustrado o processo utilizado pelo algoritmo LP.

Figura 4 – Processo utilizado pelo LP



Fonte: Adaptado de Tsoumakas, Katakis e Vlahavas (2010)

Uma desvantagem desse método é que a transformação do problema multirrótulo para problemas unirrótulo pode resultar em uma tarefa com muitas classes a serem preditas, uma vez que devem ser consideradas todas as combinações únicas de rótulos presentes no conjunto de treinamento como valores distintos do atributo classe.

Outro problema recorrente nessa abordagem é o desbalanceamento entre as classes, onde, usualmente muitas classes apresentam frequência baixa enquanto que poucas classes apresentam alta frequência (TSOUMAKAS; KATAKIS; VLAHAVAS, 2010).

A vantagem desse método, além da simplicidade, é que ele mantém a relação entre os rótulos de um mesmo exemplo, o que pode incrementar a precisão dos modelos gerados usando este método.

2.2.1.2 *Binary Relevance*

Dentre todos os métodos de transformação do problema, o mais popular é o BR, por tratar-se de um método simples e eficiente, apresentando complexidade linear com o número de monorrótulos no conjunto de dados.Essa técnica realiza a decomposição do problema multirrótulo em vários subproblemas binários.

Esse método, ainda apresenta independência entre os classificadores binários permitindo com que todos os classificadores sejam construídos em paralelo, consequentemente diminuindo ainda mais o tempo para construção do modelo final.

Inicialmente o conjunto de treinamento, cujos exemplos possuem mais de um rótulo é transformado em $|L|$ problemas de classificação unirrótulo binário, onde $|L|$ é a quantidade de rótulos contidos em L. Assim, pode-se afirmar que a predição de cada rótulo é considerada como uma tarefa independente (TSOUMAKAS; KATAKIS, 2006).

Para cada problema unirrótulo binário, o conjunto de exemplos é replicado e os rótulos desses exemplos são modificados, de modo que cada exemplo seja positivo ou negativo para esse problema. Na Figura 2 são apresentados os conjuntos de dados produzidos ao utilizar o método BR e a Figura 5 ilustra o processo utilizado pelo BR.

Quadro 2 – Conjunto de dados obtidos ao utilizar o método BR nos dados da Quadro 1

Exemplo	Rótulos
1	$\{\lambda_1\}$
2	$\{\neg\lambda_1\}$
3	$\{\lambda_1\}$
4	$\{\neg\lambda_1\}$

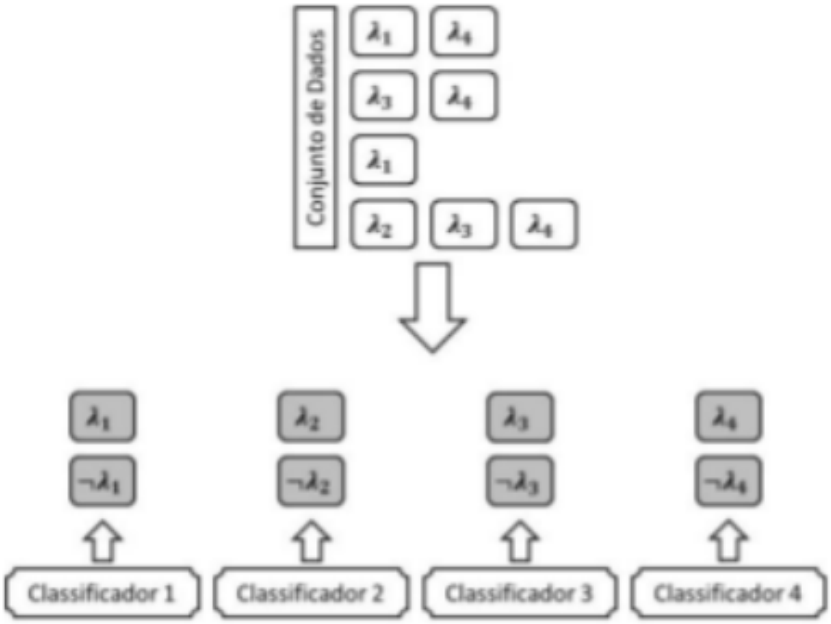
Exemplo	Rótulos
1	$\{\neg\lambda_2\}$
2	$\{\neg\lambda_2\}$
3	$\{\neg\lambda_2\}$
4	$\{\lambda_2\}$

Exemplo	Rótulos
1	$\{\neg\lambda_3\}$
2	$\{\lambda_3\}$
3	$\{\neg\lambda_3\}$
4	$\{\lambda_3\}$

Exemplo	Rótulos
1	$\{\lambda_4\}$
2	$\{\lambda_4\}$
3	$\{\neg\lambda_4\}$
4	$\{\lambda_4\}$

Fonte: Adaptado de Tsoumakas e Katakis (2006)

Figura 5 – Processo utilizado pelo BR



Fonte: Adaptado de Tsoumakas e Katakis (2006)

A principal desvantagem desse método é não considerar a dependência de rótulos na construção do modelo de classificação multirrótulo, uma vez que cada classificador binário é construído de maneira independente dos demais (TSOUMAKAS; VLAHAVAS, 2007).

2.2.1.3 *Random k-labelsets*

Tendo como objetivo minimizar os problemas do LP mencionados anteriormente, em Tsoumakas et al. (2010) foi proposta uma abordagem na qual, ao mesmo tempo em que são consideradas as correlações entre os rótulos, busca-se evitar o problema de suscetibilidade à ocorrência de muitas classes com poucos exemplos do LP. Nessa nova abordagem, chamada RAkEL, k é um parâmetro que especifica o tamanho dos subconjuntos (TSOUMAKAS; VLAHAVAS, 2007).

A evidência empírica indica que o RAkEL consegue melhorar substancialmente ao longo LP, especialmente em domínios com grande número de rótulos e apresenta um desempenho competitivo em relação a outras de alto desempenho métodos de aprendizagem multirrótulo (TSOUMAKAS; KATAKIS; VLAHAVAS, 2011).

No RAkEL é construído um comitê de classificadores LP, onde cada classificador é treinado usando um diferente subconjunto aleatório de labelsets. Assim, pode-se afirmar que no RAkEL, os classificadores unirrótulo, além de considerar as correlações entre os rótulos, são aplicados em subtarefas com um número gerenciável de rótulos e número adequado de exemplos por rótulo (TSOUMAKAS; KATAKIS; VLAHAVAS, 2010).

Ainda em Tsoumakas et al. (2010) é descrita uma extensão do RAkEL, chamada RA-

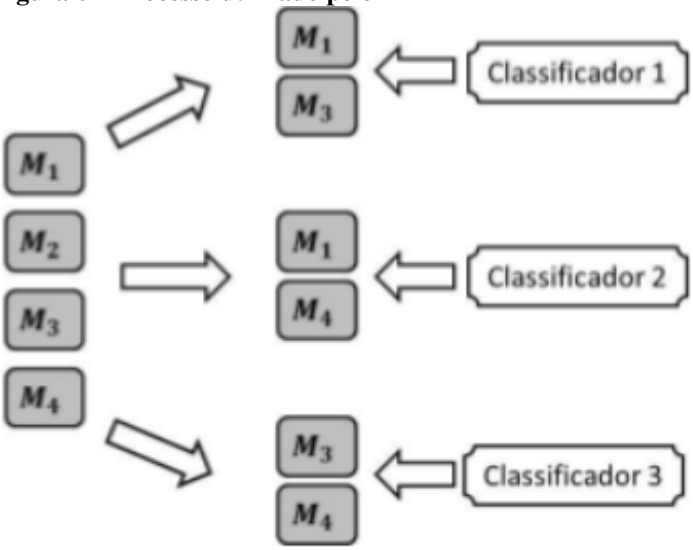
kELd. Enquanto no RAkEL original é permitida a sobreposição dos labelsets, no RAkELd, os labelsets de cada classificador do $ensemble$ são disjuntos (TSOUMAKAS; KATAKIS; VLAHAVAS, 2010). O Quadro 3 mostra o resultado da transformação do conjunto de dados do Quadro 1 usando o método RAkEL, enquanto que a Figura 6 ilustra o processo realizado.

Quadro 3 – Conjunto de dados transformados usando o método RAKEL

Exemplos	Rótulos	Metarrótulo
1	$\{\lambda_1, \lambda_4\}$	M_1
2	$\{\lambda_3, \lambda_4\}$	M_2
3	$\{\lambda_1\}$	M_3
4	$\{\lambda_2, \lambda_3, \lambda_4\}$	M_4

Fonte: Adaptado de Tsoumakas e Katakis (2006)

Figura 6 – Processo utilizado pelo RAkEL



Fonte: Adaptado de Tsoumakas e Katakis (2006)

2.2.2 Métodos de Classificação Multirrótulo: Adaptação de algoritmos

Os métodos de adaptação de algoritmos são gerados como resultado de uma extensão de um determinado algoritmo de aprendizado unirrótulo, possibilitando assim a manipulação de dados multirrótulo diretamente.

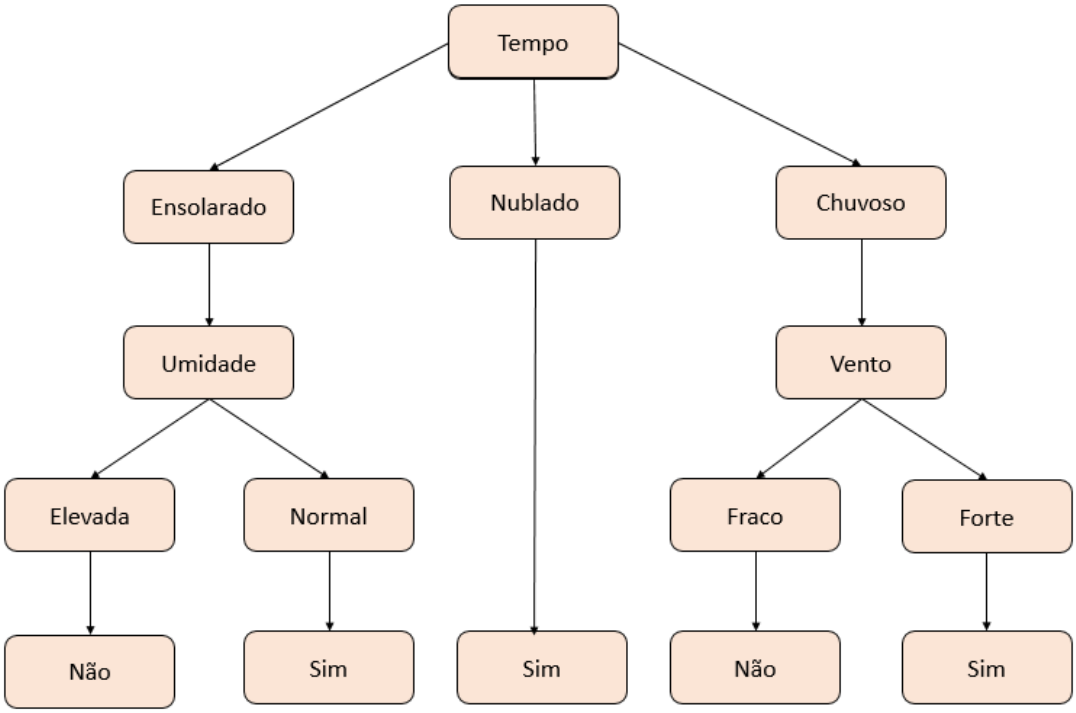
Na literatura, há uma variedade de trabalhos tratando da adaptação das mais variadas técnicas de aprendizado de máquina, como por exemplo: Algoritmos de Indução de Árvores de Decisão (CLARE; KING, 2001), k vizinhos mais próximos (ZHANG; ZHOU, 2007), AdaBoost (SCHAPIRE; SINGER, 2000), dentre outras.

2.2.2.1 Árvore de decisão

Uma maneira natural e intuitiva de se classificar um padrão é por meio de uma sequência de decisões, em que a próxima decisão depende da decisão atual (MITCHELL, 1997). Essa sequência de decisões pode ser representada por uma estrutura de dados do tipo árvore, a qual é definida recursivamente como: um nó folha que corresponde a uma classe ou um nó interno de decisão, que contém uma decisão sobre algum atributo.

Para cada resultado da decisão, existe uma aresta para uma subárvore. Essa técnica de classificação está entre as mais populares e tem sido aplicada em tarefas como, por exemplo, diagnóstico médico (MITCHELL, 1997). Na Figura 7 é apresentado um exemplo ilustrativo de uma árvore de decisão que utiliza informações climáticas para realizar a inferência se será possível jogar tênis em um determinado dia.

Figura 7 – Exemplo de árvore de decisão



Fonte: Adaptado de Mitchell (1997)

Clare et al. (2001), em seu trabalho adaptou a fórmula de cálculo da entropia de modo a viabilizar a manipulação de dados multirrótulo. Em um dos algoritmos mais utilizados para construção de árvores de decisão, chamado J48, os nós da árvore de decisão são definidos através dessa medida de entropia (CLARE; KING, 2001).

Outra modificação proposta por Clare et al. (2001) altera a funcionalidade dos nós-folha da árvore de decisão, os quais passam a representar um subconjunto de rótulos e não apenas um único rótulo como no algoritmo original (CLARE; KING, 2001).

2.2.2.2 *Multi-Label k-Nearest Neighbor*

No trabalho de Zhang et al. (2007) foi proposto o método *MultiLabel k-Nearest Neighbors*, diferenciado pelo uso de probabilidades a $priori$ e a $posteriori$. É um classificador multirrótulo construído baseado no popular método kNN (FIX; JR, 1952).

Para cada d_j elemento de teste, o algoritmo encontra seus k vizinhos mais próximos no conjunto de treino usado, isto é, encontra os k primeiros elementos ordenados pelo valor de similaridade com d_j de forma decrescente, usando por exemplo, a distância Euclidiana.

Na sequência, o algoritmo identifica quantos exemplares de cada categoria existem dentre os k vizinhos mais próximos de d_j, que chamaremos de $k(i = 1, ..., |C|)$, onde $|C|$ é o número de categorias. Seja H_1^i o evento em que d_j possui o rótulo i e H_0^i o evento em que d_j não possui o rótulo i. E mais, seja E_j^i o evento em que existem j vizinhos mais próximos de d_j pertencentes á categoria i.

Após isso, o ML-kNN faz o seguinte: para cada exemplar w_y no conjunto de treinamento, onde $(y = 1, ..., N)$, o algoritmo encontra seus k vizinhos mais próximos e calcula o número total de votos que cada categoria recebe dos k vizinhos mais próximos. Em outras palavras, seja k_i o número de votos que cada categoria i recebeu de w_y, se o exemplar w_y pertence à categoria i, então será adicionado 1, senão será adicionado 0. Finalmente com essas informações, as probabilidades posteriori são calculadas.

O ML-kNN precisa apenas de dois parâmetros: o número k de vizinhos mais próximos e a suavização σ da probabilidade. Uma generalização do método ML-kNN possibilita considerar a dependência de rótulos durante o aprendizado multirrótulo (YOUNES *et al.*, 2011). De modo similar a outros algoritmos baseados no kNN, cada exemplo a ser predito nesse método tem seus vizinhos identificados no conjunto de treinamento.

O princípio *Maximum a Posteriori* é utilizado em escopo global para atribuir um conjunto de rótulos para um exemplo a ser predito, de modo a oferecer suporte para o tratamento da dependência de rótulos. Esse princípio possibilita, por exemplo, que o número de rótulos distintos na vizinhança seja considerado durante o processo de predição, diferentemente do que ocorre no ML-kNN (ZHANG; ZHOU, 2007).

Em Almeida e Borges (2017) foi realizada uma adaptação desse algoritmo para problemas de classificação hierárquica multirrótulo em que a base de dados está estruturada em formato de DAG. Nesse trabalho foram realizados experimentos em bases de dados de funções gênicas e foram utilizadas como medida de desempenho as medidas hierárquicas de precisão e revocação (ALMEIDA; BORGES, 2017).

2.2.2.3 AdaBoost

O AdaBoost é um algoritmo de aprendizado supervisionado do tipo boost. O AdaBoost combina um conjunto de funções simples de classificação, denominadas classificadores fracos para formar um classificador forte (FREUND; SCHAPIRE, 1995).

Segundo Freund et al. (1995), um classificador forte é composto de um conjunto de classificadores fracos, associados a pesos que classificam de forma precisa dois conjuntos de dados pré-rotulados, onde as características com pesos maiores são mais significativas para a classificação de exemplos definidos como parte de um certo conjunto.

Em Schapire et al. (1999) e Schapire et al. (2000), são propostas duas extensões para o algoritmo AdaBoost, de maneira a permitir seu uso em problemas multirrótulos. Para o AdaBoost.MH, foi feita uma modificação na maneira de se avaliar o desempenho preditivo do modelo induzido, verificando sua capacidade de predizer um conjunto correto de classes para um dado exemplo. No AdaBoost.MR, foi feita uma mudança no algoritmo que faz com que ele passe a predizer um ranking de classes para cada exemplo de entrada (SCHAPIRE; SINGER, 1999; SCHAPIRE; SINGER, 2000).

2.2.2.4 Métodos Probabilísticos

Esse método assume a probabilidade de um evento (ex. um paciente doente) dado um evento ocorrer (ex. o resultado de exames realizados nesse paciente é positivo). No contexto de AM (MITCHELL, 1997), a probabilidade é a frequência relativa de objetos que contém o valor de uma classe no conjunto de dados. Os objetos são descritos por um conjunto de valores de atributos de entrada que representa o evento .

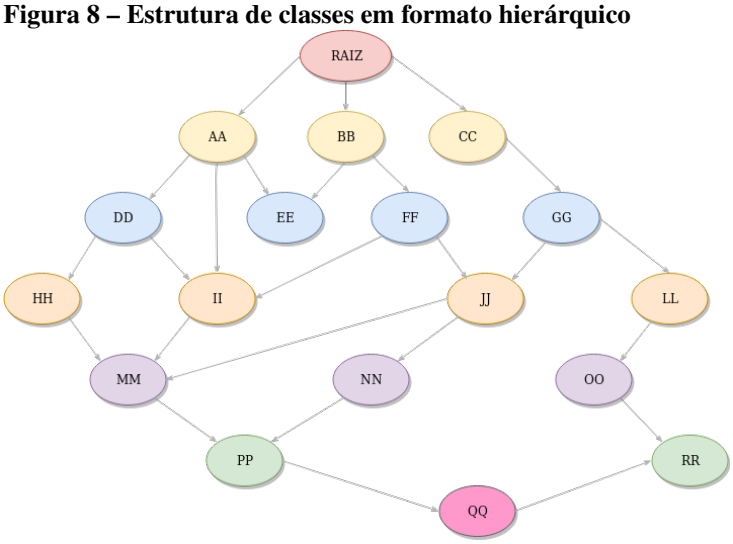
Em Mccallum (1999), é proposto um modelo probabilístico generativo, aplicado à tarefa de classificação de documentos. Neste modelo, é utilizada uma abordagem de classificação bayesiana, na qual cada rótulo gera diferentes palavras. Baseado neste modelo, um documento multirrótulo é produzido por uma mistura das distribuições das palavras de seus rótulos (MCCALLUM, 1999).

Em seu trabalho, Ghanrawi et al. (2005), explora o uso de campos aleatórios condicionais no qual dois modelos gráficos que parametrizam co-ocorrências de rótulos são apresentados. O primeiro, multirrótulo coletivo, captura padrões de co-ocorrências entre rótulos, enquanto que o segundo, multirrótulo coletivo com características, tenta capturar o impacto que uma característica individual tem sobre a probabilidade de co-ocorrência de um par de rótulos (GHAMRAWI; MCCALLUM, 2005).

2.3 CLASSIFICAÇÃO HIERÁRQUICA

Nos problemas de classificação hierárquica, as classes podem apresentar uma relação de taxonomia ou dependência, formando subclasses e superclasses (FREITAS; CARVALHO, 2007). Na Figura 8 pode-se notar um exemplo de hierarquia entre as classes de uma base de dados.

