

Zacarias Curi Filho

**UM MÉTODO PARA CLASSIFICAÇÃO
MULTIRRÓTULO DE OPINIÕES EM TEXTOS DE
NOTÍCIAS UTILIZANDO REDES LONG-SHORT
TERM MEMORY**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática.

CURITIBA

2018

Zacarias Curi Filho

**UM MÉTODO PARA CLASSIFICAÇÃO
MULTIRRÓTULO DE OPINIÕES EM TEXTOS DE
NOTÍCIAS UTILIZANDO REDES LONG-SHORT
TERM MEMORY**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática.

Área de Concentração: Descoberta de Conhecimento e Aprendizagem de Máquina

Orientador: Prof. Dr. Emerson Cabrera Paraiso

Coorientador: Prof. Dr. Alceu de Souza Britto Junior

CURITIBA

2018

Dados da Catalogação na Publicação
Pontifícia Universidade Católica do Paraná
Sistema Integrado de Bibliotecas – SIBI/PUCPR
Biblioteca Central
Luci Eduarda Wielganczuk – CRB 9/1118

C975m Curi Filho, Zacarias
2018 Um método para classificação multirrótulo de opiniões em textos de notícias utilizando redes *long-short term memory* / Zacarias Curi Filho ; orientador: Emerson Cabrera Paraiso ; coorientador: Alceu de Souza Britto Junior. – 2018. 106 f. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2018
Bibliografia: f. 90-106

1. Mineração de dados (Computação). 2. Redes neurais (Computação).
3. Algoritmos. I. Paraiso, Emerson Cabrera. II. Britto Junior, Alceu de Souza.
III. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática. IV. Título.

CDD 22. ed. – 004.312

ATA DE SESSÃO PÚBLICA

DEFESA DE DISSERTAÇÃO DE MESTRADO Nº 03/2018

**PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA – PPGIa
PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ - PUCPR**

Em sessão pública realizada às 09h00 de **02 de Março de 2018**, no Auditório Guglielmo Marconi – Bloco 8, ocorreu a defesa da dissertação de mestrado intitulada “**Um Método para Classificação Multirrotulo de Opiniões em Textos de Notícias Utilizando Redes Long-Short Term Memory**” apresentada pelo aluno **Zacarias Curi Filho**, como requisito parcial para a obtenção do título de **Mestre em Informática**, na área de concentração **Ciência da Computação**, perante a banca examinadora composta pelos seguintes membros:

Prof. Dr. Emerson Cabrera Paraiso (Orientador)- PUCPR

Prof. Dr. Alceu de Souza Britto Jr (co-orientador) – PUCPR

Prof. Dr. Julio Cesar Nievola – PUCPR

Prof.^a Dr.^a Deborah Ribeiro Carvalho – PUCPR/PPGTS

Após a apresentação da dissertação pelo aluno e correspondente arguição, a banca examinadora emitiu o seguinte parecer sobre a tese:

Membro	Parecer
Prof. Dr. Emerson Cabrera Paraiso	<input checked="" type="checkbox"/> Aprovado () Reprovado
Prof. Dr. Alceu de Souza Britto Jr	<input checked="" type="checkbox"/> Aprovado () Reprovado
Prof. Dr. Julio Cesar Nievola	<input checked="" type="checkbox"/> Aprovado () Reprovado
Prof. ^a Dr. ^a Deborah Ribeiro Carvalho	<input checked="" type="checkbox"/> Aprovado () Reprovado

Portanto, conforme as normas regimentais do PPGIa e da PUCPR, a tese foi considerada:

APROVADO

(aprovação condicionada ao atendimento integral das correções e melhorias recomendadas pela banca examinadora, conforme anexo, dentro do prazo regimental)

() **REPROVADO**

E, para constar, lavrou-se a presente ata que vai assinada por todos os membros da banca examinadora. Curitiba, 02 de Março de 2018.



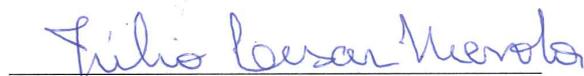
Prof. Dr. Emerson Cabrera Paraiso



Prof. Dr. Alceu de Souza Britto Junior



Prof.ª Dr.ª Deborah Ribeiro Carvalho



Prof. Dr. Julio Cesar Nievola

Resumo

Os sites da web 2.0 possibilitam que diversos usuários expressem suas emoções e reações sobre diversos temas. Um efeito da popularização desses sites é a grande quantidade de textos com informações relevantes disponibilizadas pelos usuários. A análise dessas informações pode ser realizada com técnicas de Mineração de Opiniões. Entre essas técnicas, o algoritmo Long-Short Term Memory se destaca por possuir características interessantes para a classificação de textos. Embora existam diferentes métodos e algoritmos para a tarefa de Mineração de Opiniões, poucos são capazes de realizar a classificação multirrótulo, onde várias opiniões estão presentes simultaneamente para um mesmo texto. Entre os métodos que permitem esse tipo de classificação, um grupo de técnicas baseadas em múltiplos classificadores (ensemble) se sobressai pelos resultados obtidos em diversos tipos de problemas sem a necessidade de se criar um grande número de classificadores, como em outros métodos. Visando aplicar o algoritmo Long-Short Term Memory para a tarefa de Mineração de Opiniões multirrótulo, este trabalho apresenta um novo método de transformação de problema baseado em ensemble de classes e segmentação de documentos. O método apresentado necessita de poucos classificadores, como os métodos de ensemble tradicionais, sem a necessidade de criação de novas classes. Neste trabalho, também são apresentados dois novos corpora de notícias rotulados com reações de usuários. O novo método é comparado com diversos métodos tradicionais de transformação de problema combinados com o algoritmo SVM e com o algoritmo ML-KNN. Os resultados obtidos com o novo método combinado com o algoritmo Long-Short Term Memory apresentaram um micro F1 superior a todos os outros métodos testados. Além de permitir a realização da tarefa com menos operações, o método proposto combinado com o algoritmo Long-Short Term Memory gerou um micro F1 e um jaccard index superior ao método Binary Relevance com o mesmo algoritmo. O uso do SVM com o método proposto gerou um micro F1 sem diferença estatística com o método tradicional com melhor resultado.

Palavras-chaves: Mineração de Opiniões. Classificação Multirrótulo. Deep Learning. LSTM

Abstract

The growth of the web has made common sites that allow users to share reviews and ratings on different topics. A consequence of the popularity of these sites is a large number of texts with relevant information made available by the users. An analysis of this information can be done using Opinion Mining techniques. Among these techniques, an algorithm called Long-Short Term Memory stands out because it has interesting features for Text Mining. Although there are different methods and algorithms for Opinion Mining, only a few of them are able to perform the multi-label classification, in which a set of opinions are present simultaneously for the same text. Among the methods that allow this type of classification, a group of techniques based on multiple classifiers (ensemble) stands out for the results obtained in several types of problems without the need to create a large number of classifiers, as it is in other methods. Aiming to apply the Long-Short Term Memory algorithm to the multi-label Opinion Mining task, this work presents a new problem transformation method based on class ensemble and document segmentation. The presented method needs only a few classifiers without the need to create new classes. In this research, we also present two new news corpora labeled with user reactions. The new method is compared with several problem transformation methods combined with the SVM algorithm and the ML-KNN algorithm. The results obtained with the new method combined with the Long-Short Term Memory presented to the micro F1 metric was superior to all the other methods tested. In addition to allowing the task to be performed with fewer operations, the proposed method combined with the Long-Short Term Memory algorithm generated superior results for the micro F1 and jaccard index metrics if compared to the Binary Relevance with the same algorithm. The use of SVM with the proposed method generated a micro F1 with no statistical difference to the traditional method that obtained the best result.

Keywords: Opinion Mining. Multi-Label Classification. Deep Learning. LSTM

Lista de ilustrações

Figura 1 – Exemplo de <i>review</i> (Extraído de: amazon.com.br).	14
Figura 2 – Exemplo de extração de entidades em textos (Extraído de: maps.google.com).	24
Figura 3 – Abordagens para a Mineração de Opiniões (Adaptado de: (MEDHAT; HASSAN; KORASHY, 2014)).	25
Figura 4 – Etapas para a utilização da abordagem baseada em aprendizagem de máquina (Adaptado de: (ARANHA, 2007)).	26
Figura 5 – Modelo de neurônio apresentado de McCulloch e Pitts (Extraído de: (JAIN; MAO; MOHIUDDIN, 1996)).	29
Figura 6 – Taxonomia das redes neurais (Adaptado de: (JAIN; MAO; MOHIUDDIN, 1996)).	30
Figura 7 – Funcionamento de uma Convolutional Neural Network	31
Figura 8 – Funcionamento de uma Recurrent Neural Network.	32
Figura 9 – Estrutura de uma Long Short-Term Memory	32
Figura 10 – Estrutura de um denoising autoencoder	34
Figura 11 – Estrutura de um Boltzmann Machines	35
Figura 12 – Estrutura de um Deep Boltzmann Machines	35
Figura 13 – Estrutura de um Deep Belief Networks	36
Figura 14 – Categorização dos algoritmos multirrótulo (Adaptado de: (ZHANG; ZHOU, 2014)).	37
Figura 15 – Modelo hierárquico do corpus desenvolvido por Quan e Ren (Extraído de: (QUAN; REN, 2010))	50
Figura 16 – Método utilizando divisão de imagens (Extraído de: (WEI et al., 2016))	55
Figura 17 – Estrutura da pesquisa	57
Figura 18 – Visão geral do método proposto	64
Figura 19 – Solução 2 para transformação do problema multirrótulo em simples rótulo	67
Figura 20 – Solução 3 para transformação do problema multirrótulo em simples rótulo	67
Figura 21 – Solução 4 para transformação do problema multirrótulo em simples rótulo	68
Figura 22 – Rótulos para as notícias em português do site BuzzFeed (Extraído de: BuzzFeed.com)	70
Figura 23 – Distribuição dos rótulos para a base em português	71
Figura 24 – Quantidade de instâncias por número de rótulos para o corpus em português	72

Figura 25 –Distribuição dos rótulos para o corpus em inglês	72
Figura 26 –Quantidade de instâncias por número de rótulos para o corpus em português	73

Lista de tabelas

Tabela 1	– Exemplo de uma base de dados multirrótulo	38
Tabela 2	– Dados da Tabela 1 transformados com o algoritmo Binary Relevance	38
Tabela 3	– Comparação entre as instâncias de entrada \mathbf{x} para os algoritmos Binary Relevance e Classifier Chains	39
Tabela 4	– Etapas do algoritmo Calibrated Label Ranking	41
Tabela 5	– Dados da Tabela 1 transformados com o algoritmo Label Powerset	41
Tabela 6	– Exemplo de classificação usando o algoritmo $RAkEL$ com $k = 3$ e $m = 4$	42
Tabela 7	– Abordagens e algoritmos de classificação utilizados nos trabalhos encontrados na revisão da literatura	52
Tabela 8	– Corpora e métricas de avaliação utilizados nos trabalhos encontrados na revisão da literatura	53
Tabela 9	– Características dos corpora utilizados nos artigos encontrados na revisão da literatura	54
Tabela 10	– Correlação de Pearson do corpus em português	72
Tabela 11	– Correlação de Pearson do corpus em inglês	73
Tabela 12	– Parâmetros do algoritmo LSTM para o corpus em português	75
Tabela 13	– Resultados obtidos para o corpus em português	75
Tabela 14	– Resultados obtidos com o método HOMER com o algoritmo SVM para o corpus em português	76

Lista de abreviaturas e siglas

AA	Adaptação de Algoritmo
ACC	Acurácia
ANN	Rede Neural Artificial
AP	Precisão Média
ASM	Average Simple Matching Coefficient
BM	Boltzmann Machines
BR	Binary Relevance
CA	Computação Afetiva
CC	Classifier Chains
CLR	Calibrated Label Ranking
CNN	Convolutional Neural Network
COV	Cobertura
CRF	Conditional Random Fields
DBM	Deep Boltzmann Machines
DBN	Deep Belief Networks
ECC	Ensembles of Classifier Chains
EM	Exact Match
HL	Hamming Loss
HOMER	Hierarchy Of Multilabel classifiERs
HS	Hamming Score
IDE	Integrated Development Environment
IHC	Interação Humano-Computador
KNN	K-Nearest Neighbor

LDA	Latent Dirichlet Allocation
LGR	LoGistic Regression
LP	Label Powerset
LSTM	Long Short-Term Memory
ME	Maximum Entropy
ML-KNN	Multi-Label K-Nearest Neighbor
NB	Naive Bayes
OE	One-Error
PC	Precisão
PCT	Predictive Clustering Trees
PLN	Pocessamento de Linguagem Natural
RA k EL	Random k-Labelsets
RL	Recall
RNN	Recurrent Neural Networks
RNTN	Recursive Neural Tensor Network
SAC	Subset Accuracy
SDA	Stacked Denoising Autoencoders
SVM	Support Vector Machines
TF-IDF	Term Frequency – Inverse Document Frequency
TP	Transformação de Problema

Lista de símbolos

X	conjunto de instâncias x do problema, onde $x \in X$
L	conjunto com todas as classes λ existentes no problema
Y	conjunto de conjuntos de rótulos y de cada instância x do problema, onde $Y \subseteq L$
h	modelo de classificação, onde $h(x)$ retorna as predições do classificador para x
P	conjunto de rótulos relevantes, onde $P \subseteq L$
N	conjunto de rótulos irrelevantes, onde $N \subseteq L$
m	grupo de classes para um método de transformação de problema com ensemble, onde $ m \leq k $
f	frases de uma instância x

Sumário

1	Introdução	14
1.1	Motivação	17
1.2	Objetivos	19
1.3	Hipóteses de Trabalho	20
1.4	Contribuição Científica	20
1.5	Escopo	20
1.6	Organização do Documento	20
2	Fundamentação Teórica	21
2.1	Computação Afetiva	21
2.1.1	Análise de Sentimentos e Mineração de Opiniões	22
2.1.2	Mineração de Textos na Computação Afetiva	23
2.2	Deep Learning	28
2.2.1	Redes Neurais	29
2.2.2	Deep Neural Network	30
2.3	Classificação Multirrótulo	36
2.3.1	Métodos de Transformação de Problema	37
2.3.2	Métodos de Adaptação do Algoritmo	42
2.3.3	Métricas de Avaliação	43
3	Estado da Arte	47
3.1	Classificação Multirrótulo para Mineração de Opiniões em Textos	48
3.2	Classificação Multirrótulo com Deep Learning	53
4	Procedimentos Metodológicos	57
4.1	Planejamento Inicial	57
4.2	Fase Exploratória	58
4.3	Coleta de Dados e Desenvolvimento	59
4.4	Avaliação	60
5	Método Proposto	63
5.1	Pressupostos do Método	65
5.2	Pré-processamento	65
5.3	Treinamento	66
5.4	Divisão dos Dados, Teste e Avaliação	68
5.5	Limitações do Método Proposto	69

6	Resultados	70
6.1	Construção de um corpus multirrótulo de notícias	70
6.2	Experimentos e Análise dos Resultados	73
6.2.1	Protocolo de Testes e Definição dos Parâmetros	74
6.2.2	Experimentos	75
6.2.3	Análise dos Resultados	76
7	Conclusões e Trabalhos Futuros	79
	Anexos	81
ANEXO A	Exemplos de textos do corpus em português extraído do site Buzz-Feed.com	82
ANEXO B	Exemplos de textos do corpus em inglês extraído do site Buzz-Feed.com	85
	Referências	90

1 Introdução

O avanço no estudo da interação humano-computador (IHC) tem permitido que sistemas computacionais possam ter uma comunicação mais natural e direta com o usuário. O objetivo da área de IHC é reproduzir a interação humano-humano, utilizando para isso recursos de diversas áreas, como artes, design, psicologia, sociologia, linguística e ciência da computação (DIX et al., 2005). Uma das áreas da IHC que visa melhorar a comunicação entre humanos e computadores é a Computação Afetiva (CA). No trabalho de (PICARD, 1997), a CA foi definida como a computação que se relaciona com, surge de, ou influencia as emoções. Essa tarefa pode ser útil em diversas aplicações, como na análise de opiniões de produtos e serviços, na identificação de sentimentos em textos e imagens, na reprodução de expressões corporais em avatares, entre outras (TAO; TAN, 2005).

Em paralelo aos avanços na área de IHC, tem sido registrado um aumento na quantidade de usuários utilizando a web, como apresentado em Soriano et al. (2012). Com esse aumento, torna-se cada vez mais comum o compartilhamento de opiniões, avaliações e emoções sobre diversos temas. Várias ferramentas foram criadas para gerenciar esse tipo de compartilhamento, como sites de *reviews*, microblogs e redes sociais (BERTHON et al., 2012). A Figura 1 apresenta dois exemplos de *reviews* sobre um dispositivo *e-reader* realizados no site Amazon¹.

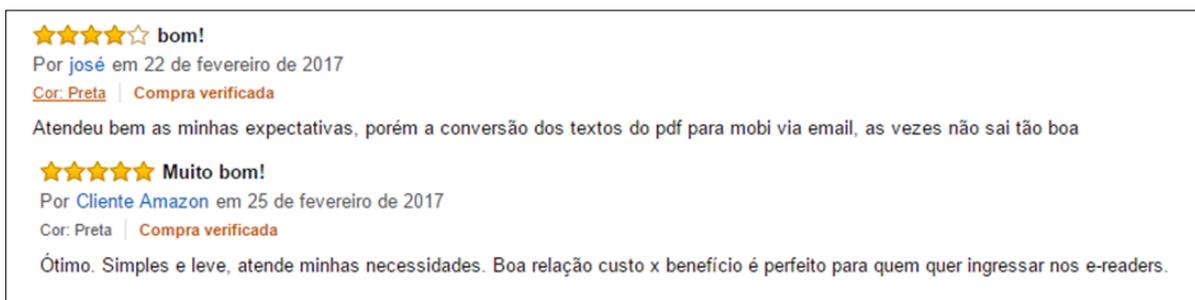


Figura 1 – Exemplo de *review* (Extraído de: amazon.com.br).

Como é possível observar na Figura 1, os *reviews* podem apresentar algumas informações destacando características do produto e as opiniões do usuário sobre essas características. Essas informações podem ser extraídas e utilizadas pelo fabricante para otimizar o produto ou pelo comerciante para destacar os principais benefícios do produto que está sendo vendido.

A união das informações apresentadas textualmente com o grande volume de dados gerados diariamente produz informações interessantes para empresas, prestadores de

¹ <https://www.amazon.com.br>

serviços e o governo (PAK; PAROUBEK, 2010). Embora exista um interesse pelas informações contidas nos textos, existem limitações causadas pelo grande volume de dados gerados diariamente pelos usuários (CAMBRIA et al., 2013). Uma das principais limitações existentes é a inviabilidade da análise manual de todos os textos. Para resolver esse problema, pesquisadores da área de Mineração de Opiniões, buscam desenvolver métodos para realizar a análise de forma automática.

A Mineração de Opiniões, em inglês *Opinion Mining*, é definida por Balazs e Velásquez (2016) como um sub-campo da área de Mineração de Textos, que tem como principal tarefa a extração de opiniões de conteúdo gerado por usuários da Web. No trabalho de Balazs e Velásquez (2016) e em outros como (DING; LIU; YU, 2008), (CAMBRIA et al., 2013), (RAVI; RAVI, 2015) o termo Análise de Sentimentos, em inglês *Sentiment Analysis*, é apresentado como sinônimo do termo Mineração de Opiniões. Em Liu (2012), os autores afirmam que alguns trabalhos utilizam outros termos para representar essa tarefa, como *opinion extraction*, *sentiment mining*, *subjectivity analysis*, *affect analysis*, *emotion analysis*, *review mining*, etc. Embora existam diversas terminologias para representar a mesma tarefa, alguns trabalhos como Munezero et al. (2014), apresentam algumas diferenças entre esses termos e indicam a utilização correta deles. Independente da nomenclatura utilizada entre as aplicações, não existem diferenças entre os métodos utilizados para a classificação, sendo eles os mesmos da área de Mineração de Textos.

Mesmo que os métodos não sofram influência do tipo de classe utilizada os termos Mineração de Opiniões e Análise de Sentimentos serão utilizados de forma distinta neste trabalho. Utilizando as definições apresentadas por Munezero et al. (2014), o termo opinião será utilizado como sinônimo do ponto de vista que uma pessoa está pensando, ou seja, a ideia ou o conhecimento que se tem sobre algo. Com base na definição elaborada por Gordon (1981 apud MUNEZERO et al., 2014), o termo sentimento será utilizado como padrões socialmente construídos de sensações, gestos expressivos e significados culturais organizados em torno de uma relação com um objeto social, geralmente outra pessoa, constituindo sua identidade. Em resumo, o termo sentimento será utilizado como algo duradouro e interno à pessoa enquanto o termo opinião será utilizado como um ponto de vista ou uma ideia expressada por uma pessoa e que pode ser facilmente alterada com estímulos externos.

Em adição aos termos opinião e sentimento, utilizaremos o termo reação como uma resposta de curto prazo manifestada sobre algo durante ou imediatamente após o contato com o mesmo. Um exemplo de reação é a expressão facial de uma pessoa imediatamente após assistir um vídeo, ou durante a sua exibição. Outro exemplo é a sensação de alegria durante e após a leitura de uma notícia feliz. Segundo Desmet (2003), as reações podem ser divididas em comportamentais (ex. se afastar de alguém), expressivas (ex. sorrir) e fisiológicas (ex. aumento do batimento cardíaco). Como visto na definição apresentada

em Weiszflog (2017), a reação está ligada a alguma ação exterior e tem como base as experiências anteriores da pessoa. Esse termo é relacionado com a atitude ou sensação adquirida por uma pessoa ao receber um estímulo de uma fonte externa. Embora não tenha o mesmo significado de opinião, para a análise de textos a opinião e a reação representam o que a pessoa pensa sobre determinado tópico ou assunto com base nas experiências anteriores.

Como visto em Medhat, Hassan e Korashy (2014), existem três abordagens para realizar a tarefa de Mineração de Opiniões: as abordagens baseadas em léxicos, as baseadas em aprendizagem de máquina e as abordagens híbridas. A abordagem baseada em léxicos, também chamada de abordagem baseada em dicionário, utiliza dicionários com palavras referentes às classes existentes para realizar a classificação. A utilização dessa abordagem é limitada às situações onde já existem os recursos necessários ou se deseja construir novos recursos para essa tarefa. Um exemplo de relato de construção desse tipo de recurso pode ser visto em Baccianella, Esuli e Sebastiani (2010). Por necessitarem de recursos léxicos, os métodos híbridos apresentam os mesmos problemas da abordagem baseada em léxicos. Os métodos baseados em aprendizagem de máquina visam utilizar uma técnica de aprendizado de máquina para descobrir regras de maneira automática. Essa abordagem possui algumas vantagens em relação a abordagem baseada em léxicos, principalmente por não necessitarem de recursos específicos para cada tipo de corpus. Devido a flexibilidade concedida pelos métodos de aprendizado de máquina essa abordagem foi escolhida para este trabalho.

Como apresentado em Tang, Qin e Liu (2015), dentre as técnicas de aprendizado de máquina um grupo de algoritmos chamado de Deep Learning, ou Aprendizagem Profunda em Português, vem conquistando resultados cada vez melhores nessa área. Segundo o trabalho de LeCun, Bengio e Hinton (2015), o Deep Learning consiste em métodos que permitem modelos computacionais compostos de várias camadas de processamento para aprender representações de dados com múltiplos níveis de abstração. Como visto em Diehl et al. (2016) e Chen e Lin (2014) esses métodos se apresentam mais eficientes que os métodos rasos (termo originado do inglês *shallow*, apresentado em vários trabalhos, como em Cho e Saul (2009) e Collobert e Weston (2008)) em diversos casos onde existe uma grande quantidade de dados. Em troca de resultados mais satisfatórios, esses métodos necessitam de um maior poder computacional, se comparados com métodos rasos tradicionais.

Dentre os algoritmos de Deep Learning existentes na literatura, o Recurrent Neural Network (RNN) se apresenta ideal para as operações com textos (LECUN; BENGIO; HINTON, 2015). Um exemplo disso pode ser visto em Sutskever, Martens e Hinton (2011), onde esse algoritmo alcançou bons resultados na tarefa de geração automática de textos. Em Mikolov et al. (2013b), os autores também obtiveram bons resultados para gerar sequências de palavras. Um tipo especial de RNN é o long short-term memory (LSTM),

apresentado inicialmente em Hochreiter e Schmidhuber (1997). Essa arquitetura adiciona uma célula de memória ao RNN tradicional, permitindo armazenar informações por um longo prazo. Como visto em Graves (2012), assim como o RNN, o LSTM é capaz de utilizar informações contextuais para realizar um mapeamento entre sequências de entrada e saída. Esse mapeamento, em conjunto com a capacidade de armazenar as informações por um longo prazo tornam esse algoritmo ideal para problemas que exigem o uso de informações contextuais distantes, como os problemas com textos extensos. O LSTM apresenta bons resultados em diversas aplicações, como em Weston, Chopra e Bordes (2014), onde os autores utilizaram esse algoritmo para responder questões. Outra possível aplicação é a apresentada no trabalho de Graves e Schmidhuber (2005), onde o LSTM foi utilizado para a classificação de fonemas. Além dos fonemas, diversos trabalhos da literatura apresentam a utilização desse algoritmo para a tarefa de Mineração de Opiniões.