Figura 8 – Estrutura de classes em formato hierárquico



Fonte: Autoria própria (2017)

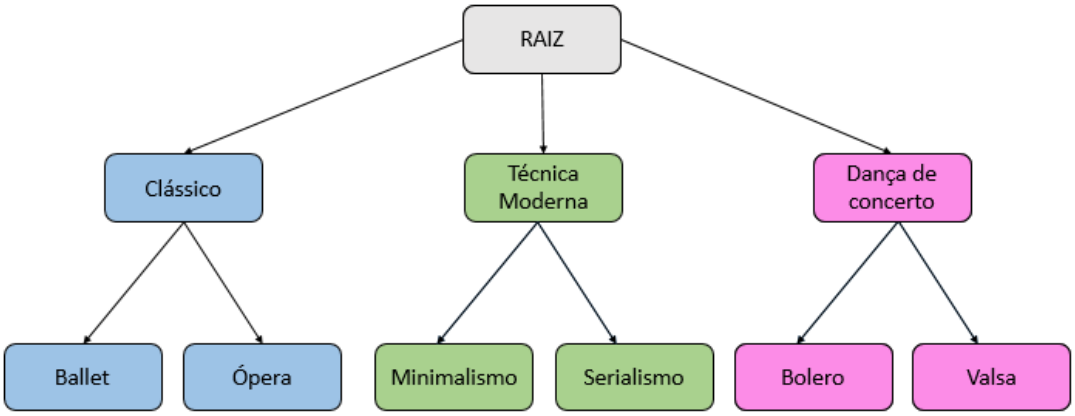
É possível distinguir diferentes tipos de problemas de classificação hierárquica, que variam de acordo com três características (SILLA JR.; FREITAS, 2011):

- Tipo de hierarquia em que as classes se organizam, que pode ser em forma de árvore ou de um DAG.
- Se os dados podem ou não seguir mais de um caminho na hierarquia. Caso possam, tem-se um problema de classificação hierárquica com múltiplas classes.
- A profundidade das rotulações dos dados. Tem-se dois casos: todos os objetos possuem rotulação até os nós folhas, que representam os níveis mais profundos da hierarquia; ou ao menos um dos objetos possui uma rotulação parcial, que não atinge um nó folha.

Problemas de classificação hierárquica têm por objetivo a classificação de cada novo dado de entrada em um dos nós-folhas, pois quanto mais profunda a classe na hierarquia, mais especifico e útil é o conhecimento (FREITAS; CARVALHO, 2007).

Um importante aspecto que caracteriza um problema de classificação hierárquica é o tipo de hierarquia empregado para representar os relacionamentos entre as classes. Existem duas maneiras em que as classes podem estar dispostas hierarquicamente: como uma árvore ou uma DAG. A Figura 9 apresenta um exemplo de classificação de gêneros musicais em que as classes estão dispostas hierarquicamente no formato de uma árvore.

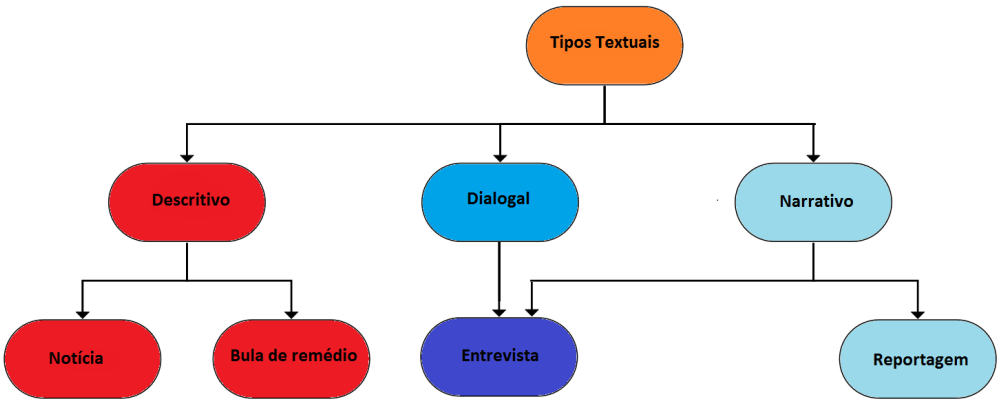
Figura 9 – Classificação de gêneros musicais



Fonte: Autoria própria (2017)

Na Figura 10 apresenta-se um exemplo de classificação de gêneros textuais, ressalta-se o fato de que ao classificar o tipo entrevista, o exemplo pertencerá tanto a classe Narrativo quanto a classe Dialogal.

Figura 10 – Classificação de gêneros textuais



Fonte: Autoria própria (2017)

A principal diferença entre a estrutura em árvore apresentada na Figura 9 e a estruturada em DAG na Figura 10 é que, na estrutura em árvore, cada nó, exceto o nó-raiz, tem somente um nó-pai, enquanto que no DAG cada nó, exceto o nó-raiz, pode ter um ou mais nós-pai.

2.4 CLASSIFICAÇÃO HIERÁRQUICA MULTIRRÓTULO

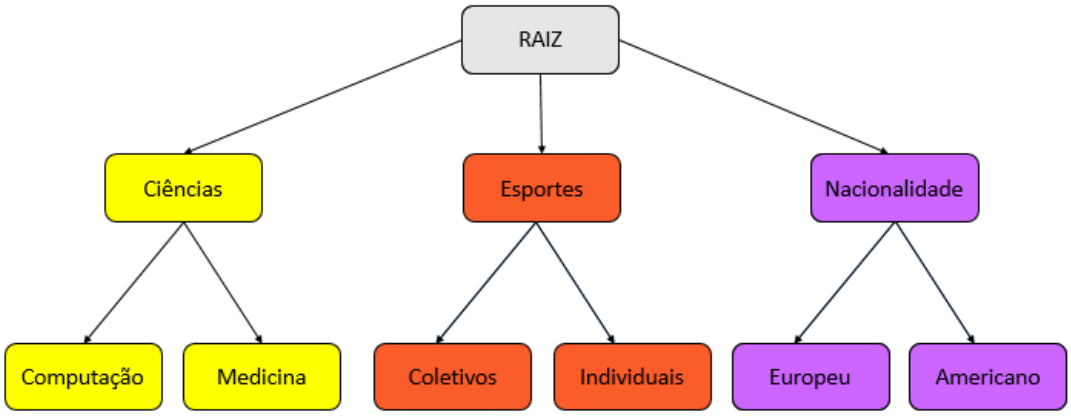
A classificação hierárquica multirrótulo tem surgido como uma nova categoria de problemas de classificação, com características tanto dos problemas de classificação multirrótulo, quanto de problemas de classificação hierárquica. Problemas pertencentes a esta nova categoria são denominados de problemas de classificação hierárquica multirrótulo.

Em um problema de classificação hierárquica multirrótulo, um exemplo pode pertencer a múltiplas classes ao mesmo tempo e essas classes são organizadas de maneira hierárquica

(CERRI, 2010). Na Figura 11 é apresentado um exemplo de problema de classificação hierárquica multirrótulo, onde a hierarquia de classes é representada através de uma árvore.

Na Figura 12 é apresentado um exemplo de predição hierárquica multirrótulo onde uma notícia de um jornal está associada à Ciência da Computação e Futebol, e pode então ser classificado tanto como Ciências/Computação, quanto como Esportes/Coletivos/Futebol. Como saída, do processo de predição é obtida uma subárvore da árvore original.

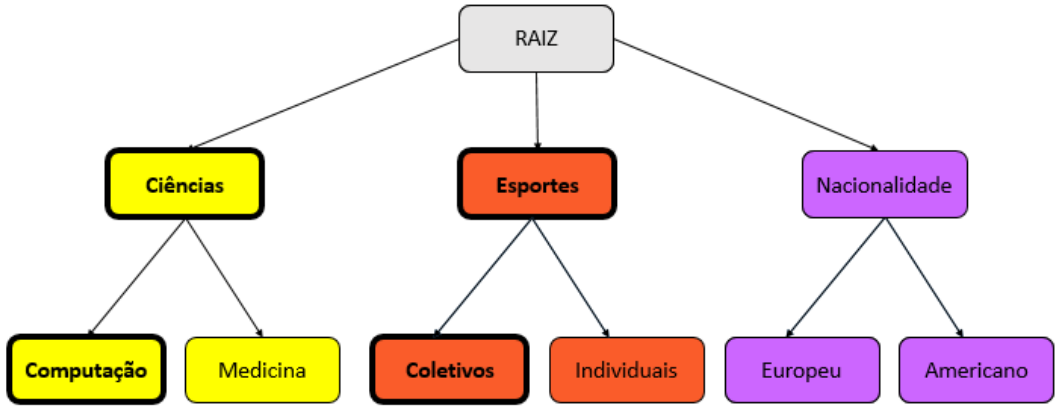
Figura 11 – Hierarquia de classes de um problema hierarquico multirrótulo estruturado como uma árvore



Fonte: Adaptado de Cerri (2010)

O critério de qualidade pode ser baseado na distância entre as classes da hierarquia, podendo ser, por exemplo, a precisão média das diferentes classes preditas, ou pode considerar que erros de classificação em níveis da hierarquia mais próximos da raiz são piores que em níveis mais profundos (STRUYF *et al.*, 2005).

Figura 12 – Predições usando uma subárvore



Fonte: Adaptado de Cerri (2010)

Esse tipo de problema é bastante comum, principalmente em problemas de categorização de texto e classificação de genes e proteínas quanto a suas funções. Pode-se dizer também que problemas de classificação hierárquica multirrótulo são mais complexos que os demais problemas de classificação, uma vez que as classes envolvidas no problema, além de estarem

estruturadas em uma hierarquia, os exemplos podem pertencer a mais de uma classe ao mesmo tempo (CERRI, 2010).

2.4.1 Métodos de Classificação Hierárquica Multirrótulo

Vários métodos podem ser utilizados no tratamento de tarefas de classificação hierárquica multirrótulo. Na literatura, há vários trabalhos propondo e analisando abordagens e métodos para tratamentos de problemas hierárquico multirrótulo de diferentes domínios, tais como: classificação de texto (ROUSU *et al.*, 2006), predição de funções genômicas e proteínas (BLOCKEEL *et al.*, 2006;CLARE, 2003;STRUYF *et al.*, 2005;VENS *et al.*, 2008;BORGES, 2012; ALMEIDA; BORGES,2017). Contudo, não há um consenso sobre que abordagem utilizar para o tratamento de problemas hierárquicos multirrótulo.

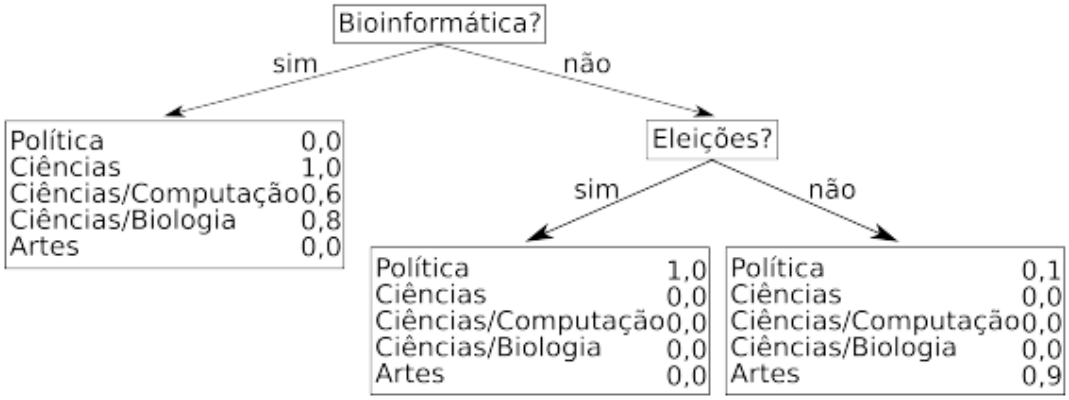
2.4.1.1 Clus-HMC

O algoritmo Clus-HMC constrói classificadores baseados em árvores de decisão e vem sendo investigado e melhorado, ao longo dos anos (RAEDT; BLOCKEEL, 1997). A versão mais utilizada, gera classificadores globais para classificação hierárquica multirrótulo. Deste modo, cada nó folha prevê uma ou mais classes (VENS *et al.*, 2008).

O Clus-HMC é uma técnica baseada na noção de *Predictive Clustering Trees* (PCT). Esse algoritmo possui a capacidade de lidar com estruturas hierárquicas na forma de árvores e DAGs. Nessa técnica, a árvore de decisão é estruturada como uma hierarquia de grupos. A idéia geral é particionar o conjunto de classes em grupos, de maneira que a distância intra-grupos seja minimizada (BLOCKEEL *et al.*, 2006).

Neste algoritmo, as árvores de decisão são vistas como uma hierarquia de grupos (clusters), na qual o nó raiz contém todos os exemplos de treinamento, e é recursivamente particionado em grupos menores, a medida que se percorre a árvore de decisão em direção ás folhas. As árvores de decisão baseadas em PCT podem ser aplicadas tanto para a tarefa de agrupamento quanto classificação (RAEDT; BLOCKEEL, 1997). A Figura 13 apresenta um exemplo de uma PCT (CERRI, 2010).

Figura 13 – Exemplo de PCT



Fonte: Adaptado de Cerri (2010)

2.4.2 Multi-Label Hierarchical Classification with an Artificial Immune System

O algoritmo MHCAIS proposto por Alves (2010) é baseado em Sistemas Imunológicos Artificiais para classificação hierárquica multirrótulo, onde os classificadores gerados são representados na forma de regras SE-ENTÃO (ALVES, 2010).

Há duas versões do classificador MHCAIS, uma com abordagem global e a outra com abordagem local. Na abordagem local, cada classificador processa apenas exemplos de classes em uma determinada região da hierarquia. Essa versão do classificador, apenas diferencia se um exemplo pode ou não ser associado á classe para a qual aquele classificador foi treinado.Na versão global, por sua vez, um único classificador processa exemplos de todas as classes ao mesmo tempo.

O algoritmo MHCAIS possui dois procedimentos fundamentais, um para Extração Sequencial de Regras (*Sequential Covering*) e outro para Evolução das Regras (*Rule Evolution*). O primeiro procedimento, Extração Sequencial de Regras, é usado em algoritmos de indução de regras para tratar problemas de predição de uma única classe.

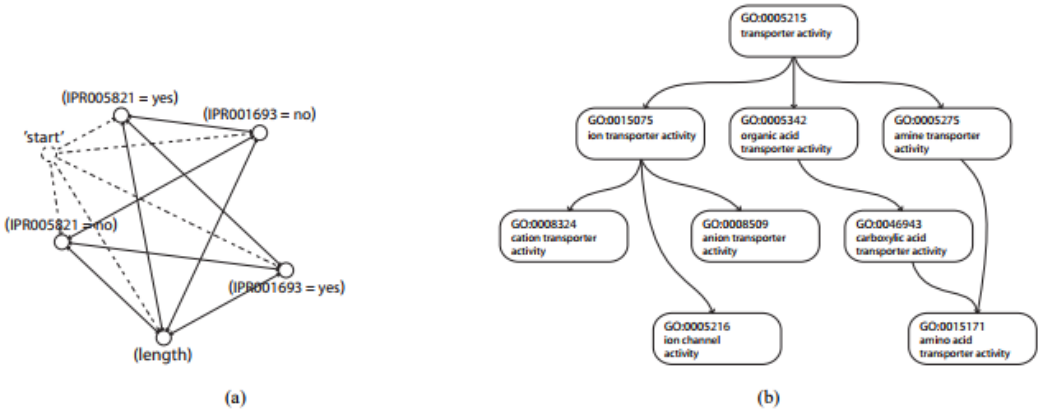
O segundo procedimento, é específico para algoritmos evolutivos ou relacionados, incluindo os SIAs baseados no princípio de seleção clonal. Os artigos participam de um processo iterativo (evolução), cujo objetivo é encontrar as melhores regras (anticorpos) que formarão a solução (classificador) para o problema. A construção do classificador termina quando o MHCAIS descobre o número de regras necessárias para classificar os exemplos da base de dados (ALVES, 2010).

2.4.2.1 Hierarchical Multi-Label Classification Anti-Miner

O algoritmo hmAnt-Miner proposto por Otero et al. (2010), trata-se de um classificador hierárquico multirrótulo global baseado no algoritmo bioinspirado Colônia de Formigas. Esse algoritmo descobre um único modelo de classificação global, através de uma lista de regras se-então, que pode prever todas as classes a partir da hierarquia de uma só vez (OTERO; FREITAS; JOHNSON, 2010).

Para guardar as informações da hierarquia de classes, o hmAnt-Miner emprega uma medida de distância baseada no procedimento de discretização dinâmica dos atributos contínuos e uma informação heurística na construção do grafo. A Figura 14 exemplifica a construção do grafo empregado no algoritmo hmAnt-Miner.

Figura 14 – Modelo do hmAnt-Miner



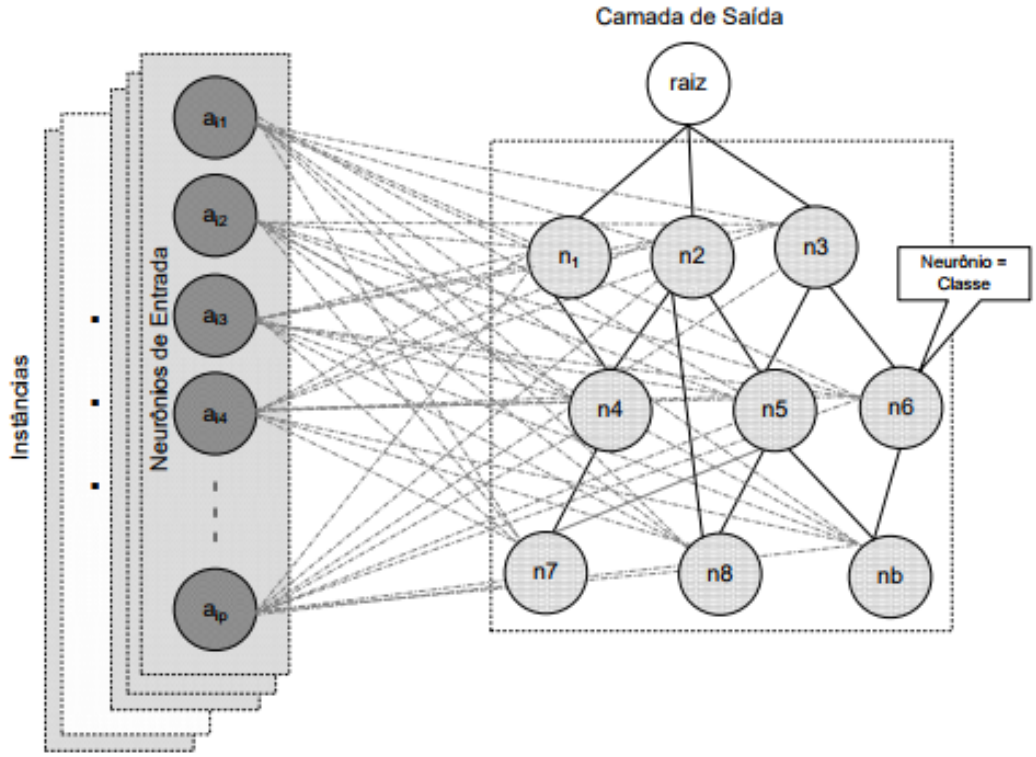
Fonte: Otero et. al(2010)

2.4.2.2 Multi-label Hierarchical Classification using a Competitive Neural Network

O algoritmo MHC-CNN é um classificador hierárquico multirrótulo usando uma abordagem de classificação global baseado em uma Rede Neural Artificial Competitiva (BORGES, 2012). Essa técnica baseia-se no aprendizado competitivo, em que os neurônios da camada de saída competem entre si para serem ativados existindo apenas um vencedor, que terá seus pesos atualizados juntamente com seus vizinhos.

Essa rede consiste de duas camadas de neurônios, sendo a camada de entrada conectada aos dados. No algoritmo MHC-CNN a topologia da camada de saída tem a estrutura de um grafo acíclico dirigido, em que cada neurônio está conectado com seus neurônios ancestrais (pais) e descendentes (filhos) e a todos os neurônios da camada de entrada. A Figura 15 apresenta o modelo do MHC-CNN.

Figura 15 – Modelo do MHC-CNN



Fonte: Borges (2012)

Os resultados das predições são avaliados através de duas abordagens de medidas de classificação hierárquica: medida baseada em distância dependente de profundidade e medida baseada na ancestralidade (sensibilidade e precisão hierárquicas).

2.4.3 Medidas de avaliação hierárquica

Apesar da maioria dos trabalhos apresentados na literatura avaliarem, de alguma maneira, o desempenho da classificação hierárquica multirrótulo, parece não haver ainda medidas definidas, ou adotadas por um grande número de trabalhos, para essa avaliação.

As medidas de avaliação podem ser divididas em algumas abordagens, sendo elas: medidas baseadas nas relações de ancestralidade e descendência, medidas baseadas em distância, medidas baseadas em similaridades. A escolha por essas medidas está em avaliar o resultado da classificação de diferentes maneiras.

2.4.3.1 Medidas baseadas nas relações de ancestralidade e descendência

Medidas baseadas em ancestralidade e descendência consideram os ancestrais e os descendentes das classes preditas no momento dos cálculos da avaliação. Dentre as medidas base-

adas nesse conceito tem-se: Precisão e Revocação Hierárquica e *Hierarchical Loss Function*.

2.4.3.1.1 Precisão e revocação hierárquica

No trabalho de Kiritchenko et al. (2004), foram propostas duas medidas de avaliação baseadas nas medidas convencionais de precisão e revocação, levando em consideração os relacionamentos hierárquicos entre as classes. Essas medidas foram chamadas de precisão e revocação hierárquicas e levam em consideração classificações nos nós internos e nós-folha (KIRITCHENKO; MATWIN; FAMILI, 2004).

Cada exemplo pertence não apenas à sua classe, mas também a todos os ancestrais dessa classe na estrutura hierárquica. Dessa maneira, dado um exemplo qualquer (x_i, Y_i), com x pertencente ao conjunto X de exemplos, Y_i o conjunto de classes preditas para o exemplo x_i e $Y_i^{'}$ o conjunto de classes verdadeiras do exemplo x_i, os conjuntos Y_i e $Y_i^{'}$ podem ser entendidos para conterem suas correspondentes classes ancestrais da seguinte maneira: $\hat{Y} = U_{y_i \epsilon Y_i} Ancestrais(y_k)$ e $\hat{Y} = U_{y_i \epsilon Y_i^{'}} Ancestrais(y_i)$

A precisão e revocação hierárquica (Prec e Rev) são calculadas utilizando as Equações 1 e 2, respectivamente.

$$Prec = \frac{\sum_i |\hat{Y_i} \cap \hat{Y_i^{'}}|}{\sum_i |\hat{Y_i}|} \tag{1}$$

$$Rev = \frac{\sum_i |\hat{Y_i} \cap \hat{Y_i^{'}}|}{\sum_i |\hat{Y_i^{'}}|} \tag{2}$$

Essas medidas contam o número de classes preditas corretamente, juntamente com o número de classes ancestrais dessas classes preditas corretamente, assumindo que exemplos também pertencem aos ancestrais de suas classes corretas (KIRITCHENKO; MATWIN; FAMILI, 2004).

A precisão e a revocação hierárquicas utilizadas sozinhas não são suficientes para a avaliação de classificadores. Sendo assim, as medidas Prec e Rev devem ser combinadas em uma extensão hierárquica da medida F-Measure, chamada FM, apresentada na Equação 3. A constante β, refere-se à importância atribuída aos valores de Prev e Rev. Quando aumenta-se o valor de β, aumenta-se o peso atribuído ao valor de Rev , e quando diminui-se β , aumenta-se o peso atribuído ao valor de Prec.

$$FM = \frac{(\beta^2 + 1) * Prec * Rev}{\beta^2 * Prec * Rev} \tag{3}$$

2.4.3.2 Hierarchical Loss Function

No trabalho de Cesa-Bianchi et al. (2006), foi proposta uma nova medida de avaliação chamada de Hierarchical Loss Function (H-Loss). Essa medida utiliza a noção intuitiva de que sempre que um erro de classificação é cometido em um nó da hierarquia de classes, não devem haver penalizações adicionais para erros cometidos na subárvore desse nó (CESA-BIANCHI; GENTILE; ZANIBONI, 2006).

Segundo os autores, dada uma estrutura hierárquica G, essa estrutura pode ser considerada uma floresta, composta por árvores definidas sobre o conjunto de classes do problema. Uma classificação multirrótulo $v \, \epsilon \, \{0,1\}^L$ respeita essa estrutura G se e somente se v for a união de um ou mais caminhos de G, onde cada caminho começa em uma raiz e necessariamente termina em uma folha. Assim, todos os caminhos em G, de uma raiz até a folha, são examinados.

Sempre que um nó i é encontrado, tal que $\hat{y}_i \neq v_i$, o valor 1 é adicionado à função H-Loss, e todas as predições na subárvore enraizada no nó i são descartadas. Dada essa definição, pode-se dizer que $l_{0/1} \leq l_H \leq l_\Delta$. A função H-Loss que leva em consideração os relacionamentos hierárquicos é definida na Equação 4, na qual $ANC\,(i)$ denota o conjunto de ancestrais do nó i.

$$l_H\,(\hat{y}, v) = \sum_{i=1}^{L} \{\hat{y}_i \neq v_i \wedge \hat{y}_j = v_j \, \epsilon \, ANC\,(i)\} \tag{4}$$

2.4.3.3 Medidas baseadas em distância

Como classes que estão mais perto umas das outras na hierarquia tendem a ser mais próximas entre si do que de outras classes, esse método considera a distância entre a classe verdadeira e a classe predita na hora de medir o desempenho do classificador.

No trabalho de Sun et al. (2001), foram utilizadas medidas chamadas de Micro/Macro Precisão Hierárquica e Micro/Macro Revocação Hierárquica. Segundo os autores, as medidas Micro Precisão e Revocação atribuem igual importância a todos os exemplos, enquanto as medidas Macro Precisão e revocação atribuem igual importância a todas as classes (SUN; LIM, 2001).

As medidas macro precisão/ revocação hierárquicas primeiramente medem o desempenho obtido em cada classe do problema separadamente, e então obtém uma média desses desempenhos. As medidas micro precisão/revocação hierárquicas, por outro lado, medem o desempenho médio obtido em cada exemplo do conjunto de dados. Assim, as macro medidas são consideradas como a média do desempenho por classe, enquanto as micro medidas são consi-

deradas a média do desempenho por exemplo do conjunto de dados (YANG, 1999).

O cálculo das medidas é realizado computando-se, para cada classe, a contribuição dos exemplos erroneamente atribuídos àquela classe. Para o cálculo dessa contribuição, é necessário que se defina uma distância aceitável entre duas classes, dada por , que deve ser maior que 0. A distância é igual ao número de arestas que separam uma classe predita de uma classe verdadeira na estrutura hierárquica.

A contribuição de um exemplo x_j a uma classe y_i é formalmente definida pelas Equações 5 e 6, nas quais $x_j.agd$ e $x_j.lbd$ são respectivamente as classes preditas e verdadeiras do exemplo x_j.

Se x_j é um Falso Positivo:

$$Con\left(x_j, y_i\right) = \sum_{y^{'} \epsilon\ x_j.agd} \left(1.0 - \frac{Dis\left(y^{'}, y_i\right)}{Dis_0}\right) \tag{5}$$

Se x_j é um Falso Negativo:

$$Con\left(x_j, y_i\right) = \sum_{y^{'} \epsilon\ x_j.lbd} \left(1.0 - \frac{Dis\left(y^{'}, y_i\right)}{Dis_0}\right) \tag{6}$$

Após seu cálculo, a contribuição de um exemplo x_j é então restringida á faixa de valores $[-1, 1]$. Esse refinamento, denotado por $RCon\left(x_j, y_j\right)$, é definido na Equação 7.

$$RCon\left(x_j, y_j\right) = min\left(1, max\left(-1, Con\left(x_j, y_j\right)\right)\right) \tag{7}$$

Para todos os exemplos, a contribuição total de falsos positivos $(FpCon_i)$ e falsos negativos $(FnCon_i)$ é definida nas Equações 8 e 9.

$$FpCon_i = \sum_{x_j\ \epsilon\ Fp_i} RCon\left(x_j, y_j\right) \tag{8}$$

$$FnCon_i = \sum_{x_j\ \epsilon\ Fn_i} RCon\left(x_j, y_j\right) \tag{9}$$

Após os cálculos das contribuições de cada exemplo, são calculados os valores da Precisão e da Revocação Hierárquica de cada classe. Os cálculos são apresentados nas Equações 10 e 11.

$$Pr_i^{CD} = \frac{max\left(0, |TP_i| + FpCon_i + FnCon_i\right)}{|TP_i| + |FP_i| + FnCon_i} \tag{10}$$

$$Re_i^{CD} = \frac{max\left(0, |TP_i| + FpCon_i + FnCon_i\right)}{|TP_i| + |FN_i| + FpCon_i} \tag{11}$$

Os valores estendidos da Precisão e Revocação Hierárquicas (Micro Precisão e Revocação Hierárquicas) são apresentadas nas Equações 12 e 13, em que m representa o número de classes do problema.

$$\hat{Pr}^{\mu CD} = \frac{\sum_{i=1}^{m} max\left(0, |TP_i| + FpCon_i + FnCon_i\right)}{\sum_{i=1}^{m} |TP_i| + |FP_i| + FnCon_i} \tag{12}$$

$$\hat{Re}^{\mu CD} = \frac{\sum_{i=1}^{m} max\left(0, |TP_i| + FpCon_i + FnCon_i\right)}{\sum_{i=1}^{m} |TP_i| + |FN_i| + FpCon_i} \tag{13}$$

De acordo com o valor escolhido para Dis_0, as contribuições $FpCon_i$ e $FnCon_i$ podem ser negativas. Portanto, uma função max é aplicada ao numerador das Equações 12 e 13 para fazer com que seus valores não sejam menores que 0. Como $FpCon_i \leq |FP_i|$. quando $|TP_i| + |FP_i| + FnCon_i \leq 0$, o numerador $max\left(0, |TP_i| + FpCon_i + FnCon_i\right) = 0$ e $\hat{Pr}^{\mu CD}$ pode ser considerado 0 nesse caso. A mesma regra é aplicada ao cálculo de $\hat{Re}^{\mu CD}$ (SUN; LIM, 2001).

Além dos cálculos da Micro Precisão e Revocação Hierárquicas, pode-se ainda obter os valores estendidos da Macro Precisão e Revocação Hierárquicas, por meio das Equações 14 e 15. Nas equações, m refere-se ao número de classes envolvidas.

$$\hat{Pr}^{\mu CD} = \frac{\sum_{i=1}^{m} Pr_i^{CD}}{m} \tag{14}$$

$$\hat{Re}^{\mu CD} = \frac{\sum_{i=1}^{m} Re_i^{CD}}{m} \tag{15}$$

Assim como as medidas Precisão e Revocação Hierárquicas, utilizadas por Kiritchenko et al. (2004), as medidas Micro/Macro Precisão e Revocação Hierárquicas também podem ser combinadas na medida FM, calculada por meio da Equação 3.

Medidas baseadas em distância possuem a desvantagem de não considerarem o fato de que classificações nos níveis mais profundos da hierarquia são mais difíceis e levam a informações mais especificas do que classificações nos níveis mais altos. É possível contornar essa desvantagem, tornando mais altos os custos de classificações erradas em níveis mais elevados do que em níveis mais profundos da hierarquia.

2.4.3.4 Medidas baseadas na Curva de Precisão e Revocação

Vens et. al. (2008) propôs uma medida baseada na análise de curvas de precisão e revocação (curvas PR). Nessa medida é escolhido um conjunto de limiares entre 0 e 1, sendo que cada limiar corresponde a um ponto no espaço da curva PR e variando esses limiares obtêm-se a curva PR.

Para um cálculo mais aproximado da área da curva, é necessário a utilização de algum método de interpolação, como por exemplo, o método de interpolação não-linear (DAVIS; GOADRICH, 2006).

Para um determinado limiar um ponto de Precisão e Revocação no espaço da curva PR são obtidos através da Equação 16 e da Equação 17, respectivamente, em que i representa as classes.

$$\overline{Prec} = \frac{\sum_i VP_i}{\sum_i VP_i + \sum_i FP_i} \tag{16}$$

$$\overline{Rev} = \frac{\sum_i VP_i}{\sum_i VP_i + \sum_i FN_i} \tag{17}$$

Essa medida funciona apenas para classificadores que possuem em sua saída um valor contínuo. Dessa maneira, avalia-se a saída desse classificador e compara com um determinado limiar. Assim, se uma saída é maior que o limiar então se atribui a instância à classe, caso contrario a instância não é atribuída a classe.Quanto menor for o valor do limiar mais exemplos são atribuídos a uma classe aumentando a medida de revocação (BORGES, 2012).

Outra questão levantada pelos autores é que as curvas PR podem ser construídas individualmente para cada classe em um problema multirrótulo, obtendo como positivos os exemplos pertencente à classe e como negativo os outros exemplos. Para combinar os desempenhos individuais em cada classe de forma a obter o desempenho geral, dois métodos podem ser utilizados: a área abaixo da curva PR (AUPRC) e a área média abaixo da curva PR (VENS *et al.*, 2008).

A área média abaixo da curva é descrita por $\overline{AUPRC}$. Nessa medida é calculada a média ponderada das áreas abaixo da curva PR individuais. A Equação 18 mostra como é calculada essa abordagem.

$$\overline{AUPRC_{w_1,\ldots,w|C|}} = \sum_i w_i * AUPRC_i \tag{18}$$

Os autores utilizaram essa abordagem de duas maneiras. A primeira, denominada de $\overline{AUPRC}$, os pesos w_i são inicializados com o valor de $\frac{1}{|C|}$, em que C representa todas as classes. A segunda abordagem, denominada de $\overline{AUPRC_w}$, consiste em atribuir um peso para uma classe

conforme a sua frequência. Essa frequência é calculada por $w_i = \frac{v_i}{\sum_j v_j}$, em que v_i é a frequência de classe c_i do conjunto de dados. Nesse caso, segundo os autores é que em alguns problemas de classificação hierárquica, classes que são mais frequentes podem ser mais importantes.

2.5 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste Capítulo foi apresentada uma breve introdução nos conceitos fundamentais de classificação e classificação de dados hierárquica, além dos conceitos referentes a classificação de dados multirrótulo, abordando as principais técnicas utilizadas para a resolução desse tipo de problema de classificação.

Por fim, foram apresentados os conceitos fundamentais de classificação hierárquica multirrótulo, tema do presente trabalho. Esse problema pode ser tratado como uma combinação dos problemas de classificação hierárquica com os problemas de classificação multirrótulo. Foram apresentadas as técnicas utilizadas para a resolução desse tipo de problema de classificação, bem como algumas medidas especificas utilizadas para a avaliação de classificadores hierárquicos multirrótulo. Foi feita também uma breve revisão bibliográfica dos principais trabalhos da literatura.

3 SELEÇÃO DE ATRIBUTOS

Neste Capítulo são apresentados conceitos básicos referente a seleção de atributos. A Seção 3.1 apresenta os conceitos fundamentais de seleção de atributos. Na Seção 3.2 são descritas as principais técnicas de seleção de atributos. Por fim, a Seção 3.3 apresenta as considerações finais do Capítulo.

3.1 CONCEITOS FUNDAMENTAIS DE SELEÇÃO DE ATRIBUTOS

A seleção de atributos é uma técnica muito explorada na área de mineração de dados, principalmente na tarefa de classificação (GUYON; ELISSEEFF, 2006). Tem como objetivo identificar atributos relevantes, para reduzir o tempo de execução do processo de classificação, aumentar a capacidade preditiva do classificador e obter uma representação compacta do conceito a ser aprendido (remoção de ruídos e diminuição da dimensionalidade dos dados).

Através disso, ao decorrer dos anos, uma riqueza de técnicas de seleção de atributos vem sendo desenvolvida por pesquisadores em Bioinformática, Aprendizagem de Máquina e *Data Mining* (SAEYS; INZA; LARRAÑAGA, 2007).

Na literatura existem várias definições formais para atributos relevantes, subdividindo-os em atributos de fraca e forte relevância (JOHN *et al.*, 1994). Alguns algoritmos utilizam a relevância de cada atributo para auxiliar durante a seleção, como mostrado em (BOZ, 2002).

Além disso, experimentos comprovam que o número de exemplos utilizados para garantir uma certa taxa de classificação cresce exponencialmente com o número de atributos irrelevantes presentes (LANGLEY; IBA, 1993).

Do ponto de vista teórico, pode ser demonstrado que uma seleção de atributos ótima para problemas de classificação requer uma busca exaustiva de todos os possíveis subconjuntos de atributos (REUNANEN, 2003), tornando-se impraticável quando o número de atributos é muito alto.

Por esse motivo foram desenvolvidas diferentes técnicas de seleção de atributos, que utilizam critérios de seleção e algoritmos de busca distintos para avaliar e encontrar de forma heurística o subconjunto de atributos mais adequado. Essas técnicas são divididas em duas abordagens principais, sendo estas a Filtro e $Wrapper$.

As estratégias $Wrapper$ e Filtro são executadas em uma fase de pré-processamento dos dados, e procuram pelo conjunto de atributos mais adequado para ser utilizado pelo algoritmo de classificação ou no processo de indução do modelo de classificação.

Em linhas gerais, a diferença entre algoritmos de seleção de atributos do tipo filtro e do tipo $Wrapper$ é que na primeira o subconjunto de atributos é avaliado por uma medida

independente, enquanto na última é utilizado o próprio algoritmo de classificação para essa avaliação.

A abordagem $Wrapper$ seleciona subconjuntos de atributos que atingem uma acurácia preditiva maior que a abordagem filtro, já que ela avalia os atributos utilizando o mesmo algoritmo que será utilizado na fase de classificação. Contudo, como a abordagem $Wrapper$ necessita de várias execuções do algoritmo de classificação, o seu custo computacional tende a ser bem maior que o custo da abordagem Filtro.

Alguns exemplos de utilização de seleção de atributos, em bases de dados e domínios específicos, podem ser encontrados na área de classificação hierárquica. Em Koller e Sahami (1997), implementa-se um classificador de documentos hierárquico $top - down$ no qual a seleção de atributos é executada antes do treinamento do classificador para cada nó da hierarquia (KOLLER; SAHAMI, 1997). Em Secker et al. (2010), foi proposto um classificador hierárquico $top - down$ com seleção de atributos para um problema da área de bioinformática (SECKER *et al.*, 2010).

3.2 ABORDAGENS DE SELEÇÃO DE ATRIBUTOS

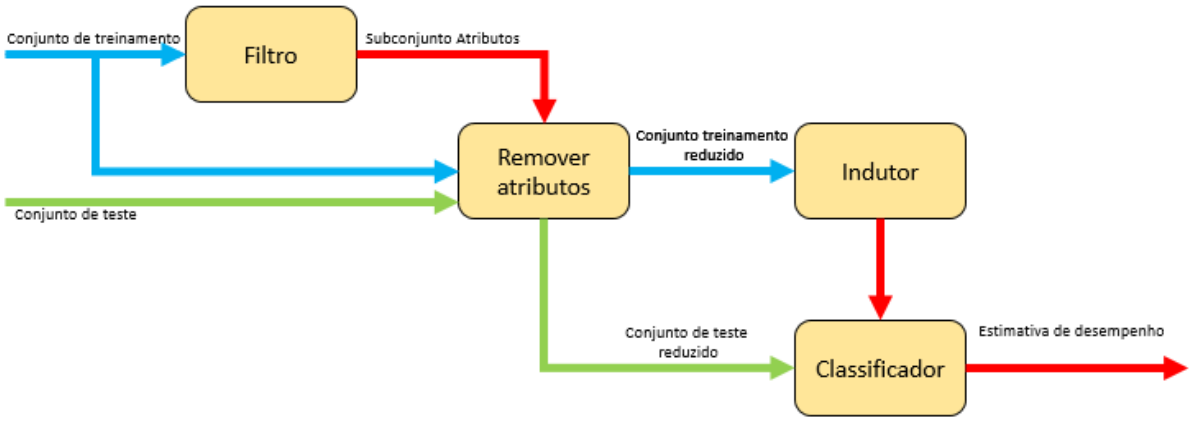
Os métodos de seleção de atributos podem ser divididos em dois grupos, sendo estes: Filtro e $Wrapper$ (KOHAVI; JOHN, 1997).

3.2.1 Abordagem de Seleção de Atributos do tipo Filtro

As abordagens de seleção de atributos do tipo Filtro são independentes do algoritmo de classificação. Esses métodos podem avaliar cada atributo independente dos outros, determinando o grau de correlação entre cada atributo e a classe (YANG; PEDERSEN, 1997), ou podem avaliar subconjuntos de atributos, buscando através de estratégias e heurísticas, aqueles que, em conjunto, melhor identificam as classes (HALL, 2000; LIU; SETIONO *et al.*, 1996).

O termo filtro deriva da ideia de que os atributos irrelevantes são filtrados da base de dados antes da aplicação do algoritmo de classificação (BLUM; LANGLEY, 1997). Os filtros usam as informações da própria base de treinamento para escolher os atributos a serem utilizados posteriormente. Esse processo pode ser visualizado na Figura 16.

Figura 16 – Técnica de seleção de atributos: Filtro



Fonte: Adaptado de Blum et. al (1997)

Técnicas do tipo filtro podem avaliar os atributos individualmente e escolher os melhores, como exemplificado pelas técnicas *Information Gain Attribute Ranking* (YANG; PEDERSEN, 1997) e $Relief$ (KIRA; RENDELL, 1992); ou podem avaliar subconjuntos de atributos, buscando de forma heurística o melhor subconjunto. As técnicas mais conhecidas desse último grupo são a *Correlation-based Feature Selection* (HALL, 2000) e a *Consistency-based Feature Selection* (LIU; MOTODA, 2007).

Essa técnica avalia a relevância dos atributos verificando somente para as propriedades intrínsecas dos dados. Na maioria dos casos, um ponto relevância de atributo é calculado, e atributos de baixa pontuação são removidos. Depois, este subconjunto de atributos é apresentado como entrada para o algoritmo de classificação.