Um exemplo do uso de Deep Learning para a Mineração de Opiniões é o trabalho de Socher et al. (2013), onde os autores introduziram uma arquitetura chamada Recursive Neural Tensor Network para realizar a classificação de polaridade em *reviews* de filmes. Nesse tipo de classificação cada *review* é classificado como positivo ou negativo com base nas opiniões expressas no texto. A arquitetura proposta por Socher e colegas possibilitou uma acurácia de 85,4%, superando algoritmos como NB, com 81,8% e SVM, com 79,4%. Em Munkhdalai e Yu (2016) os autores utilizaram um método com o algoritmo LSTM para a mesma base e obtiveram uma acurácia de 89,3%. Em Santos e Gatti (2014), uma Convolutional Neural Network (CNN) foi utilizada para realizar a classificação de *reviews* de filmes e tweets. Os autores alcançaram o estado da arte nas duas bases utilizadas. Além do idioma inglês utilizado nos trabalhos citados anteriormente, alguns autores utilizaram outros idiomas. Um exemplo pode ser visto em Brum, Araujo e Kepler (2016), onde os autores utilizaram um corpus com *reviews* de livros do Brasil, escritos em Português para realizar a classificação com o algoritmo Recursive Neural Tensor Network.

1.1 Motivação

Após a realização de uma revisão com os trabalhos que utilizam Deep Learning para a tarefa de Mineração de Opiniões foi possível observar que diferentes tipos de classificação podem ser utilizados para essa tarefa. A maioria dos trabalhos realiza uma classificação binária, na qual considera apenas se o texto é positivo ou negativo, como em Socher et al. (2012). A segunda abordagem mais utilizada é a classificação de forma multi classe, na qual existem mais de duas opções de classes para cada instância, mas apenas uma é correta. Um exemplo desse tipo de classificação é o caso de *reviews* que são classificados com base em estrelas, onde para cada texto existe um valor relacionado a ele, como realizado no trabalho de Munkhdalai e Yu (2016). Essas abordagens realizam a classificação de forma simples rótulo, onde existe apenas uma classe correta para cada

exemplo.

Em contrapartida aos tipos de classificação utilizados atualmente, é possível realizar a classificação de forma multirrótulo. Nesse tipo de classificação um grupo de classes corretas pode existir para cada instância (TSOUMAKAS; KATAKIS, 2006). Um exemplo da classificação multirrótulo é o caso em que a reação dos leitores é utilizada para classificar uma notícia. Como cada pessoa expressa um comportamento diferente ao ler a notícia múltiplos rótulos podem ser selecionados simultaneamente para o mesmo texto. Assim como no exemplo das notícias, a classificação multirrótulo se apresenta necessária na maioria dos casos onde é analisado um texto sobre a perspectiva do leitor, situação na qual é observado o comportamento de diversas pessoas ao ler o mesmo texto. O trabalho de Bhowmick, Basu e Mitra (2009) é um exemplo de como notícias podem ser classificadas sob essa perspectiva.

Como apresentado em Zhang e Zhou (2014), existem dois métodos para resolver problemas multirrótulo: os métodos de adaptação de algoritmos e os métodos de transformação de problema. Os métodos de adaptação de algoritmos consideram a realização de transformações nos métodos tradicionais de classificação com simples rótulo para possibilitar a utilização com problemas multirrótulo. Os métodos de transformação de problema consideram a realização de transformações no problema para transformar o mesmo em um problema simples rótulo.

Embora a classificação multirrótulo apresente uma grande gama de aplicações para a tarefa de Mineração de Opiniões, poucos trabalhos foram encontrados na literatura utilizando esse tipo de classificação com os métodos tradicionais. Já com a utilização de algoritmos de Deep Learning, até o presente momento, apenas dois trabalhos foram encontrados na literatura. Simultaneamente a pequena quantidade de trabalhos explorando esse tema, não foram encontradas bases grandes o suficiente para possibilitar um bom desempenho desses algoritmos.

Uma das principais limitações para a utilização da maioria dos métodos de transformação de problema com Deep Learning é a necessidade de se utilizar vários classificadores simultaneamente (BI; KWOK, 2013). Como os algoritmos de Deep Learning necessitam de um alto poder computacional, a utilização dessas transformações se torna inviável em muitos casos. Alguns métodos de transformação de problema possibilitam reduzir o número de classificadores com a realização de um ensemble das classes. Outra forma de resolver esse problema é realizando uma alteração nos dados de entrada, possibilitando que uma base multirrótulo seja classificada com apenas um classificador simples rótulo. Essa alteração foi apresentada em alguns trabalhos utilizando imagens, como (WEI et al., 2016) e (KINGHORN; ZHANG; SHAO, 2018).

Embora não tenham sido encontrados trabalhos dividindo textos para a classificação, o trabalho de Liu e Chen (2015) apresenta uma técnica para classificação multirrótulo

de opiniões em textos utilizando a segmentação do texto para possibilitar a extração de características. No trabalho foi utilizado um sistema de classificação baseado em três componentes principais, segmentação do texto, extração das características e classificação multirrótulo. O processo de segmentação apresentado por Liu e Chen é utilizado para dividir o texto em palavras ou sentenças com base em sua frequência e, após a extração das características, realizar a classificação com um método de classificação multirrótulo tradicional. Uma demonstração da possibilidade de se dividir hierarquicamente um texto também pode ser observada no corpus desenvolvido por Quan e Ren (2010). Nesse trabalho os autores apresentam um corpus composto por textos de blogs em chinês rotulado a nível de documento, parágrafo e sentença. O trabalho de Wei et al. (2014) apresenta um método para classificação multirrótulo de imagens utilizando Deep Learning. No método os autores apresentam um método de transformação de problema, onde realiza a segmentação da imagem original em pequenas imagens com apenas um objeto e as classifica individualmente, utilizando para isso o algoritmo CNN.

Visando verificar a eficiência das técnicas de Deep Learning para a classificação multirrótulo de textos e a possibilidade de se dividir textos para a classificação, o presente trabalho apresenta um método de transformação de problema com ensemble para a classificação multirrótulo de textos. Além da apresentação do novo método, neste trabalho são apresentados testes com o algoritmo LSTM e dois novos corpora multirrótulo de notícias de entretenimento.

1.2 Objetivos

Esta pesquisa tem como objetivo desenvolver um método de transformação de problema voltado ao uso de Deep Learning para a classificação multirrótulo de textos.

Objetivos específicos:

- desenvolver um método de transformação de problema para classificação multirrótulo de textos;
- determinar os principais métodos de transformação de problema para classificação multirrótulo existentes na literatura para comparações com o método proposto;
- construir um corpus de textos multirrótulo anotado com reações para o treinamento e avaliação do método proposto;
- desenvolver um protótipo computacional para a realização de testes e avaliação do método proposto.

1.3 Hipóteses de Trabalho

As hipóteses desta pesquisa são:

- H1) É possível realizar a classificação multirrótulo de textos utilizando Deep Learning.
- H2) O uso de Deep Learning para a classificação multirrótulo de notícias com a utilização de métodos de transformação de problema possibilita um resultado superior ao obtido com a utilização de algoritmos tradicionais.
- H3) O método de transformação de problema apresentado neste trabalho possibilita um resultado superior aos métodos existentes na literatura para a classificação multirrótulo de notícias.

1.4 Contribuição Científica

A principal contribuição deste trabalho é a proposta de um método de transformação de problema para a classificação multirrótulo de textos utilizando Deep Learning. Outra contribuição é a disponibilização de um novo corpus multirrótulo anotado com reações com tamanho adequado para o correto funcionamento dos algoritmos de Deep Learning.

1.5 Escopo

O escopo desta pesquisa limita-se a classificação multirrótulo de informações textuais, escritas em português do Brasil e anotadas com reações, utilizando algoritmos rasos e de Deep Learning em conjunto com métodos de transformação de problema.

1.6 Organização do Documento

O presente documento está organizado em 7 capítulos. No Capítulo 2 é apresentada uma revisão sobre as áreas de Computação Afetiva, Deep Learning e classificação multirrótulo. O Capítulo 3 apresenta o estado da arte do tema proposto. Os procedimentos metodológicos utilizados nesta pesquisa são apresentados no Capítulo 4. Um novo método para classificação multirrótulo de textos voltado para a Mineração de Opiniões em notícias é apresentado no Capítulo 5. O Capítulo 6 apresenta os resultados obtidos. Por fim, o Capítulo 7 conclui o trabalho e apresenta os trabalhos futuros.

2 Fundamentação Teórica

Neste capítulo são apresentadas as principais definições e conceitos necessários para o entendimento do método proposto e dos resultados obtidos. As seções a seguir apresentam informações sobre Computação Afetiva (CA), Deep Learning e Classificação Multirrótulo.

2.1 Computação Afetiva

A área de IHC tem como objetivo estudar e aperfeiçoar a interação entre pessoas e computadores. Pesquisadores dessa área procuram reproduzir a comunicação humano-humano, desenvolvendo para isso novas tecnologias com o objetivo de otimizar a comunicação humano-computador. A IHC utiliza recursos de diversas áreas, como artes, design, psicologia, sociologia, linguística e ciência da computação (DIX et al., 2005). No trabalho de Dix et al. (2005) os autores apresentam um estudo sobre essa área e definem que a principal fonte de entrada dos dados em humanos ocorre por meio dos sentidos e a principal fonte de saída por meio dos controles motores. Embora diversos estudos apresentem novos meios de saída de dados, a maioria dos usuários utiliza o teclado como principal fonte de comunicação. Esse fato faz com que uma grande quantidade de dados em forma de textos seja gerada. Visando utilizar esses dados para otimizar a comunicação humano-computador, pesquisadores realizam estudos na área de CA.

A CA foi definida em Picard (1997) como a computação que se relaciona com, surge de, ou influencia as emoções. Segundo Picard (1999) a CA expande a IHC, incluindo a comunicação emocional e os meios adequados de manipulação de informações afetivas. Em conjunto com a área de Processamento de Linguagem Natural (PLN) e Mineração de Dados é possível realizar a análise de opiniões, sentimentos e emoções em informações textuais. Como apresentado em Cambria et al. (2013), a análise dessas informações possui aplicações que vão além da IHC, podendo ser utilizadas em diversas finalidades nos meios comerciais, no ensino e nas tarefas multimídia.

Um exemplo dessas aplicações é a apresentada no trabalho de Rosa, Rodriguez e Bressan (2015), no qual os autores propuseram um sistema de recomendação de músicas para o português com base na intensidade do sentimento do usuário. Outro exemplo de sistema de recomendação utilizado Análise de Sentimentos é o apresentado por Yang et al. (2013). Nesse trabalho, os autores apresentaram um sistema de recomendações de locais com base em experiências anteriores e comentários do usuário. Algumas abordagens, como a apresentada em Poria et al. (2016) apresentam formas de realizar esse tipo de análise utilizando diversas fontes de entrada ao mesmo tempo. No trabalho realizado por Poria

e colegas, foi utilizada uma fusão de áudio, imagem e texto para realizar a Análise de Sentimentos em vídeos extraídos da Web. Para provar o conceito apresentado no artigo, os autores desenvolveram um avatar atualizado em tempo real com base nos sentimentos classificados durante a execução de um vídeo.

Devido a diversidade de aplicações dessa área, alguns autores utilizam termos diferentes para representar a mesma tarefa. As subseções a seguir apresentam informações sobre a terminologia utilizada nessa área e algumas características da Mineração de Textos na CA.

2.1.1 Análise de Sentimentos e Mineração de Opiniões

A Web 2.0 representa a geração de softwares que consideram a rede como uma plataforma e não mais como um conjunto de sites estáticos. Essa geração possibilitou que usuários e desenvolvedores enxergassem a Web de outra maneira, permitindo novas formas de interação e participação (O'REILLY, 2007). Segundo Zabin e Jefferies (2008), entre essas plataformas existem diversos sites nos quais o usuário pode contribuir diretamente com a manipulação do conteúdo, como sites de *reviews*, fóruns, *blogs* e redes sociais. Uma característica em comum entre esses sites é a capacidade do usuário expressar textualmente suas próprias opiniões e experiências sobre diversos temas. Devido ao crescente uso dessas ferramentas a análise das informações publicadas pelos usuários se tornou de grande interesse para diversas companhias (PAK; PAROUBEK, 2010). Visando executar a análise dessas informações de forma automática pesquisadores realizaram a união das áreas de PLN com a CA, desenvolvendo assim trabalhos na área de Mineração de Opiniões.

O termo Mineração de Opiniões foi apresentado inicialmente por Dave, Lawrence e Pennock (2003) e em muitos trabalhos possui o mesmo significado que o termo Análise de Sentimentos, introduzido por Nasukawa e Yi (2003). Com o passar do tempo, diversos outros termos foram criados para representar esse tipo de tarefa, como *opinion extraction*, *sentiment mining*, *subjectivity analysis*, *affect analysis*, *emotion analysis*, *review mining*, etc (LIU, 2012).

Segundo Liu (2012) essa área abrange todo estudo envolvendo uma análise computacional de opiniões, sentimentos, avaliações, atitudes e emoções relacionados a entidades. As entidades utilizadas para a análise podem ser produtos, serviços, organizações, indivíduos, questões, eventos, tópicos, etc. Como apresentado por Pang, Lee et al. (2008), existem diversas aplicações para essa área, como a análise de *reviews*, análise de notícias, utilização em sistemas de recomendação, utilização em sistemas de *business intelligence*, entre outros.

Devido à grande quantidade de objetos de estudo e aplicações a utilização dos

termos Mineração de Opiniões e Análise de Sentimentos acaba se tornando inadequada em alguns casos. No trabalho de Munezero et al. (2014), os autores apresentaram definições específicas para cada um dos tipos de análises realizadas, possibilitando uma melhor diferenciação entre os trabalhos da área.

Os termos estudados por Munezero et al. (2014) foram o afeto, a emoção, o sentimento e a opinião. Segundo os autores, o termo afeto é raramente utilizado na linguagem, isso ocorre pois esse termo se refere a algo anterior ou externo a consciência. O termo emoção é apresentado por Brave e Nass (2003) como uma reação a algum estímulo que ocorre com base nas memórias do indivíduo. Alguns trabalhos estudaram as emoções e criaram modelos de representação, como em Ekman e Oster (1979) e Plutchik (1984). Como visto em Brave e Nass (2003) o termo sentimento pode gerar confusões com o termo emoção, o sentimento não apresenta apenas um estado, como a emoção, mas um período longo podendo ter influência de diversos fatores simultaneamente. Alguns exemplos de sentimento são amor, ódio, alegria, tristeza e medo. O termo opinião pode ser definido simplesmente como o que uma pessoa pensa sobre algo. Muitas vezes uma opinião pode conter emoções e sofrer influência dos sentimentos de uma pessoa sobre o tema.

2.1.2 Mineração de Textos na Computação Afetiva

A Mineração de Textos na CA possui alguns desafios que tornam essa tarefa complexa. Uma demonstração do nível de dificuldade existente pode ser visto no processo de anotação do corpus, onde humanos geralmente apresentam divergências ao realizar uma anotação em pares. Essa divergência pode ser vista no trabalho de Bermingham e Smeaton (2010), onde os autores registraram uma média de 0,65 para o índice Kappa na classificação de textos de microblogs com sete classes. Esse resultado demonstra como a identificação automática de sentimentos em textos pode possuir resultados limitados.

Uma das principais causas da dificuldade no processo de identificação é a falta de informações explícitas sobre as opiniões no texto. Em vários casos são apresentadas apenas informações subjetivas, necessitando de uma análise complexa para realizar a tarefa de classificação (HU; LIU, 2004). Diversas vezes essas informações ainda sofrem influência da ironia ou sarcasmo, alterando o significado do sentimento expresso e podendo gerar erros na classificação. Alguns trabalhos estudam a identificação desses casos (CARVALHO et al., 2009) e a influência desses casos na tarefa de classificação (MAYNARD; GREENWOOD, 2014). Outro problema existente é a falta de estrutura dos textos. Muitos dos corpus encontrados na literatura, principalmente os extraídos de redes sociais, apresentam variações em relação ao tamanho e formas de escrita de cada texto. Em conjunto com os problemas citados anteriormente existem os casos onde são utilizados gírias e palavras incorretas, impedindo que todas as palavras sejam identificadas para a análise caso um tratamento não seja realizado, como em Balahur e Turchi (2013). Esses problemas podem

ser ampliados em alguns casos onde o idioma nativo do usuário é diferente do idioma utilizado para escrever o texto, como apresentado por Yim e Warschauer (2017).

Além dos problemas citados anteriormente, muitos métodos existentes na literatura não podem ser utilizados em todos os níveis de granularidade. No trabalho de Feldman (2013) são apresentados três níveis de granularidade: nível de documento, nível de sentença e nível de aspecto. A análise a nível de documento é a mais comum na literatura. Esse nível considera o texto como um todo, apresentando uma opinião sobre uma entidade central (LIU, 2010). Outra forma de realizar a análise é considerando o nível de sentença, onde são utilizados textos divididos em frases ou sentenças que possam conter uma opinião individualmente. Um exemplo de utilização é a base apresentada por Socher et al. (2013), onde os autores realizaram a rotulação das sentenças da base rotulada a nível de documento apresentada por Pang e Lee (2005). Uma das formas de resolver o problema onde o documento possui várias entidades é realizando a análise a nível de aspecto, também chamado de nível de entidade, como apresentado em Schouten e Frasincar (2016). Enquanto os modelos de documento e sentença consideram apenas uma entidade, o nível de análise baseado em aspecto considera que existem diversas entidades no texto, podendo conter múltiplos sentimentos associados a ela. Um exemplo de identificação de termos para a análise nesse nível de granularidade pode ser visto na Figura 2, que apresenta um resumo das principais *reviews* sobre um local existente no site Google Maps¹. Na figura as palavras em negrito foram identificadas automaticamente pelo sistema e representam as entidades encontradas.

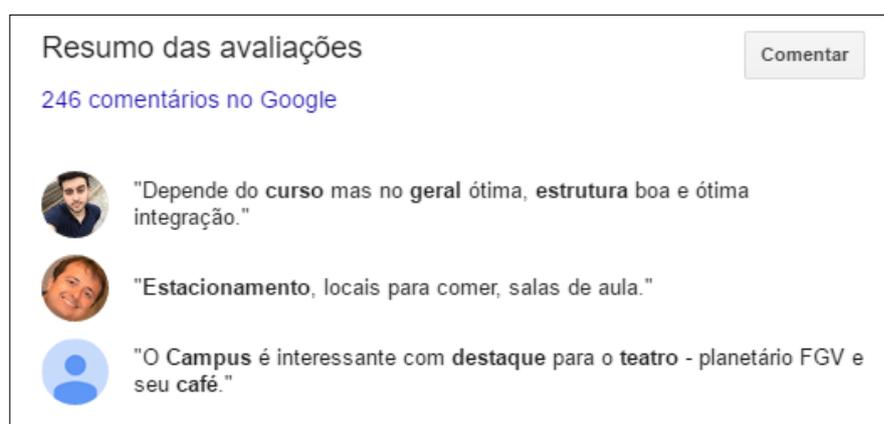


Figura 2 – Exemplo de extração de entidades em textos (Extraído de: maps.google. com).

Como visto em Medhat, Hassan e Korashy (2014), existem duas abordagens principais para realizar a tarefa de Mineração de Opiniões, as baseadas em aprendizagem de máquina e as baseadas em léxicos. Um terceiro tipo de abordagem combina a abordagem baseada em léxicos e a abordagem baseada em aprendizagem de máquina. A Figura 3 apresenta essas abordagens e como são realizadas as suas divisões.

¹ <http://maps.google.com>

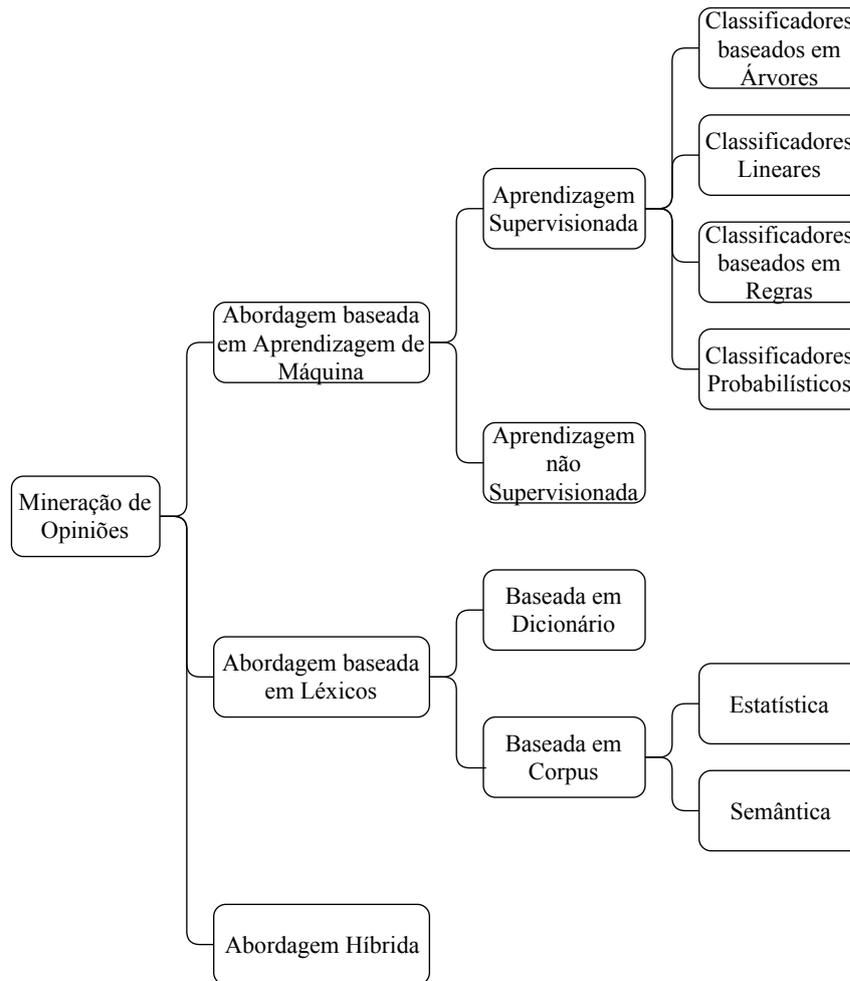


Figura 3 – Abordagens para a Mineração de Opiniões (Adaptado de: (MEDHAT; HASSAN; KORASHY, 2014)).

A abordagem baseada em léxicos, utiliza dicionários com palavras referentes as classes existentes para realizar a classificação. A utilização dessa abordagem é limitada a situações onde já existam recursos disponíveis ou exista a possibilidade de construção de novos recursos para o problema apresentado. Essa abordagem se divide em duas outras: a abordagem baseada em dicionário e a baseada em corpus. A abordagem baseada em dicionário é realizada com a construção de dicionários com base em grupos de palavras rotuladas manualmente e seus sinônimos. A abordagem baseada em corpus utiliza o mesmo princípio da abordagem baseada em dicionário, mas utiliza métodos estatísticos ou semânticos para orientação específica de contexto na busca de sinônimos. Diversos trabalhos exploram a utilização da abordagem baseada em léxicos para a Mineração de Opinião, como (KIM; HOVY, 2004), (ASGHAR, 2014) e (AGARWAL et al., 2016).

A abordagem baseada em aprendizagem de máquina visa utilizar técnicas de aprendizado de máquina para descobrir regras de maneira automática. Como apresentado no trabalho de Aranha (2007), na maioria dos casos a abordagem baseada em aprendizagem de máquina para análise a nível de documento segue um padrão de etapas, apresentado

na Figura 4. No caso das análises a nível de sentença e aspecto uma etapa adicional é realizada para a divisão dos documentos.

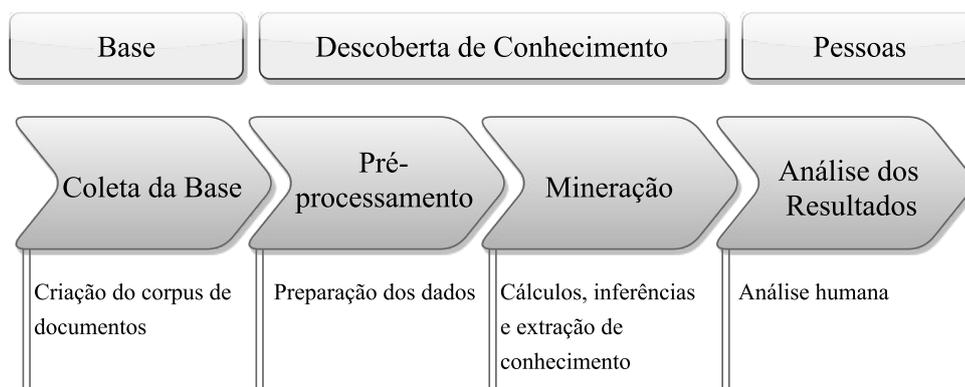


Figura 4 – Etapas para a utilização da abordagem baseada em aprendizagem de máquina (Adaptado de: (ARANHA, 2007)).

A primeira etapa desse processo se refere a coleta dos dados para a análise. Diversas fontes de dados podem ser utilizadas nessa etapa, como twitter, *reviews*, redes sociais e notícias. Alguns desses corpus podem ser anotados com informações extraídas no momento da coleta, como as estrelas de *reviews* (PANG; LEE; VAITHYANATHAN, 2002) e emoticons do twitter (GO; BHAYANI; HUANG, 2009) (PURVER; BATTERSBY, 2012). Em alguns casos, como em Dosciatti, Ferreira e Paraiso (2015) a fonte de origem do corpus não apresenta recursos que possam ser utilizados para a rotulação automática, necessitando assim ser rotulado manualmente.

Na etapa de pré-processamento diversas estratégias podem ser utilizadas para fornecer dados com maior qualidade e para aumentar a eficiência dos processos de mineração subsequentes. Como apresentado por Bakliwal et al. (2012), uma dessas estratégias é a substituição de símbolos por tokens com o seu significado, como em alguns casos onde links são substituídos por 'link', ou respostas de um tweet são substituídas por 'RT'. O objetivo dessa operação é representar de forma genérica o máximo de palavras possíveis evitando assim expressões que só acontecem uma vez no corpus, como o link de um site. Outra operação com o mesmo objetivo é a aplicação de um lematizador, essa técnica visa extrair a raiz da palavra, ignorando o tempo verbal, o gênero, o plural, entre outros. Em alguns casos, como em redes sociais, é necessário utilizar alguma técnica para remover letras repetidas, substituindo palavras como 'oiiii' por sua forma original 'oi'. Outra operação importante para possibilitar uma maior eficiência na classificação é a remoção dos stop words, onde todas as palavras que não apresentarem importância para a classificação são removidas. Assim como os stop words as palavras menos frequentes também podem ser removidas, diminuindo assim o tamanho das instâncias. Em alguns trabalhos também é realizada uma etapa de indexação, onde são criados índices para cada instância, possibilitando um acesso futuro mais rápido.

As estratégias apresentadas anteriormente são responsáveis por generalizar e limpar os dados. Após essa etapa, é necessário transformar o texto em uma entrada numérica aceita pelo classificador. Uma das estratégias apresentadas no trabalho de Salton e Buckley (1988) é a transformação de cada palavra presente no corpus em um atributo, técnica chamada de bag-of-words. Com essa transformação, é possível fornecer ao classificador o valor 1 caso a palavra esteja presente na instância ou 0 caso contrário. Outra maneira de realizar essa transformação utilizando cada palavra como um atributo é apresentar a quantidade de ocorrências de cada palavra em cada instância. A representação das palavras pode ir além da quantidade de ocorrências no texto. A representação TF-IDF (frequência do termo–inverso da frequência nos documentos, do inglês term frequency–inverse document frequency) possibilita calcular a importância de uma palavra em relação ao restante do corpus. O uso do TF-IDF possibilita distinguir as palavras com base em sua frequência, fornecendo um valor menor as palavras com um grande número de ocorrências.

Embora o método TF-IDF considere todo o corpus, as ligações entre as palavras não são consideradas. Como visto em Guthrie et al. (2006), uma forma de resolver esse problema é a utilização de n-grams. Um n-gram é a representação de n itens de um texto, onde um item pode ser um fonema, uma letra, uma sílaba, uma palavra, entre outros. Um exemplo do funcionamento do método 2-grams (também chamado de bigrams) com palavras pode ser observado com a divisão da frase “Chuva alaga penitenciária de Cascavel”. Com a aplicação do método a frase seria dividida em: “Chuva alaga”, “alaga penitenciária”, “penitenciária de” e “de Cascavel”. Para aumentar o alcance entre os termos existentes na frase também é possível realizar pulos em conjunto com os n-grams. Essa técnica é chamada de skip-grams e consiste na realização de ligação entre palavras com distância de no máximo k posições. A aplicação do método 2-skip-2-grams no exemplo citado anteriormente gera as divisões: “Chuva de”, “Chuva penitenciária”, “Chuva alaga”, “alaga Cascavel”, “alaga de”, “alaga penitenciária”, “penitenciária Cascavel”, “penitenciária de” e “de Cascavel”.

As técnicas skip-grams e bag-of-words também podem ser utilizadas para a criação de modelos mais complexos de representação, como o Word2Vec, apresentado nos trabalhos de Mikolov et al. (2013a) e Mikolov et al. (2013b). A ideia principal por trás desse modelo é a adição do contexto da palavra à sua representação. Os métodos de representação citados anteriormente tratam cada palavra de forma individual, enquanto o Word2Vec e outros algoritmos similares, como Glove (PENNINGTON; SOCHER; MANNING, 2014) e o Fasttext (BOJANOWSKI et al., 2016), utilizam as palavras mais próximas para interpretar o significado da palavra atual. O Word2Vec utiliza uma rede neural de três camadas (entrada, camada oculta e saída) para classificar uma palavra a partir do contexto em que ela ocorre. A rede treinada tem como objetivo retornar os vizinhos após adicionar a palavra principal como entrada ou retornar a palavra principal após usar os vizinhos como entrada. Os vetores numéricos são obtidos com o uso do softmax como camada de

saída, isso é possível pois os pesos existentes entre as sinapses dos neurônios representam as palavras em seus contextos, ou seja, a incorporação de palavras que forma a palavra principal.

Após o pré-processamento e a criação da representação numérica, um algoritmo de aprendizagem deve ser utilizado para realizar a classificação. Segundo Medhat, Hassan e Korashy (2014), os algoritmos mais comuns para esse tipo de tarefa são: Support Vector Machines (SVM), Naive Bayes (NB), K-Nearest Neighbour (KNN), Maximum Entropy (ME) e Redes Neurais (ANN). Entre os algoritmos baseados em Redes Neurais existe um grupo chamado de Deep Learning. Esses algoritmos vem conquistando resultados cada vez melhores para essa tarefa e são apresentados com mais detalhes na próxima seção.

2.2 Deep Learning

Uma rede neural convencional é composta pela ligação de diversos neurônios simples, onde cada um produz uma sequência de ativações. O aprendizado dessas redes tem como objetivo encontrar os pesos corretos para permitir que as ativações apropriadas sejam realizadas e a rede tenha o comportamento desejado. A aquisição desse comportamento pode necessitar de longas cadeias de estágios, onde cada estágio realiza transformações nas ativações da rede. Em algumas situações, uma única camada não é o suficiente para representar os dados de entrada. Para atender essas situações, foram criados algoritmos de um grupo chamado Deep Learning, que são redes com múltiplas camadas (SCHMIDHUBER, 2015). No trabalho de LeCun, Bengio e Hinton (2015) o Deep Learning é definido como os métodos que permitem modelos computacionais compostos de várias camadas de processamento para aprender representações de dados com múltiplos níveis de abstração.

Nos últimos anos diversos trabalhos relataram a melhora do estado da arte com a utilização desses algoritmos. O Deep Learning foi utilizado para diversas tarefas, como reconhecimento de imagens (HE et al., 2016), reconhecimento de discurso (DENG; HINTON; KINGSBURY, 2013), identificação de opiniões em textos (IRSOY; CARDIE, 2014), geração automática de imagens a partir de textos (REED et al., 2016), detecção de pedestres em ruas (OUYANG; WANG, 2013), previsão de sequências de DNA (ALIPANAHI et al., 2015), entre diversas outras.

Além das aplicações já existentes, LeCun, Bengio e Hinton (2015) afirmam que a melhora nos recursos computacionais e o aumento na quantidade de dados disponíveis pode possibilitar mais casos de sucesso no futuro. Além desses avanços pesquisadores buscam apresentar novos algoritmos e arquiteturas, o que pode acelerar ainda mais esse progresso. Para demonstrar o funcionamento desses algoritmos, o restante desta seção apresenta uma breve explicação sobre rede neurais e sobre os principais algoritmos de

Deep Learning.

2.2.1 Redes Neurais

As redes neurais apresentaram um grande avanço desde o seu primeiro modelo apresentado em McCulloch e Pitts (1943). Esse tipo de arquitetura é baseada no funcionamento do sistema nervoso central de animais. Na biologia, o neurônio é um tipo especial de célula no qual um conjunto é capaz de armazenar informações. Essa célula é composta basicamente pelo corpo e dois canais de comunicação: os dendritos que recebem os impulsos e os axônios que transmitem os impulsos.

Na computação um modelo dessa célula foi criado por McCulloch e Pitts (1943). Assim como o modelo biológico esse modelo é composto de um núcleo central e dois canais de comunicação. O modelo computacional, ilustrado na Figura 5, é composto por um vetor com as entradas x e pesos w para representar as sinapses. O corpo realiza a soma de x com base em w e utiliza uma função de ativação com o apoio de um limiar u para determinar se o neurônio é ativado ou não. Caso o neurônio seja ativado o retorno é 0, caso contrário 1. Matematicamente essa operação pode ser definida por $y = \theta \left(\sum_{j=1}^n w_j x_j - u \right)$, onde θ é uma função contendo uma etapa (JAIN; MAO; MOHIUDDIN, 1996).

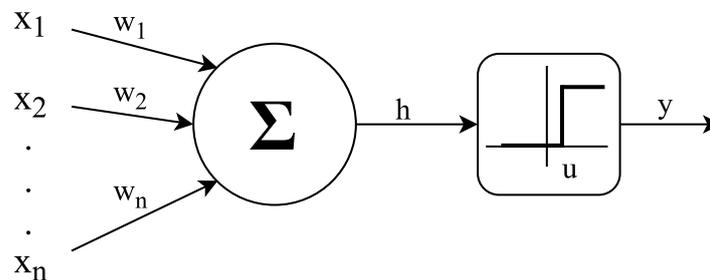


Figura 5 – Modelo de neurônio apresentado de McCulloch e Pitts (Extraído de: (JAIN; MAO; MOHIUDDIN, 1996)).

O modelo criado por McCulloch e Pitts (1943) necessita que os pesos w sejam ajustados manualmente, gerando diversas dificuldades. No trabalho de Rosenblatt (1958), os autores propõem um modelo probabilístico chamado de Perceptron para realizar esses ajustes automaticamente. Com o passar do tempo diversas outras alterações do neurônio apresentado por McCulloch e Pitts foram apresentadas. A principal alteração foi a criação de diferentes funções de ativação. No trabalho de Karlik e Olgac (2011) as funções Bi-polar sigmoid, Uni-polar sigmoid, Tanh, Conic Section e Radial Bases Function são apresentadas e seus desempenhos são avaliados. Além das funções apresentadas por Karlik e Olgac diversas outras podem ser encontradas na literatura.

Outra modificação realizada foi em relação a arquitetura de conexão dos neurônios. No trabalho de Jain, Mao e Mohiuddin (1996), os autores apresentam dois grupos de ar-

quitetas, as Feed-forward networks e as Recurrent networks. As Feed-forward networks são as arquiteturas onde todas as conexões são unidirecionais, como o single layer perceptron e o multilayer perceptron. As arquiteturas Recurrent networks, também chamadas de feedback networks, apresentam laços para permitir a utilização da experiência dos neurônios anteriores. Alguns exemplos das redes Recurrent são as competitive networks e as Hopfield network. A taxonomia dessas redes é apresentada na Figura 6.

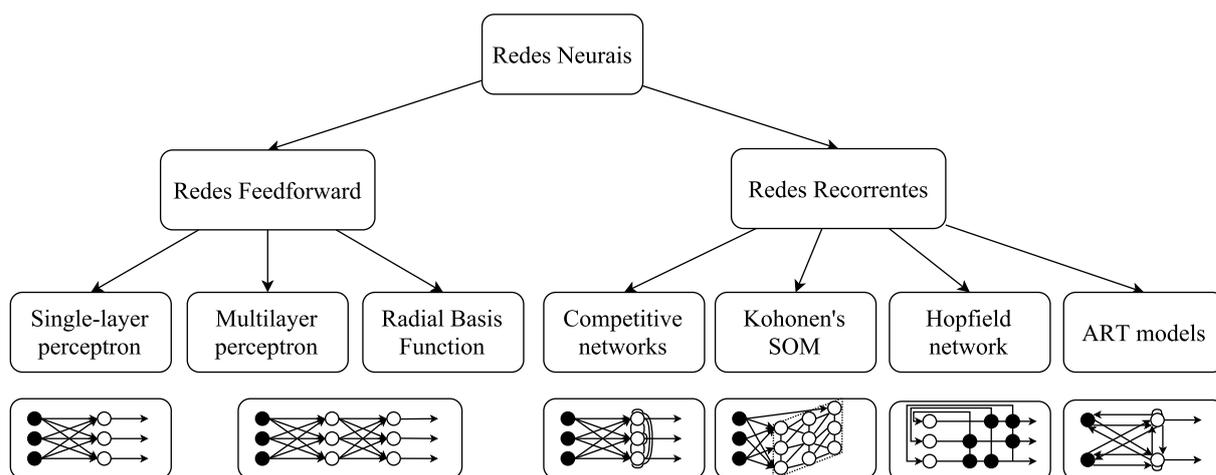


Figura 6 – Taxonomia das redes neurais (Adaptado de: (JAIN; MAO; MOHIUDDIN, 1996)).

Além das arquiteturas demonstradas na Figura 6 diversas outras foram apresentadas na literatura. Como visto em Deng, Hinton e Kingsbury (2013) a escolha da melhor arquitetura depende do tipo de problema a ser resolvido. Em alguns casos essas arquiteturas demandam um grande número de camadas para resolver o problema, como os algoritmos de Deep Learning.

2.2.2 Deep Neural Network

Segundo o trabalho realizado por Schmidhuber (2015), a principal inspiração para a criação de algoritmos de Deep Learning foi o princípio apresentado por Hubel e Wiesel (1959) e Hubel e Wiesel (1962), onde o córtex visual de gatos foi estudado. Nos trabalhos de Hubel e Wiesel foi identificado que as células do córtex são ativadas com pequenas características visuais, como a orientação de linhas, que se comunicam entre si para reconhecer os padrões dos objetos.

O avanço nos estudos do funcionamento dos neurônios e os estudos de redes neurais com múltiplas camadas por Ivakhnenko e Lapa (1966) permitiram que pesquisadores utilizassem novas formas de representação dos dados. Fukushima e Miyake (1980) apresentam um método capaz de dividir o problema original da mesma forma que o cérebro de animais. Atualmente esse método é chamado de Convolutional Neural Network (CNN) e

diversos trabalhos utilizam uma versão melhorada dessa rede. As principais modificações realizadas com o passar do tempo foram o ajuste dos pesos com o algoritmo backpropagation por LeCun et al. (1990) e o downsampling com o algoritmo Max-Pooling.

Uma ilustração do funcionamento dos modelos atuais da CNN é apresentada na Figura 7. Como visto no modelo apresentado por Krizhevsky, Sutskever e Hinton (2012), as CNNs são redes do tipo feed-forward compostas por uma ou mais camadas de convolução. Cada camada possui um conjunto de máscaras que após serem aplicadas tem seus resultados combinados. Outra função da convolução é criar subamostras da imagem original, permitindo a criação de imagens menores, com resultados diferentes. Após o fim das etapas de convolução a operação de max pooling é realizada sobre os mapas de características para se obter o maior valor existente em cada mapa, com isso as características mais importantes são extraídas. Por fim as características obtidas na função de max pooling são utilizadas como entrada em uma única camada cuja saída é a probabilidade de cada rótulo ser verdadeiro.

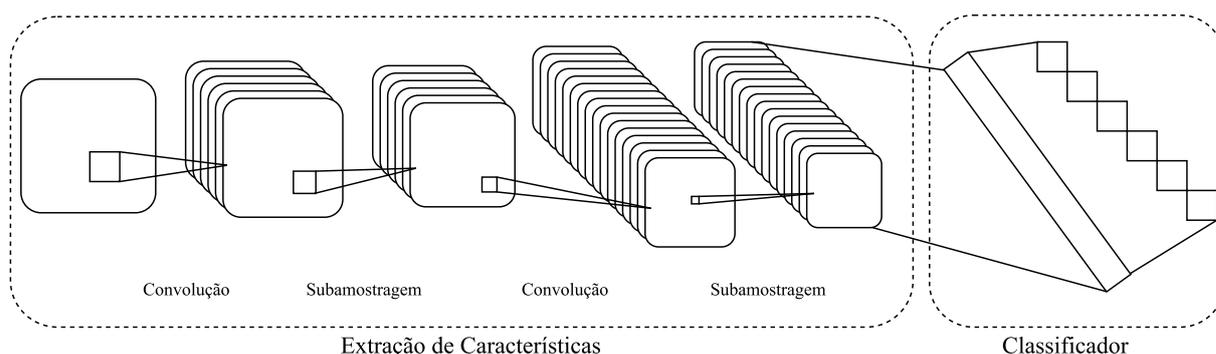


Figura 7 – Funcionamento de uma Convolutional Neural Network

Com o passar do tempo pesquisadores estudaram outras formas de representação com múltiplas camadas. Um exemplo é o algoritmo chamado Recurrent Neural Network (RNN). Segundo Irsoy e Cardie (2014), esse algoritmo possibilita uma ligação entre os dados de entrada, realizando a classificação com diversos pedaços distintos da entrada ao mesmo tempo. Um exemplo disso é a leitura, para compreender um texto é necessário o conhecimento das palavras anteriores e não apenas algumas palavras do texto como o utilizado pelo CNN. Para realizar essa tarefa a RNN utiliza múltiplas cópias da mesma rede, onde cada uma passa uma mensagem a seu sucessor, como ilustrado na Figura 8.

Uma variação do RNN é o Long Short-Term Memory (LSTM), apresentado inicialmente em Hochreiter e Schmidhuber (1997). O LSTM é um tipo de RNN capaz de utilizar informações armazenadas a longo prazo para o treinamento. Nas RNN convencionais é possível realizar uma ligação com algumas informações anteriores, mas esse algoritmo não é capaz de realizar essa ligação com informações distantes. Para resolver o problema da dependência de termos distantes o modelo introduz uma nova estrutura chamada célula de

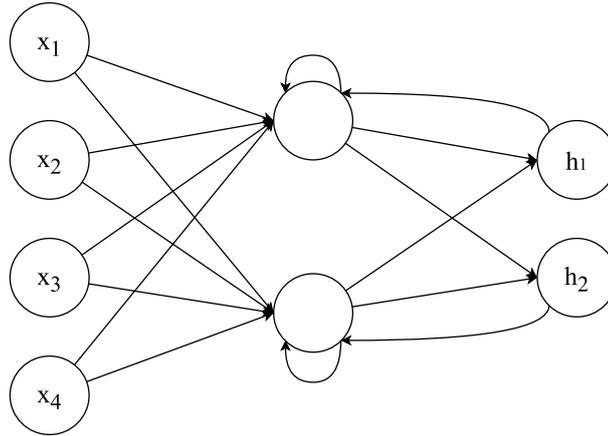


Figura 8 – Funcionamento de uma Recurrent Neural Network.

memória, apresentada na Figura 9. Essa estrutura é composta de um portão de entrada, neurônios com conexões recorrentes, um portão de esquecimento e um portão de saída. A primeira etapa da LSTM é utilizar uma camada sigmoide para decidir qual informação será descartada da célula atual. Após a escolha do que será descartado outra camada sigmoide é utilizada para decidir quais novas informações devem ser armazenadas na célula. Em seguida um vetor *tanh* com os candidatos a serem armazenados é criado. Após a criação do vetor uma função sigmoide é utilizada para decidir quais as melhores informações para a próxima célula. Com todos os passos realizados a célula antiga é atualizada e o processo é realizado novamente com os novos dados.

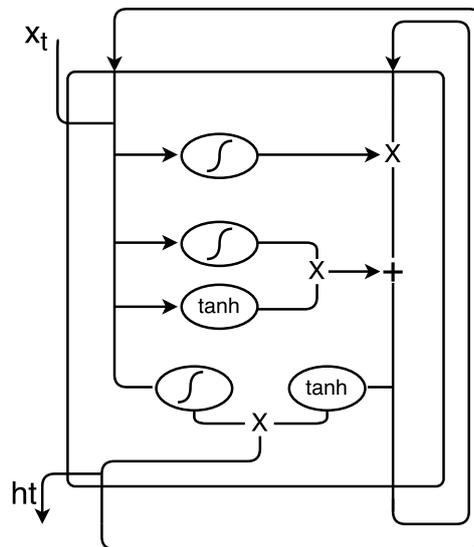


Figura 9 – Estrutura de uma Long Short-Term Memory

As etapas realizadas pelo LSTM também podem ser representadas pelas equações 2.1. Nessas equações os caracteres subscritos representam vetores e os caracteres em maiúsculo matrizes. Na notação utilizada, f representa o portão de esquecimento, i representa o portão de entrada, o o portão de saída, c representa a célula de memória e h

a unidade LSTM. As matrizes existentes são W que armazenam os pesos da entrada e U que armazenam as conexões recorrentes.

$$\begin{aligned}
 f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
 i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
 o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
 c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
 h_t &= o_t \circ \sigma_h(c_t)
 \end{aligned} \tag{2.1}$$

Como apresentado na equação 2.1, todas as equações podem ser observadas como etapas para a atualização da unidade LSTM h_t , realizada através da atualização da célula de memória atual c_t . Essa atualização é realizada diversas vezes, a quantidade de repetições desses procedimentos é definido por um parâmetro chamado época. A quantidade de épocas define a quantidade de passos para frente (*forward*) e para trás (*backward*) para todos os exemplos existentes. A quantidade de iterações necessárias para a realização dessa tarefa é definida com base na quantidade de épocas utilizadas em conjunto com o tamanho do batch definido. O tamanho do batch define a quantidade de exemplos utilizados simultaneamente em uma passagem para frente e para trás. Esse parâmetro é utilizado para adequar a quantidade de exemplos com a quantidade de memória e capacidade de processamento disponível. Outro parâmetro existente na configuração dessas redes é a quantidade de camadas ocultas (*hidden size*). Esse valor representa a quantidade de unidades LSTM existentes na rede.

Na primeira etapa de cada atualização de cada célula de memória do problema uma função sigmoide é utilizada para descartar as informações antigas com base nos pesos da entrada, suas conexões na camada anterior e a tendência (*bias*). Após a eliminação das informações outra função sigmoide é utilizada para a introdução de novas informações com base nos mesmos pesos e conexões utilizados para o descarte na etapa anterior. A próxima etapa utiliza os mesmos dados das etapas anteriores como entrada de uma função sigmoide que define o que vai ser eliminado pelo portão de saída. A célula de memória c é atualizada tendo como base os dados obtidos pelas funções sigmoides das etapas anteriores. Os pesos da entrada, as conexões na camada anterior e a tendência também são utilizados nessa etapa em uma função de tangente hiperbólica. O resultado dessa função, os dados do portão de esquecimento f , os dados do portão de entrada i e os dados da célula de memória da camada anterior são utilizados para definir o novo estado da célula que será passado para a próxima camada.

Uma alternativa as operações realizadas pelo algoritmo LSTM pode ser desenvolvida com a utilização de denoising autoencoders. O objetivo dessa operação é forçar a camada oculta a descobrir características mais robustas, evitando que características sim-

ples sejam aprendidas. Os autoencoders são utilizados para recriar a entrada dos dados, alterando assim a forma como o problema é recebido pelo algoritmo (BENGIO et al., 2007). No trabalho de Vincent et al. (2008), os autores apresentam uma forma de utilizar os denoising autoencoders em uma rede com várias camadas, chamada de Stacked Denoising Autoencoders (SDA). A primeira etapa desse processo é o treinamento de forma não supervisionada. Nessa etapa a saída do denoising autoencoder de uma camada é utilizada como entrada da próxima camada. Após o treinamento não supervisionado a rede passa por um segundo treinamento com o objetivo de realizar um ajuste fino supervisionado. Para essa etapa é adicionada uma camada de regressão logística e a rede é treinada da mesma maneira que um multilayer perceptron. Uma ilustração de um denoising autoencoder é apresentado na Figura 10. Nesse exemplo a entrada x é corrompida por q , gerando a nova entrada \tilde{x} . O encoder f cria a nova entrada y e o decoder g produz a reconstrução z . O erro da reconstrução em comparação com a entrada original é calculado utilizando a métrica $L_H(x, z)$.

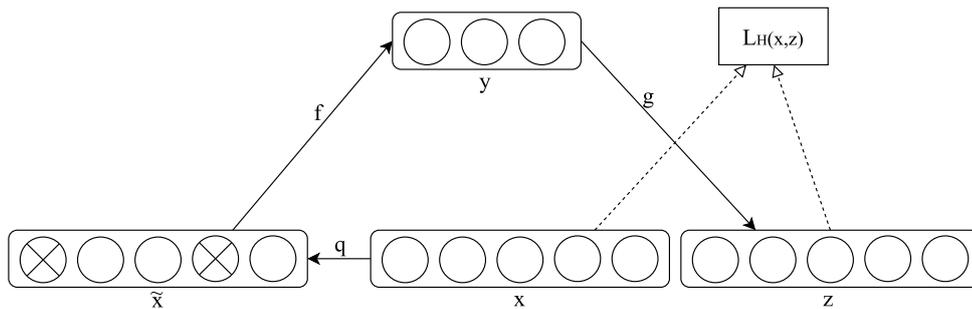


Figura 10 – Estrutura de um denoising autoencoder

Outro algoritmo existente na literatura é o Boltzmann Machines (BM), apresentado por Ackley, Hinton e Sejnowski (1985). Esse algoritmo é um tipo de rede recorrente e estocástica, que utiliza um conjunto de vetores de dados binários com modelos estatísticos para calcular a probabilidade de serem ativados ou desativados. Em diversas situações os neurônios da rede são divididos em visíveis e ocultos. Os neurônios visíveis são aqueles que possuem alguma ligação com os dados de entrada, enquanto os ocultos não sofrem influências externas. Uma ilustração da arquitetura de uma BM é apresentada na Figura 11. Na figura v representa os neurônios visíveis e h os ocultos.

Outra forma de organização dessa arquitetura é a apresentada por Salakhutdinov e Hinton (2009), onde existe uma série de BMs empilhadas. Essa arquitetura recebe o nome de Deep Boltzmann Machines (DBM). Nas DBMs, as unidades ocultas são organizadas de forma hierárquica, de forma que apenas os vizinhos possam se comunicar. A Figura 12 apresenta uma ilustração dessa arquitetura, onde v representa os neurônios visíveis, h os ocultos e w os pesos calculados pelos modelos estatísticos.