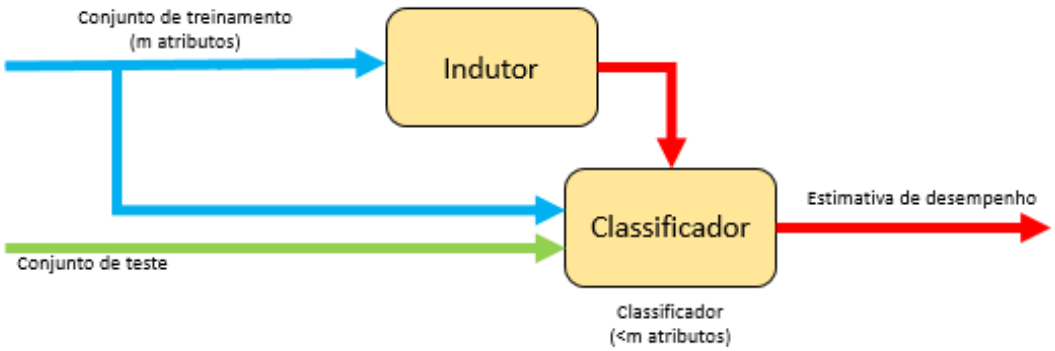
3.2.2 Abordagem de Seleção de atributos do tipo $Wrapper$

Assim como as abordagens do tipo filtro que avaliam subconjuntos de atributos, as abordagens Wrapper precisam realizar uma busca entre os possíveis subconjuntos a serem avaliados. No filtro os subconjuntos são avaliados por meio de uma métrica; já no $Wrapper$, o algoritmo de classificação é executado para cada subconjunto e a avaliação geralmente é feita em termos da acurácia preditiva retornada pelo algoritmo.

Normalmente, possuem boa capacidade preditiva, no entanto, requerem várias execuções do algoritmo de classificação, o que eleva o custo computacional em relação aos outros métodos e podem ser inviáveis em bases de dados com um número grande de atributos (GUYON; ELISSEEFF, 2006).

Os algoritmos de seleção de atributos do tipo $Wrapper$ além de empregarem diferentes classificadores para avaliar o subconjunto de atributos, também possuem critérios de parada e estratégias de busca distintos (LIU; MOTODA, 2007). Esse processo pode ser observado na Figura 17.

Figura 17 – Técnica de seleção de atributos: $Wrapper$



Fonte: Adaptado de Blum et. al (1997)

Alguns dos primeiros trabalhos com essa abordagem de seleção de atributos focaram nos classificadores de árvores de decisão, analisando diferentes estratégias de busca para selecionar o subconjunto de atributos candidatos, além de comparar o impacto de cada estratégia no comportamento final do indutor da árvore.

O algoritmo $Obvilion$, proposto em Langley e Iba (1993), combina a abordagem $Wrapper$ com o método de classificação kNN. Na seleção de atributos do $Obvilion$, é feita uma busca conhecida como *backward elimination*, que começa a execução com todos os atributos e retira gradualmente aqueles julgados menos relevantes, ou seja, que não fazem com que a acurácia estimada decline.

Esse método de seleção de atributos é considerado um $Wrapper$, pois envolve a execução do algoritmo do kNN para uma parte dos dados de treinamentos, utilizando um sistema de validação cruzada $leave - one - out$ para estimar a acurácia de cada subconjunto de atributos.

Outros algoritmos foram desenvolvidos em conjunto com o método kNN, mas utilizando-se estratégias distintas para compor o subconjunto de atributos a ser avaliado, como: buscas aleatórias, subida da montanha (*hill climbing*) e algoritmos genéticos.

Também existem estratégias híbridas que tentam combinar as vantagens das abordagens filtro e $Wrapper$ (LIU; MOTODA, 2007). Uma técnica muito utilizada é o uso de filtros para reduzir o conjunto inicial de atributos, viabilizando o uso de um $Wrapper$ em seguida.

3.2.2.1 Estratégias de busca

A estratégia de busca envolve a escolha de uma representação adequada para os possíveis subconjuntos de variáveis, uma condição inicial e um critério de interrupção da busca (HOLSCHUH *et al.*, 2008).

Existem diversos mecanismos de buscas na literatura, entre eles destacam-se a busca exaustiva, a seleção para frente (seleção $forward$), a retroalimentação e as buscas heurísticas (KOHAVI; JOHN, 1997; GUYON; ELISSEEFF, 2006).

3.2.2.1.1 Busca exaustiva

Uma maneira de se avaliar os subconjuntos de atributos é percorrer, de modo exaustivo, todo o espaço de busca do problema. O fato de se percorrer todo o espaço de busca garante que ao final do algoritmo terá sido encontrado o melhor subconjunto de atributos possível.

Em contrapartida, para problemas com um número elevado de atributos, esse tipo de busca torna-se inviável pois o espaço de busca é da ordem de 2^n, onde n representa o número total de atributos.Um exemplo em que a busca exaustiva é inviável, é o problema do caixeiro viajante.

3.2.2.1.2 Seleção para frente

Na seleção para frente a exploração do espaço de estados é realizada adicionando-se novas variáveis a um subconjunto selecionado em um passo anterior.

De forma geral, inicia-se a busca a partir do conjunto vazio e utiliza-se um operador simples que gera os "estados filhos" adicionando uma das variáveis ausentes no "estado pai". Cada "estado filho" é avaliado e o melhor avaliado é selecionado como o "pai" do próximo passo.

Um critério de interrupção comum neste caso é a falha em encontrar um "estado filho" de melhor avaliação que o "pai" (KOHAVI; JOHN, 1997). Trata-se de um método guloso, que sempre adiciona a variável que mais contribui para o incremento de performance do subconjunto atual, razão pela qual o melhor subconjunto pode não ser obtido no final.

A estratégia de seleção para frente é computacionalmente menos custosa, pois evita o treinamento excessivo de modelos com muitas entradas (que são de treinamento mais custoso que os com poucas entradas), já que inicia a busca por subconjuntos pequenos.

3.2.2.1.3 Retroalimentação

No processo de retroalimentação, a exploração do espaço de busca acontece pela remoção de variáveis de um "estado pai". O estado inicial mais comum é o equivalente ao subconjunto que contém todas as variáveis, e o operador simples associado gera os "estados filhos" eliminando uma das variáveis presentes no "estado pai". O "estado filho" melhor avaliado torna-se o "estado pai" do passo seguinte. Como critério de interrupção, costuma-se adotar também a falha em encontrar um "estado filho" de melhor avaliação que o "estado pai" (KOHAVI; JOHN, 1997).

O método de retroalimentação é considerado um método guloso que sofre da mesma deficiência da seleção para frente: o melhor subconjunto pode não ser obtido no final. A retro-eliminação tem como característica, o fato de permitir que as variáveis sejam avaliadas desde o início em conjunto com todas as outras, o que pode evitar que certas combinações de variáveis

que contribuem para o bom desempenho do classificador ou regressor sejam desfeitas (KOHAVI; JOHN, 1997; GUYON; ELISSEEFF, 2006).

Mesmo possuindo bons resultados, tanto o método de seleção para frente quanto o de retroalimentação, possuem o problema de que o número de subconjuntos potencialmente necessários para encontrar a melhor combinação de variáveis, cresce muito rapidamente com o aumento do número de variáveis, tornando estas estratégias de busca pouco factíveis.

Formas de se reduzir a vulnerabilidade a mínimos locais e aumentar a eficiência da exploração do espaço de busca, tanto da seleção para frente quanto da retroeliminação, incluem diferentes critérios de interrupção ($k > 1$ passos sem melhorias de avaliação, por exemplo) e o uso de operadores mais complexos, e mesmo a combinação de operadores de adição e de eliminação de variáveis, dando origem a um processo de busca bidirecional (REUNANEN, 2003).

3.2.2.1.4 Buscas heurísticas

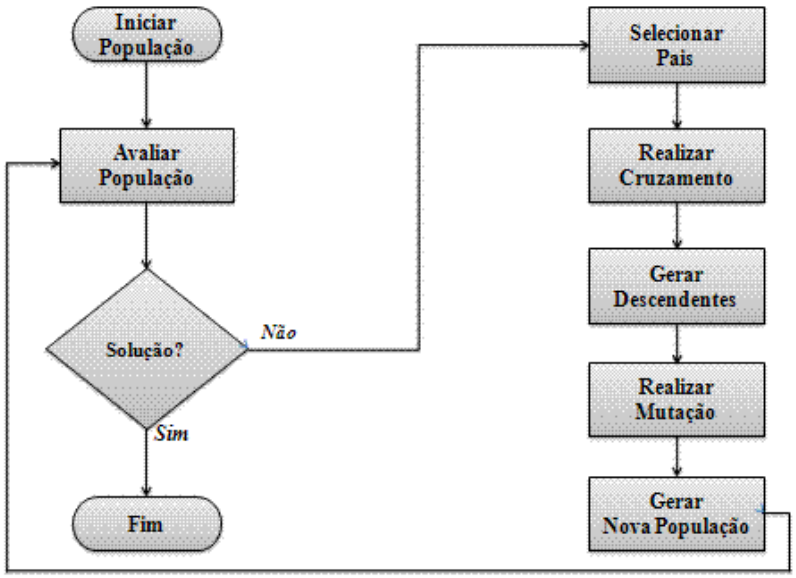
A utilização de heurísticas e meta-heurísticas são alternativas para encontrar boas soluções sem a necessidade de exploração de todo o espaço de busca. Em aplicações do mundo real estas técnicas têm se mostrado adequadas às necessidades visto que uma solução é satisfatória quando tem qualidade aproximada à uma ótima mas encontrada com tempo factível.

Este tipo de busca é frequentemente implementado utilizando-se métodos de computação evolutiva, como por exemplo o Algoritmo Genético, pois apresentam baixa vulnerabilidade a mínimos locais (HOLLAND, 1992).

Esses algoritmos simulam processos naturais de sobrevivência e reprodução das populações, essenciais em sua evolução. Na natureza, indivíduos de uma mesma população competem entre si, buscando principalmente a sobrevivência, seja através da busca de recursos como alimento, ou visando a reprodução. Os indivíduos mais aptos terão um maior número de descendentes.

A idéia básica de funcionamento dos algoritmos genéticos é a de tratar as possíveis soluções do problema como "indivíduos"de uma "população", que irá "evoluir"a cada iteração ou "geração". Para isso é necessário construir um modelo de evolução onde os indivíduos sejam soluções de um problema. O funcionamento deste algoritmo pode ser observado na Figura 18.

Figura 18 – Funcionamento do Algoritmo Genético



Fonte: Adaptado de Holland (1992)

O tamanho da população pode afetar o desempenho global e a eficiência dos algoritmos genéticos, pois populações muito pequenas podem convergir rapidamente a uma solução que não é a melhor possível, já populações grandes exigem um custo computacional elevado.

Além do tamanho da população, é importante que durante a modelagem do problema, seja realizada a escolha correta da forma de representação dos indíviduos, sendo as mais conhecidas a representação binária e a real.

A partir da geração da população inicial, é necessário que todos os indíviduos sejam avaliados pela função de aptidão, também conhecida como $fitness$. É através desta função que se mede quão próximo um indivíduo está da solução desejada ou quão boa é esta solução. É essencial que esta função seja representativa e diferencie na proporção correta as más soluções das boas.

A próxima etapa a ser realizada é a seleção dos melhores indíviduos para permanecerem na população, nos quais serão aplicados os operadores genéticos. Os piores indivíduos são descartados e novos indíviduos são gerados, de maneira que a população sempre mantenha o mesmo tamanho.

Há dois métodos principais utilizados para a seleção dos indivíduos, sendo estes, a seleção por Roleta e a seleção por Torneio. Na seleção por Roleta, cada indivíduo é representado na roleta proporcionalmente ao seu $fitness$. Assim, os melhores indivíduos tem maior chance de permanecerem na população. Um dos problemas encontrados pode ser o tempo de processamento, já que o método exige duas passagens por todos os indivíduos da população.

No método por Torneio escolhe-se um número n de indivíduos para formar uma subpopulação temporária. Deste grupo, o indivíduo selecionado dependerá de uma probabilidade k definida previamente. Este método é o mais utilizado, pois oferece a vantagem de não exigir

que a comparação seja feita entre todos os indivíduos da população (BANZHAF *et al.*, 1998).

O princípio básico dos operadores genéticos é transformar a população através de sucessivas gerações, estendendo a busca até chegar a um resultado satisfatório. Os operadores genéticos são necessários para que a população se diversifique e mantenha características de adaptação adquiridas pelas gerações anteriores. Os operadores de cruzamento e de mutação têm um papel fundamental em um algoritmo genético.

Através do cruzamento são criados novos indivíduos misturando características de dois indivíduos pais. Esta mistura é feita tentando imitar (em um alto nível de abstração) a reprodução de genes em células. Trechos das características de um indivíduo são trocados pelo trecho equivalente do outro. O resultado desta operação é um indivíduo que potencialmente combine as melhores características dos indivíduos usados como base. Alguns tipos de cruzamento bastante utilizados são o cruzamento em um ponto e o cruzamento em dois pontos.

A mutação, por sua vez, simplesmente modifica aleatoriamente alguma característica do indivíduo sobre o qual é aplicada. Esta troca é importante, pois acaba por criar novos valores de características que não existiam ou apareciam em pequena quantidade na população em análise. O operador de mutação é necessário para a introdução e manutenção da diversidade genética da população. Desta forma, a mutação assegura que a probabilidade de se chegar a qualquer ponto do espaço de busca possivelmente não será zero. O operador de mutação é aplicado aos indivíduos através de uma taxa de mutação geralmente pequena.

3.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste Capítulo foi apresentada a principal motivação na aplicação de técnicas de seleção de atributos, bem como os conceitos principais referentes a seleção de atributos.

Por fim, foram apresentadas as duas principais abordagens de seleção de atributos, sendo estas a abordagem Filtro e a *Wrapper*. A abordagem Filtro é independente do algoritmo de classificação que está sendo utilizado, enquanto na abordagem *Wrapper*, são utilizadas as medidas do próprio algoritmo de classificação. Devido a isso, normalmente os resultados obtidos com os algoritmos da abordagem *Wrapper* são superiores aos da abordagem Filtro. Também foram apresentadas as principais estratégias de busca utilizadas pelos algoritmos de seleção de atributos.

4 REVISÃO SISTEMÁTICA DA LITERATURA

Neste Capítulo é apresentado o levantamento bibliográfico visando estabelecer o estado da arte atual a respeito das técnicas de seleção de atributos em classificação hierárquica. Na Seção 4.1 está descrito o método de revisão sistemática utilizado. A Seção 4.2 descreve a realização da revisão sistemática. Na Seção 4.3 são apresentados trabalhos relacionados de outras natureza, como artigos de conferência, dissertações e teses. Por fim, a Seção 4.4 apresenta as considerações finais do Capítulo.

4.1 MÉTODO DE REVISÃO SISTEMÁTICA

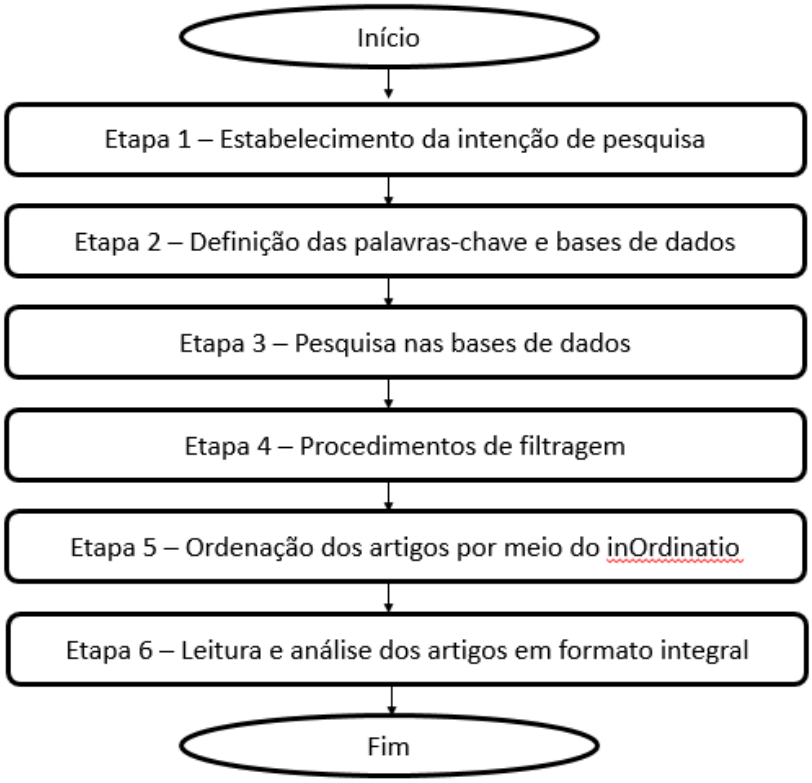
O método de revisão sistemática utilizado nesse trabalho é baseado no *Methodi Ordinatio* proposto por Pagani, Kovaleski e Resende (2015). Esse método foi adotado por apresentar uma etapa em que os trabalhos são classificados em um *ranking* de importância, através de critérios de avaliação e validação. A ordem de importância baseia-se em um cálculo onde são considerados o fator de impacto, o ano de publicação e o número de citações, permitindo assim a diminuição e refinamento dos artigos a serem lidos na íntegra (PAGANI; KOVALESKI; RESENDE, 2015). A Figura 19 apresenta todas as etapas do método de revisão sistemática utilizado nesse estudo.

Na primeira etapa é definida a intenção de pesquisa, na qual deve-se delimitar o tema a ser trabalhado juntamente com a criação das questões de pesquisa a serem respondidas pela revisão sistemática. Também é definida nessa etapa quais as características (ano de publicação, tipo de trabalho, onde foi publicado, idioma da publicação, etc.) que um trabalho deve ter para que este seja considerado na revisão sistemática.

Ao definir-se o tema da pesquisa deve-se realizar uma busca preliminar exploratória com palavras-chave com o intuito de validar o resultado de busca obtido a partir das mesmas, verificando assim a necessidade de alteração ou combinação das mesmas. Esse processo é realizado na segunda etapa, juntamente com a definição das bases de dados. Vale ressaltar que a escolha das bases de dados varia dependendo do contexto onde está inserida a intensão de pesquisa.

Na terceira etapa deve-se escolher um gerenciador de bibliografia, para assim realizar a busca efetiva nas bases de dados com as palavras-chave definidas. Com os resultados obtidos em todas as bases de dados, na quarta etapa são realizadas as primeiras filtragens. São descartados os resultados duplicados, os trabalhos em que o título, resumo ou palavras-chave não tem relação com o tema especificado na etapa 1, trabalhos apresentados em conferências e capítulos de livros. Outros filtros definidos pelo pesquisador, que não estão presentes no método também devem ser aplicados nessa etapa.

Figura 19 – Etapas da Revisão Sistemática



Fonte: Adaptado de Pagani, Kovaleski e Resende (2015)

Após a filtragem, deve-se realizar a identificação do ano, número de citações e fator de impacto, através dos índices JCR ($JournalCitationReports$) e SJR ($ScientificJournalRankings$), de todos os trabalhos para facilitar o cálculo que cria a ordem de importância dos trabalhos. A ordenação dos resultados é realizada na quinta etapa e ocorre por meio da equação $InOrdinatio$ definida por Pagani et al. (2015), essa equação é apresentada na Equação 19.

$$InOrdinatio = \frac{F_i}{1000} + \alpha * [10 - (Aa - Ap)] + C_i \tag{19}$$

em que F_i representa o Fator de impacto, Aa o ano atual em que a revisão sistemática está sendo realizada, Ap o ano da publicação do artigo e C_i o número total de citações do artigo. O α é um número no intervalo entre 1 e 10 que representa quão mais importante um estudo novo é em relação a um mais velho, sendo 1 pouca importância e 10 mais importância, em resumo quanto maior o parâmetro α mais bem colocados irão ficar os estudos mais novos.

A última etapa inicia-se com a busca dos artigos selecionados no formato completo. Se o artigo for localizado então realiza-se a leitura e análise sistêmica, caso não seja localizado deve-se descartá-lo. O objetivo da leitura dos artigos é responder as perguntas definidas na etapa 1.

4.2 APLICAÇÃO DO MÉTODO DE REVISÃO SISTEMÁTICA

Para a aplicação do método $Methodi\ Ordinatio$ foram realizadas as fases de estabelecimento da intenção de pesquisa, definição das palavras-chave e base de dados, pesquisa nas bases de dados, procedimentos de filtragem, ordenação dos artigos por meio do $inOrdinatio$ e a leitura e análise dos artigos em formato integral.

4.2.1 Estabelecimento da intenção de pesquisa

Esse trabalho foi desenvolvido para realizar uma revisão no que diz respeito á seleção de atributos em classificação hierárquica multirrótulo. Para tal foram definidas as perguntas á serem respondidas a cerca de cada trabalho encontrado, sendo que as mesmas encontram-se no Quadro 4.

Quadro 4 – Definição do protocolo da revisão sistemática

Nº	**Pergunta**
1	Quais são as abordagens e técnicas de seleção de atributos utilizadas em classificação hierárquica?
2	Qual foi a contribuição científica do trabalho?
3	Quais foram as áreas e as bases de dados em que a seleção de atributos foi aplicada?
4	Quais foram os classificadores e as medidas de avaliação utilizados?
5	Os resultados obtidos pelas técnicas propostas foram superiores as técnicas existentes?

Fonte: Autoria própria

O período de interesse para a revisão foi definido como sendo os últimos 17 anos, assim foram considerados apenas trabalhos publicados entre os anos de 2000 até 2017. Também foram desconsiderados todos os livros e capítulos de livros, sendo considerados apenas artigos.

4.2.2 Definição das palavras-chave e base de dados

Nessa etapa foram realizados testes com várias palavras-chave em diversas bases de dados a fim de determinar quais as melhores combinações de palavras-chave e quais as bases de dados mais adequadas para o tema da revisão.

As palavras-chave escolhidas foram: "*Hierarchical Multi-label Classification*", *Global Hierarchical Classification*, *Hierarchical Classification*, *Feature Selection* e *Attribute Selection*. Essas palavras-chave foram combinadas formando a seguinte *string* de busca: "(("*Feature Select*"*OR* "*Attribute Select*") *AND* "*Classification Hierarchical*")".

É importante ressaltar que foi realizada uma busca nas bases de dados com as palavras-chave na língua portuguesa e nenhuma das bases retornou algum resultado da busca. A escolha

das bases de dados foi realizada baseada em uma busca no site da CAPES[1].O Quadro 5 apresenta a relação das bases de dados selecionadas para a revisão com seus respectivos sites.

Quadro 5 – Bases de dados utilizadas

Base de dados	Sites
Science Direct	<http://www.sciencedirect.com>
Inderscience	<http://www.inderscience.com/>
Springer	<http://link.springer.com/>
Science Domain	<http://www.sciencedomain.org/>
Emerald Insight	<http://www.emeraldinsight.com/>
Scientific Periodicals Electronic Library	<http://www.spell.org.br>
ArXiv.org	<https://arxiv.org/>
CogPrints	<http://cogprints.org/>
IEEEXplore	<http://ieeexplore.ieee.org>
SciELO.ORG	<http://www.scielo.org/>

Fonte: Autoria Própria

4.2.3 Pesquisa nas bases de dados

A $string$ de busca definida na Seção 4.2.2 foi aplicada a todas as bases de dados e foram considerados no filtro disponíveis: o período de publicação, entre 2000 e 2017, e o tipo de publicação somente artigos. A Tabela 1 mostra a quantidade de resultados obtidos em cada base de dados com o uso da $string$ de busca.

Tabela 1 – Quantidade de resultados por base

Base de dados	Resultados
Science Direct	90
Inderscience	1
Springer	392
Science Domain	0
Emerald Insight	6
Scientific Periodicals Electronic Library	0
ArXiv.org	13
CogPrints	0
IEEEXplore	226
SciELO.ORG	0

Fonte: Autoria Própria

Ao final dessa etapa foram encontrados 728 artigos e os mesmos foram importados para o gerenciador de bibliografia EndNote[2].

[1] https://www.periodicos.capes.gov.br/index.php
[2] http://endnote.com/

4.2.4 Procedimentos de filtragem

Com o auxílio do software EndNote, dos 728 artigos inicialmente obtidos foram removidos os resultados duplicados, os livros, os artigos de conferência, capítulos de livros, restando apenas 259 artigos.

A leitura do título, palavras-chave e resumos desses trabalhos possibilitou identificar e remover todos aqueles que não tem relação com o tema dessa revisão sistemática. Apenas 16 artigos chegaram ao final da etapa de filtragem.

4.2.5 Classificação e ordenação dos artigos

Todos os trabalhos resultantes da etapa de filtragem foram ordenados de acordo com a equação $InOrdinatio$, apresentada na Equação 19 (PAGANI; KOVALESKI; RESENDE, 2015). O valor do Fator de impacto (F_i) de cada trabalho foi obtido através dos índices JCR e SJR. O número de citações C_i de cada artigo foi obtido através do Google Acâdemico. Foi considerado para o cálculo o valor do parâmetro α = 10.

Para a leitura e análise, descartou-se apenas o artigo "*A Hierarchical Neural Network Document Classifier with Linguistic Feature Selection"*, pois obteve o valor negativo no *ranking* de classificação dos trabalhos relevantes, conforme pode ser observado na Tabela 2.

4.2.6 Leitura e análise dos artigos em formato integral

Na sexta e última etapa do método *Methodi Ordinatio* são apresentadas as respostas obtidas para as perguntas definidas na Seção 4.2.1 através da leitura e análise dos trabalhos resultantes após a etapa de filtragem da Seção 4.2.4.

4.2.6.1 Quais são as abordagens e técnicas de seleção de atributos utilizadas em classificação hierárquica?

A leitura e análise dos estudos identificou a utilização de 16 técnicas de seleção de atributos nos 15 trabalhos selecionados. Na Tabela 3 são apresentados os métodos e a abordagem de seleção de atributos e a quantidade de trabalhos por técnica.

Pode-se verificar por meio dos dados coletados que não há um consenso na escolha de técnicas para seleção de atributos em classificação hierárquica.

Tabela 2 – Ranking de classificação dos trabalhos após avaliação

Título do artigo	Fi	Ap	Ci	InOrdinatio
A novel hierarchical selective ensemble classifier with bioinformatics application	2,01	2017	5	105,0020
Distributed incremental fingerprint identification with reduced database penetration rate using a hierarchical classification based on feature fusion and selection	4,53	2017	0	100,0045
Novel feature selection and classification of Internet video traffic based on a hierarchical scheme	2,52	2017	0	100,0025
Hierarchical k-nearest neighbours classification and binary differential evolution for fault diagnostics of automotive bearings operating under variable conditions	2,89	2016	6	96,0028
Hierarchical emotion classification and emotion component analysis on chinese micro-blog posts	3,93	2015	14	94,0039
Efficient and accurate face detection using heterogeneous feature descriptors and feature selection	2,50	2013	33	93,0025
Hierarchical classification strategy for Phenotype extraction from epidermal growth factor receptor endocytosis screening	2,45	2016	0	90,0024
Accurate mapping of forest types using dense seasonal Landsat time-series	6,39	2014	19	89,0064
Inductive Model Generation for Text Classification Using a Bipartite Heterogeneous Network	0,96	2014	11	81,0010
Exploring Attribute Selection in Hierarchical Classification	0	2014	3	73,0000
Feature-Selected Tree-Based Classification	4,94	2013	9	69,0049
Fast and efficient lung disease classification using hierarchical one-against-all support vector machine and cost-sensitive feature selection	1,84	2012	10	60,0018
Hierarchical classification of G-Protein-Coupled Receptors with data-driven selection of attributes and classifiers	0,67	2010	30	60,0010
Multivariate feature selection and hierarchical classification for infrared spectroscopy: serum-based detection of bovine spongiform encephalopathy	3,43	2007	43	43,0034
Improving classification in protein structure databases using text mining	2,45	2009	14	34,0024
A Hierarchical Neural Network Document Classifier with Linguistic Feature Selection	1,90	2005	18	-1,9981

Fonte: Autoria Própria

4.2.6.2 Qual foi a contribuição científica do trabalho?

Os artigos selecionados podem ser categorizados em dois tipos principais, com relação ao tipo de contribuição científica. O primeiro grupo inclui os artigos que adaptaram técnicas já existentes na literatura, enquanto no segundo grupo estão aqueles que apresentaram novas técnicas de seleção de atributos para o problema de classificação hierárquica. Essa divisão dos trabalhos pode ser observada no Quadro 6.

Tabela 3 – Técnicas e abordagens de seleção de atributos x Quantidade de trabalhos

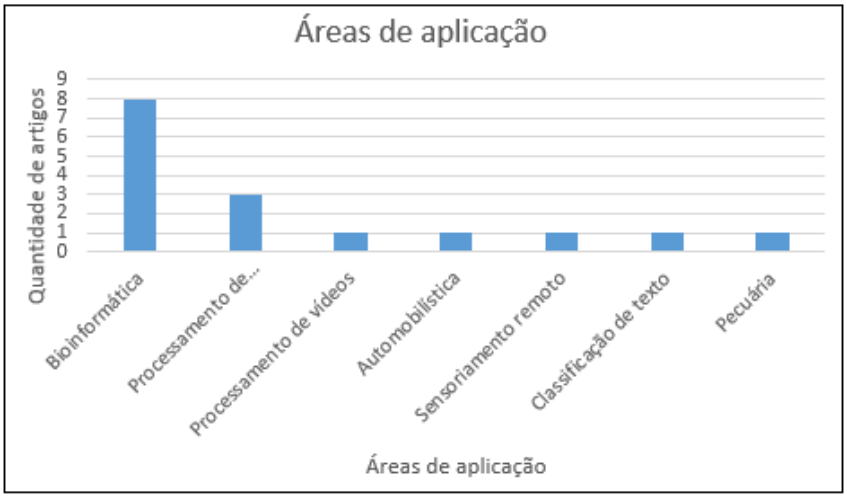
Técnica	Abordagem	Quantidade de artigos
MSRD (Maximize the Sum of Relevance and Distance)	Embutida	1
RD (Random Forest)	Wrapper	2
CFS (Consistency-based Feature Selection)	Filtro	2
BDE (Binary Differencial Evolution)	Wrapper	1
$\chi^2 - test$	Filtro	1
PMI (Pointwise Mutual Information)	Filtro	1
PSO-Adaboost (Particle Swarm Otimization Adaboost)	Wrapper	1
B&B (Branch and Bound procedure)	Filtro	1
Best individual-N features	Filtro	1
SVM-RFE (Support Vector Machine Recursive Feature Elimination)	Wrapper	1
Feature Selection based on thresholding metrics	Embutida	1
Soma das ocorrências dos termos	Embutida	1
Information Gain Attribte Ranking	Filtro	1
FSHC (Feature Selected Hierarchical Classifier)	Híbrida	1
cSFS (Cost-Sensitive Feature Selection)	Embutida	1
Best First	Filtro	1

Fonte: Autoria própria

4.2.6.3 Quais foram as áreas e as bases de dados em que a seleção de atributos foi aplicada?

A Figura 20 apresenta a distribuição dos trabalhos nas seguintes áreas: Bioinformática, Processamento de imagens, Processamento de vídeos, Automobilística, Sensoriamento remoto, Classificação de texto e Pecuária. Como pode-se observar as áreas predominantes no estudo de seleção de atributos em classificação hierárquica multirrótulo foram a Bioinformática e o Processamento de Imagens.

Figura 20 – Áreas de aplicação x Quantidade de artigos



Fonte: Autoria própria

Quadro 6 – Tipos de contribuição científica x Artigos selecionados

Tipo de contribuição	Título do artigo
Aprimoramento de técnicas existentes	Novel feature selection and classification of Internet video traffic based on a hierarchical scheme
	Efficient and accurate face detection using heterogeneous feature descriptors and feature selection
	Hierarchical classification strategy for Phenotype extraction from epidermal growth factor receptor endocytosis screening
	Exploring Attribute Selection in Hierarchical Classification
	Feature-Selected Tree-Based Classification
	Fast and efficient lung disease classification using hierarchical one-against-all support vector machine and cost-sensitive feature selection
	Hierarchical classification of G-Protein-Coupled Receptors with data-driven selection of attributes and classifiers
	Multivariate feature selection and hierarchical classification for infrared spectroscopy: serum-based detection of bovine spongiform encephalopathy
Criação de uma nova técnica	Distributed incremental fingerprint identification with reduced database penetration rate using a hierarchical classification based on feature fusion and selection
	Hierarchical k-nearest neighbours classification and binary differential evolution for fault diagnostics of automotive bearings operating under variable conditions
	A novel hierarchical selective ensemble classifier with bioinformatics application
	Hierarchical emotion classification and emotion component analysis on chinese micro-blog posts
	Accurate mapping of forest types using dense seasonal Landsat time-series
	Inductive Model Generation for Text Classification using a Bipartite Heterogeneous Network
	Improving classification in protein structure databases using text mining

Fonte: Autoria própria

Com relação as bases de dados, cada trabalho utilizou-se um conjunto diferente do outro. No Quadro 7 pode-se observar cada uma das bases separadas por área de aplicação.

4.2.6.4 Quais foram os classificadores e as medidas de avaliação utilizados?

Em Menze, Wolfgang e Hamprecht (2007) o método de seleção de atributos proposto, o Random Forest com base na medida *Gini Importance*, foi utilizado em um conjunto de dados da área de pecuária. Com o intuito de comparar o método proposto utilizou-se outros classificadores que também passaram pela etapa de seleção de atributos, sendo estes o LDA (*Linear*

Quadro 7 – Áreas de aplicação x Base de dados utilizadas

Área de aplicação	Base de dados
Bioinformática	Base criada pelos próprios autores (FREEMAN; KULIC; BASIR, 2013)
	Base criada pelos próprios autores (CAO *et al.*, 2016)
	Cellcycle (PAES; PLASTINO; FREITAS, 2014)
	Church (PAES; PLASTINO; FREITAS, 2014)
	SPO (PAES; PLASTINO; FREITAS, 2014)
	Sequence (PAES; PLASTINO; FREITAS, 2014)
	Phenotype (PAES; PLASTINO; FREITAS, 2014)
	Gasch1 (PAES; PLASTINO; FREITAS, 2014)
	Gasch2 (PAES; PLASTINO; FREITAS, 2014)
	Expr (PAES; PLASTINO; FREITAS, 2014)
	Eisen (PAES; PLASTINO; FREITAS, 2014)
	Derisi (PAES; PLASTINO; FREITAS, 2014)
	Spambase (WEI *et al.*, 2017)
	BDWorld e-mails (WEI *et al.*, 2017)
	Letter (WEI *et al.*, 2017)
	Thoracic Surgery (WEI *et al.*, 2017)
	CATH (KOUSSOUNADIS; REDFERN; JONES, 2009)
	SCOP (KOUSSOUNADIS; REDFERN; JONES, 2009)
	GPCR (SECKER *et al.*, 2010)
	HRCT (CHANG *et al.*, 2012)
Pecuária	VLA (MENZE; PETRICH; HAMPRECHT, 2007)
Classificação de textos	ACM (ROSSI *et al.*, 2014)
	Ohscal (ROSSI *et al.*, 2014)
	Polarity (ROSSI *et al.*, 2014)
	Reviews (ROSSI *et al.*, 2014)
	WAP (ROSSI *et al.*, 2014)
Sensoriamento Remoto	SRTMDEM (ZHU; LIU, 2014)
	Landsat images (ZHU; LIU, 2014)
Processamento de imagens	FERET (PAN; ZHU; XIA, 2013)
	BIOID (PAN; ZHU; XIA, 2013)
	Extended Yale Face (PAN; ZHU; XIA, 2013)
	CAS-PEAL (PAN; ZHU; XIA, 2013)
	Sina Weibo (XU; YANG; WANG, 2015)
	SFinge (PERALTA *et al.*, 2017)
	NIST-SD4 (PERALTA *et al.*, 2017)
	NIST-SD14 (PERALTA *et al.*, 2017)
Automobilística	CWRUBD (BARALDI *et al.*, 2016)
Processamento de vídeos	Base criada pelos próprios autores (DONG; ZHAO; JIN, 2017)

Fonte: Autoria própria

Discriminant Analysis), R-LDA (*Robust Linear Discriminant Analysis*), SVM e ANN (*Artificial Neural Network*). Os autores adotaram as medidas de sensibilidade e especificidade para comparar o desempenho entre as técnicas (MENZE; PETRICH; HAMPRECHT, 2007).

Em Koussounadis, Redfern e Jones (2009) foi proposta uma nova abordagem para a predição de proteínas. Para isso foram utilizadas as técnicas SVM, SSAP+Text (*Sequencial Struc-*

tural Aligment Program - Text), SSAP (*Sequencial Structural Aligment Program*). Os experimentos foram conduzidos em dois conjuntos de bases de dados e adotou-se como medidas de desempenho a precisão e revocação, além das curvas ROC (KOUSSOUNADIS; REDFERN; JONES, 2009).

Em Secker et. al (2010) foi realizada uma comparação entre técnicas de classificação e seleção de atributos. Os classificadores analisados nesse trabalho foram o Naive Bayes, Redes Bayesianas, SVM, 1-*Nearest Neighbor*, PART, J48, *Naive Bayes Tree* e AIRS2 (*Artificial Immune Recognition System*). Os experimentos foram conduzidos em um conjunto de dados de proteínas e adotou-se a medida de acurácia como base para a comparação entre os algoritmos (SECKER *et al.*, 2010).

Em Chang et. al (2012) foi desenvolvido um método de classificação, denominada OAA-SVM (*One-Against-All Support Vector Machine*). Os experimentos foram realizados em um conjunto de dados de pacientes que possuíam ou não uma determinada doença. O desempenho do algoritmo foi medido através da acurácia e foram realizadas comparações com o classificador OAO-SVM (One-Against-One Support Vector Machine) (CHANG *et al.*, 2012).

Em Pan, Zhu e Xia (2013) foi criado um detector facial baseado no método PSO-AdaBoost (*Particle Swarm Optimization - AdaBoost*). Os autores validaram realizaram um comparativo do desempenho desse método com as técnicas SVM e AdaBoost. Os experimentos foram conduzidos em conjuntos de dados de imagens com e sem a presença de faces, sendo o desempenho dos algoritmos avaliado através da taxa de acerto na detecção das faces (PAN; ZHU; XIA, 2013).

Em Freeman, Kulic e Basir (2013) foi proposto um método de classificação hierárquica, o FSHC (*Feature Selected Hierarchical Classifier*). A fim de comparar o desempenho do método proposto foram utilizados outros classificadores, sendo estes o *(*One versus Rest, Fuzzy One versus Rest, One versus One), DAG-SVM (*Directed Acyclic Graph SVM*), SVM-BDT (*Support Vector Machine Binary Decision Tree*), o ABT (*Adaptive Binary Tree*) e o CART (*Classification and Regression Tree*). Os experimentos foram conduzidos em conjuntos de dados de diversos domínios. Para comparação de desempenho entre os algoritmos foram utilizadas as medidas de acurácia, precisão e revocação (FREEMAN; KULIC; BASIR, 2013).

Os autores Zhu e Liu (2014) desenvolveram um método para o mapeamento de regiões onde estão situadas florestas, sendo este o SVM-RFE (*Support Vector Machine Recursive Feature Selection*). Os experimentos para validação do método foram realizados em dois conjuntos de dados e o desempenho do método foi comparado com a versão tradicional do método SVM. A métrica utilizada para verificar o desempenho entre os algoritmos foi a acurácia (ZHU; LIU, 2014).

Em Rossi et.al (2014) validou o método criado, o IMBHN (*Inductive Model based on Bipartite Heterogeneous Networks*), através de uma avaliação empírica usando uma grande quantidade de coleções de texto de diferentes domínios. A medida escolhida pelos autores para comparação dos algoritmos foi a acurácia. Esse algoritmo produziu resultados significativamente

melhores que os algoritmos kNN, C4.5, SVM e Naive Bayes (ROSSI *et al.*, 2014).

Em Paes, Plastino e Freitas (2014) foram avaliadas duas abordagens de classificação hierárquica: local por nó pai e local por nível. Esse trabalho propôs um método de seleção de atributo que produz um ranking dos atributos através da medida *Information Gain*. Além disso, esse trabalho também avaliou uma estratégia *lazy* de seleção de atributos, a qual posterga a seleção de atributos ao momento da classificação de um anova instância. Os experimentos foram conduzidos a partir de bases de dados provenientes da área de bioinformática e adotou-se a medida f-measure para a comparação de desempenho entre os algoritmos (PAES; PLASTINO; FREITAS, 2014).

Xu, Yang e Wang (2015) utilizaram uma base de dados de um blog chines, o Sina Weibo, com o intuito de reconhecer as emoções expressadas pelos usuários. Neste trabalho foi desenvolvido o método ECA (*Emotion Component Analysis*), sendo seu desempenho medido através das medidas de precisão hierárquica, revocação hierárquica e f-measure hierárquica. Os autores dividiram os experimentos em quatro grupos, no primeiro grupo foram aplicados somente classificadores planos, no segundo somente classificadores hierárquicos, no terceiro aplicou-se técnicas de seleção de atributos juntamente com os classificadores hierárquicos e no último, utilizou-se uma técnica de dicionário de emoções (XU; YANG; WANG, 2015).

Em Baraldi et. al (2016) foram realizados experimentos com um conjunto de dados coletados na Universidade da Reserva Ocidental e comparou-se os resultados com experimentos realizados anteriormente nas mesmas bases. Nesses experimentos foram utilizados os classificadores kNN, SVM e SVMkNN e a métrica *misclassification rate* (taxa de exemplos classificados incorretamente). O objetivo desse trabalho era otimizar o processo de detecção de defeitos em rolamentos automotivos (BARALDI *et al.*, 2016).