Um algoritmo com funcionamento parecido ao DBM é o Deep Belief Networks

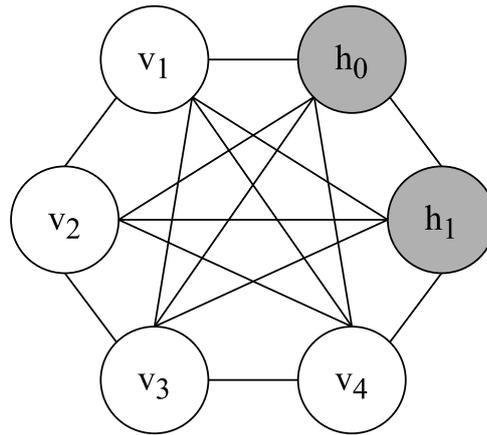


Figura 11 – Estrutura de um Boltzmann Machines

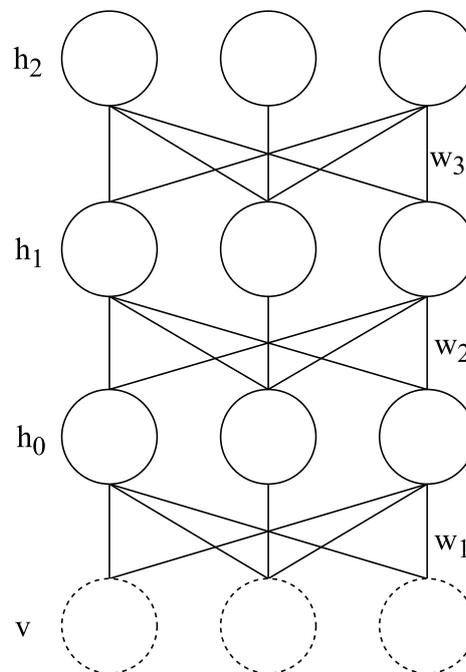


Figura 12 – Estrutura de um Deep Boltzmann Machines

(DBN). A principal diferença entre esses algoritmos é o fato do DBN ser direto, sem realimentação, enquanto o DBM possui conexões indiretas. Uma representação dessa arquitetura é apresentada na Figura 13

Como visto nesta seção, diversos algoritmos de Deep Learning foram apresentados ao longo do tempo. Uma série de ferramentas surgiu para auxiliar desenvolvedores a utilizar esses recursos tanto comercialmente quanto no âmbito de pesquisa. Entre as principais ferramentas existentes estão: *caffe*², *CNTK*³, *MXNet*⁴, *Deeplearning4j*⁵, *theano*⁶,

² <http://caffe.berkeleyvision.org/>

³ <https://www.microsoft.com/en-us/cognitive-toolkit/>

⁴ <https://mxnet.apache.org/>

⁵ <https://deeplearning4j.org/>

⁶ <http://deeplearning.net/software/theano/>

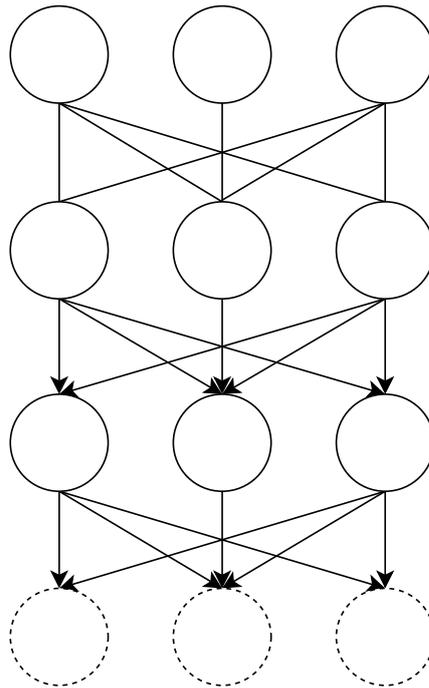


Figura 13 – Estrutura de um Deep Belief Networks

TensorFlow⁷, Torch⁸, keras⁹ e Lasagne¹⁰. O trabalho de Shi et al. (2016) apresenta um comparativo do desempenho de algumas dessas ferramentas utilizando diferentes plataformas e bases.

Embora diversos trabalhos da área de Mineração de Opiniões relatem a melhora do estado da arte com a utilização de algoritmos de Deep Learning, poucos trabalhos exploram a utilização desses algoritmos com dados multirrótulos. A próxima seção apresenta uma revisão sobre esse tipo de classificação.

2.3 Classificação Multirrótulo

As abordagens mais utilizadas para as tarefas tradicionais de aprendizado supervisionado tratam da classificação chamada simples rótulo. Nesse tipo de classificação cada exemplo é representado por apenas um rótulo. Considerando λ como um único rótulo para uma instância da base de dados utilizada e L como o conjunto de classes do problema, temos a classificação chamada de binária para os casos onde $|L| = 2$. Em algumas situações, como para a tarefa de identificação de emoções em textos abordada neste trabalho, esse tipo de classificação também é chamada de classificação de polaridade. Nos casos onde $|L| > 2$ a classificação é chamada de multi classe. Diferente da classificação binária e multi-classe, onde existe apenas um rótulo λ para cada instância, a classificação

⁷ <https://www.tensorflow.org/>

⁸ <http://torch.ch/>

⁹ <https://keras.io/>

¹⁰ <https://lasagne.readthedocs.io>

multirrótulo aceita um conjunto de rótulos Y para representar cada instância, tal que $Y \subseteq L$ (TSOUMAKAS; KATAKIS, 2006).

Como apresentado por Tsoumakas, Katakis e Vlahavas (2009), diversas informações textuais, como documentos e web sites, possuem mais de um rótulo. Um exemplo desse tipo de informação é a classificação de notícias sobre medidas econômicas de outros países, que devem pertencer as categorias economia, internacional e em alguns casos política. Em conjunto com a análise de informações textuais, diversas outras aplicações podem ser realizadas com esse tipo de classificação, como a identificação de objetos em imagens e vídeos, a categorização de músicas por emoções e o marketing direcionado.

Os métodos existentes para a classificação multirrótulo podem ser agrupados em métodos de transformação de problema e métodos de adaptação de algoritmo. A Figura 14 apresenta a categorização dos algoritmos multirrótulo apresentada por Zhang e Zhou (2014). O restante desta seção apresenta uma definição e os principais algoritmos desses dois métodos, em seguida as principais métricas para a avaliação desse tipo de classificação são apresentadas.

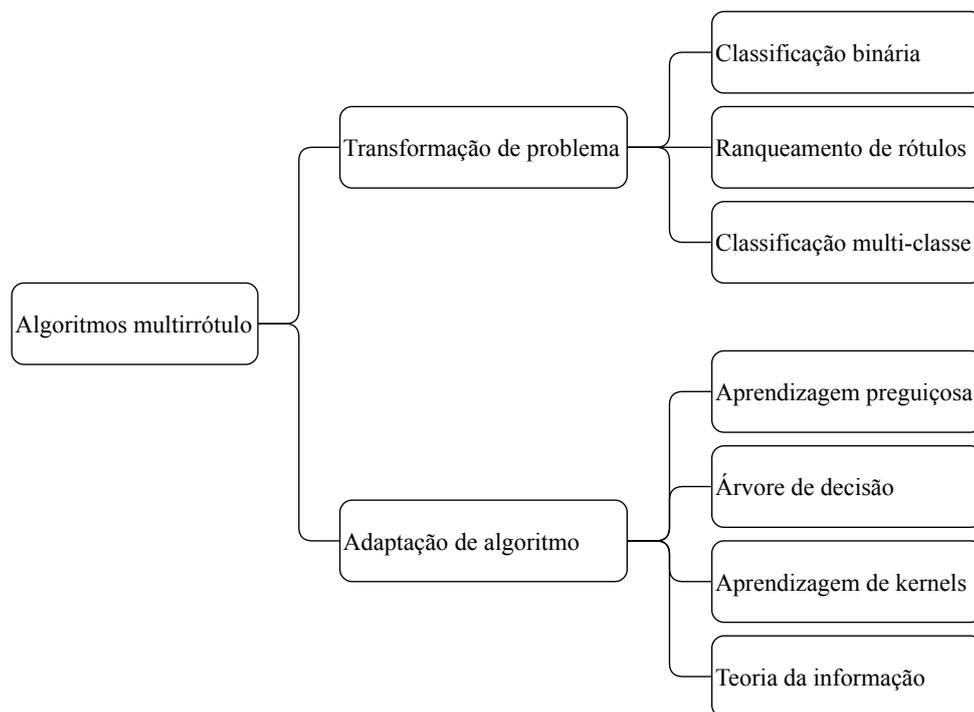


Figura 14 – Categorização dos algoritmos multirrótulo (Adaptado de: (ZHANG; ZHOU, 2014)).

2.3.1 Métodos de Transformação de Problema

Cherman, Monard e Metz (2011) definem os métodos de transformação de problema como todos os métodos que realizam a transformação de um problema multirrótulo em um ou mais problemas simples rótulo. No trabalho de Zhang e Zhou (2014), os autores apresentam os quatro principais métodos de transformação de problema existentes na

literatura. Para demonstrar o funcionamento desses métodos será utilizado o exemplo de base multirrótulo apresentado na Tabela 1.

Tabela 1 – Exemplo de uma base de dados multirrótulo

Ex.	Amor	Ódio	Alegria	Tristeza
1	X		X	
2	X			
3	X			X
4		X		X

Como visto na Figura 14, os principais métodos de transformação de problema são divididos em: binary classification (classificação binária), label ranking (ranqueamento de rótulos) e multi-class classification (classificação multi-classe). Zhang e Zhou (2014) apresentam dois métodos de binary classification: o Binary Relevance e o Classifier Chains.

Segundo Alvares-Cherman, Metz e Monard (2012), o algoritmo Binary Relevance transforma um problema multirrótulo em vários problemas binários com apenas um rótulo. Com essa transformação, cada rótulo é classificado individualmente por um classificador binário como relevante ou irrelevante. Para ilustrar o funcionamento desse método, a transformação da Tabela 1 com o algoritmo Binary Relevance é apresentada na Tabela 2. Como é possível observar na Tabela 2, um classificador é treinado individualmente para cada rótulo λ , gerando os modelos de classificação binária h_i . Quando uma nova instância \mathbf{x} é fornecida para ser classificada, os modelos h_i predizem as classificações y_i . Após realizar todas as predições os resultados y_i são agrupados com base na instância gerando o conjunto de rótulos final.

Tabela 2 – Dados da Tabela 1 transformados com o algoritmo Binary Relevance

Ex.	Amor	\sim Amor	Ex.	Ódio	\sim Ódio
1	X		1		X
2	X		2		X
3	X		3		X
4		X	4	X	

Ex.	Alegria	\sim Alegria	Ex.	Tristeza	\sim Tristeza
1	X		1		X
2		X	2		X
3		X	3	X	
4		X	4	X	

No trabalho de Cherman, Monard e Metz (2011), os autores apresentam como principal vantagem do algoritmo Binary Relevance o fato de não ser tão complexo computacionalmente como os outros métodos de transformação do problema. Embora a transformação utilize poucos recursos esse método necessita de um modelo para cada classe, gerando um alto custo computacional nos casos em que existem muitas classes. Uma

forma de resolver esse problema é a utilização de um método dividir-para-conquistar. O problema pode ser transformado em uma árvore hierárquica possibilitando a criação de um problema com menos classes. Um exemplo onde a transformação em um problema hierárquico possibilitou melhores resultados foi em Tsoumakos, Katakis e Vlahavas (2008), onde os autores apresentam um algoritmo chamado HOMER (Hierarchy Of Multilabel classifiERs). Outro problema existente no algoritmo Binary Relevance é a classificação das classes de forma independente, em muitos casos algumas classes impedem que outras seja corretas. Um exemplo disso pode ser observado em uma base de textos rotulada com emoções, onde dificilmente existe um caso em que as classes alegria e medo são verdadeiras ao mesmo tempo.

Para resolver o problema da classificação independente Read et al. (2009) apresentam um método chamado Classifier Chains. Esse método permite modelar correlações de rótulos, mantendo a mesma complexidade computacional do algoritmo Binary Relevance. Para realizar essa tarefa, o método transforma o problema multirrótulo em uma cadeia de problemas binários, onde a previsão de um classificador influencia a decisão do próximo a ser utilizado. Cheng, Hüllermeier e Dembczynski (2010) explicam que nesse método um classificador é treinado individualmente para cada rótulo, gerando os modelos h_i . Dada uma nova instância \mathbf{x} a ser classificada, o modelo h_1 prediz y_1 , ou seja, a relevância do rótulo λ_1 para a instância \mathbf{x} . Em seguida h_2 prevê a relevância de λ_2 , utilizando para isso o valor de \mathbf{x} em conjunto com o valor y_1 predito anteriormente. Essa operação é realizada até as i classes existentes serem preditas, onde, no ultimo caso h_i utiliza todos os valores y_{i-1} preditos anteriormente como entrada adicional. Um exemplo da diferença entre os valores de entrada \mathbf{x} para cada classificador h nos métodos Binary Relevance e Classifier Chains é apresentado na Tabela 3. Para demonstração é utilizada a instância \mathbf{x} com os valores $[0,1,0,1,0,1]$ e os rótulos y da instância 1 do exemplo apresentado da Tabela 1.

Tabela 3 – Comparação entre as instâncias de entrada \mathbf{x} para os algoritmos Binary Relevance e Classifier Chains

Binary Relevance			Classifier Chains		
h	\mathbf{x}	y	h	\mathbf{x}	y
h_1	$[0,1,0,1,0,1]$	1	h_1	$[0,1,0,1,0,1]$	1
h_2	$[0,1,0,1,0,1]$	0	h_2	$[0,1,0,1,0,1,1]$	0
h_3	$[0,1,0,1,0,1]$	1	h_3	$[0,1,0,1,0,1,1,0]$	1
h_4	$[0,1,0,1,0,1]$	0	h_4	$[0,1,0,1,0,1,1,0,1]$	0

Um dos desafios existentes no método Classifier Chains é a escolha da melhor ordem dos classificadores h . Nos trabalhos de Read et al. (2009) e Read et al. (2011), os autores propoem o uso de um Ensemble de Classifier Chains para construir n permutações randômicas, possibilitando assim a utilização de diversas ordens de classificadores h . Com o passar do tempo outros métodos foram apresentados para a escolha da melhor sequência de classificadores, como em Cheng, Hüllermeier e Dembczynski (2010), onde um

otimizador probabilístico chamado Bayes Optimal Prediction é utilizado. Em Goncalves, Plastino e Freitas (2013), os autores utilizam um algoritmo genético para essa tarefa. Outra abordagem foi a utilizada por Read, Martino e Luengo (2014), onde foi apresentado um método utilizando o algoritmo de monte carlo para encontrar a melhor sequência.

Como visto na Figura 14, existem métodos de transformação de problema que exploram abordagens diferentes da simples transformação do problema multirrótulo em um problema binário. Um desses grupos de métodos é o Label Ranking, apresentado com mais detalhes por Vembu e Gärtner (2010). Um método pertencente a esse grupo é o Calibrated Label Ranking, introduzido por Fürnkranz et al. (2008). A ideia básica desse método é transformar o problema multirrótulo em um problema de ranqueamento de rótulos, onde a posição dos rótulos é decidida com base em técnicas de comparação por pares e utilizada para realizar a classificação. Ou seja, o algoritmo Label Ranking tradicional é utilizado para realizar a ordenação dos rótulos com base em sua relevância, após a ordenação uma calibração de rótulos é adicionada, possibilitando a separação dos rótulos relevantes dos irrelevantes. Para realizar essa tarefa Fürnkranz et al. (2008) realizaram a combinação de uma técnica de comparação por pares com uma técnica de classificação tradicional, onde um classificador separado é utilizado para prever os rótulos relevantes. Para o conjunto de rótulos L de determinada instância \mathbf{x} , temos $\{\lambda_1, \lambda_2, \dots, \lambda_i\}$. O algoritmo Label Ranking realiza a ordenação desses rótulos com base na sua relevância, gerando $\lambda_{o_1} \succ \lambda_{o_2} \succ \dots \succ \lambda_{o_i}$. Para realizar a classificação multirrótulo, o algoritmo Calibrated Label Ranking adiciona um classificador a esse problema, criando assim um calibration label λ_0 , que tem por objetivo separar os conjuntos de rótulos relevantes P dos conjuntos irrelevantes N . Um exemplo do funcionamento dessa divisão é a adição de λ_0 no conjunto ordenado, onde é gerado $\lambda_{o_1} \succ \dots \succ \lambda_{o_j} \succ \lambda_0 \succ \lambda_{o_{j+1}} \succ \dots \succ \lambda_{o_i}$. Com isso é possível obter os conjuntos de rótulos $P = \{\lambda_{o_1}, \dots, \lambda_{o_j}\}$ e $N = \{\lambda_{o_{j+1}}, \dots, \lambda_{o_i}\}$. Uma ilustração desse método pode ser vista na Tabela 4, onde é demonstrado o funcionamento do algoritmo Calibrated Label Ranking na instância 1 do exemplo apresentado da Tabela 1.

O principal problema apresentado por esse algoritmo é a necessidade de diversas classificações em pares para realizar o ranqueamento. Esse tipo de classificação necessita de $i(i-1)/2$ classificadores binários, gerando um problema em escala quadrática em relação ao número de rótulos. Alguns trabalhos estudam formas de reduzir o número de classificadores, como (MENCÍA; PARK; FÜRNRANZ, 2010), (MADJAROV; GJORGJEVIKJ; DELEV, 2010) e (MADJAROV; GJORGJEVIKJ; DŽEROSKI, 2012).

Outra forma existente para adaptar problemas multirrótulos é transformando o mesmo em um problema multi classe, como o método Label Powerset. Como apresentado em Tsoumakas e Vlahavas (2007), esse algoritmo considera todas as combinações de rótulos existentes no conjunto de treinamento como um valor de classe individual de uma

Tabela 4 – Etapas do algoritmo Calibrated Label Ranking

(1)	Ex.	Amor	Ódio	Alegria	Tristeza
	1	X		X	

(2)	Ex.	Amor	Alegria	Ódio	Tristeza
	1	X	X		

(3)	Ex.	Amor	Alegria	λ_0	Ódio	Tristeza
	1	X	X	O		

(4)		<i>P</i>		<i>N</i>	
	Ex.	Amor	Alegria	Ódio	Tristeza
	1	X	X		

tarefa de classificação multi classe e simples rótulo. Ou seja, o algoritmo utiliza todas as combinações existentes em nos conjuntos Y como rótulos individuais do problema. A Tabela 5 apresenta o resultado dessa transformação nos dados apresentados na Tabela 1.

Tabela 5 – Dados da Tabela 1 transformados com o algoritmo Label Powerset

Ex.	Amor \wedge Alegria	Amor	Amor \wedge Tristeza	Ódio \wedge Tristeza
1	X			
2		X		
3			X	
4				X

Como apresentado em Tsoumakias, Katakis e Vlahavas (2011), o principal problema existente nesse tipo de classificação é causado pela diferença existente entre os domínios. Em muitos casos o algoritmo Label Powerset cria uma grande quantidade de classes com poucos exemplos associados a ela. Para resolver esse problema os autores apresentaram um método chamado Random k -Labelsets (RA k EL), que divide o conjunto inicial de rótulos em m subconjuntos randômicos com k classes chamados labelsets. Após essa divisão, o algoritmo Label Powerset é utilizado para realizar a transformação do problema e possibilitar o treinamento. Um exemplo da divisão em subconjuntos realizada pelo RA k EL pode ser vista na Tabela 6, que apresenta a transformação da instância 1 do exemplo apresentado da Tabela 1.

Uma característica em comum entre todos os métodos de transformação de problema apresentados anteriormente é a necessidade de utilização de múltiplos classificadores para resolver o problema (BI; KWOK, 2013). Essa característica torna inviável a utilização de classificadores mais robustos, fazendo com que a maioria dos autores optem por classificadores que utilizam menos recursos computacionais. Como em Mahdavi-Shahri et al. (2016), onde os autores utilizam KNN, Naïve Bayes, Random Tree, Reptree e C4.5 e em Cherman, Monard e Metz (2011), onde são utilizados os algoritmos C4.5, Naïve

Tabela 6 – Exemplo de classificação usando o algoritmo RAKEL com $k = 3$ e $m = 4$

Ex.	modelo	labelset	predições				
			Amor	Ódio	Alegria	Tristeza	
1	h_1	{Amor, Ódio, Alegria}	1	0	1	-	
	h_2	{Amor, Alegria, Tristeza}	1	-	1	0	
	h_3	{Ódio, Alegria, Tristeza}	-	0	1	1	
	h_4	{Amor, Ódio, Tristeza}	1	0	-	0	
	voto			3/3	0/3	3/3	1/3
	predição final			1	0	1	0

Bayes, KNN e SVM. Uma forma de resolver esse problema é realizando a adaptação das instâncias \mathbf{x} com base no tipo de entrada utilizada. No trabalho de Wei et al. (2016), os autores apresentam uma forma de dividir imagens para realizar a classificação utilizando pequenos pedaços da imagem original com objetos. Em Liu e Chen (2015), os autores propõem um método para divisão de textos para realizar a tarefa de Análise de Sentimentos. Além dos métodos de adaptação de problema apresentados existem os métodos de transformação de algoritmos, apresentados a seguir.

2.3.2 Métodos de Adaptação do Algoritmo

Os métodos de adaptação de algoritmo são definidos por Zhang e Zhou (2014) como todos os algoritmos tradicionais de mineração de dados que são adaptados para trabalhar diretamente com um problema multirrótulo. Uma dessas alterações é o ML-KNN, apresentado por Zhang e Zhou (2005) e Zhang e Zhou (2007). Nesses trabalhos, os autores realizam uma adaptação do algoritmo KNN para possibilitar a utilização com dados multirrótulos. Na primeira etapa do algoritmo ML-KNN, todos os k vizinhos mais próximos de cada instancia são identificados. Após essa identificação, são utilizadas informações estatísticas obtidas a partir dos conjuntos de rótulos dos vizinhos para utilização com o princípio maximum a posteriori, que é utilizado para determinar os rótulos.

Outra abordagem existente na literatura é a adaptação de algoritmos de árvores de decisão. Clare e King (2001) apresentam uma adaptação do algoritmo C4.5, que permite vários rótulos nos nós folhas da árvore. Para possibilitar isso, os autores apresentam uma modificação na fórmula da entropia realizando a soma das entropias de cada rótulo, permitindo assim solucionar problemas multirrótulo. A nova fórmula apresentada é $entropia(E) = -\sum_{i=1}^N (p(c_i) \log p(c_i) + q(c_i) \log q(c_i))$. Onde $p(c_i)$ é a probabilidade da classe c_i e $q(c_i) = 1 - p(c_i)$

Elisseff, Weston et al. (2001) apresentam uma modificação do algoritmo SVM. A estratégia usada por esse algoritmo é adaptar a margem máxima para aceitar dados multirrótulos. Uma série de classificadores lineares são utilizados para minimizar o ranking loss e permitir a utilização em casos não lineares.

Outro algoritmo adaptado para a utilização com multirrótulo foram as redes neurais. No trabalho de Zhang e Zhou (2006), os autores apresentam uma alteração do algoritmo feed-forward neural networks para aceitar dados multirrótulos. A principal modificação realizada por esse método é a alteração da função de erro.

2.3.3 Métricas de Avaliação

As métricas de avaliação para problemas simples rótulo, como acurácia, precisão, recall e F-Measure consideram apenas se um exemplo é correto ou não. Esse tipo de métrica não é adequado para os problemas multirrótulo devido a existência de acertos parciais, como no caso em que um modelo classifica corretamente 2 rótulos, mas existem 3 corretos. O trabalho de Zhang e Zhou (2014) apresenta dois grupos de métricas, as baseadas em exemplo (GHAMRAWI; MCCALLUM, 2005), (GODBOLE; SARAWAGI, 2004), (SCHAPIRE; SINGER, 2000) e as baseadas em rótulo (TSOUMAKAS; VLAHAVAS, 2007).

As métricas dos dois grupos apresentados por Zhang e Zhou são responsáveis por avaliar a qualidade do classificador. Além dessas métricas outras podem ser utilizadas para analisar a base de dados utilizada, no trabalho de Herrera et al. (2016) três métricas são apresentadas para essa tarefa. Uma dessas métricas é responsável por calcular a quantidade média de rótulos por instância. Essa métrica é chamada de cardinalidade dos rótulos (do inglês label cardinality) e está representada na equação 2.2.

$$cardinalidade(X) = \frac{1}{|X|} \sum_{i=1}^{|X|} |Y_i| \quad (2.2)$$

A cardinalidade dos rótulos está diretamente ligada a quantidade de rótulos existente no problema. A forma normalizada dessa métrica é chamada de densidade dos rótulos (do inglês label density). Essa métrica consiste da divisão da cardinalidade pela quantidade de classes do problema $|L|$, essa métrica é apresentada na equação 2.3.

$$densidade(X) = \frac{1}{|L|} \frac{1}{|X|} \sum_{i=1}^{|X|} |Y_i| \quad (2.3)$$

Outra forma de verificar se a base de dados utilizada é multirrótulo pode ser realizada com o uso da quantidade de instâncias com apenas um rótulo. A métrica P_{min} apresentada inicialmente por Turner et al. (2013), representa a proporção de instâncias simples rótulo existentes na base. Essa métrica é representada na equação 2.4.

$$P_{min}(X) = \sum_{y' \in Y / |y'|=1} \frac{|y'|}{|X|} \quad (2.4)$$

Após a verificação da base e aplicação de um método de classificação multirrótulo é possível avaliar a qualidade desse método com uma métrica de avaliação multirrótulo. As métricas baseadas em rótulo e em exemplo identificadas por Zhang e Zhou (2014) são apresentadas a seguir.