Em Cao et. al (2016) foi realizado um comparativo entre classificadores e técnicas de seleção de atributos com o intuito de encontrar a melhor combinação para a resolução do problema de extração do fenótipo de imagens. Os classificadores abordados nesse trabalho são o LDC (*Linear Classifier*), QDC (*Quadratic Classification*), kNNC (*k-Nearest Neighbor Classifier*), SVM e NEURC (*Neural Network Classifier*). Os experimentos foram conduzidos com uma base de dados criada pelos próprios autores e adotou-se as medidas de precisão hierárquica, revocação hierárquica e f-measure hierárquica (CAO *et al.*, 2016).

Em Wei et. al (2017) foi criado o modelo de classificação PTHS (*Parallel opTimization and Hierarchical Selection*). Esse classificador foi comparado com outros três existentes da literatura, sendo estes o *Bagging, Boosting e Random Forest (RF)*. Os experimentos foram conduzidos com bases de proteínas e os autores adotaram as medidas de acurácia (ACC), raiz do erro quadrático médio (RMSE) e curva ROC (AUC) (WEI *et al.*, 2017).

No trabalho de Peralta et. al (2017) foi desenvolvido um método de identificação de impressão digital, denominado AFIS (*Automatic Fingerprint Identification System*). O desempenho desse método é comparado com outros dois, o RF e o SVM, sendo adotadas as métricas de acurácia (ACC) e taxa de rejeição (PERALTA *et al.*, 2017).

Em Dong, Zhao e Jin (2017) utiliza-se o classificador hierárquico kNN. Nesse trabalho é realizado um comparativo do kNN hierárquico com outros classificadores, como o AVG-NN, C4-5, Naive Bayes, RF, MLP (Multi-Layer Perceptron), RBF (Radial Basis Function), SVM, Hier-SVM. A comparação desses classificadores se dá através das métricas de precisão, revocação, f-measure e acurácia. Para a realiação dos experimentos, os autores coletaram os dados do trafego de rede durante vários períodos do dia durante um mês (DONG; ZHAO; JIN, 2017).

4.2.6.5 Os resultados obtidos pelas técnicas propostas foram superiores as técnicas existentes?

Em Menze, Wolfgang e Hamprecht (2007), os autores verificaram que os classificadores hierárquicos utilizados obtiveram um desempenho superior comparados aos classificadores planos, visto que os mesmos levam em consideração a estrutura hierárquica em que as classes estão dispostas. Além disso, observou-se também que o método de seleção de atributos proposto o Random Forest baseado na medida *Gini Importance* trouxe melhoras significativos no poder preditivo dos classificadores analisados (MENZE; PETRICH; HAMPRECHT, 2007).

Em Koussonadis, Redfern e Jones (2009) verificou-se que o método aplicado SSAP-Text apresentou um desempenho superior ao classificador SVM que é largamente utilizado no dominio de classificação de proteínas. Além disso, observou-se o ganho no poder preditivo dos dois métodos analisados quando aplicou-se técnicas de seleção de atributos, consequentemente reduzindo a taxa de erro durante o processo de classificação (KOUSSOUNADIS; REDFERN; JONES, 2009).

Em Secker et. al (2010), os experimentos demonstraram que o sistema top-down proposto reduziu significativamente o tempo de processamento necessário para treinar e testar o modelo de classificação sem reduzir a acurácia preditiva (SECKER *et al.*, 2010).

Em Chang et. al (2012), pode-se observar que o método de seleção de atributos proposto, o cSFS, permitiu uma significativa redução do tempo de execução dos classificadores OAA-SVM e OAO-SVM. Além disso, o classificador OAA-SVM teve um desempenho superior ao OAO-SVM levando-se em consideração a medida de acurácia (CHANG *et al.*, 2012).

Em Freeman, Kulic e Basir (2013), pode-se observar que o método proposto FSHC teve um desempenho igual ou superior aos demais métodos, nos dez conjuntos de bases de dados de diferentes domínios, quando levando em consideração a medida de precisão. Além disso, os resultados demonstraram que o algoritmo cria soluções com menos classificadores, menos recursos e com um tempo de teste mais curto que as outras técnicas (FREEMAN; KULIC; BASIR, 2013).

Em Pan, Zhu e Xia (2013), o algoritmo proposto PSO-Adaboost obteve bons resultados ao ser testado com múltiplos conjuntos de dados e com outras técnicas existentes. Devido ao bom desempenho, o método foi extendido para a detecção de mais de uma face por imagem (PAN;

ZHU; XIA, 2013).

Em Paes, Plastino e Freitas (2014) foram realizados experimentos aplicando a seleção de atributos em duas abordagens diferentes de classificadores hierárquicos, sendo estas a local por nó pai e a local por nível. Observou-se que a aplicação das técnicas de seleção de atributos aumentou o poder preditivo dos classificadores hierárquicos (PAES; PLASTINO; FREITAS, 2014).

Em Zhu e Liu (2014), o método proposto SVM-RFE apresentou bons resultados quando testado em conjunto de dados com muitos atributos. Além disso, o desempenho do SVM-RFE foi comparado com a abordagem tradicional do SVM e o mesmo mostrou-se bastante competitivo. A técnica tem potencial para ser utilizada em outras áreas e aplicadas a imagens de sensoriamento remoto de diferentes fontes para mapear tipos de floresta em escalas regionais ou globais (ZHU; LIU, 2014).

Em Rossi et. al (2014), o método proposto IMBHN apresentou bons resultados quando comparado com outras técnicas já existentes na literatura, com desempenho significativamente melhor que os algoritmos kNN, C4.5, SVM e Naive Bayes (ROSSI *et al.*, 2014).

Em Xu, Yang e Wang (2015), conforme descrito anteriormente, os experimentos foram divididos em quatro grupos, e verificou-se que o terceiro grupo o qual aplicou técnicas de seleção de atributos em classificadores hierárquicos teve um desempenho superior aos demais levando-se em consideração a medida *F-measure*. No geral, o classificador de emoções obteve altas taxas de acerto em todos os experimentos realizados (XU; YANG; WANG, 2015).

Em Cao et. al (2016), o classificador hierárquico proposto juntamente com a aplicação da técnica de seleção de atributos apresentou melhorias no processo de análise temporal de endocitose quando comparado com outras técnicas existentes através da medida de precisão (CAO *et al.*, 2016).

Em Baraldi et. al (2016), o modelo de detecção de defeitos em rolamentos automobilísticos apresentou ganhos significativos no processo de identificação e categorização ao ser comparado com outros classificadores, como o kNN, SVM, SVMkNN. Além disso, o algoritmo proposto mostrou-se competitivo com outros de abordagem de evolução diferencial binária (BARALDI *et al.*, 2016).

Em Dong, Zhao e Jin (2017), o algoritmo proposto apresentou um desempenho superior ao AVG-kNN com relação a métrica f-measure. Seu desempenho foi inferior somente na comparação de tempo de execução. Os autores consideram o desempenho do tempo de execução significativo, visto que no geral, o desempenho do algoritmo foi superior aos demais (DONG; ZHAO; JIN, 2017).

Em Peralta et. al (2017) o método proposto AFIS obteve bons resultados quando comparados os valores da medida de precisão dos classificadores RF e SVM. O modelo de detecção de impressão digital teve desempenho satisfatório em todos os conjuntos de dados utilizados, com alta taxa de precisão (PERALTA *et al.*, 2017).

Em Wei et. al (2017), o método proposto PTHS, apresentou-se um desempenho competitivo ao ser comparado com os métodos *Bagging, Boosting* e RF. Além disso, os autores criaram o método de seleção de atributos MSRD, que mostrou-se eficiente quando combinado com os classificadores para a redução de dimensionalidade das bases utilizadas (WEI *et al.*, 2017).

4.3 OUTROS TRABALHOS

Além da revisão sistemática da literatura realizou-se uma busca na ferramenta *Google Scholar* com o intuito de filtrar trabalhos relacionados de outras naturezas, como dissertações, teses, artigos publicados em conferências e pesquisas em andamento.

A pesquisa foi realizada com as seguintes palavras-chave: "feature selection in hierarchical classification"e "seleção de atributos em classificação hierárquica". Os resultados das buscas estão apresentados na Tabela 4, assim como as técnicas utilizadas, área de aplicação e principais resultados.

Tabela 4 – Trabalhos de seleção de atributos para classificação hierárquica

Trabalhos	**Técnicas**	**Aplicação**
(SANTOS; NIEVOLA, 2016)	Combinação de métodos de seleção e classificadores hierárquicos (C4.5, SMO, rip e Naive Bayes)	Bioinformática
(PEREIRA; NIEVOLA, 2016)	Classificação hierárquica por nó, por nó pai e por camadas juntamente com técnicas de seleção de atributos	Bioinformática
(DIAS; MERSCHMANN, 2015)	Classificador Hierárquico Global-Model Naive Bayes	Bioinformática
(KOLLER; SAHAMI, 1997)	Técnicas de seleção de atributos em classificação hierárquica top-down	Categorização de textos
(SLAVKOV *et al.*, 2013)	Técnica de seleção de atributos do tipo Filtro para problemas de classificação hierárquica multirrótulo.	bioinformática

Fonte: Autoria Própria

Em Santos e Nievola (2016), são exploradas técnicas de seleção de atributos em conjunto com classificadores hierárquicos de diferentes categorias objetivando melhorar suas respectivas performances. As bases utilizadas para os experimentos estão estruturadas no formato de árvore. Para realizar o processo de extração de atributos, são definidas medidas associadas ao objeto que se deseja extrair, de forma que as medidas sejam similares para objetos similares e diferentes para objetos distintos (SANTOS; NIEVOLA, 2016).

Em Pereira e Nievola (2016), o foco do trabalho é a redução da dimensionalidade da base de dados de proteínas para previsão de funções proteicas.As bases de dados utilizadas estão estruturadas em formato de árvore. Este trabalho explora a aplicação das abordagens de classifi-

cação hierárquica por nó, por nó pai e por camadas em conjunto à redução de dimensionalidade de atributos das relações utilizadas (PEREIRA; NIEVOLA, 2016).

Em Dias e Merschmann (2015) propõe-se uma adaptação da medida de incerteza simétrica para problemas hierárquicos monorrótulos. A avaliação da adaptação foi realizada por meio de uma abordagem do tipo Filtro onde numa etapa de pré-processamento a medida adaptada avalia a capacidade preditiva de cada atributo individualmente, permitindo a construção de um ranking dos mesmos a partir de suas avaliações. Em seguida, o classificador hierárquico Global-Model Naive Bayes foi utilizado em um procedimento incremental de seleção dos k melhores atributos do ranking para se avaliar a qualidade do ranking construído (DIAS; MERSCHMANN, 2015).

Em Koller e Sahami (1997) é desenvolvido por meio da abordagem de classificação local por nó em conjunto com a seleção de atributos, um *framework* probabilístico. Para cada nó da hierarquia de classes é construído um classificador binário e aplica-se um método de seleção de atributos, visando identificar os atributos mais relevantes para a construção de cada um dos classificadores locais. Como resultado, tem-se a melhora da acurácia preditiva e a obtenção de classificadores mais robustos e complexos (KOLLER; SAHAMI, 1997).

Em Slavkov et. al (2013), foi proposto um método de seleção de atributos para classificação hierárquica multirrótulo, sendo este capaz de lidar com a hierarquia de classes como um todo, ou seja, sem a decomposição do problema hierárquico em diversos problemas de classificação plana. Esse trabalho realizou uma adaptação do algoritmo *ReliefF* para o contexto hierárquico multirrótulo (SLAVKOV *et al.*, 2013).

4.4 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste Capítulo foram apresentados os trabalhos correlatos através da revisão sistemática e busca complementar na literatura por artigos, teses, dissertações e/ou pesquisas em andamento no tema deste trabalho.

A revisão sistemática permitiu com que fossem identificados, avaliados e interpretados todos os artigos de periódicos disponíveis e relevantes para a questão de pesquisa definida. Já a busca complementar, permitiu com que fossem verificados e incluídos no levantamento bibliográfico outros trabalhos correlatos a esse trabalho.

Após o levantamento bibliográfico foi verificado que há na literatura somente um artigo publicado referente a seleção de atributos em classificação hierárquica multirrótulo global, o qual utiliza a abordagem filtro.

5 METODOLOGIA

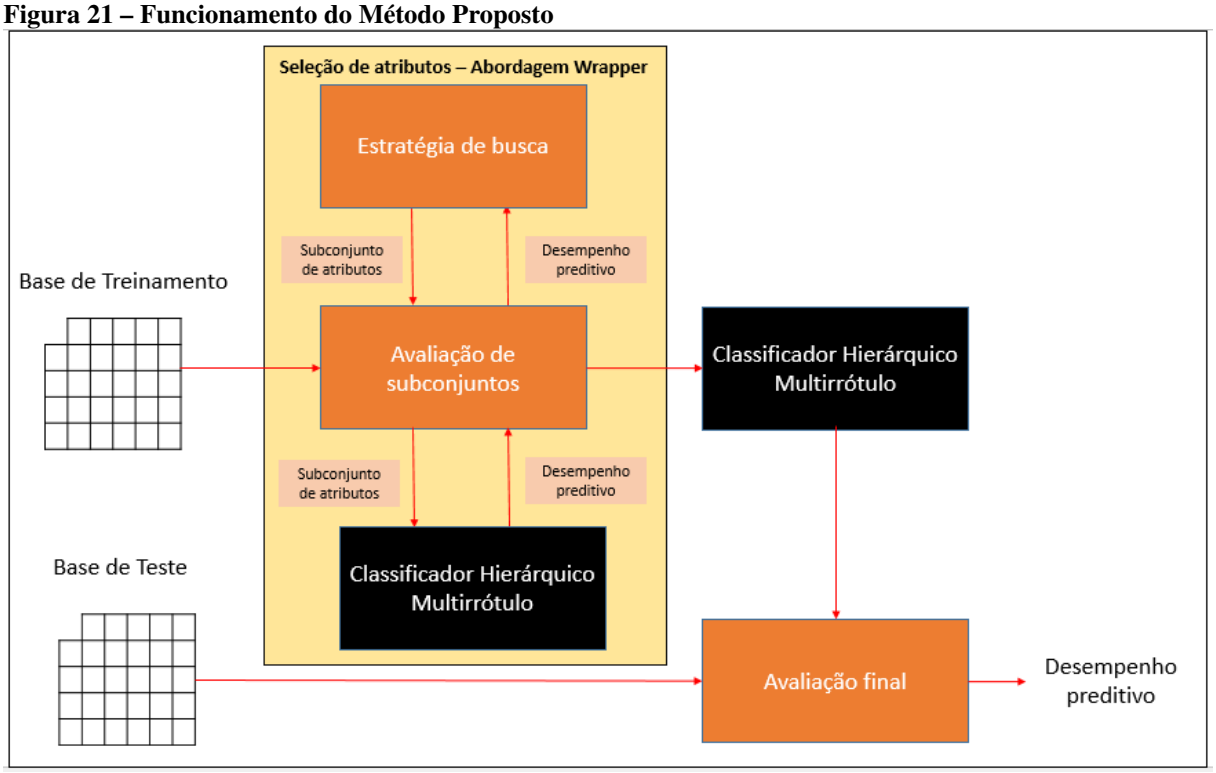
Neste Capítulo apresenta-se o método proposto para seleção de atributos em classificação hierárquica multirrótulo. Para tal, a Seção 5.1 descreve as etapas do método, a Seção 5.2 apresenta um exemplo da aplicação do método proposto. Por fim, a Seção 5.3 apresenta as considerações finais do Capítulo.

5.1 DESCRIÇÃO DO ALGORITMO

O método proposto para a seleção de atributos utiliza a abordagem *Wrapper*. Nesse tipo de abordagem, o próprio algoritmo de aprendizagem (classificador hierárquico multirrótulo) é usado como uma "caixa-preta"para obter uma medida de avaliação para cada subconjunto candidato.

Para a execução de um algoritmo que utiliza a abordagem $Wrapper$ exige-se a utilização de uma estratégia de busca e de um algoritmo de aprendizado capaz de avaliar a qualidade dos subconjuntos de atributos gerados. Com base nisso, na Figura 21 é apresentado o modelo do método proposto.

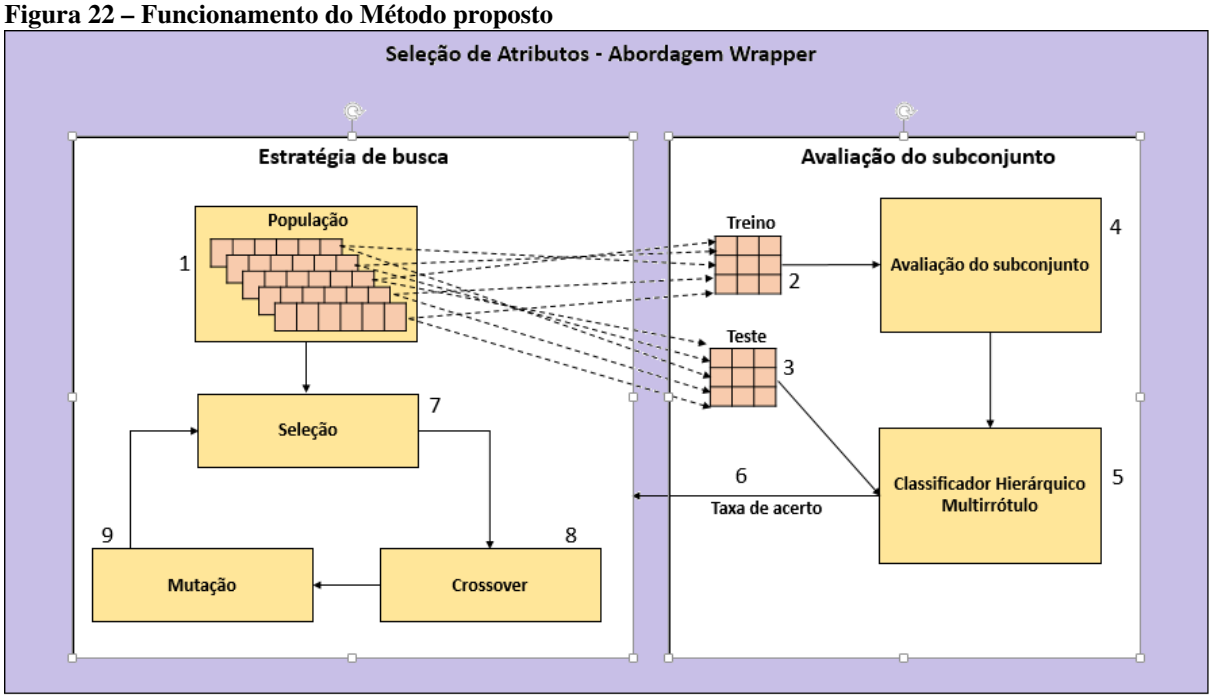
Figura 21 – Funcionamento do Método Proposto



Fonte: Autoria própria

O processo de seleção de atributos foi divido em 9 etapas principais, conforme pode ser observado na Figura 22.

Figura 22 – Funcionamento do Método proposto



Fonte: Autoria própria

Para que o processo de seleção de atributos seja iniciado, é necessário que seja definido o conjunto de dados e que o mesmo seja dividido em dois arquivos, a base de treinamento e a base de teste, como pode ser observado na Figura 21.

Na fase de treinamento, utiliza-se a estratégia de busca para explorar o espaço de possibilidades de subconjuntos de atributos, com o objetivo de encontrar o subconjunto que apresente o maior valor referente a medida de desempenho que foi escolhida.

A cada novo subconjunto encontrado, é criada uma base de dados reduzida contendo apenas os atributos contidos neste subconjunto. Essa base é utilizada pelo classificador hierárquico multirrótulo para que o mesmo possa criar o modelo de aprendizagem.

Com o modelo de aprendizagem criado, então inicia-se a fase de teste, nesta a base de dados de teste é reduzida contendo os mesmos atributos da base de treinamento e inicia-se a fase de avaliação do modelo.

Todas as instâncias da base de teste são analisadas e classificadas, gerando a medida de desempenho do classificador hierárquico multirrótulo. Essa medida é utilizada para avaliar o subconjunto de atributos, que originou as bases de treinamento e teste, é bom ou não. O processo encerra-se quando o critério de parada estabelecido para a estratégia de busca é atingido.

5.1.1 Estratégica de busca

Para este trabalho, optou-se pela escolha do algoritmo genético como estratégia de busca, sendo esta escolha justificada pela eficiência deste método em encontrar boas soluções

em grandes espaços de busca, como é o caso do espaço de busca relativo ao problema de seleção de atributos, que é descrito por 2^n estados, para uma situação com n atributos (SANTORO; NICOLETTI, 2004).

Outra característica importante com relação aos algoritmos genéticos é que eles são versáteis e podem utilizar diversos critérios para a seleção dos melhores atributos.São técnicas utilizadas para otimização, ou seja, procuraram várias soluções possíveis e utilizam a informação obtida nesse processo a fim de encontrar soluções cada vez melhores. A Tabela 5 apresenta o algoritmo criado de forma simplificada.