As métricas baseadas em exemplos avaliam o classificador em cada exemplo do conjunto de testes e retornam o valor médio de todas as avaliações. O Subset Accuracy (SAC), apresentado na equação 2.5, considera as predições corretas. O seu funcionamento é parecido com a métrica acurácia tradicional.

$$SAC(h) = \frac{1}{|X|} \sum_{i=1}^{|X|} \mathbb{I}[h(x_i) = Y_i] \quad (2.5)$$

A métrica Hamming Loss (HL) considera os erros do classificador, ou seja quando um rótulo correto não é predito e quando um rótulo incorreto é predito. Quanto menor o resultado obtido por essa métrica melhor é o resultado obtido pelo classificador. A métrica Hamming Loss é definida pela equação 2.6, onde Δ representa a diferença simétrica entre dois conjuntos.

$$HL(h) = \frac{1}{|X|} \frac{1}{|L|} \sum_{i=1}^{|X|} |h(x_i) \Delta Y_i| \quad (2.6)$$

As métricas Acurácia_{ml} (ACC), Precisão_{ml} (PC), Recall_{ml} (RL) e F-Measure_{ml} (FM) representam as adaptações das métricas existentes para problemas simples rótulo em problemas multirrótulo. Essas métricas são definidas nas equações 2.7, 2.8, 2.21 e 2.10, respectivamente. A métrica acurácia apresentada na equação 2.7 também é chamada de Jaccard Index em alguns trabalhos, como Read, Puurula e Bifet (2014) e Gouk, Pfahringer e Cree (2016). A métrica de avaliação F^β possui um fator de balanceamento $\beta > 0$, onde o valor mais comum é $\beta = 1$.

$$ACC_{ml}(h) = \frac{1}{|X|} \sum_{i=1}^{|X|} \frac{|h(x_i) \cap Y_i|}{|h(x_i) \cup Y_i|} \quad (2.7)$$

$$PC_{ml}(h) = \frac{1}{|X|} \sum_{i=1}^{|X|} \frac{|h(x_i) \cap Y_i|}{|h(x_i)|} \quad (2.8)$$

$$RL_{ml}(h) = \frac{1}{|X|} \sum_{i=1}^{|X|} \frac{|h(x_i) \cap Y_i|}{|Y_i|} \quad (2.9)$$

$$F_{ml}^\beta(h) = \frac{(1 + \beta^2) \cdot PC_{ml}(h) \cdot RL_{ml}(h)}{\beta^2 \cdot PC_{ml}(h) + RL_{ml}(h)} \quad (2.10)$$

Nos casos em que é possível calcular a função de valores reais $f(., .)$ quatro métricas podem ser utilizadas. Uma dessas métricas é o one-error (OE), equação 2.11, que tem como objetivo avaliar os rótulos melhores classificados que não estão no conjunto de classes relevantes do problema.

$$OE(f) = \frac{1}{|X|} \sum_{i=1}^{|X|} \mathbb{I}[\arg \max_{y \in Y} f(x_i, y)] \notin Y_i \quad (2.11)$$

A Cobertura (COV do inglês coverage), equação 2.12, avalia quantos passos em média são necessários para cobrir todos os rótulos relevantes não classificados.

$$COV(f) = \frac{1}{|X|} \sum_{i=1}^{|X|} \max_{y \in Y_i} \text{rank}_f(x_i, y) - 1 \quad (2.12)$$

A métrica Ranking Loss, equação 2.13, avalia a quantidade de rótulos classificados incorretamente.

$$\text{rankingloss}(f) = \frac{1}{|X|} \sum_{i=1}^{|X|} \frac{1}{|Y_i| |\bar{Y}_i|} |\{(y', y'') \mid f(x_i, y') \leq (x_i, y''), (y', y'') \in Y_i \times \bar{Y}_i\}| \quad (2.13)$$

A métrica Precisão média (AP do inglês average precision), equação 2.14, calcula a média de rótulos relevantes classificados como superiores a um rótulo específico $y \in Y_i$

$$AP(f) = \frac{1}{|X|} \sum_{i=1}^{|X|} \frac{1}{|Y_i|} \sum_{y \in Y_i} \frac{|\{y' \mid \text{rank}_f(x, y') \leq \text{rank}_f(x_i, y), y' \in Y_i\}|}{\text{rank}_f(x_i, y)} \quad (2.14)$$

As métricas baseadas em rótulos avaliam o classificador em cada rótulo j e retornam o valor médio de todas as avaliações realizadas no conjunto de teste. Essas métricas utilizam os valores de verdadeiro positivo (VP_j), falso positivo (FP_j), verdadeiro negativo (VN_j) e falso negativo (FN_j), apresentados nas equações 2.15, 2.16, 2.17 e 2.18, respectivamente. Com o cálculo desses valores é possível utilizar a maioria das métricas existentes para classificação simples rótulo.

$$VP_j = |x_i \mid y_j \in Y_i \wedge y_j \in h(x_i), 1 \leq i \leq |X|| \quad (2.15)$$

$$FP_j = |x_i \mid y_j \notin Y_i \wedge y_j \in h(x_i), 1 \leq i \leq |X|| \quad (2.16)$$

$$VN_j = |x_i \mid y_j \notin Y_i \wedge y_j \notin h(x_i), 1 \leq i \leq |X|| \quad (2.17)$$

$$FN_j = |x_i | y_j \in Y_i \wedge y_j \notin h(x_i), 1 \leq i \leq |X|| \quad (2.18)$$

Com base nesses valores é possível utilizar as métricas de avaliação Acurácia (ACC), Precisão (PC), Recall (RL) e F-Measure (FM), apresentados nas equações equações 2.19, 2.20, 2.21 e 2.22, respectivamente.

$$ACC(VP_j, FP_j, VN_j, FN_j) = \frac{VP_j + VN_j}{VP_j + FP_j + VN_j + FN_j} \quad (2.19)$$

$$PC(VP_j, FP_j, VN_j, FN_j) = \frac{VP_j}{VP_j + FP_j} \quad (2.20)$$

$$RL(VP_j, FP_j, VN_j, FN_j) = \frac{VP_j}{VP_j + FN_j} \quad (2.21)$$

$$F^\beta(VP_j, FP_j, VN_j, FN_j) = \frac{(1 + \beta^2) \cdot VP_j}{(1 + \beta^2) \cdot VP_j + \beta^2 \cdot FN_j + FP_j} \quad (2.22)$$

Utilizando $B(VP_j, FP_j, VN_j, FN_j)$ para representar alguma métrica de classificação binária ($B \in \{ \text{acurácia, precisão, recall, fmeasure} \}$), podemos calcular os valores de Macro-averaging, como visto na equação 2.23 e Micro-averaging, equação 2.24.

$$B_{macro}(h) = \frac{1}{|L|} \sum_{j=1}^{|L|} B(VP_j, FP_j, VN_j, FN_j) \quad (2.23)$$

$$B_{micro}(h) = B \left(\sum_{j=1}^{|L|} VP_j, \sum_{j=1}^{|L|} FP_j, \sum_{j=1}^{|L|} VN_j, \sum_{j=1}^{|L|} FN_j \right) \quad (2.24)$$

Nos casos em que é possível calcular a função de valores reais $f(.,.)$ é possível usar métricas de ranqueamento baseadas em rótulos. Entre essas métricas, temos a macro-averaged AUC, equação 2.25 e micro-averaged AUC, equação 2.26

$$AUC_{macro} = \frac{1}{|L|} \sum_{j=1}^{|L|} \frac{|\{(x', x'') | f(x', y_j) \geq f(x'', y_j), (x', x'') \in Z_j \times \bar{Z}_j\}|}{|Z_j| |\bar{Z}_j|} \quad (2.25)$$

Onde, $Z_j = \{x_i | y_j \in Y_i, 1 \leq i \leq |X|\}$ e $\bar{Z}_j = \{x_i | y_j \notin Y_i, 1 \leq i \leq |X|\}$

$$AUC_{micro} = \frac{|\{(x', x'', y', y'') | f(x', y') \geq f(x'', y''), (x', y') \in S^+, (x'', y'') \in S^-\}|}{|S^+| |S^-|} \quad (2.26)$$

Onde, S^+ corresponde aos rótulos relevantes, tal que $S^+ = \{(x_i, y) | y \in Y_i, 1 \leq i \leq |X|\}$ e S^- corresponde aos rótulos irrelevantes, tal que $S^- = \{(x_i, y) | y \notin Y_i, 1 \leq i \leq |X|\}$

3 Estado da Arte

Neste capítulo são apresentados os trabalhos que se relacionam com o escopo desta pesquisa. São apresentados os trabalhos encontrados na literatura relacionados a classificação multirrótulo para a tarefa de Mineração de Opiniões em textos utilizando Deep Learning, classificação multirrótulo para a tarefa de Mineração de Opiniões em textos utilizando outros algoritmos de aprendizagem de máquina e classificação multirrótulo utilizando Deep Learning para outras tarefas.

Para a coleta dos trabalhos foram utilizadas as bases: ACL Anthology¹, ACM Digital Library², arXiv³, IEEE Xplore⁴, ScienceDirect⁵, Scopus⁶ e SpringerLink⁷. As *strings* de busca utilizadas foram criadas com base nos termos “multi-label”, “Opinion Mining”, “text” e “Deep Learning”. A versão final da *string* utilizada para selecionar os trabalhos relacionados a classificação multirrótulo para a tarefa de Mineração de Opiniões em textos utilizando Deep Learning foi:

(“multi-label” OR “multi-output”) AND (“deep learning” OR “deep structured learning” OR “deep machine learning” OR “deep neural networks”) AND (“opinion” OR “sentiment” OR “emotion”) AND (“analysis” OR “annotation” OR “classification” OR “mining”) AND (“texts” OR “lyrics” OR “news” OR “reviews” OR “twitter” OR “tweets” OR “sina” OR “social networks”)

Para a seleção de trabalhos que utilizam classificação multirrótulo para a tarefa de Mineração de Opiniões em textos utilizando algoritmos rasos foi utilizada a *string*:

(“multi-label” OR “multi-output”) AND (“opinion” OR “sentiment” OR “emotion”) AND (“analysis” OR “annotation” OR “classification” OR “mining”) AND (“texts” OR “lyrics” OR “news” OR “reviews” OR “twitter” OR “tweets” OR “sina” OR “social networks”)

Para os trabalhos que utilizam classificação multirrótulo com Deep Learning aplicada em outras tarefas foi utilizada a *string*:

(“multi-label” OR “multi-output”) AND (“deep learning” OR “deep structured learning” OR “deep machine learning” OR “deep neural networks”)

Foram coletados artigos em inglês publicados até o dia 10 de Janeiro de 2018 que

¹ <http://aclweb.org/anthology/>

² <http://dl.acm.org/>

³ <https://arxiv.org/>

⁴ <http://ieeexplore.ieee.org/>

⁵ <http://www.sciencedirect.com/>

⁶ <https://www.scopus.com/>

⁷ <https://link.springer.com/>

contêm em seu título, resumo ou palavras chaves os termos das *strings* de busca. As seções a seguir apresentam os artigos encontrados.

3.1 Classificação Multirrotulo para Mineração de Opiniões em Textos

Wu e Ren (2009b) apresentam a utilização do algoritmo SVM em conjunto com o método de transformação de problema BR para a classificação de oito emoções em uma parte do corpus Ren-CECps (Chinese Emotion Corpus of RenLab), com 12742 sentenças extraídas de um blog chinês. Em Wu e Ren (2009a) os autores utilizam o algoritmo maximum entropy com o método BR para classificar todas as sentenças com emoção desse corpus.

O corpus Ren-CECps, apresentado por Quan e Ren (2009), é constituído de textos extraídos de blogs em chinês rotulados a nível de sentença. O corpus contém 1487 documentos, 11255 parágrafos, 35096 sentenças e 878164 caracteres em chinês. No trabalho de Wang, Ren e Miao (2013), os autores utilizam 1000 documentos dessa base para avaliar um novo método chamado de multi-label emotion topic model (MLETM). Wu et al. (2013) utilizam esse corpus para avaliar um modelo de predição baseado no algoritmo CRF. Wu e colegas também apresentam o uso dos algoritmos SVM e LGR em conjunto com o método de transformação de problema BR. O melhor resultado para todas as métricas utilizadas foi obtida com a fusão dos métodos com CRF e LGR, que gerou um micro-f1 de 51,45 e uma acurácia de 42,36.

Wang, Miao e Zhao (2015) utilizam advérbios negativos, características dos tópicos e a teoria das decisões de três vias para realizar a classificação do corpus Ren-CECps. O método possibilitou um macro-accuracy de 75,2 e um all-match-accuracy de 11,3. Esses resultados foram superiores ao algoritmo Naive Bayes com o método de transformação de problema BR em relação ao all-match-accuracy e inferiores em relação ao macro-accuracy. O trabalho de Wang et al. (2016) apresenta uma estratégia utilizando um algoritmo de Deep Learning para a classificação multirrotulo de emoções. No trabalho, os autores apresentam um método utilizando CNN para classificar textos de microblog do corpus EACWT e o corpus Ren-CECps. A estratégia utilizada por Wang e colegas foi transformar o problema multirrotulo em um problema de ranqueamento, utilizando para isso o algoritmo CLR. Essa estratégia possibilitou um micro-f1 de 28,89 para o corpus Ren-CECps e 61,23 para o corpus EACWT.

O corpus EACWT (Emotion Analysis in Chinese Weibo Texts) foi desenvolvido para a conferência NLPCC 2014⁸ (*The conference on Natural Language Processing and*

⁸ http://tcci.ccf.org.cn/conference/2014/pages/page04_eva.html

Chinese Computing). Esse corpus é composto por 10139 sentenças extraídas de um microblog chinês rotuladas com sete classes de emoção. Todas as sentenças possuem um ou dois rótulos, sendo um majoritário e um minoritário. Assim como Wang e colegas, o trabalho de Xu et al. (2015) apresenta o uso desse corpus com um algoritmo de Deep Learning. Nesse trabalho, os autores apresentam uma abordagem para o balanceamento do corpus com base na matriz de embedding. O artigo propõe um método com o algoritmo RNTN (Recursive Neural Tensor Network) para a criação da matriz de embedding utilizada na classificação. O algoritmo SMOTE é aplicado nessa matriz para produzir uma base de treinamento totalmente balanceada. Após o balanceamento o algoritmo ML-KNN é aplicado sobre essa base, gerando uma precisão média de 52.

Wang et al. (2014) utilizam o corpus EACWT para avaliar um novo método baseado na técnica de transformação de problema CLR. Os autores utilizam os algoritmos Naive Bayes, SVM, léxicos e as combinações entre eles. A melhor configuração testada foi obtida com o uso de todos os algoritmos simultaneamente. Essa configuração possibilitou uma precisão média de 53,30. Outra abordagem com a utilização de léxicos foi apresentada por Song et al. (2015), onde emoticons foram combinados com palavras para a criação desse recurso. Os autores utilizaram um grafo de três camadas para representar emoticons, palavras sementes, palavras candidatas e as suas ligações. Um algoritmo de caminhada aleatória multirrótulo foi utilizado para encontrar as melhores combinações. O léxico gerado possibilitou uma precisão média de 57,9 para o corpus EACWT.

Outro corpus construído a partir de textos de blogs em chinês foi desenvolvido por Quan e Ren (2010). A principal diferença entre esse corpus e os outros existentes na literatura é rotulação em três níveis: documento, parágrafo e sentença. A Figura 15 ilustra a estrutura hierárquica formada por esses três níveis. Além das 8 classes de emoções utilizadas, esse corpus está anotado com expressões linguísticas detalhadas. O corpus de Quan e Ren contém 500 documentos, 4004 parágrafos, 12742 sentenças e 324571 caracteres chineses. Após a publicação, os autores continuaram o processo de anotação, gerando uma nova versão do corpus com 1487 documentos, 11953 parágrafos, 38051 sentenças e 971628 caracteres chineses. O trabalho de Wang et al. (2015) utiliza esse corpus para avaliar um novo método de classificação baseado em *Fuzzy-Rough Set*, que possibilitou um subset accuracy de 99,76 para documento, 98,61 para parágrafo e 96,38 para sentença.

Um corpus em chinês também foi utilizado no trabalho de Liu e Chen (2015), onde é apresentado um estudo sobre métodos de transformação de problema para classificação multirrótulo de emoções em textos de microblogs. O método apresentado nesse trabalho possui três componentes principais, a segmentação do texto, a extração de características e a classificação. No trabalho, os autores apresentam uma comparação com o uso de 11 métodos de transformação de problema e três algoritmos de classificação. Liu e Chen utilizam os métodos BR, CC, CLR, HOMER, RAkEL, ECC com os algoritmos PCT,

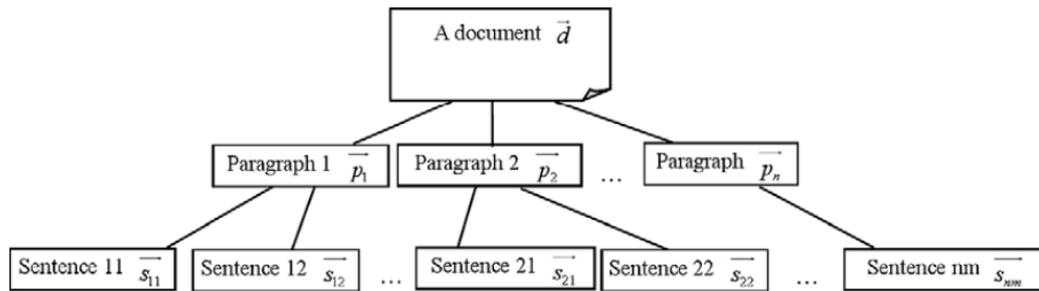


Figura 15 – Modelo hierárquico do corpus desenvolvido por Quan e Ren (Extraído de: (QUAN; REN, 2010))

SVM e KNN e os algoritmos adaptados BRkNN, BRkNN-a, BRkNN-b, ML-KNN, RF-PCT. As duas bases utilizadas nos experimentos foram construídas com postagens em chinês do microblog Sina, sendo uma delas rotulada com 5 emoções e outra com 10. Os melhores desempenhos foram obtidos pelos métodos RAKE e ECC para a base menor e CLR, HOMER e ECC para a base maior.

Zhang, Li e Lu (2017) apresentam uma comparação dos algoritmos DT, SVM e LR para o método de transformação de problema BR. Com o intuito de realizar a comparação dos algoritmos os autores utilizam uma base composta por aproximadamente 840 tweets em inglês rotulados com as 6 emoções definidas por Ekman. O trabalho de Khan, Khan e Khan (2016) também apresenta uma abordagem utilizando transformação de problema, mas utiliza uma base rotulada a nível de aspecto para avaliação do método. Outros trabalhos, como Huang et al. (2013a), Pham et al. (2013) e Yamamoto e Satoh (2013), Lim e Buntine (2014) e Ruder, Ghaffari e Breslin (2016) utilizam o mesmo tipo de base para a classificação.

Em Ye, Xu e Xu (2012), os autores exploram a utilização das abordagens de transformação de problema BR e RAKE e adaptação de algoritmo ML-KNN para classificar emoções em textos extraídos de um site de notícias em chinês pertencente ao grupo Sina. Além de Ye e colegas outros autores utilizam abordagens de adaptação de algoritmo para essa tarefa, como em Luyckx et al. (2012), onde os autores utilizam apenas instâncias simples rótulo com o algoritmo SVM para gerar probabilidades e classificar múltiplas emoções em notas de suicídio do corpus apresentado por Pestian et al. (2012). Em seu trabalho, Pestian e colegas apresentam os resultados de outros métodos participantes de uma competição com esse corpus. Outro trabalho empregando um algoritmo adaptado para problemas multirrótulo é apresentado por Li et al. (2016a), onde notícias, tweets e questões em inglês são classificados utilizando o algoritmo Maximum Entropy. Em Li et al. (2016b), os autores apresentam a classificação de notícias em inglês e textos do microblog sina em chinês utilizando o novo método proposto chamado weighted multi-label classification model (WMCM). No trabalho de Wang, Ren e Miao (2016) uma adaptação do algoritmo Naive Bayes é utilizada para classificação de emoções em textos extraídos

de blogs da china.

O trabalho de Herzig et al. (2016) apresenta a utilização do método de transformação de problema BR em conjunto com o algoritmo SVM para a identificação de sentimentos em diálogos entre consumidores e prestadores de serviços em redes sociais. Outro trabalho utilizando diálogos é apresentado por Phan, Shindo e Matsumoto (2016), onde emoções em diálogos de filmes são identificadas por meio de um algoritmo de Deep Learning com a ajuda de léxicos. Phan e colegas apresentam a criação de um novo léxico e a adaptação do mesmo para o corpus utilizado. O novo léxico é utilizado para extrair as características da base e gerar a entrada da rede neural utilizada. A classificação multirótulo é realizada por meio de uma alteração na última camada da rede, onde um limiar é aplicado para selecionar os rótulos corretos a partir da probabilidade gerada.

Bobicev e Sokolova (2017) também utilizam léxicos para auxiliar um método de aprendizagem de máquina. Os autores utilizam 11 léxicos em conjunto com os métodos de transformação de problema BR e CC com os algoritmos SVM e Naive Bayes. O corpus utilizado para as comparações entre os métodos foi coletado de mensagens postadas em um fórum sobre saúde e é rotulado com 4 classes.

Todos os trabalhos apresentados realizam a classificação de textos em Inglês ou Chinês. Embora o Inglês e o Chinês sejam os idiomas mais utilizados, alguns trabalhos apresentam o uso de corpora compostos por outros idiomas. Duan et al. (2014) apresenta o relato da utilização de *crowdsourcing* para a anotação de duas histórias infantis em Japonês. O mesmo método foi utilizado para anotar 1398 tweets relacionados a um festival japonês. Além da base os autores apresentam duas técnicas baseadas nos métodos de transformação de problema BR e LP com o algoritmo Naive Bayes. No trabalho Duan et al. (2015) os dois corpora de histórias infantis são utilizados para avaliar uma versão otimizada do método proposto no primeiro artigo.

Outro idioma utilizado foi o holandês, no trabalho Zwaan et al. (2015). Nesse trabalho, os autores apresentam uma nova base com 29 textos de peças teatrais rotuladas manualmente com os rótulos de emoção criados para um modelo de estudo entre partes do corpo e expressões emocionais em textos dos séculos XVII e XVIII. Os experimentos realizados pelos autores se baseiam na comparação entre os métodos de transformação de problema BR e RAKE com o algoritmo SVM. O método RAKE obteve um melhor resultado na maior parte dos experimentos realizados.

Uma visão geral das abordagens utilizadas nos trabalhos apresentados nesta seção pode ser vista na Tabela 7. Dentre os trabalhos analisados 15 utilizam um método de transformação de problema, 12 um método de adaptação de algoritmo e 3 recursos léxicos. Embora alguns trabalhos utilizem um algoritmo de Deep Learning para o pré-processamento apenas dois trabalhos encontrados apresentam a utilização desses algoritmos para a tarefa de classificação. Em contraparte, o algoritmo SVM e o método de

transformação de problema BR foram os mais utilizados entre os trabalhos analisados.

Tabela 7 – Abordagens e algoritmos de classificação utilizados nos trabalhos encontrados na revisão da literatura

Artigo	Abordagem	Algoritmos
(BHOWMICK et al., 2009)	AA	ML-KNN
(BHOWMICK; BASU; MITRA, 2009)	TP (RA k EL)	C 4.5
(WU; REN, 2009a)	TP (Binary Pruning)	Maximum Entropy
(WU; REN, 2009b)	TP (BR)	SVM
(LUYCKX et al., 2012)	AA	SVM
(YE; XU; XU, 2012)	AA e TP (BR e RA k EL)	SVM, NB e ML-KNN
(WANG; REN; MIAO, 2013)	AA	MLETM
(WU et al., 2013)	AA e TP (BR)	CRF, SVM e LGR
(DUAN et al., 2014)	TP (novo)	NB
(WANG et al., 2014)	TP (CLR) e Léxico	NB, SVM
(DUAN et al., 2015)	TP (novo)	NB
(LIU; CHEN, 2015)	AA e TP (diversos)	PCT, SVM e KNN
(SONG et al., 2015)	Léxico	-
(ZWAAN et al., 2015)	TP (BR, RA k EL)	SVM
(WANG; MIAO; ZHAO, 2015)	TP (BR, TWD)	NB, LDA
(WANG et al., 2015)	AA	Fuzzy-Rough
(XU et al., 2015)	AA	ML-KNN
(HERZIG et al., 2016)	TP (BR)	SVM
(LI et al., 2016a)	AA	Maximum Entropy
(LI et al., 2016b)	AA	WMCM
(PHAN; SHINDO; MATSUMOTO, 2016)	AA e Léxico	DNN
(WANG; REN; MIAO, 2016)	AA	NB
(WANG et al., 2016)	TP (CLR)	CNN
(BOBICEV; SOKOLOVA, 2017)	TP (BR, CC)	SVM, NB
(ZHANG; LI; LU, 2017)	TP (BR)	DT, SVM, LR

A Tabela 8 apresenta informações sobre os corpora e as métricas utilizadas nos artigos analisados. A acurácia foi a métrica utilizada com mais frequência, sendo empregada em 10 dos 25 trabalhos analisados. A métrica precisão média foi utilizada em 9 trabalhos e as métricas F1, F1_{micro} e precisão em 8 trabalhos. As características dos corpora utilizados nos trabalhos são apresentadas na Tabela 9.

Como é possível observar na Tabela 9, os principais idiomas utilizados para a classificação multirrótulo de emoções são o Inglês e o Chinês. Esses idiomas também são os mais utilizados para a classificação simples rótulo, como visto no trabalho de Liu (2012). Dos 19 corpora encontrados na literatura 10 são compostos por textos em inglês e 6 chinês. Em relação ao nível de análise, 11 corpora são rotulados a nível de documento e 7 a nível de sentença. Apenas o corpus apresentado por Quan e Ren (2010) é rotulado nos dois níveis e em parágrafo. A origem mais comum de dados são os microblogs, como o Twitter e o Sina. Todos os trabalhos encontrados utilizaram textos rotulados com classes de emoções. A maioria dos trabalhos utiliza bases com mais de 6 classes de emoção para a rotulação.

Tabela 8 – Corpora e métricas de avaliação utilizados nos trabalhos encontrados na revisão da literatura

Artigo	Corpus utilizado	Métricas utilizadas
(BHOWMICK et al., 2009)	Times of India	AP, COV, F1, HL, OE, P-ACC, SAC
(BHOWMICK; BASU; MITRA, 2009)	Times of India	AP, COV, F1, HL, OE, P-ACC, RL, SAC
(WU; REN, 2009a)	Ren-CECps (parcial)	ACC, $F1_{macro}$, $F1_{micro}$, PC, RL
(WU; REN, 2009b)	Ren-CECps (parcial)	$F1_{macro}$, $F1_{micro}$, SAC
(LUYCKX et al., 2012)	Medical NLP Challenge	ACC, $F1_{micro}$
(YE; XU; XU, 2012)	Sina News	AP, COV, HL, OE, RL
(WANG; REN; MIAO, 2013)	Ren-CECps (parcial)	ACC_{macro} , OH-ACC
(WU et al., 2013)	Ren-CECps (parcial)	ACC, $F1_{macro}$, $F1_{micro}$, HL, PC, RL
(DUAN et al., 2014)	Little Masa and a Red Apple, Although we are in love	ACC, ASM
(WANG et al., 2014)	EACWT	AP
(DUAN et al., 2015)	Little Masa and a Red Apple, Although we are in love	ASM
(LIU; CHEN, 2015)	Huangpu River Incident, Influenza A	AP, COV, F1, $F1_{macro}$, $F1_{micro}$, HL, OE, SAC
(SONG et al., 2015)	EACWT	AP, F1, $f1_{macro}$, PC, PC_{macro} , RL, RL_{macro}
(ZWAAN et al., 2015)	Dutch theater texts	F1, $F1_{macro}$, $F1_{micro}$
(WANG; MIAO; ZHAO, 2015)	Ren-CECps (parcial)	AM-ACC, ACC_{macro} , OM-ACC, TM-ACC
(WANG et al., 2015)	Quan e Ren (2010)	AP, HL, OE, SAC
(XU et al., 2015)	EACWT	AP
(HERZIG et al., 2016)	General Customer Care, Technical Customer Support	F1, PC, RL
(LI et al., 2016a)	Semeval, SSTweet, ISEAR	$F1_{micro}$
(LI et al., 2016b)	Semeval, Sina News (parcial)	F1
(PHAN; SHINDO; MATSUMOTO, 2016)	Movie Dialog Corpus	F1, HS, PC, RL
(WANG; REN; MIAO, 2016)	Ren-CECps	ACC, ACC_{macro} , AM-PC, OM-PC, PC_{macro} , TM-PC
(WANG et al., 2016)	Ren-CECps (parcial), EACWT	AP, HL, $F1_{macro}$, $F1_{micro}$, OE, RL, SAC
(BOBICEV; SOKOLOVA, 2017)	InVitroFertilization Forum	EM, F1
(ZHANG; LI; LU, 2017)	Twitter Social Relationships	ACC, F1, PC, RL

3.2 Classificação Multirrótulo com Deep Learning

Diversos trabalhos existentes na literatura registram bons resultados com a utilização de algoritmos de Deep Learning em dados simples rótulo. Várias áreas foram exploradas por esses trabalhos, como o reconhecimento de imagens (HE et al., 2016; SHEN; WU; SUK, 2017), reconhecimento de discurso (DENG; HINTON; KINGSBURY, 2013; ZHANG et al., 2017), identificação de opiniões em textos (IRSOY; CARDIE, 2014; TANG; QIN; LIU, 2015), tradução de textos (SUTSKEVER; VINYALS; LE, 2014; JOHNSON et al., 2017) e criação de legendas para imagens (CHEN; ZITNICK, 2015; HE; DENG, 2017). Em diversos domínios existe mais de um rótulo associado a uma única instância, tornando necessário a alteração do problema ou do algoritmo de classificação utilizado. Os algoritmos de Deep Learning possuem algumas limitações quando utilizados em conjunto com métodos de transformação de problema, principalmente devido ao alto poder computacional necessário para a criação do modelo. O restante desta seção apresenta informações sobre os trabalhos encontrados durante a revisão da literatura que utilizam Deep Learning em bases multirrótulo.

Como visto na seção anterior, uma das formas para solucionar problemas de classificação multirrótulo é a utilização de um método de transformação de problema com um algoritmo de classificação simples rótulo. Um desafio gerado pelos algoritmos de Deep Learning é a demanda de um alto poder computacional para realizar o treinamento da

Tabela 9 – Características dos corpora utilizados nos artigos encontrados na revisão da literatura

Nome do Corpus	# Artigos	Origem	Idioma	# Classes	# Instâncias	Nível de Análise
Ren-CECps	6	Blogs	Chinês	8	35096	Sentença
EACWT	4	Microblog	Chinês	7	19493	Sentença
Although we are in love	2	Histórias infantis	Japonês	10	63	Sentença
Little Masa and a Red Apple	2	Histórias infantis	Japonês	10	78	Sentença
Semeval	2	Manchetes de Notícias	Inglês	6	1246	Documento
Sina News	2	Notícias	Chinês	8	8802	Documento
Times of India	2	Notícias	Inglês	4	1305	Sentença
Dutch theater texts	1	Peças de teatro	Holandês	38	27993	Sentença
General Customer Care	1	Microblog	Inglês	9	4243	Documento
Huangpu River Incident	1	Microblog	Chinês	5	385	Documento
Influenza A	1	Microblog	Chinês	10	762	Documento
InVitroFertilization Forum	1	Fórum	Inglês	4	1321	Documento
ISEAR	1	Questões	Inglês	7	7666	Sentença
Medical NLP Challenge	1	Notas de Suicídio	Inglês	15	600	Documento
Movie Dialog Corpus	1	Diálogos de filmes	Inglês	8	11610	Documento
Quan e Ren (2010)	1	Blogs	Chinês	8	1487, 11953, 38051	Documento, Parágrafo, Sentença
SSTweet	1	Microblog	Inglês	2	4242	Documento
Technical Customer Support	1	Microblog	Inglês	9	4016	Documento
Twitter Social Relationships	1	Microblog	Inglês	6	16424	Documento

rede. Essa demanda causa uma limitação em relação a algumas estratégias tradicionais, onde geralmente algoritmos simples são utilizados. Para resolver esse problema Yu et al. (2017), Yang et al. (2017) e Wei et al. (2016) apresentam uma abordagem onde dividem os dados de entrada do problema. Esses trabalhos realizam a divisão de uma imagem em hipóteses contendo pedaços da imagem original com objetos. Os objetos das hipóteses são classificados de forma simples rótulo por um classificador treinado com uma base separada e os resultados são reunidos. A Figura 16 apresenta o funcionamento do método proposto por Wei et al. (2016).

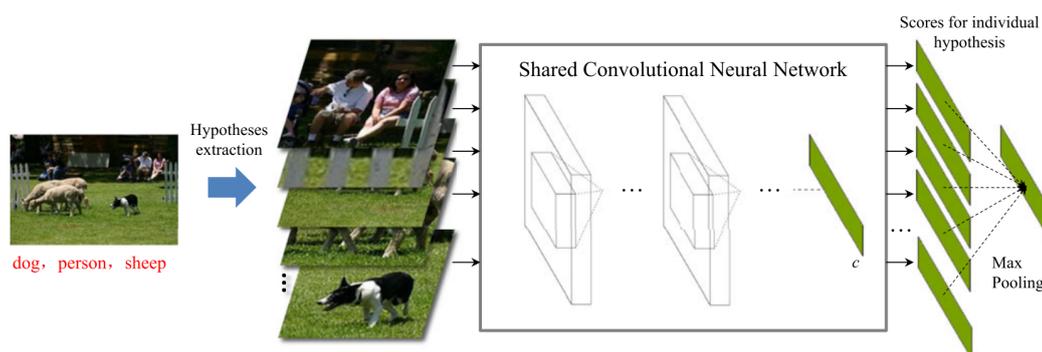


Figura 16 – Método utilizando divisão de imagens (Extraído de: (WEI et al., 2016))

Kinghorn, Zhang e Shao (2018) apresentam um método parecido com o retratado pelos trabalhos utilizando imagens citados anteriormente. Após a detecção de objetos e criação de sub-imagens os autores utilizam os algoritmos CNN e RNN para a criação de legendas. O método apresentado por Kinghorn e colegas é dividido em quatro etapas, inicialmente é realizada a detecção e reconhecimento dos objetos presentes na imagem, em seguida um classificador é aplicado para identificar cada objeto detectado. Após classificar os objetos a cena é detectada e por fim uma legenda é criada para a imagem.

Outra estratégia para classificar imagens com múltiplos rótulos foi apresentada por Zhao et al. (2015). Nesse trabalho os autores apresentam um método onde uma CNN é incorporada em funções de hash para aprender representações de características e mapeamentos a partir de códigos de hash. Enquanto esse processo é realizado uma lista de classificação codifica as informações de similaridade de vários níveis e é utilizada para orientar o aprendizado das funções de hash. Assim como o modelo de divisão das imagens em hipóteses esse processo permite a utilização de apenas uma rede evitando multiplicar o tempo de treinamento.

Karalas et al. (2015) apresentam um método usando sparse autoencoders para realizar a classificação de imagens obtidas por satélites. Outra estratégia para classificação de imagens foi a apresentada em Huang et al. (2013b), onde os autores transformam o problema multirrótulo em diversos problemas binários com o algoritmo BR e utilizam o algoritmo RBM para a classificação. No trabalho de Wang et al. (2016), os autores

utilizam uma combinação das redes CNN e RNN para possibilitar a criação de uma matriz de embedding formada por conjuntos de imagens e rótulos. Diferente de Wanag e colegas, o trabalho de Lenc e Král (2017) apresenta a utilização de uma base formada apenas por textos. Os autores utilizam um corpus de notícias em Checo rotulado em categorias para avaliar o método de transformação de problema BR com os algoritmos multi-layer perceptron e CNN. Em Lin et al. (2017) o método de transformação de problema BR é utilizado para classificar categorias de reviews multirrótulo. O algoritmo utilizado pelos autores é baseado em dois algoritmos LSTMs e um CNN.

No trabalho de Read e Perez-Cruz (2014) os autores apresentam um método utilizando RBM para a classificação de dados obtidos de diversos domínios diferentes. No trabalho, os autores utilizam áudio, imagem, dados biológicos e textos. Assim como Read e Perez-Cruz outros autores exploraram o uso do Deep Learning para a classificação multirrótulo de dados de diversas origens. Guan et al. (2018) apresenta um método chamado Stacked Denoising Autoencoder Multi-Label Learning (SdaMLL) para classificação de funções de genes. Cakir et al. (2015) apresentam um método de adaptação de algoritmo para realizar a classificação multirrótulo de eventos sonoros. Outra utilização de áudio foi apresentada no trabalho de Kukanov et al. (2016), onde os autores exploram a utilização de reconhecimento de fala. Outra possível aplicação desses métodos são os dados médicos, como em Gao et al. (2017), onde os autores utilizam uma CNN para detectar doenças pulmonares e em Fitriawan et al. (2016), onde é utilizado um DBN para a descoberta de novas drogas.

Em Lenc e Král (2016) os autores apresentam um método utilizando CNN para a classificação automática de documentos. Assim como o método para classificar emoções apresentado por Wang et al. (2016), esse trabalho apresenta uma vasta aplicação comercial devido a grande quantidade de informações textuais geradas diariamente por usuários e devido a grande quantidade de dados armazenados. Visando ampliar o estudo nessa área, este trabalho apresenta um novo método de transformação de problema adaptado para as características dos algoritmos de Deep Learning, assim como dois novos corpora multirrótulo para a avaliação desse método. O novo método proposto neste trabalho se baseia na divisão dos dados para a classificação, como no trabalho de Wei et al. (2016) e na divisão das classes, como no método de transformação de problema *RA_kEL*. Esse método se diferencia dos demais métodos para classificação multirrótulo de textos existentes na literatura por necessitar de um número reduzido de classificadores, demandando menos recursos computacionais e menos tempo para a criação dos classificadores. O Capítulo a seguir apresenta as etapas utilizadas no processo de criação desse novo método.

4 Procedimentos Metodológicos

Este capítulo apresenta a abordagem metodológica utilizada no desenvolvimento desta pesquisa. A estrutura da pesquisa é dividida em quatro fases: Planejamento inicial, Fase exploratória, Desenvolvimento e Avaliação. A Figura 17 apresenta uma visão geral dessas fases e suas tarefas.

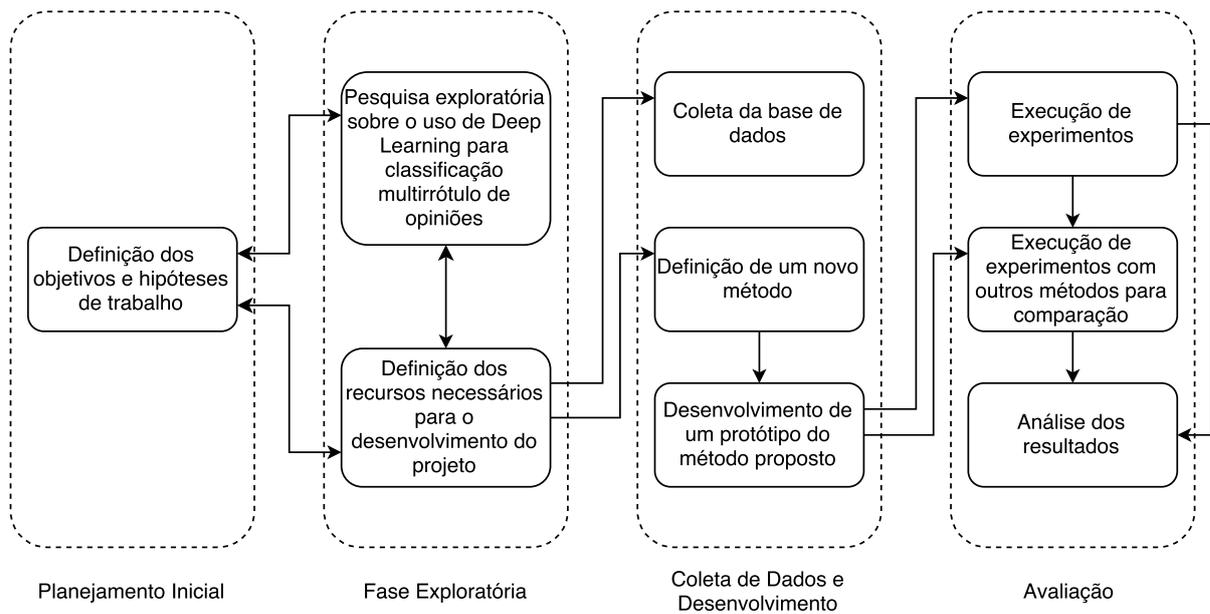


Figura 17 – Estrutura da pesquisa

As seções a seguir apresentam as fases e as tarefas do método de pesquisa utilizado.

4.1 Planejamento Inicial

Visando dar continuidade à pesquisa iniciada por Martinazzo (2010), Dosciatti (2015) e Reis (2017) propôs-se a aplicação de um algoritmo de Deep Learning para a classificação de opiniões em textos. A motivação para a escolha desse grupo de algoritmos foi o grande número de trabalhos relatando bons resultados para diversas aplicações, como apresentado em LeCun, Bengio e Hinton (2015) e Tang, Qin e Liu (2015). Após a realização de uma revisão com os trabalhos existentes na literatura que apresentam a tarefa de Mineração de Opiniões com Deep Learning, observou-se a pequena quantidade de trabalhos utilizando esses algoritmos para a classificação multirrotulo de opiniões em textos. Visando obter os bons resultados relatados na literatura, foi definido como objetivo geral deste trabalho desenvolver um método utilizando Deep Learning para a classificação

multirrótulo de opiniões em textos. Nessa fase, também procuramos identificar as melhores aplicações dos algoritmos de Deep Learning existentes na literatura e qual seria o mais adequado para a tarefa apresentada neste trabalho, onde optamos pela utilização do LSTM. Em paralelo a fase exploratória definimos os objetivos específicos da pesquisa.

O primeiro desafio encontrado durante a fase exploratória foi a inexistência de uma base de dados pública grande o suficiente para possibilitar a utilização de forma eficiente dos algoritmos de Deep Learning. Com isso, foi definido como o primeiro objetivo específico a construção de um corpus de textos multirrótulos anotado com reações para o treinamento e avaliação do método proposto.

O segundo objetivo específico desta pesquisa é o desenvolvimento de um método de transformação de problema para classificação multirrótulo de textos. A criação de um novo método é proposta devido a algumas limitações relacionadas a quantidade de classificadores utilizados. O método de transformação de problema apresentado neste trabalho tem como principal característica uma quantidade menor de classificadores que os métodos tradicionais, evitando assim multiplicar o número de classificadores pelo número de classes.

Para realizar a comparação do método proposto com os métodos existentes na literatura definimos como terceiro objetivo específico a realização de um levantamento sobre os principais métodos de transformação de problema para classificação multirrótulo existentes na literatura. Esse levantamento é necessário para permitir a criação de um protocolo de testes capaz de viabilizar a avaliação adequada do método proposto.

Por fim, o último objetivo específico deste trabalho é o desenvolvimento de um protótipo computacional para a realização de testes e avaliação do método proposto. A seção a seguir apresenta as tarefas realizadas na fase exploratória desta pesquisa.

4.2 Fase Exploratória

A primeira tarefa foi a realização de uma pesquisa exploratória sobre o uso de Deep Learning para classificação multirrótulo de opiniões. Essa etapa foi necessária para proporcionar a compreensão do campo de estudo, possibilitando assim a definição dos objetivos de pesquisa e a criação do método.

A pesquisa exploratória foi realizada na forma de uma revisão da literatura. Nesta revisão foram explorados os trabalhos relacionados ao uso de Deep Learning para Mineração de Opiniões em textos, classificação multirrótulo para a tarefa de Mineração de Opiniões em textos utilizando Deep Learning, classificação multirrótulo para a tarefa de Mineração de Opiniões em textos utilizando algoritmos rasos e classificação multirrótulo utilizando Deep Learning para outras tarefas. No Capítulo 3 são apresentados detalhes

da realização dessa pesquisa e uma descrição dos artigos encontrados.

Após a coleta e leitura dos artigos foi possível realizar a segunda tarefa da fase exploratória, onde foi realizada a definição dos recursos necessários para o desenvolvimento do projeto. Nessa etapa foram verificados quais recursos deveriam ser construídos na fase de Coleta de Dados e Desenvolvimento, apresentada na próxima seção.

4.3 Coleta de Dados e Desenvolvimento

A fase de Coleta de Dados e Desenvolvimento é dividida em três tarefas: a pesquisa e coleta da base de dados, o desenvolvimento de um novo método e o desenvolvimento de um protótipo do novo método.

A necessidade da coleta de um novo corpus foi definida durante a fase exploratória. Nessa fase também foram definidas quais características a nova base deve possuir para a correta avaliação do método proposto. Embora nenhum trabalho tenha sido encontrado na literatura apresentando qual o tamanho ideal da base de dados necessária, alguns trabalhos com bons resultados como Kim (2014) e Santos e Gatti (2014) possibilitaram uma melhor compreensão da área. Após a análise das bases utilizadas foi possível definir que o uso de ao menos 10 mil exemplos permite a utilização desses algoritmos, mesmo sendo considerado uma base pequena. Melhores resultados podem ser obtidos com bases maiores, como 50 mil exemplos. Mesmo que muitos trabalhos apresentem bons resultados com bases desse tamanho, o trabalho de Zhang e LeCun (2015) afirma que essas bases são relativamente pequenas e apresenta testes com bases de até 20 milhões de exemplos. Após a definição da quantidade de exemplos necessária foi realizada uma busca por sites de notícias e redes sociais com textos já rotulados pelo usuário. Após a realização de uma pesquisa empírica pelo melhor site para a coleta, um *crawler* foi desenvolvido para a montagem da base. Informações sobre as bases coletadas são apresentadas na Seção 6.1.

Na etapa de desenvolvimento de um novo método foram estudados os métodos de transformação de problema existentes na literatura e quais características um método deveria possuir para a utilização de forma eficiente com algoritmos de Deep Learning. Após esse estudo foram definidas as etapas do método, apresentado com mais detalhes no Capítulo 5.

Com o método definido iniciou-se a montagem de um protótipo com a utilização das ferramentas mais citadas em relatos da literatura. Após a construção do método diversos testes empíricos foram realizados para definir as melhores configurações do método e do algoritmo utilizado. Esses testes foram realizados na fase de avaliação, apresentada na seção a seguir.

4.4 Avaliação

Após a construção do protótipo diversos testes foram realizados para delimitar a melhor configuração do método e da base construída. Adicionalmente a esses testes, experimentos com os métodos do estado da arte foram realizados para possibilitar comparações com o método proposto. A primeira etapa é referente a eliminação das notícias sem informações textuais. Como o novo corpus foi coletado de um site de entretenimento diversas notícias apresentam apenas imagens ou vídeos, sendo o único texto a origem da mídia. Essas notícias foram removidas pois não apresentam textos referentes ao conteúdo, impossibilitando a classificação. Para a remoção dessas instâncias, todas as notícias com 3 palavras ou menos foram eliminadas automaticamente e uma verificação manual foi realizada nas notícias restantes. A escolha de 3 palavras foi realizada devido a característica do site de coleta manter, em algumas notícias, apenas o texto referente a fonte da imagem.

Com o fim da etapa de limpeza das instâncias, o corpus foi dividido utilizando o método validação cruzada com 3 folds. Esse método de avaliação foi escolhido por ser utilizado em vários trabalhos apresentados na literatura e recomendado em diversos estudos, como (KOHAVI, 1995), (KIM, 2009) e (BORRA; CIACCIO, 2010). Como visto em Refaeilzadeh, Tang e Liu (2009), a ideia básica do cross-validation é dividir os dados em dois segmentos, um para treinamento do modelo e um para a validação, de tal forma que os dados presentes em um segmento não estejam presentes no outro. No cross-validation tradicional os dados de treinamento e validação são cruzados k vezes, permitindo que cada instancia seja avaliada uma vez. A escolha de 3 folds foi realizada para possibilitar uma melhor avaliação das bases desbalanceadas. Segundo o trabalho apresentado por Batista, Carvalho e Monard (2000), o uso de poucos folds possibilita uma base de teste maior, garantindo que vários exemplos das classes minoritárias estejam presentes nessa base. Outra motivação para a escolha de 3 divisões é a quantidade de testes realizados. Zeng e Martinez (2000) recomendam o uso de menos folds em muitos casos onde algoritmos que demandam um alto tempo de execução são utilizados.

Com a criação das divisões da base, um teste empírico foi realizado para a obtenção da melhor configuração do algoritmo utilizado e da operação de embedding, onde um vetor de números reais é criado para representar as palavras. Como neste trabalho foi utilizado o algoritmo LSTM, foram definidos os melhores valores para a quantidade de hidden layers, a quantidade de células de memória e o número de épocas. Para a operação de embedding foram definidos o tamanho dos vetores utilizados e o tamanho máximo da entrada. A escolha desses valores foi realizada através de busca por força bruta.

Para realizar a comparação dos resultados obtidos com as principais técnicas existentes na literatura foi utilizado um protocolo de testes parecido com o apresentado por Liu e Chen (2015). Neste trabalho utilizamos os métodos de transformação de problema BR, CC, CLR, HOMER, LP e RAKEL com o algoritmo SVM e o método de adaptação

de algoritmo ML-KNN com o algoritmo KNN. Os algoritmos HOMER e RAKEL foram executados com todas as configurações disponibilizadas, para todos os outros algoritmos a configuração padrão foi utilizada. Esses testes foram realizados com o software Meka¹ (READ et al., 2016) em conjunto com o software mulan² (TSOUMAKAS et al., 2011). A utilização desse protocolo permite responder a hipótese de trabalho H2. Para possibilitar a validação da hipótese H3 outro teste foi realizado utilizando o método de transformação de problema BR com o algoritmo LSTM e o método proposto com o algoritmo SVM.

O pré-processamento utilizado para os testes com o algoritmo SVM foi baseado no realizado por Dosciatti (2015). Inicialmente todos os caracteres foram convertidos para o minúsculo e os caracteres especiais foram removidos. Todas as palavras presentes na lista de stopwords fornecida pelo sistema SnollBall³ foram eliminadas. Um lematizador, também fornecido pelo sistema snowball, foi aplicado para extrair os radicais das palavras. Todos os links, emails, números, símbolos monetários e porcentagens foram substituídos por tokens. Por fim, o método TF-IDF (sigla para frequência do termo - inverso da frequência no documento, derivado do inglês *term frequency - inverse document frequency*) foi aplicado para representar as palavras de forma vetorial.

As métricas de avaliação utilizadas para analisar os métodos são: hamming loss, micro F1 e jaccard index. A escolha dessas métricas foi realizada devido a frequência com que são utilizadas nos trabalhos encontrados na literatura e as suas características, que permitem avaliar as hipóteses definidas. Como apresentado por Herrera et al. (2016), a métrica hamming loss é um indicador dos erros cometidos pelo classificador em relação ao total de classes existentes no problema. Essa métrica permite avaliar o quanto o classificador errou indiferente da classe. Diferente do hamming loss, o jaccard index calcula a quantidade de acertos do classificador. Em problemas multirrótulo, o jaccard index é definido como a proporção entre o número de rótulos previstos corretamente e a soma do total de rótulos previstos e existentes no problema. A métrica micro F1 para bases multirrótulo é uma média harmônica entre precisão e o recall. Assim como a métrica para problemas simples rótulo o F1 para multirrótulo se apresenta eficiente para a avaliação de bases desbalanceadas, como é o caso das bases apresentadas neste trabalho.