Tabela 5 – Passos para Redução do subconjunto de atributos utilizando a abordagem *Wrapper*

ENTRADA DO ALGORITMO
- Conjunto de dados de treinamento $BD_{trein} = [e_1, e_2, e_3, ..., e_q] formado por n atributos$
- Parâmetros de configuração do algoritmo genético: taxa de mutação tm, taxa de descarte td e quantidade de gerações g.
- Quantidade de genes s que compõe o tamanho do cromossomo
PASSO 1: INICIALIZAÇÃO
- Determinar o cálculo do tamanho t da população do Algoritmo Genético: $t = \frac{n!}{s!(n-s)!}$
- Extrair do conjunto de dados de treinamento a quantidade total de atributos n
- Determinar os valores da tm, td e g
PASSO 2: CRITÉRIO DE PARADA
- Número de gerações g
PASSO 3: SELEÇÃO DE ATRIBUTOS
- Inicializar o conjunto de indivíduos $P = \{i_1, i_2, i_3, ..., i_t\}$ que representa a população inicial
- Avaliar cada indivíduo i_j do conjunto P, em que o $fitness$ é a medida obtida pelo classificador hierárquico multirrótulo
- Selecionar os melhores indivíduos
- Reproduzir a população utilizando a técnica de $crossover$
- Verificar existência de mutação
- Retornar ao 3 até que a condição de parada seja satisfeita.
SAÍDA DO ALGORITMO
- Subconjuntos dos atributos selecionados

Fonte: Autoria própria

Inicialmente são gerados t possíveis conjuntos de atributos (cromossomos) randomicamente, sendo t o tamanho da população determinada pelo cálculo apresentado na Equação 20:

$$t = \frac{n!}{s!(n-s)!} \tag{20}$$

em que n representa a quantidade total de atributos da base de dados escolhida e s é o tamanho do cromossomo, variando no intervalo de $1\ a\ n$.

Os subconjuntos gerados durante a execução do algoritmo genético são avaliados conforme descrito na Seção 5.1.2. A cada geração são descartados os piores indivíduos da população naquele momento, o qual depende da taxa informada pelo usuário.

Com o intuito de garantir a diversidade da população são aplicados os operadores de cruzamento e mutação, com taxas de aplicação determinadas na execução do algoritmo.

5.1.2 Avaliação do subconjunto

Para avaliar o subconjunto utilizou-se um classificador hierárquico multirrótulo e foi escolhida uma medida que representa o desempenho preditivo desse classificador, podendo esta ser a taxa de acerto, precisão hierárquica, AUPRC, entre outras.

Neste trabalho, definiu-se que a função de aptidão de cada indivíduo do algoritmo genético será representada pela medida de desempenho do classificador hierárquico multirrótulo. Para isso, a cada indivíduo gerado, cria-se uma base de treinamento e de teste com apenas os atributos pertencentes ao subconjunto.

Tabela 6 – Passos para o funcionamento da fase de teste do algoritmo

ENTRADA DO ALGORITMO
- Conjunto de dados de teste reduzida $BD_{teste} = [e_1, e_2, e_3, ..., e_q]$ em que q é o total de instâncias de teste.
PASSO 1: CRITÉRIO DE PARADA
- Até que todas as instâncias tenham sido selecionadas e testadas
PASSO 3: TESTE
- Selecionar uma instância e_i do conjunto de dados de entrada $BD_{teste} = [e_1, e_2, e_3, ..., e_q]$ - Classificar a instância e_i conforme o modelo de aprendizado criado na fase de treinamento
SAÍDA DO ALGORITMO
- Medida de desempenho do classificador hierárquico multirrótulo.

Fonte: Autoria própria

Conforme demonstrado na Tabela 5 e na Tabela 6 espera-se que ao final da execução do algoritmo proposto, o melhor individuo represente a melhor escolha de atributos possível para aquele cenário.

5.2 SIMULAÇÃO DO ALGORITMO

Para melhor entendimento do método desenvolvido, será utilizada uma base hierárquica multirrótulo fictícia, sendo esta representada na Tabela 7.

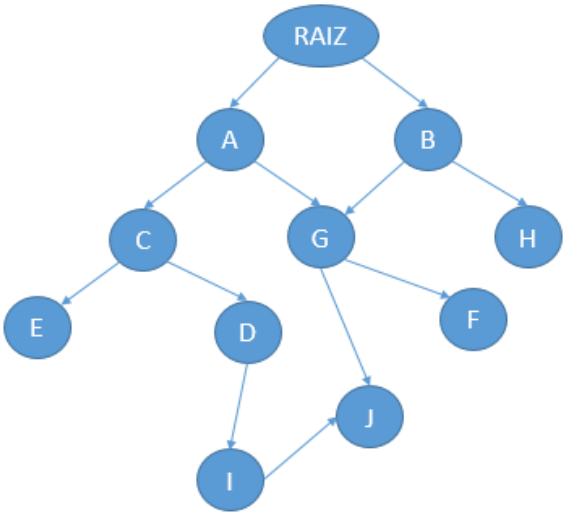
Tabela 7 – Base de dados

	A1	A2	A3	A4	A5	A6	A7	A8	A9	CLASSE
E1	0.67	0.49	0.06	0.54	0.35	0.01	0.63	0.19	0.16	A
E2	0.45	0.42	0.56	0.92	0.95	0.34	0.21	0.88	0.16	C@E@I
E3	0.08	0.93	0.20	0.36	0.76	0.05	0.58	0.88	0.90	B@F
E4	0.84	0.37	0.10	0.35	0.47	0.78	0.92	0.81	0.01	B@H@I
E5	0.79	0.89	0,26	0,22	0,44	0,57	0,21	0,75	0,46	A@D
E6	0.27	0.49	0,04	0,41	0,83	0,19	0,36	0,67	0,91	G
E7	0.75	0.32	0,35	0,74	0,07	0,28	0,42	0,24	0,18	A@D@C@F
E8	0.85	0.11	0,95	0,89	0,69	0,47	0,16	0,35	0,67	B@I
E9	0.26	0.82	0,53	0,95	0,74	0,89	0,33	0,41	0,49	D
E10	0.36	0.76	0,88	0,04	0,11	0,61	0,77	0,58	0,24	H@J

Fonte: Autoria Própria

Cada linha da Tabela 7 representa uma instância (exemplo), totalizando assim 10 exemplos. Cada exemplo está representado pela letra E seguido da numeração sequencial. As colunas representam os atributos (características) de cada exemplo, são representados pela sigla A, com exceção da última coluna, que representa o atributo classe. A hierarquia das classes está representada por meio de um DAG conforme observa-se na Figura 23.

Figura 23 – Hierarquia das classes



Fonte: Autoria Própria

Conforme descrito anteriormente, para a aplicação do método proposto é necessário que seja aplicada alguma técnica de divisão da base de dados, na base de treinamento e na base de teste. A Tabela 8 e a Tabela 9 apresentam as bases obtidas de treinamento e teste, respectivamente.

Tabela 8 – Base de treinamento

	A1	A2	A3	A4	A5	A6	A7	A8	A9	CLASSE
E2	0.45	0.42	0.56	0.92	0.95	0.34	0.21	0.88	0.16	C@E@I
E4	0.84	0.37	0.10	0.35	0.47	0.78	0.92	0.81	0.01	B@H@I
E6	0.27	0.49	0.04	0.41	0.83	0.19	0.36	0.67	0.91	G
E8	0.85	0.11	0.95	0.89	0.69	0.47	0.16	0.35	0.67	B@I
E9	0.26	0.82	0.53	0.95	0.74	0.89	0.33	0.41	0.49	D
E10	0.36	0.76	0.88	0.04	0.11	0.61	0.77	0.58	0.24	H@J

Fonte: Autoria própria

Tabela 9 – Base de teste

	A1	A2	A3	A4	A5	A6	A7	A8	A9	CLASSE
E1	0.67	0.49	0.06	0.54	0.35	0.01	0.63	0.19	0.16	A
E3	0.08	0.93	0.20	0.36	0.76	0.05	0.58	0.88	0.90	B@F
E5	0.79	0.89	0.26	0.22	0.44	0.57	0.21	0.75	0.46	A@D
E7	0.75	0.32	0.35	0.74	0.07	0.28	0.42	0.24	0.18	A@D@C@F

Fonte: Autoria própria

Após a divisão da base, conforme pode ser observado na Tabela 8 e Tabela 9, a base de treinamento ficou com 6 amostras e a a base de teste com 4 amostras.

5.2.1 Passos para redução do subconjunto de atributos utilizando a abordagem $Wrapper$ e Classificação Hierárquica Multirrótulo

A primeira etapa do método proposto consiste na inicialização dos parâmetros da estratégia de busca. Para isso, foram definidos os parâmetros tm, td e g, conforme os valores apresentados na Tabela 10.

Tabela 10 – Parâmetros de configuração do AG

Parâmetro	Valor
tm	0.01
td	0.5
g	1
s	8

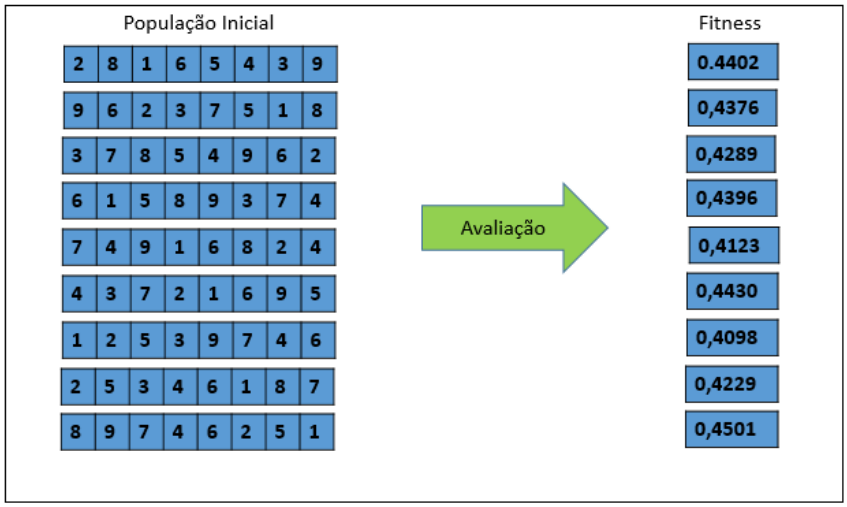
Fonte: Autoria própria

A partir da definição do parâmetro s e a leitura da base de treinamento para contagem do total de atributos n, é possível calcular o tamanho da população. Para o valor de $s = 8$ e de $n = 9$, tem-se pela aplicação da Equação 20, $t = 9$.

Cada indivíduo da população, também chamado de cromossomo, possui um conjunto de tamanho fixo de bits, ou genes, que são representados em uma base de codificação. Nesse exemplo esse tamanho representa o valor do parâmetro s.

Com a definição e inicialização dos parâmetros, já pode-se iniciar o processo de seleção de atributos. O algoritmo inicia-se com a criação da população de maneira aleatória. A Figura 24 representa uma possível população e o processo de avaliação.

Figura 24 – População inicial do AG



Fonte: Autoria Própria

A cada geração, é necessário selecionar os piores indivíduos, ou seja, aqueles que possuem o menor valor da função de aptidão. Para os indivíduos gerados na Figura 24, o descarte aconteceria conforme descrito na Figura 25.

Figura 25 – Descarte da população

Fonte: Autoria Própria

Para manter a população sempre do mesmo tamanho, novos indivíduos são gerados através do processo de recombinação ou *crossover*. Ainda na etapa de reprodução, ocorre o processo de mutação nos descendentes recém-gerados, permitindo assim uma diversidade na população.

Após o descarte são gerados novos indivíduos na população através dos operadores de *crossover* e mutação. Esse processo de seleção, reprodução e mutação são repetidos até que o critério de parada seja atingido.

Ao final da execução do algoritmo de seleção, considerando que os indivíduos apresentados na Figura 24, por exemplo, representassem a população final, o indivíduo com o valor de aptidão 0.4501 seria o melhor indivíduo encontrado pelo AG e os atributos contidos nesse indivíduo seriam utilizados para a fase de teste.

Na fase de teste, ao considerar o indivíduo com o valor de aptidão 0.4501 teria-se a base de dados apresentada na Tabela 11.

Tabela 11 – Base de teste reduzida

	A1	A2	A4	A5	A6	A7	A8	A9	CLASSE
E1	0.67	0.49	0.54	0.35	0.01	0.63	0.19	0.16	A
E2	0.08	0.93	0.36	0.76	0.05	0.58	0.88	0.90	B@F
E3	0.79	0.89	0.22	0.44	0.57	0.21	0.75	0.46	A@D
E4	0.75	0.32	0.74	0.07	0.28	0.42	0.24	0.18	A@D@C@F

Fonte: Autoria própria

Toda instância da base de dados de teste é analisada e classificada obtendo-se no final do processo de classificação, a medida da área abaixo da curva de PR (AUPRC).

5.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste Capítulo foi apresentado o método proposto para a seleção de atributos utilizando a abordagem *Wrapper* em classificação hierárquica multirrótulo. O método desenvolvido utiliza como estratégia de busca o Algoritmo Genético e um classificador hierárquico multirrótulo para avaliação dos subconjuntos, sendo definida uma métrica para o desempenho preditivo.

Também foram apresentados os pseudocódigos da etapa de redução do conjunto de atributos e da etapa de teste. Por fim, realizou-se uma simulação mostrando o passo a passo da execução do método proposto em uma base de dados fictícia.

6 EXPERIMENTOS E RESULTADOS

Neste Capítulo são apresentados os experimentos e resultados preliminares alcançados com o método proposto. Para isso, a Seção 6.1 descreve as bases de dados utilizadas para a realização dos experimentos, enquanto a Seção 6.2 faz uma breve descrição sobre os experimentos iniciais que foram feitos utilizando conjunto de dados estruturados na forma de DAG. Por fim, a Seção 6.3 apresenta as considerações finais do trabalho.

6.1 BASE DE DADOS

O conjunto de dados utilizados nos experimentos são dados biológicos da área genômica funcional. Eles foram utilizados por Borges (2012), a qual normalizou as bases e fez a imputação dos atributos faltantes

A Tabela 12 mostra as principais características das bases de dados usadas nos experimentos. Uma descrição mais detalhada sobre cada uma delas podem ser encontradas em Borges (2012).

Tabela 12 – Características da base de dados GO

Base de dados	**Qtde. Amostras**	**Qtde. Atributos**	**Qtde. Classes**	**Qtde. Max. Níveis**	**Qtde. Min/-Max Classes por Amostra**	**Qtde. Min/-Max Amostras por Classe**
Cellcycle	3751	77	4125	13	3/28	0/785
Church	3749	27	4125	13	3/28	0/786

Fonte: Adaptado de Borges (2012)

6.2 EXPERIMENTOS INICIAIS

Os experimentos foram realizados em um notebook com o Sistema Operacional Windows 10, processador intel core i5, 8 GB de memória RAM e 1 TB de HD.

Conforme descrito na Seção 3, a abordagem *Wrapper* utiliza uma estratégia de busca para encontrar o melhor subconjunto de atributos, ou seja, o subconjunto que resulta no maior valor preditivo do classificador hierárquico multirrótulo. Nos experimentos iniciais optou-se pela escolha do algoritmo genético como estratégia de busca e o do Clus-HMC (VENS *et al.*, 2008), como classificador hierárquico multirrótulo, bem como a medida de desempenho AUPRC.

O Clus HMC foi escolhido devido ao fato de ser bastante utilizado em classificação hierárquica multirrótulo, possuindo assim resultados consolidados na literatura para que posteriormente possa ser realizada uma comparação do desempenho do método de seleção de atributos

proposto.

Além disso, os métodos de seleção de atributos baseados na abordagem $Wrapper$ costumam utilizar algoritmos de aprendizado que exigem menor poder computacional tal como kNN ou árvores de decisão, critério no qual o Clus-HMC está enquadrado. É importante ressaltar que ainda assim, é necessário a utilização de técnicas de otimização na busca, pra garantir um melhor desempenho.

Os parâmetros utilizados para o Algoritmo Genético estão apresentados na Tabela 13.

Tabela 13 – Parâmetros do Algoritmo Genético

Parâmetro	Valor utilizado
tm	0.01
td	0.25
g	30
t	100

Fonte: Autoria Própria

O tamanho da população foi fixado em 100 indivíduos, pois o objetivo era somente a validação dos resultados obtidos pelo algoritmo e também pelo tempo de processamento de cada experimento.

A condição de parada do algoritmo foi estabelecida como sendo o número de gerações. Na Tabela 14 e na Tabela 15 são apresentados os resultados obtidos nas bases de dados Cellcycle e Church, respectivamente. Os valores obtidos em cada um dos experimentos variando-se o valor do tamanho do subconjunto de atributos para as bases.

Tabela 14 – Resultado obtido na base Cellcycle

Base de dados: Cellcycle										
Atributos										
Método proposto										**Clus-HMC**
7	14	21	28	35	42	49	56	63	70	77
0,439	0,443	0,439	0,440	0,437	0,443	0,443	0,442	0,437	0,434	0,440

Fonte: Autoria própria

Tabela 15 – Resultado obtido na base Church

Base de dados: Church			
Atributos			
Método proposto			**Clus-HMC**
7	14	21	27
0,449	0,451	0,452	0,450

Fonte: Autoria própria

O resultado obtido pela primeira versão do algoritmo para as bases Cellcycle e Church considera-se significativo, embora não tenha sido realizado ainda um teste estatístico, visto que o algoritmo com apenas 7 atributos consegue um resultado muito próximo do valor da medida AUPRC obtido pelo classificador utilizando todos os atributos.

6.3 CONSIDERAÇÕES FINAIS

Neste capítulo apresentou-se os resultados preliminares do método proposto. Foi realizada uma comparação do desempenho do classificador hierárquico multirrótulo Clus-HMC com todos os atributos das bases de dados e após a seleção de atributos, onde variou-se o tamanho do subconjunto de atributos.

O método proposto apresentou resultados similares ao obtido pelo Clus-HMC sem a seleção de atributos, porém o valor obtido da medida AUPRC foi para conjunto de atributos de tamanho inferiores, permitindo assim com que houvesse um ganho computacional no tempo de processamento das bases de dados.

7 CRONOGRAMA

Neste Capítulo estão descritas as atividadae para o desenvolvimento do trabalho, bem como as considerações finais do mesmo. Para isso, a Seção 7.1 apresenta o cronograma e por fim, a Seção 7.2 apresenta as considerações finais deste trabalho.

7.1 ATIVIDADES A SEREM REALIZADAS

O Quadro 8 apresenta as atividades realizadas e programadas para que o projeto proposto seja desenvolvido. O símbolo * indica que a atividade já foi realizada, enquanto o símbolo + indica as atividades em andamento ou que ainda não foram iniciadas.

Quadro 8 – Cronograma de desenvolvimento do trabalho

Atividade	2016					2017												2018					
	AGO	SET	OUT	NOV	DEZ	JAN	FEV	MAR	ABR	MAI	JUN	JUL	AGO	SET	OUT	NOV	DEZ	JAN	FEV	MAR	ABR	MAI	JUN
Realização das disciplinas	*	*	*	*	*	*	*	*	*	*	*	*											
Pesquisa e levantamento bibliográfico				*	*	*	*	*	*	*	*	*	*	*	+	+	+	+	+	+	+		
Escrita do trabalho de Qualificação							*	*	*	*	*	*											
Qualificação																+							
Criação de um protótipo do método proposto							*	*	*	*	*												
Desenvolvimento do método											*	*	*	*	+	+	+						
Experimentos											*	*	*	*	+	+	+						
Análise estatística dos experimentos																	+	+					
Desenvolvimento da dissertação															+	+	+	+	+	+	+	+	
Defesa do trabalho																							+

Fonte: Autoria Própria

Segue o detalhamento das principais atividades definidas no Quadro 8:

- Realização das disciplinas: Durante o primeiro ano do mestrado foram cursadas 6 disciplinas, sendo estas divididas em Núcleo Comum, Núcleo Básico e Tópicos Avançados. Para o primeiro grupo foram realizadas as disciplinas de Análise e Projeto de Algoritmos e Metodologia da Pesquisa. Já para o segundo grupo foram cursadas as disciplinas de Estrutura de Dados e Processamento de Imagens. Por fim, para o terceiro grupo foram cursadas as disciplinas de Tópicos Avançados em Métodos Computacionais e Modelagem e Simulação Computacional.

- Pesquisa e levantamento bibliográfico: Para esta atividade foi realizada uma revisão sistemática da literatura, conforme descrito no capítulo 4. Essa atividade ainda está em aberto, pois durante o andamento da pesquisa deve-se realizar novamente uma pesquisa na literatura com o intuito de verificar trabalhos publicados recentemente.

- Escrita do trabalho de qualificação: A escrita do exame de qualificação iniciou-se juntamente com as atividades do mestrado na disciplina de Metodologia da Pesquisa e foi finalizada no mês de setembro do ano seguinte.