Para a implementação do método proposto neste trabalho foi utilizada a linguagem python 3.6 com o framework TensorFlow⁴ e a biblioteca scikit-learn⁵. A execução dos testes com os softwares Meka e mulan foi realizada em um desktop com processador Intel Core I7 4770 com 3,4GHz e 16GB de memória RAM com 1600MHz. Os testes com o método proposto foram realizados em um servidor com processador Intel Xeon E5-

¹ <http://meka.sourceforge.net/>

² <http://mulan.sourceforge.net/>

³ <http://snowballstem.org/>

⁴ <https://www.tensorflow.org/>

⁵ <http://scikit-learn.org/stable/>

1620 v4 com 3,5GHz, 16GB de memória RAM com 2400MHz em conjunto com 150GB virtualizados em um HD com velocidade de 7200RPM. A GPU utilizada nos testes foi uma NVIDIA GeForce GTX TITAN X com 12GB de memória e 3072 CUDA Cores. O Sistema Operacional utilizado em ambos os computadores foi o Windows 10. Os testes com o método proposto foram realizados com a versão 1.1 do TensorFlow, disponibilizada pelo Anaconda, e com a com o CUDA na versão 8.0.60. Para as implementações em python foi utilizada a IDE (sigla para Ambiente de Desenvolvimento Integrado, derivado do inglês *Integrated Development Environment*) PyCharm⁶, as implementações em Java foram realizadas na IDE Eclipse⁷.

⁶ <https://www.jetbrains.com/pycharm/>

⁷ <http://www.eclipse.org/>

5 Método Proposto

Este capítulo apresenta um método de transformação de problema voltado ao uso de Deep Learning para classificação multirrótulo de reações em textos extraídos de notícias. Diferente da opinião, utilizada na maioria dos trabalhos, o termo reação utilizado neste trabalho representa a resposta sobre um texto que não foi escrito pelo usuário. Como cada pessoa possui sua própria identidade, diferentes respostas podem ser fornecidas por diferentes pessoas. Essa característica torna necessária a utilização de um tipo de classificação que permite múltiplos rótulos simultaneamente, chamada de classificação multirrótulo. Embora poucos trabalhos de classificação multirrótulo com o uso de Deep Learning tenham sido encontrados, o método proposto neste trabalho apresenta o uso de um desses algoritmos. A motivação para a escolha desse grupo de algoritmos é o sucesso obtido na tarefa de classificação simples rótulo apresentado em diversos trabalhos da literatura (TANG; QIN; LIU, 2015). Como visto em LeCun, Bengio e Hinton (2015), um algoritmo em especial, chamado LSTM, vem se apresentando apropriado para as tarefas envolvendo textos. Embora a maioria dos testes realizados utilizem o LSTM, outros algoritmos de classificação podem ser utilizados no método proposto. Para uma melhor avaliação, testes com o algoritmo SVM também foram realizados.

Uma das principais dificuldades da maioria dos métodos de transformação de problema para classificação multirrótulo é a necessidade de utilização de múltiplos classificadores simples rótulo (BI; KWOK, 2013). Essa necessidade faz com que, em muitos casos, torne-se inviável o uso de algoritmos mais robustos, como os algoritmos de Deep Learning. Esse problema ocorre pois esses algoritmos necessitam de um alto poder computacional para o treinamento da rede neural, como apresentado em Diehl et al. (2016).

Uma família de métodos de transformação de problema possibilita minimizar essa dificuldade realizando um ensemble de classes. Esses métodos criam grupos com as classes do problema e resolvem cada grupo individualmente utilizando outro método de transformação de problema sem ensemble, como o Label Powerset utilizado pelo método RAKEL. Uma limitação existente na maioria desses métodos é a necessidade de criação de novas classes, formadas pela combinação das classes originais. Outra forma de resolver o problema sem a necessidade de múltiplos classificadores é apresentada em alguns trabalhos onde os dados do problema são divididos, transformando o problema multirrótulo em apenas um problema multi classe. Em Yu et al. (2017), Yang et al. (2017) e Wei et al. (2016) os autores apresentam abordagens fundamentadas na divisão de uma imagem em hipóteses contendo pedaços da imagem original com objetos. Os objetos das hipóteses são classificados de forma simples rótulo e os resultados são reunidos.

O método de transformação de problema apresentado neste capítulo visa usufruir dos benefícios disponibilizados pelos métodos de ensemble sem a necessidade de criação de novas classes. Para isso, o método proposto consiste na união da estratégia utilizada pelos métodos que utilizam ensemble com uma adaptação para textos dos métodos de divisão de imagens. As etapas do método proposto seguem as mesmas etapas dos métodos de Descoberta de Conhecimento em Bancos de Dados tradicionais, com a adição de uma etapa de divisão de classes para o treinamento, uma etapa de divisão de textos em frases no teste e uma etapa de união de classes e frases para a avaliação. Uma visão geral do método é apresentada na Figura 18.

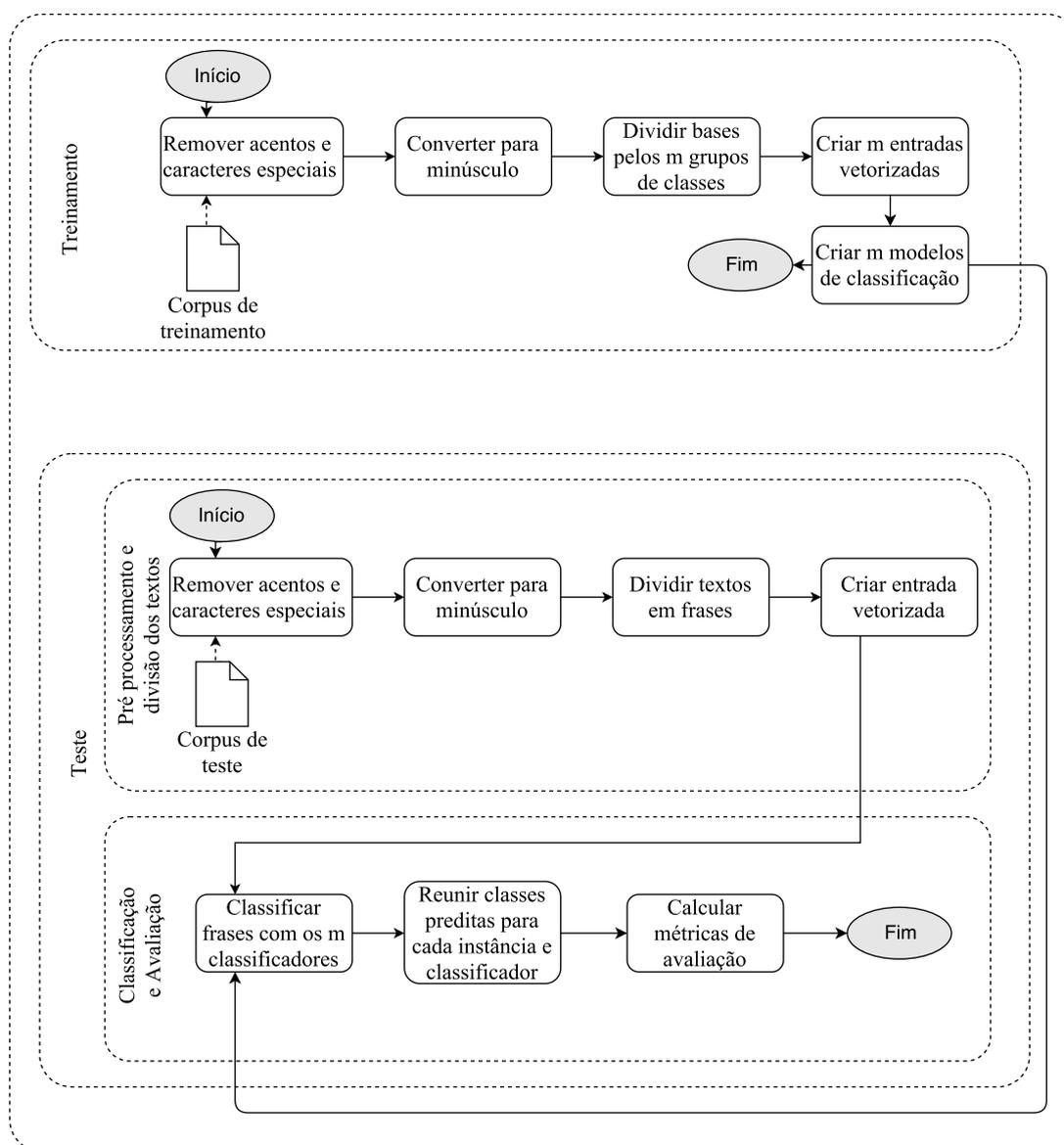


Figura 18 – Visão geral do método proposto

5.1 Pressupostos do Método

O método proposto possui os seguintes pressupostos:

- O método deve aceitar dados textuais com múltiplos rótulos.
- O método deve identificar automaticamente as opiniões ou reações em textos de diversos idiomas.
- O método não deve necessitar de recursos léxicos e ontologias para o seu funcionamento.
- O método deve ser utilizado em textos com diversas frases, como notícias.

5.2 Pré-processamento

A primeira etapa do modelo proposto é a realização do pré-processamento dos dados. Como um dos pressupostos é a possibilidade de se utilizar diversos idiomas, poucas alterações são realizadas nessa etapa. As únicas alterações realizadas para os testes utilizando o LSTM são a transformação de todos os caracteres para minúsculo e remoção de caracteres especiais. Adicionalmente, para o corpus português, são realizadas as transformações necessárias para o modelo de embedding utilizado. Essas alterações são apresentadas no código do projeto Portuguese Word Embeddings¹ e são compostas pela transformação de alguns termos em tokens, como emails e links. Para os testes utilizando o algoritmo SVM o procedimento de remoção de caracteres especiais, conversão para minúsculo e tokenização de termos também foi realizado. Adicionalmente, foi realizada a remoção de stopwords e aplicação do lematizador com os recursos disponibilizados pelo sistema snowball². Os testes com a utilização do SVM foram realizados com a quantidade de ocorrências de cada palavra no texto, enquanto para o LSTM uma operação de embedding foi realizada.

A operação de embedding consiste na criação de um vetor denso composto de valores reais para representar as palavras. Diversos modelos de representação são apresentados na literatura, como os trabalhos de Tang et al. (2014) e Melamud, Goldberger e Dagan (2016). Neste trabalho é utilizado o modelo Word2Vec, apresentado em Mikolov et al. (2013a). A proposta do Word2Vec é criar uma representação vetorial das palavras de forma que as palavras pertencentes a contextos similares fiquem alocadas no mesmo espaço vetorial. Na literatura existem poucos vetores de embedding pré-treinados para o Português, o trabalho de Hartmann et al. (2017) se destaca entre os vetores para esse

¹ https://github.com/nathanshartmann/portuguese_word_embeddings

² <http://snowballstem.org/>

idioma pela quantidade de tokens utilizados no corpus de treinamento. Por esse motivo, optamos pela utilização do vetor apresentado por Hartmann e colegas³.

O pré-processamento apresentado nesta seção tem como objetivo transformar o texto em uma entrada apropriada para o algoritmo de classificação utilizado. O pré-processamento utilizado neste trabalho é inspirado em trabalhos da literatura que apresentam bons resultados com a utilização do mesmo. Essa etapa pode ser modificada para adequar os dados ao algoritmo utilizado para a classificação ou aceitar características do texto utilizado, como um idioma diferente. Após o pré-processamento os dados são utilizados como entrada para o treinamento dos classificadores, essa tarefa é apresentada na próxima seção.

5.3 Treinamento

A principal diferença existente na etapa de treinamento entre o método proposto e métodos de aprendizagem de máquina tradicionais é a necessidade da criação de um ensemble com m classificadores multi classe contendo todas as L classes do problema. A ideia de se criar diversos classificadores tem como influência os métodos de transformação de problema com o uso de ensemble existentes na literatura, como os apresentados por Tsoumakas e Vlahavas (2007), Tsoumakas, Katakis e Vlahavas (2011), Zhou e Tao (2012), Rokach, Schclar e Itach (2014) e Gharroudi, Elghazel e Aussem (2015). Esses métodos tem como princípio a criação de m classificadores, contendo cada um uma parte das classes L do problema. A estratégia mais utilizada para possibilitar que todas as classes possam ser abrangidas pelo classificador é incluir em cada uma das m divisões todas as combinações existentes de classes, criadas pelo método Label Powerset. Um exemplo desses algoritmos é o RAKEL, apresentado na seção 2.3.1.

A principal vantagem dos métodos de transformação de problema com ensemble é a necessidade de se utilizar menos classificadores que as outras técnicas. Porém, a maioria desses métodos utiliza uma técnica chamada Label Powerset para possibilitar que a classificação inclua todas as classes do problema simultaneamente. Essa técnica necessita que combinações das classes originais sejam criadas para permitir a classificação multirrótulo. A criação de novas classes gera algumas limitações em bases pequenas, onde uma combinação existente na base de teste pode não ocorrer na base de treinamento. As novas classes também apresentam problemas em bases com muitas instâncias, onde é possível que todas as combinações de classes para cada grupo existam, criando muitas classes novas. Para evitar esses problemas, o método proposto neste trabalho não realiza a criação de novas classes e apresenta uma etapa adicional na fase de teste, onde as frases são divididas e classificadas individualmente.

³ <http://www.nilc.icmc.usp.br/nilc/index.php/repositorio-de-word-embeddings-do-nilc>

Como o método proposto classifica as frases de forma simples rótulo um classificador simples rótulo deve ser criado na etapa de treinamento. A solução ideal, chamada de Solução 1, para essa tarefa é utilizar uma base criada de forma simples rótulo, como o realizado pelos trabalhos utilizando imagens. Um problema existente nas bases de textos rotuladas com opinião ou reação é a diferença entre as classes utilizadas. Esse problema é ampliado devido a existência de diferentes idiomas e a pequena quantidade de bases multirrótulo existentes na literatura.

Devido a incompatibilidade entre as bases existentes propomos três soluções alternativas de treinamento para utilização do método. A primeira forma de resolver o problema das bases, chamada de Solução 2, é realizando a transformação de dados multirrótulo em dados simples rótulo eliminando as classes com menos votos. Esse método é apresentado em Tsoumakas e Katakis (2006) e uma ilustração é apresentada na Figura 19.

Figura 19 – Solução 2 para transformação do problema multirrótulo em simples rótulo

Ex.	Amor	Ódio	Alegria	Tristeza
1	42		13	
2	20			
3	10			84
4		12		3

 \Rightarrow

Ex.	Amor	Ódio	Alegria	Tristeza
1	X			
2	X			
3				X
4		X		

Outra solução, chamada de Solução 3, é repetir as instâncias onde existe mais de um rótulo ativo ao mesmo tempo, ou seja, em todos os casos onde $|Y| \geq 2$ deve-se repetir a instância x para cada rótulo y . Essa solução também é apresentada por Tsoumakas e Katakis (2006) e uma ilustração é apresentada na Figura 20.

Figura 20 – Solução 3 para transformação do problema multirrótulo em simples rótulo

Ex.	Amor	Ódio	Alegria	Tristeza
1	42		13	
2	20			
3	10			84
4		12		3

 \Rightarrow

Ex.	Amor	Ódio	Alegria	Tristeza
1	X			
1			X	
2	X			
3	X			
3				X
4		X		
4				X

Em alguns casos, onde a base apresenta dados simples rótulo misturados com dados multirrótulo é possível utilizar uma quarta solução apresentada por Tsoumakas e Katakis (2006), chamada de Solução 4, onde apenas os dados simples rótulos são utilizados para o treinamento. Nesse caso todas as instâncias x , onde $|Y| \geq 2$ são eliminadas. A Figura 21 apresenta uma ilustração desse método.

Para essa etapa é ideal a utilização da Solução 1, mas em muitos casos as soluções alternativas são necessárias. O principal problema ao se utilizar a Solução 2 é a perda

Figura 21 – Solução 4 para transformação do problema multirrótulo em simples rótulo

Ex.	Amor	Ódio	Alegria	Tristeza
1	42		13	
2	20			
3	10			84
4		12		3

⇒

Ex.	Amor	Ódio	Alegria	Tristeza
2	X			

de informações causada pela eliminação dos rótulos. O principal problema da Solução 3 é a duplicação de instâncias com rótulos diferentes, o que pode gerar imprecisão no modelo gerado. O uso da Solução 4 está limitado aos casos onde dados multirrótulo e dados multi classe com apenas um rótulo estejam presentes na mesma base. Outra limitação dessa solução é o tamanho da base resultante, que pode se tornar insuficiente para bons resultados com algoritmos de Deep Learning. As soluções 2, 3 e 4 devem ser utilizadas apenas em casos onde não existam bases simples rótulo compatíveis com a base multirrótulo do problema.

Além da escolha do melhor método para transformar a base, uma etapa importante para o método proposto e também para os outros métodos com ensemble é a seleção dos melhores grupos de classes para cada um dos m classificadores. Com o objetivo de evitar conflito entre as classes, o método de seleção de grupos utilizado neste trabalho tem como fundamento as correlações entre as classes do problema. O objetivo dessa tarefa é garantir que as classes mais correlacionadas sejam alocadas em grupos separados e as classes com menos correlação no mesmo grupo. No método de seleção proposto, inicialmente seleciona-se as m classes com mais exemplos. Essa separação garante que as classes mais frequentes não concorram entre si, diminuindo o efeito do desbalanceamento das classes. Após a separação das m classes mais frequentes as classes restantes são adicionadas a cada grupo com base nas menores correlações de pearson, respeitando a quantidade máxima de classes k definida.

5.4 Divisão dos Dados, Teste e Avaliação

Como visto na Figura 18, a etapa de teste do método proposto necessita da divisão dos documentos x_i da base de teste em f frases $\{x_{i_1}, x_{i_2}, \dots, x_{i_f}\}$. Essa divisão possui o mesmo objetivo da divisão de imagens em hipóteses realizada pelos trabalhos apresentados na seção 3.2. Nessa operação assume-se que cada pedaço da entrada possui m rótulos y , obtidos pelos m classificadores gerados na etapa de treinamento, e a união desses rótulos para todas as f frases cria o conjunto de rótulos L com a solução do problema.

Considerando p as instâncias da base de teste, f as frases de cada instância e $h_m(f)$ o resultado do classificador m para a frase f , temos o Algoritmo 1:

Algoritmo 1 Classificação das frases

```

function CLASSIFICAÇÃO( $p$ )
  for  $a \leftarrow 1$  to  $p$  do
    Cria o vetor de frases  $f$ 
    for  $b \leftarrow 1$  to  $m$  do
      for  $c \leftarrow 1$  to  $f$  do
         $Ty[a][h_b(c)] ++$ 
      end for
    end for
  end for
  return  $Ty$ 
end function

```

Um exemplo dessa operação é o seguinte trecho retirado de um site de notícias de entretenimento: “(1)A prefeitura proibiu mesas na calçada mas os clientes do bar tiveram uma ideia. (2)A prefeitura de Caxias do Sul proibiu os bares do centro da cidade de terem mesas na calçada, mas não disse nada sobre caminhões. (3)Os frequentadores do bar Zanuzi, de Caxias do Sul (RS), fizeram sucesso ao tomar cerveja sentados em um CAMINHÃO após o local ser notificado pela prefeitura para que retirasse as mesas da calçada. (4)Os frequentadores do bar já haviam feito algo parecido antes, colocando tapetes e almofadas na calçada, já que não havia mesas...”

Considerando as seis emoções básicas propostas por Ekman (Tristeza, Medo, Surpresa, Desgosto, Raiva e Alegria) divididas em dois grupos $m_1 = (\text{Alegria, Desgosto e Raiva})$ e $m_2 = (\text{Medo, Surpresa e Tristeza})$, podemos supor que um classificador obteve os resultados Desgosto e Tristeza para a frase 1, Raiva e Surpresa para a frase 2, Raiva e Surpresa para 3 e Desgosto e Surpresa para 4. Com isso o resultado final da classificação seria $Y = \{ \text{Desgosto, Raiva, Surpresa e Tristeza} \}$.

5.5 Limitações do Método Proposto

O método apresentado neste capítulo apresenta algumas limitações causadas pela dificuldade de se encontrar bases simples rótulo e multirrótulo com as mesmas classes. Essas limitações são externas ao método e podem ser resolvidas com a criação de novos recursos ou com a utilização das soluções propostas. A principal limitação do método apresentado é a impossibilidade de utilizar o mesmo em dados que não sejam informações textuais. Outra limitação é a necessidade de textos com múltiplas frases na base de teste. Esse tipo de texto se torna necessário pois a utilização de textos pequenos pode gerar casos em que o total de rótulos corretos é maior que o total de frases. O capítulo a seguir apresenta informações sobre a base construída para avaliação do método e os resultados obtidos com a aplicação do método proposto e de outros métodos existentes na literatura.

6 Resultados

Neste capítulo são descritos os resultados obtidos nesta pesquisa. Em resumo, são apresentadas as informações sobre os novos corpora desenvolvidos e as comparações do desempenho obtido pelo novo método com os principais métodos existentes na literatura.

6.1 Construção de um corpus multirrótulo de notícias

A primeira etapa para a construção da base foi uma pesquisa empírica em diversos sites de notícias em português e inglês. O resultado dessa etapa foi a seleção do site BuzzFeed¹ para a coleta dos dados. Esse site foi criado em 2006 e seu foco são notícias de entretenimento e diversão. Uma das características existentes nesse site até o início de 2017 era a possibilidade do leitor expressar a sua opinião sobre a notícia através de rótulos predefinidos, como visto na figura 22. Essa figura demonstra as opções de opiniões que os usuários podiam selecionar após ler determinada notícia. Para selecionar uma das opções o usuário devia clicar sobre o rótulo escolhido, sendo que somente uma opinião era aceita por usuário. Os valores existentes sobre os rótulos apresentam a quantidade de usuários que tiveram aquela reação. Segundo BuzzFeed (2017) essa funcionalidade era fornecida aos usuários para possibilitar a criação, classificação e caracterização do melhor conteúdo.

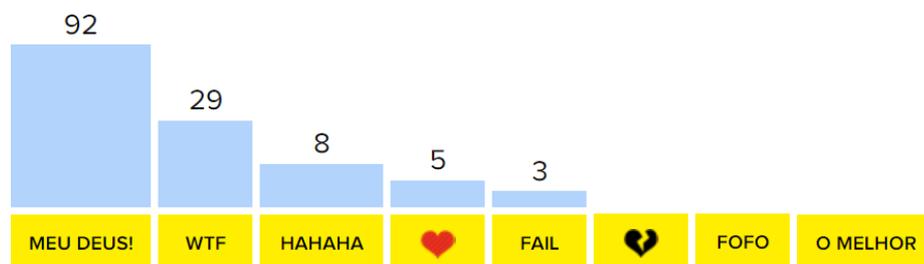


Figura 22 – Rótulos para as notícias em português do site BuzzFeed (Extraído de: BuzzFeed.com)

Outra característica que torna esse site uma fonte interessante para a coleta do corpus é a sua importância no cenário mundial. Segundo a ferramenta de monitoramento de tráfego alexa², em Dezembro de 2017 o site BuzzFeed.com, estava na posição 174 no rank mundial de acessos e na posição 54 no rank de acessos nos Estados Unidos.

Após a escolha do site e dos idiomas português e inglês iniciou a etapa de coleta da base. Como não foram encontradas APIs para essa tarefa um *crawler* foi desenvolvido. Todas as notícias compostas de textos e com no mínimo um rótulo publicadas em

¹ <https://www.buzzfeed.com/>

² <http://www.alexa.com/>

português do Brasil ou Inglês Americano no período de origem do site até o dia 10 de Fevereiro de 2017 foram coletadas. Neste trabalho foram utilizados apenas os rótulos disponíveis para todas as notícias. Alguns rótulos como amazing, blimey e ahhhhh estavam disponíveis apenas para algumas notícias do site, por esse motivo foram descartados.

A base com notícias em português é composta por 8080 notícias com uma média de 35,85 frases e um desvio padrão de 28,79. A média de palavras é de 223,33 com desvio padrão de 204,35. Esses dados demonstram que a base possui notícias com tamanhos variados. A base possui oito classes de rótulos. Os rótulos para as notícias são: Amável (opção com imagem de coração), Encantador (opção fofo), Engraçado (opção hahaha), Falho (opção fail), Odiável (opção com imagem de coração partido), Repulsivo (opção com a gíria wtf), Surpreendente (opção meu deus!) e Vencedor (opção o melhor). A quantidade de ocorrências de cada rótulo nos textos é apresentada na Figura 23.

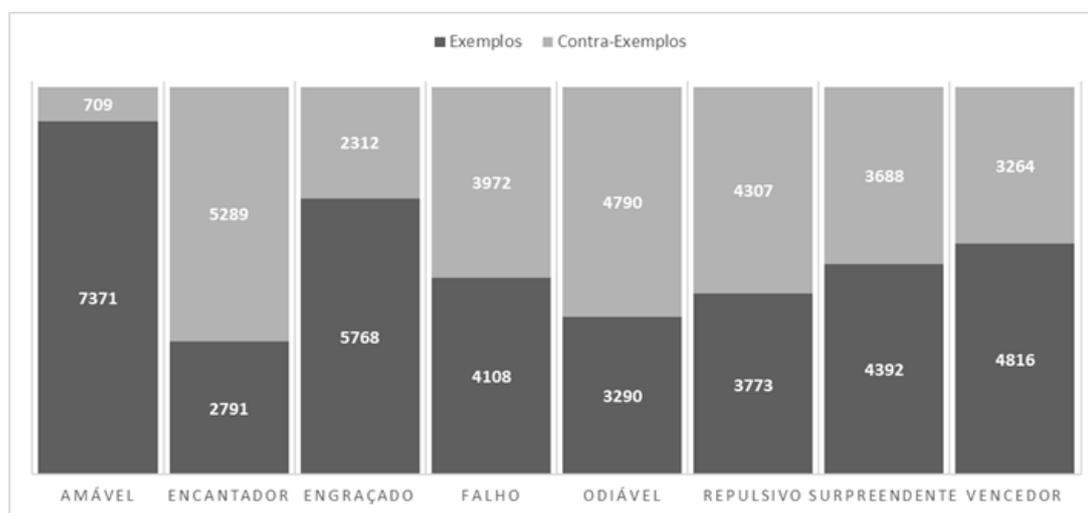


Figura 23 – Distribuição dos rótulos para a base em português

Em relação a quantidade de rótulos por notícia a base possui uma cardinalidade dos rótulos (Label Cardinality) de 4,49 e uma densidade (Label Density) de 0,56. A Figura 24 apresenta distribuição das instâncias com base na quantidade de rótulos.