- Criação de um protótipo do método proposto: A primeira versão do método de seleção foi elaborado e foram realizados os experimentos iniciais, sendo que estes permitem o levantamento dos pontos a serem melhorados ou repensados.
- Desenvolvimento do método: Iniciou-se o processo de aprimoramento da técnica desenvolvida. A ideia é que sejam testadas outras técnicas bioinspiradas para efeito de comparação com o algoritmo genético.
- Análise estatística dos experimentos, Término da escrita e Defesa: Estas atividades ainda não foram iniciadas.

7.2 CONSIDERAÇÕES FINAIS DO TRABALHO

Com base na realização da revisão sistemática da literatura pode-se perceber a originalidade da pesquisa em andamento, pois encontrou-se apenas uma técnica desenvolvida para a seleção de atributos em problemas de classificação hierárquica multirrótulo e o mesmo utiliza a abordagem Filtro enquanto o método proposto neste trabalho utiliza a abordagem Wrapper.

As atividades propostas para que seja alcançado o objetivo geral e específico estão sendo realizadas dentro do prazo estabelecido e a primeira versão do método proposto apresentou resultados significativos nos primeiros experimentos realizados.

Como próxima etapa, serão ampliados os testes com outras bases de dados hierárquicas para comprovar a eficácia do método proposto. Além da comparação do algoritmo genético com outra abordagem bioinspirada afim de verificar de ganho no processo de busca pelo melhor subconjunto de atributos.

REFERÊNCIAS

AHA, David W; BANKERT, Richard L. A comparative evaluation of sequential feature selection algorithms. In: **Learning from data**. [S.l.]: Springer, 1996. p. 199–206.

ALMEIDA, Thissiany Beatriz; BORGES, Helyane Bronoski. An adaptation of the ml-knn algorithm to predict the number of classes in hierarchical multi-label classification. In: SPRINGER. **Modeling Decisions for Artificial Intelligence**. [S.l.], 2017. p. 77–88.

ALVES, Roberto Teixeira. Um sistema imunológico artificial para classificação hierárquica e multi-label de funções de proteínas. Universidade Tecnológica Federal do Paraná, 2010.

BANZHAF, Wolfgang *et al.* **Genetic programming: an introduction**. [S.l.]: Morgan Kaufmann San Francisco, 1998. v. 1.

BARALDI, Piero *et al.* Hierarchical k-nearest neighbours classification and binary differential evolution for fault diagnostics of automotive bearings operating under variable conditions. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 56, p. 1–13, 2016.

BARUTCUOGLU, Zafer; SCHAPIRE, Robert E; TROYANSKAYA, Olga G. Hierarchical multi-label prediction of gene function. **Bioinformatics**, Oxford Univ Press, v. 22, n. 7, p. 830–836, 2006.

BEN-BASSAT, M. Pattern recognition and reduction of dimensionality. **Handbook of Statistics**, North Holland, v. 2, n. 1982, p. 773–910, 1982.

BLOCKEEL, Hendrik *et al.* Decision trees for hierarchical multilabel classification: A case study in functional genomics. In: SPRINGER. **European Conference on Principles of Data Mining and Knowledge Discovery**. [S.l.], 2006. p. 18–29.

BLUM, Avrim L; LANGLEY, Pat. Selection of relevant features and examples in machine learning. **Artificial intelligence**, Elsevier, v. 97, n. 1, p. 245–271, 1997.

BORGES, HELYANE BRONOSKI. **CLASSIFICADOR HIERÁRQUICO MULTIRRÓTULO USANDO UMA REDE NEURAL COMPETITIVA**. Tese (Doutorado) — Pontifícia Universidade Católica do Paraná, 2012.

BOUTELL, Matthew R *et al.* Learning multi-label scene classification. **Pattern recognition**, Elsevier, v. 37, n. 9, p. 1757–1771, 2004.

BOZ, Olcay. Feature subset selection by using sorted feature relevance. In: **ICMLA**. [S.l.: s.n.], 2002. p. 147–153.

CAO, Lu *et al.* Hierarchical classification strategy for phenotype extraction from epidermal growth factor receptor endocytosis screening. **BMC bioinformatics**, BioMed Central, v. 17, n. 1, p. 196, 2016.

CARVALHO, André de; FREITAS, Alex. A tutorial on multi-label classification techniques. **Foundations of Computational Intelligence Volume 5**, Springer, p. 177–195, 2009.

CERRI, Ricardo. **Técnicas de classificação hierárquica multirrótulo**. Tese (Doutorado) — Universidade de São Paulo, 2010.

CERRI, Ricardo *et al.* Reduction strategies for hierarchical multi-label classification in protein function prediction. **BMC bioinformatics**, BioMed Central, v. 17, n. 1, p. 373, 2016.

CESA-BIANCHI, Nicolò; GENTILE, Claudio; ZANIBONI, Luca. Incremental algorithms for hierarchical classification. **Journal of Machine Learning Research**, v. 7, n. Jan, p. 31–54, 2006.

CHANDRASHEKAR, Girish; SAHIN, Ferat. A survey on feature selection methods. **Computers & Electrical Engineering**, Elsevier, v. 40, n. 1, p. 16–28, 2014.

CHANG, Yongjun *et al.* Fast and efficient lung disease classification using hierarchical one-against-all support vector machine and cost-sensitive feature selection. **Computers in biology and medicine**, Elsevier, v. 42, n. 12, p. 1157–1164, 2012.

CLARE, Amanda. **Machine learning and data mining for yeast functional genomics**. Tese (Doutorado) — The University of Wales, 2003.

CLARE, Amanda; KING, Ross D. Knowledge discovery in multi-label phenotype data. In: SPRINGER. **European Conference on Principles of Data Mining and Knowledge Discovery**. [S.l.], 2001. p. 42–53.

DAVIS, Jesse; GOADRICH, Mark. The relationship between precision-recall and roc curves. In: ACM. **Proceedings of the 23rd international conference on Machine learning**. [S.l.], 2006. p. 233–240.

DIAS, Thieres Nardy; MERSCHMANN, Luiz Henrique de C. Adaptação da medida incerteza simétrica para a seleção de atributos no contexto de classificação hierárquica monorrótulo. **Anais do Encontro Nacional de Inteligência Artificial e Computacional, Natal, RN, Brazil**, p. 142–149, 2015.

DIMITROVSKI, Ivica *et al.* Hierarchical classification of diatom images using ensembles of predictive clustering trees. **Ecological Informatics**, Elsevier, v. 7, n. 1, p. 19–29, 2012.

DONG, Yu-ning; ZHAO, Jia-jie; JIN, Jiong. Novel feature selection and classification of internet video traffic based on a hierarchical scheme. **Computer Networks**, Elsevier, v. 119, p. 102–111, 2017.

ELISSEEFF, André; WESTON, Jason. A kernel method for multi-labelled classification. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2002. p. 681–687.

FACELI, Katti *et al.* Inteligência artificial: Uma abordagem de aprendizado de máquina. **Rio de Janeiro: LTC**, v. 2, p. 192, 2011.

FIX, Evelyn; JR, Joseph L Hodges. **Discriminatory analysis-nonparametric discrimination: Small sample performance**. [S.l.], 1952.

FOROUTAN, Iman; SKLANSKY, Jack. Feature selection for automatic classification of non-gaussian data. **IEEE Transactions on Systems, Man, and Cybernetics**, IEEE, v. 17, n. 2, p. 187–198, 1987.

FREEMAN, Cecille; KULIC, Dana; BASIR, Otman. Feature-selected tree-based classification. **IEEE transactions on cybernetics**, IEEE, v. 43, n. 6, p. 1990–2004, 2013.

FREITAS, Alex A; CARVALHO, Andre CPLF de. A tutorial on hierarchical classification with applications in bioinformatics. In: **In: D. Taniar (Ed.) Research and Trends in Data Mining Technologies and Applications, Idea Group, 2007**. [S.l.: s.n.], 2007.

FREUND, Yoav; SCHAPIRE, Robert E. A desicion-theoretic generalization of on-line learning and an application to boosting. In: SPRINGER. **European conference on computational learning theory**. [S.l.], 1995. p. 23–37.

GHAMRAWI, Nadia; MCCALLUM, Andrew. Collective multi-label classification. In: ACM. **Proceedings of the 14th ACM international conference on Information and knowledge management**. [S.l.], 2005. p. 195–200.

GOLDBERG, David E. Genetic algorithms in search, optimization, and machine learning, 1989. **Reading: Addison-Wesley**, 1989.

GONÇALVES, Teresa; QUARESMA, Paulo. A preliminary approach to the multilabel classification problem of portuguese juridical documents. In: SPRINGER. **Portuguese Conference on Artificial Intelligence**. [S.l.], 2003. p. 435–444.

GOPAL, Siddharth; YANG, Yiming. Recursive regularization for large-scale classification with hierarchical and graphical dependencies. In: ACM. **Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining**. [S.l.], 2013. p. 257–265.

GUYON, Isabelle; ELISSEEFF, André. An introduction to feature extraction. In: **Feature extraction**. [S.l.]: Springer, 2006. p. 1–25.

HALL, Mark A. Correlation-based feature selection of discrete and numeric class machine learning. University of Waikato, Department of Computer Science, 2000.

HOLLAND, John H. **Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence**. [S.l.]: MIT press, 1992.

HOLSCHUH, Leonardo de Moraes *et al.* Contribuições para o aprendizado por busca de projeção. [sn], 2008.

HUANG, Phoenix X; BOOM, Bastiaan J; FISHER, Robert B. Hierarchical classification with reject option for live fish recognition. **Machine Vision and Applications**, Springer, v. 26, n. 1, p. 89–102, 2015.

JOHN, George H *et al.* Irrelevant features and the subset selection problem. In: **Machine learning: proceedings of the eleventh international conference**. [S.l.: s.n.], 1994. p. 121–129.

KIRA, Kenji; RENDELL, Larry A. The feature selection problem: Traditional methods and a new algorithm. In: **AAAI**. [S.l.: s.n.], 1992. v. 2, p. 129–134.

KIRITCHENKO, Svetlana; MATWIN, Stan; FAMILI, Fazel. Hierarchical text categorization as a tool of associating genes with gene ontology codes. 2004.

KOHAVI, Ron; JOHN, George H. Wrappers for feature subset selection. **Artificial intelligence**, Elsevier, v. 97, n. 1-2, p. 273–324, 1997.

KOLLER, Daphne; SAHAMI, Mehran. **Hierarchically classifying documents using very few words**. [S.l.], 1997.

KOUSSOUNADIS, Antonis; REDFERN, Oliver C; JONES, David T. Improving classification in protein structure databases using text mining. **BMC bioinformatics**, BioMed Central, v. 10, n. 1, p. 129, 2009.

LANGLEY, Pat; IBA, Wayne. Average-case analysis of a nearest neighbor algorithm. In: **IJCAI**. [S.l.: s.n.], 1993. p. 889–894.

LAUSER, Boris; HOTHO, Andreas. Automatic multi-label subject indexing in a multilingual environment. In: SPRINGER. **International Conference on Theory and Practice of Digital Libraries**. [S.l.], 2003. p. 140–151.

LIU, Huan; MOTODA, Hiroshi. **Computational methods of feature selection**. [S.l.]: CRC Press, 2007.

LIU, Huan; SETIONO, Rudy *et al.* A probabilistic approach to feature selection-a filter solution. In: **ICML**. [S.l.: s.n.], 1996. v. 96, p. 319–327.

LUO, Xiao; ZINCIR-HEYWOOD, A Nur. Evaluation of two systems on multi-class multi-label document classification. In: SPRINGER. **International Symposium on Methodologies for Intelligent Systems**. [S.l.], 2005. p. 161–169.

MCCALLUM, Andrew. Multi-label text classification with a mixture model trained by em. In: **AAAI workshop on Text Learning**. [S.l.: s.n.], 1999. p. 1–7.

MENZE, Bjoern H; PETRICH, Wolfgang; HAMPRECHT, Fred A. Multivariate feature selection and hierarchical classification for infrared spectroscopy: serum-based detection of bovine spongiform encephalopathy. **Analytical and bioanalytical chemistry**, Springer, v. 387, n. 5, p. 1801–1807, 2007.

MITCHELL, Tom M. Machine learning. 1997. **Burr Ridge, IL: McGraw Hill**, v. 45, n. 37, p. 870–877, 1997.

MONARD, MC; BARANAUSKAS, JA. Conceitos sobre aprendizado de máquina. capítulo 4. **REZENDE, SO Sistemas Inteligentes: Fundamentos e Aplicações. Barueri: Manole**, 2002.

NAM, Jinseok *et al.* Large-scale multi-label text classification—revisiting neural networks. In: SPRINGER. **Joint european conference on machine learning and knowledge discovery in databases**. [S.l.], 2014. p. 437–452.

NANCULEF, Ricardo; FLAOUNAS, Ilias; CRISTIANINI, Nello. Efficient classification of multi-labeled text streams by clashing. **Expert Systems with Applications**, Elsevier, v. 41, n. 11, p. 5431–5450, 2014.

OTERO, Fernando EB; FREITAS, Alex A; JOHNSON, Colin G. A hierarchical multi-label classification ant colony algorithm for protein function prediction. **Memetic Computing**, Springer, v. 2, n. 3, p. 165–181, 2010.

PAES, Bruno C; PLASTINO, Alexandre; FREITAS, Alex A. Seleção de atributos aplicada à classificação hierárquica. In: **Symposium on Knowledge Discovery, Mining and Learning-KDMiLe**. [S.l.: s.n.], 2013.

____. Exploring attribute selection in hierarchical classification. **Journal of Information and Data Management**, v. 5, n. 1, p. 124, 2014.

PAGANI, Regina Negri; KOVALESKI, João Luiz; RESENDE, Luis Mauricio. Methodi ordinatio: a proposed methodology to select and rank relevant scientific papers encompassing the impact factor, number of citation, and year of publication. **Scientometrics**, Springer, v. 105, n. 3, p. 2109–2135, 2015.

PAN, Hong; ZHU, Yaping; XIA, Liangzheng. Efficient and accurate face detection using heterogeneous feature descriptors and feature selection. **Computer Vision and Image Understanding**, Elsevier, v. 117, n. 1, p. 12–28, 2013.

PAPPA, Gisele Lobo. **Seleção de atributos utilizando Algoritmos Genéticos multiobjetivos**. Tese (Doutorado) — Pontifícia Universidade Católica do Paraná, 2002.

PERALTA, Daniel *et al.* Distributed incremental fingerprint identification with reduced database penetration rate using a hierarchical classification based on feature fusion and selection. **Knowledge-Based Systems**, Elsevier, v. 126, p. 91–103, 2017.

PEREIRA, Leonardo Henrique; NIEVOLA, Júlio Cesar. **Análise de Técnicas de Classificação Hierárquica usando Seleção de Atributos para Previsão da Função de Proteínas**. [S.l.], 2016.

RAEDT, Luc De; BLOCKEEL, Hendrik. Using logical decision trees for clustering. **Inductive Logic Programming**, Springer, p. 133–140, 1997.

REUNANEN, Juha. Overfitting in making comparisons between variable selection methods. **Journal of Machine Learning Research**, v. 3, n. Mar, p. 1371–1382, 2003.

REZENDE, Solange Oliveira. **Sistemas inteligentes: fundamentos e aplicações**. [S.l.]: Editora Manole Ltda, 2003.

ROMAO, LUIZ MELO. **Classificaçao Global Hierárquica Multirrótulo da Funçao de Proteınas Utilizando Sistemas Classificadores**. Tese (Doutorado) — PhD thesis, Pontificia Universidade Católica do Paraná, 2012.

ROSSI, Rafael Geraldeli *et al.* Inductive model generation for text classification using a bipartite heterogeneous network. **Journal of Computer Science and Technology**, Springer, v. 29, n. 3, p. 361–375, 2014.

ROUSU, Juho *et al.* Kernel-based learning of hierarchical multilabel classification models. **Journal of Machine Learning Research**, v. 7, n. Jul, p. 1601–1626, 2006.

SAEYS, Yvan; INZA, Iñaki; LARRAÑAGA, Pedro. A review of feature selection techniques in bioinformatics. **bioinformatics**, Oxford Univ Press, v. 23, n. 19, p. 2507–2517, 2007.

SANTORO, Daniel Monegatto; NICOLETTI, Maria do Carmo. Sobre o processo de seleção de atributos utilizando algoritmo genético direcionado por uma rede neural construtiva. 2004.

SANTOS, Joelma Carla. Extração de atributos de forma e seleção de atributos usando algoritmos genéticos para classificação de regiões. **Instituto Nacional de Pesquisas Espaciais (INPE)**, 2007.

SANTOS, Maria Angela Moscalewski Roveredo dos; NIEVOLA, Júlio Cesar. **SELEÇÃO DE ATRIBUTOS PARA CLASSIFICAÇÃO HIERÁRQUICA**. [S.l.], 2016.

SCHAPIRE, Robert E; SINGER, Yoram. Improved boosting algorithms using confidence-rated predictions. **Machine learning**, Springer, v. 37, n. 3, p. 297–336, 1999.

____. Boostexter: A boosting-based system for text categorization. **Machine learning**, Springer, v. 39, n. 2, p. 135–168, 2000.

SECKER, Andrew *et al.* Hierarchical classification of g-protein-coupled receptors with data-driven selection of attributes and classifiers. **International journal of data mining and bioinformatics**, Inderscience Publishers, v. 4, n. 2, p. 191–210, 2010.

SHEN, Xipeng *et al.* Multilabel machine learning and its application to semantic scene classification. In: **storage and retrieval methods and applications for multimedia**. [S.l.: s.n.], 2004. p. 188–199.

SILLA JR., Carlos N.; FREITAS, Alex A. A survey of hierarchical classification across different application domains. **Data Min. Knowl. Discov.**, Kluwer Academic Publishers, Hingham, MA, USA, v. 22, n. 1-2, p. 31–72, jan. 2011. ISSN 1384-5810. Disponível em: <http://dx.doi.org/10.1007/s10618-010-0175-9>.

SLAVKOV, Ivica *et al.* Relieff for hierarchical multi-label classification. In: SPRINGER. **International Workshop on New Frontiers in Mining Complex Patterns**. [S.l.], 2013. p. 148–161.

STRUYF, Jan *et al.* Hierarchical multi-classification with predictive clustering trees in functional genomics. In: SPRINGER. **EPIA**. [S.l.], 2005. p. 272–283.

SUN, Aixin; LIM, Ee-Peng. Hierarchical text classification and evaluation. In: IEEE. **Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on**. [S.l.], 2001. p. 521–528.

TAN, Pang-Ning *et al.* **Introduction to data mining**. [S.l.]: Pearson Education India, 2006.

TROHIDIS, Konstantinos *et al.* Multi-label classification of music into emotions. In: **ISMIR**. [S.l.: s.n.], 2008. v. 8, p. 325–330.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis. Multi-label classification: An overview. **International Journal of Data Warehousing and Mining**, v. 3, n. 3, 2006.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis; VLAHAVAS. Random k-labelsets for multilabel classification. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 23, n. 7, p. 1079–1089, 2011.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis; VLAHAVAS, Ioannis. Mining multi-label data. **Data mining and knowledge discovery handbook**, Springer, p. 667–685, 2010.

TSOUMAKAS, Grigorios; VLAHAVAS, Ioannis. Random k-labelsets: An ensemble method for multilabel classification. **Machine learning: ECML 2007**, Springer, p. 406–417, 2007.

VENS, Celine *et al.* Decision trees for hierarchical multi-label classification. **Machine Learning**, Springer, v. 73, n. 2, p. 185–214, 2008.

WEI, Leyi *et al.* A novel hierarchical selective ensemble classifier with bioinformatics application. **Artificial Intelligence in Medicine**, Elsevier, 2017.

WITTEN, Ian H *et al.* **Data Mining: Practical machine learning tools and techniques**. [S.l.]: Morgan Kaufmann, 2016.

XU, Hua; YANG, Weiwei; WANG, Jiushuo. Hierarchical emotion classification and emotion component analysis on chinese micro-blog posts. **Expert systems with applications**, Elsevier, v. 42, n. 22, p. 8745–8752, 2015.

YANG, Yiming. An evaluation of statistical approaches to text categorization. **Information retrieval**, Springer, v. 1, n. 1, p. 69–90, 1999.

YANG, Yiming; PEDERSEN, Jan O. A comparative study on feature selection in text categorization. In: **Icml**. [S.l.: s.n.], 1997. v. 97, p. 412–420.

YOUNES, Zoulficar *et al.* A dependent multilabel classification method derived from the k-nearest neighbor rule. **EURASIP Journal on Advances in Signal Processing**, Springer Science & Business Media, 2011.

ZHANG, Min-Ling; ZHOU, Zhi-Hua. Ml-knn: A lazy learning approach to multi-label learning. **Pattern recognition**, Elsevier, v. 40, n. 7, p. 2038–2048, 2007.

ZHOU, Denny; XIAO, Lin; WU, Mingrui. Hierarchical classification via orthogonal transfer. 2011.

ZHU, Xiaolin; LIU, Desheng. Accurate mapping of forest types using dense seasonal landsat time-series. **ISPRS Journal of Photogrammetry and Remote Sensing**, Elsevier, v. 96, p. 1–11, 2014.