A Tabela 10 apresenta a correlação de Pearson para os rótulos da base em português. Segundo Fitz-Gibbon e Morris (1987) é possível considerar que a correlação entre os rótulos wtf e fail é moderada e em todos os outros casos a correlação entre os rótulos é fraca ou desprezível.

A base em inglês contém 118150 notícias com uma média de 34,23 frases e desvio padrão de 37,05. A média de palavras é de 252,13 com desvio padrão de 293,99. Esses dados demonstram que, assim como na base em português, a base em inglês apresenta textos com diversos tamanhos. A base contém 11 rótulos: cute, ew, fail, hate, lol, love, omg, trashy, win, wtf e yaaass. A Figura 25 apresenta a quantidade de ocorrências de cada rótulo. A quantidade de instâncias por quantidade de rótulos é apresentada na Figura 26.

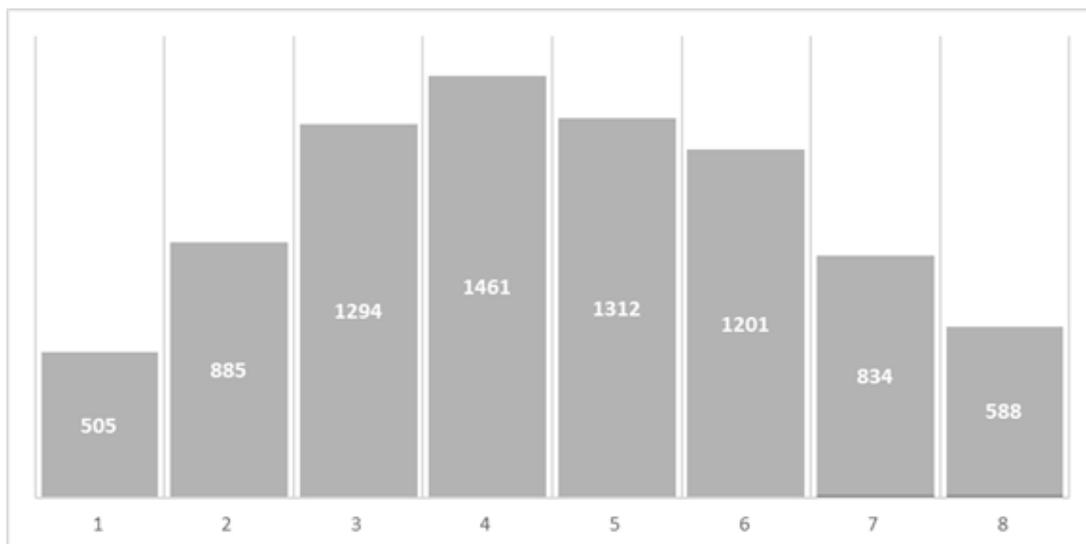


Figura 24 – Quantidade de instâncias por número de rótulos para o corpus em português

Tabela 10 – Correlação de Pearson do corpus em português

	Amável	Encantador	Engraçado	Falho	Odiável	Repulsivo	Surpreendente	Vencedor
Amável	1,0000	0,1885	-0,0037	-0,0232	-0,0564	-0,0297	0,0363	0,2243
Encantador	0,1885	1,0000	0,0418	0,0318	0,0310	0,0708	0,1421	0,2023
Engraçado	-0,0037	0,0418	1,0000	0,2928	0,1708	0,3088	0,1846	0,2065
Falho	-0,0232	0,0318	0,2928	1,0000	0,2838	0,4316	0,2212	0,1334
Odiável	-0,0564	0,0310	0,1708	0,2838	1,0000	0,2816	0,1996	0,0822
Repulsivo	-0,0297	0,0708	0,3088	0,4316	0,2816	1,0000	0,3218	0,1750
Surpreendente	0,0363	0,1421	0,1846	0,2212	0,1996	0,3218	1,0000	0,2452
Vencedor	0,2243	0,2023	0,2065	0,1334	0,0822	0,1750	0,2452	1,0000

A cardinalidade dos rótulos da base é 7,62 e a densidade dos rótulos é 0,69.

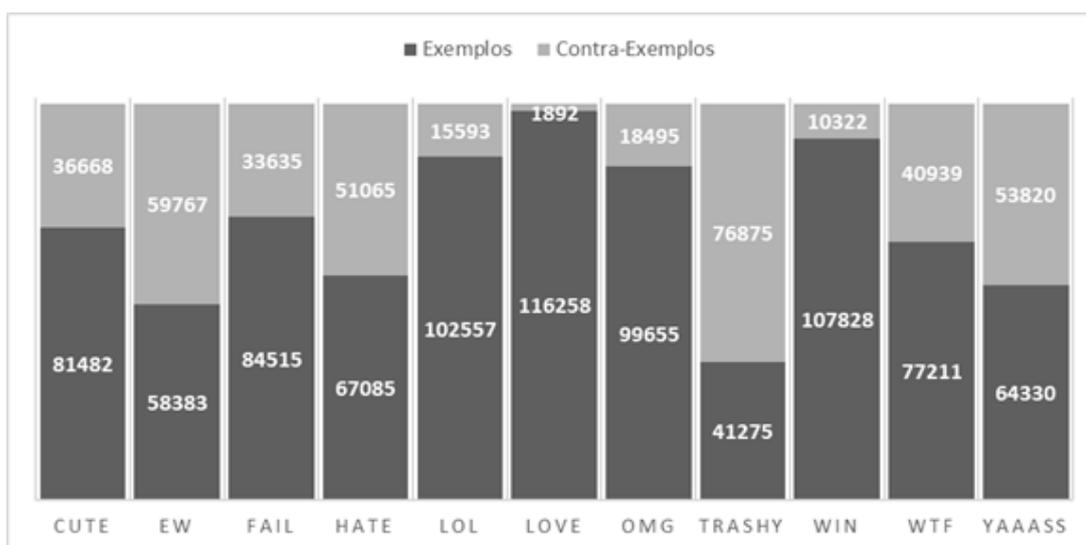


Figura 25 – Distribuição dos rótulos para o corpus em inglês

Assim como na base em português os rótulos wtf e fail apresentam uma correlação moderada enquanto as correlações entre os outros rótulos podem ser consideradas fracas ou desprezíveis se considerarmos o trabalho de Fitz-Gibbon e Morris (1987). As correlações

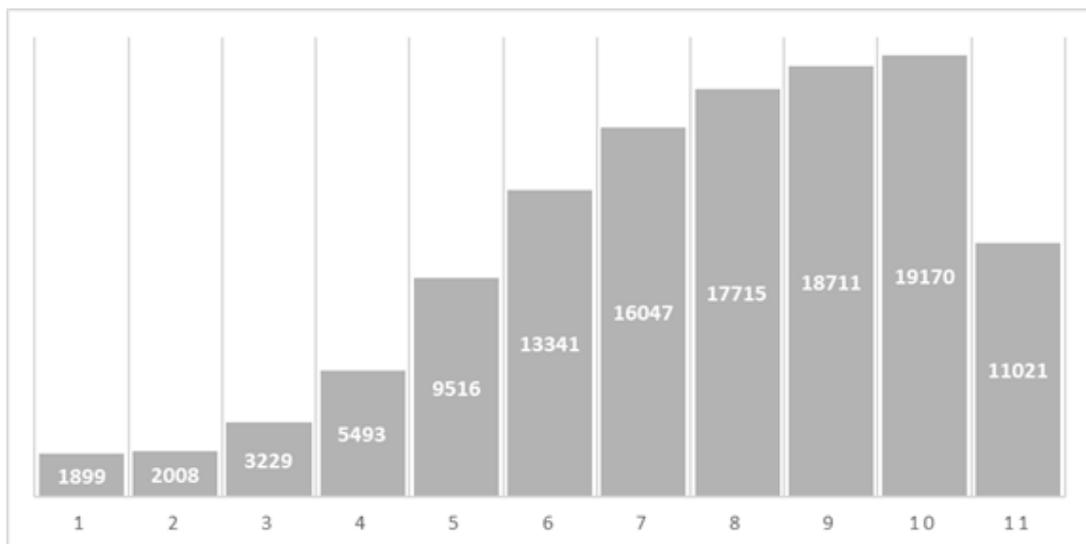


Figura 26 – Quantidade de instâncias por número de rótulos para o corpus em português

fracas indicam que existem independência entre os rótulos, o que garante que as bases sejam multirrótulo. A Tabela 11 apresenta os valores das correlações de Pearson do corpus em inglês.

Tabela 11 – Correlação de Pearson do corpus em inglês

	cute	ew	fail	hate	lol	love	omg	trashy	win	wtf	yaaass
cute	1,0000	0,1459	0,1209	0,1286	0,2410	0,1625	0,2731	0,1156	0,2999	0,1105	0,2288
ew	0,1459	1,0000	0,3817	0,3351	0,1863	0,0661	0,1790	0,3043	0,1018	0,3818	0,1660
fail	0,1209	0,3817	1,0000	0,3304	0,2602	0,0816	0,1748	0,2793	0,1162	0,4277	0,1655
hate	0,1286	0,3351	0,3304	1,0000	0,1478	0,0712	0,1920	0,2243	0,0950	0,3196	0,1742
lol	0,2410	0,1863	0,2602	0,1478	1,0000	0,1419	0,2544	0,1598	0,2378	0,2698	0,1612
love	0,1625	0,0661	0,0816	0,0712	0,1419	1,0000	0,1852	0,0450	0,2898	0,0559	0,1272
omg	0,2731	0,1790	0,1748	0,1920	0,2544	0,1852	1,0000	0,1400	0,3226	0,2168	0,2301
trashy	0,1156	0,3043	0,2793	0,2243	0,1598	0,0450	0,1400	1,0000	0,0846	0,2663	-0,0387
win	0,2999	0,1018	0,1162	0,0950	0,2378	0,2898	0,3226	0,0846	1,0000	0,0865	0,2482
wtf	0,1105	0,3818	0,4277	0,3196	0,2698	0,0559	0,2168	0,2663	0,0865	1,0000	0,1244
yaaass	0,2288	0,1660	0,1655	0,1742	0,1612	0,1272	0,2301	-0,0387	0,2482	0,1244	1,0000

Um exemplo de notícia do corpus em português é apresentado no Apêndice A e em um exemplo do corpus em inglês no Apêndice B. Esses exemplos permitem uma melhor visualização da estrutura dos dados e da forma como os rótulos são organizados.

6.2 Experimentos e Análise dos Resultados

Esta seção tem o intuito de detalhar os experimentos realizados com o corpus em português, apresentado na seção 6.1. Esta seção está dividida em três partes, inicialmente é apresentada uma revisão do protocolo de testes apresentado na seção 4.4. e os resultados dos ajustes dos parâmetros dos algoritmos e métodos utilizados. A segunda parte apresenta

os resultados dos experimentos realizados. Por fim, a terceira parte desta seção apresenta a análise dos resultados obtidos.

6.2.1 Protocolo de Testes e Definição dos Parâmetros

Atendendo o objetivo desta pesquisa, o principal resultado obtido é a proposta de um novo método de transformação de problema para a classificação multirrótulo de textos. Uma descrição desse método é apresentada no Capítulo 5. Além da proposta do novo método, foi realizado um levantamento na literatura sobre os principais métodos e algoritmos existentes para a tarefa de classificação multirrótulo. Com base nesse levantamento, um protocolo de testes foi elaborado para a avaliação do método proposto. Para o corpus em português, o novo método é comparado com os métodos de transformação de problema BR, CC, CLR, HOMER, LP e RAKE com o algoritmo SVM. Também são realizados testes com o método BR com o algoritmo LSTM e o método de adaptação de algoritmo ML-KNN. O método proposto neste trabalho é avaliado com os algoritmos LSTM e SVM. Para todos os testes, o corpus é dividido utilizando validação cruzada com 3 folds. Os testes completos para o corpus em inglês serão realizados em uma etapa futura a este trabalho, seguindo o mesmo protocolo utilizado para o corpus em português.

Os parâmetros utilizados no algoritmo LSTM foram definidos com a utilização do método BR. A entrada do classificador foi criada a partir de uma operação de embedding realizada com o uso do modelo Word2Vec. A matriz utilizada para o corpus em português é apresentada por Hartmann et al. (2017). O ajuste dos melhores valores para cada parâmetro foi realizado por meio de uma busca por força bruta. A tabela 12 apresenta os parâmetros modificados e os valores obtidos. Nessa tabela, a coluna *range* representa as faixas de valores testadas para cada parâmetro. O incremento dos valores foi realizado gradativamente, inicialmente foram utilizadas faixas maiores de valores e a partir do momento em que o maior F1 era encontrado uma faixa menor era utilizada. Um exemplo é a seleção do melhor valor para a camada oculta, inicialmente foram utilizados os valores 25, 50, 75, 100, 125, etc. Após a criação e avaliação de todos os modelos o melhor resultado foi obtido com o valor 75. Com a obtenção desse valor foi utilizada uma nova faixa com os valores mais próximos, 50 a 100, com um incremento de 10. Essa operação foi realizada em todos os parâmetros e gerou os resultados apresentados na coluna melhor valor. Os valores padrão de cada parâmetro foram definidos com base no exemplo³ de RNN para classificação de texto do framework TensorFlow.

Outro parâmetro existente no método proposto neste trabalho, e nos outros métodos de transformação de problema com ensemble existentes na literatura, é a seleção dos melhores grupos de classes. O método apresentado neste trabalho realiza a seleção dos

³ https://github.com/tensorflow/tensorflow/blob/master/tensorflow/examples/learn/text_classification.py

Tabela 12 – Parâmetros do algoritmo LSTM para o corpus em português

Parâmetro	Descrição	Range	Melhor Valor
Tamanho da matriz de embedding	Tamanho do vetor de embedding pré-treinado utilizado	50, 300	300
Tamanho do documento para treinamento	Quantidade de palavras usadas para treinar o modelo	10-1500 (inc. de 100, 50, 10 e 5)	35
Tamanho do documento para teste	Quantidade de palavras utilizadas para testar o modelo	10-1500 (inc. de 100, 50, 10 e 5)	15
Camadas ocultas	Quantidade de células LSTM	5-250 (inc. de 25, 10 e 5)	75
Batch	Quantidade de exemplos utilizados simultaneamente	100-1000 (inc. de 100)	400
Épocas	Quantidade de passos realizados	5-150 (incremento de 25, 10 e 5)	65

grupos com base na correlação existente entre as classes. O método de seleção proposto (apresentado na seção 5.3) gerou três grupos de classes, com tamanho máximo $k=3$, onde $m_1 =$ (amável, falho e odiável), $m_2 =$ (encantador, engraçado e surpreendente) e $m_3 =$ (repulsivo e vencedor).

6.2.2 Experimentos

Após a execução dos experimentos apresentados na seção anterior, foram obtidos os resultados apresentados na tabela 13. A parte superior dessa tabela apresenta os resultados obtidos pelos métodos já consolidados na literatura. A parte inferior apresenta os resultados obtidos no método proposto neste trabalho com os algoritmos SVM e LSTM.

Tabela 13 – Resultados obtidos para o corpus em português

Método	Algoritmo	Jaccard index	Micro F1 ↓	Hamming loss
RA k EL	SVM	0,5172	0,6832	0,3694
HOMER	SVM	0,5173	0,6804	0,4139
CLR	SVM	0,5055	0,6701	0,3868
BR	LSTM	0,4974	0,6643	0,3849
ML-KNN	KNN	0,4993	0,6631	0,3813
CC	SVM	0,4862	0,6583	0,3840
BR	SVM	0,4860	0,6570	0,3830
LP	SVM	0,4078	0,5991	0,5126
Novo S2	LSTM	0,5433	0,7039	0,4337
Novo S3	LSTM	0,5256	0,6890	0,4376
Novo S4	LSTM	0,5224	0,6862	0,4339
Novo S2	SVM	0,5049	0,6709	0,4172
Novo S4	SVM	0,4923	0,6598	0,4206
Novo S3	SVM	0,4347	0,6059	0,4177

*Os testes com o algoritmo LSTM foram executados com a utilização de uma GPU. Os melhores valores são apresentados em negrito.

O valor obtido para o método RA k EL, apresentado na tabela 13, foi obtido após a seleção dos melhores parâmetros. Foram testadas todas as combinações pré-estabelecidas

fornecidas pelo software meka. O melhor valor foi obtido com a criação de três grupos com $L * 2$ classes. O ajuste de parâmetros também foi realizado para o método HOMER. Foram testados três diferentes métodos de seleção de classes (*Random*, *BalancedClustering* e *Clustering*), três diferentes quantidades de partições (2, 4, 6) e três métodos de transformação de problema (BR, CC, LP). Os resultados obtidos para todas as configurações são apresentados na tabela 14.

Tabela 14 – Resultados obtidos com o método HOMER com o algoritmo SVM para o corpus em português

Tipo	# partições	Método	Jaccard index	Micro F1 ↓	Hamming loss
Random	4	LP	0,5173	0,6804	0,4139
BalancedClustering	4	LP	0,5031	0,6699	0,4354
Clustering	4	BR	0,4921	0,6625	0,3852
Clustering	6	BR	0,4916	0,6619	0,3831
Clustering	4	CC	0,4908	0,6612	0,3862
Clustering	6	CC	0,4891	0,6595	0,3836
BalancedClustering	6	BR	0,4891	0,6588	0,3869
Random	4	CC	0,4847	0,6562	0,3920
Random	6	CC	0,4836	0,6550	0,3878
Clustering	2	BR	0,4844	0,6550	0,3873
BalancedClustering	6	CC	0,4829	0,6544	0,3903
Random	6	BR	0,4828	0,6543	0,3874
BalancedClustering	4	CC	0,4827	0,6538	0,3962
BalancedClustering	4	BR	0,4823	0,6536	0,3961
Clustering	2	CC	0,4841	0,6533	0,3879
Random	4	BR	0,4812	0,6528	0,3927
BalancedClustering	2	BR	0,4793	0,6504	0,3975
Clustering	2	LP	0,4778	0,6498	0,4250
Random	2	BR	0,4770	0,6492	0,3965
Random	2	CC	0,4767	0,6483	0,3963
BalancedClustering	2	LP	0,4733	0,6463	0,4412
Clustering	6	LP	0,4709	0,6462	0,4142
BalancedClustering	2	CC	0,4755	0,6461	0,3987
Random	2	LP	0,4742	0,6460	0,4382
Random	6	LP	0,4695	0,6422	0,4117
Clustering	4	LP	0,4588	0,6329	0,4226
BalancedClustering	6	LP	0,4465	0,6251	0,4554

Os melhores valores são apresentados em negrito.

6.2.3 Análise dos Resultados

Analisando os resultados obtidos pelos métodos já consolidados na literatura com o uso do corpus em português, apresentados na parte superior da tabela 13, é possível observar que o método RA k EL obteve o melhor desempenho em relação as métricas micro F1 e hamming loss. O melhor desempenho em relação a métrica micro F1 indica que o

método obteve mais acertos nas classes com menos exemplos, tratando melhor o desbalanceamento dos dados. O menor valor obtido com a métrica hamming loss indica que o método RAKEL gerou menos instâncias classificadas incorretamente. Entre os melhores métodos, também é possível destacar o HOMER. Esse método possibilitou o maior resultado para a métrica jaccard index, responsável por avaliar as classes positivas classificadas corretamente. Esses dados demonstram que os métodos de transformação de problema com ensemble apresentaram os melhores resultados se comparados aos outros métodos tradicionais testados. O melhor desempenho desses métodos pode ser justificado pela divisão de classes gerada, que possibilita uma maior diversidade na distribuição das classes. Essa diversidade permite que as classes sejam distribuídas de forma que todas possam ser classificadas corretamente, evitando que classes incompatíveis pertençam ao mesmo grupo.

Um desafio existente nos métodos baseados em ensemble é a seleção dos melhores grupos de classes. Um exemplo disso são os testes realizados para o ajuste dos parâmetros do método HOMER para o corpus em português. Embora esse método tenha obtido o melhor resultado em relação ao jaccard index e o segundo melhor em relação ao micro F1, 27 configurações desse método foram testadas, gerando diferentes resultados. Os resultados obtidos com as diferentes configurações variam até 5,5 p.p. em relação ao micro F1 e 7,08 p.p. para o jaccard index. A tabela 14 apresenta todas as configurações utilizadas e os resultados obtidos. Como é possível observar nessa tabela, a melhor configuração em relação as métricas jaccard index e micro F1 foi obtida com a criação de 4 grupos randômicos de classes, classificadas com o método LP. Embora tenham sido testados métodos que consideram o balanceamento das classes, o melhor resultado obtido foi gerado a partir de um método de seleção randômico. Esse fato demonstra a dificuldade para a criação de um método automático de seleção dos melhores grupos de classes para o ensemble.

Diferente do método randômico de seleção de classes, que possibilitou o melhor resultado para o método HOMER, o método de seleção utilizado neste trabalho apresenta o uso da correlação existente entre as classes para a criação dos grupos. Outra diferença existente entre o HOMER e o método de transformação de problema proposto é a necessidade de transformação do corpus multirrótulo em simples rótulo antes da etapa de treinamento. As três soluções propostas são apresentadas na seção 5.3 e se baseiam no uso da classe mais votada (solução S2), duplicação das instâncias com mais de um rótulo (solução S3) e remoção de todas as instâncias com mais de um rótulo (solução S4). A parte inferior da tabela 13 apresenta os resultados obtidos pelas três soluções com os algoritmo SVM e LSTM.

O uso do LSTM no método proposto neste trabalho possibilitou um jaccard index e um micro F1 superior ao SVM para o mesmo método em todas as soluções apresentadas. O uso do método com o algoritmo LSTM também possibilitou um jaccard index e um

micro F1 superior a todos os métodos tradicionais testados. É possível observar que a melhor solução do método proposto (S2) com o algoritmo LSTM apresentou um micro F1 3,96 p.p. superior ao método BR com o mesmo algoritmo. Assim como para o LSTM, a melhor solução apresentada com o uso do algoritmo SVM foi a S2, onde apenas a classe com mais votos (reação mais clicada) é considerada. Ao comparar os valores obtidos pela métrica micro F1 é possível observar que o método foi inferior aos métodos com ensemble existentes na literatura (RAkEL e HOMER), mas se apresentou superior aos métodos sem ensemble que utilizam o mesmo algoritmo.

Além das diferenças existentes nos resultados, as soluções propostas apresentam diferenças na quantidade de processamento necessária para o treinamento do classificador. As três diferentes soluções propostas para a transformação do problema multirrótulo em um problema simples rótulo (S2, S3 e S4) geram diferentes quantidades de exemplos nas bases de treinamento. Essa diferença faz com que a solução S4 seja a mais eficiente em relação ao processamento. Isso ocorre devido a remoção de instâncias da base de treinamento, demandando assim menos recursos para o treinamento do classificador. A solução S2 mantém o mesmo número de exemplos da base de teste, permitindo uma maior estabilidade na quantidade de recursos necessários. A solução S3 apresenta o pior desempenho em relação a quantidade de processamento. Isso ocorre pois essa solução tende a duplicar diversos exemplos. O aumento na quantidade de exemplos na etapa de treinamento exige mais processamento e pode demandar um maior tempo de execução para alguns algoritmos de classificação. Um desses algoritmos é o SVM utilizado neste trabalho.

O algoritmo SVM foi utilizado com as configurações padrões do framework meka, que utiliza o kernel polinomial. A principal limitação desse algoritmo para bases com uma vasta quantidade de amostras é a necessidade de comparação de cada par de exemplos para cada característica do problema. Essa comparação demanda um aumento exponencial na quantidade de processos quando se aumenta a quantidade de exemplos de treinamento. Embora o algoritmo LSTM não seja tão sensível quanto o SVM com kernel polinomial, esse algoritmo apresenta a necessidade do ajuste de alguns parâmetros. A Tabela 12 ilustra os testes realizados para a obtenção do melhor ajuste desse algoritmo. A necessidade desses ajustes também apresenta uma limitação em relação ao desempenho do classificador

Além da diferença na quantidade de processamento na etapa de treinamento, os diferentes métodos apresentam diferenças na etapa de teste. O método proposto neste trabalho necessita que os documentos sejam divididos em frases. Após a divisão as frases são testadas individualmente pelo classificador para gerar o resultado do documento após reagrupadas. Essa divisão torna necessária a execução de f testes para cada documento existente, gerando a necessidade de mais processamento nessa etapa.

7 Conclusões e Trabalhos Futuros

Em diversos casos, as informações contidas em textos podem causar diferentes reações em cada pessoa. A identificação de todas as possíveis reações de forma simultânea, chamada de multirrótulo, apresenta um desafio para a área de Mineração de Opiniões. A maioria dos trabalhos dessa área realiza a classificação chamada de simples rótulo, onde identifica apenas uma opinião ou sentimento por texto. Entre os métodos existentes para a classificação simples rótulo de opiniões, um grupo de algoritmos chamado Deep Learning apresenta características que possibilitaram melhores resultados em diversos relatos existentes na literatura.

Embora diversos trabalhos apresentem o uso de algoritmos de Deep Learning para a identificação de opiniões em textos, poucos trabalhos são encontrados realizando essa tarefa de forma multirrótulo. Uma das estratégias existentes para realizar esse tipo de classificação é a transformação do problema multirrótulo em diversos problemas simples rótulo. Um grupo de métodos, baseados em ensemble, possibilitam a realização dessa tarefa com a criação de poucos classificadores. Para possibilitar o menor número de classificadores, muitos desses métodos apresentam a necessidade de criação de novas classes, gerando algumas limitações dependendo das características da base de dados utilizada. Outra possível estratégia, específica para imagens, permite resolver problemas multirrótulo com a utilização de apenas um classificador. Essa operação é realizada com a criação de hipóteses, através de sub amostras da imagem original.

Visando obter os benefícios obtidos pelos métodos de ensemble sem a necessidade de criação de novas classes, este trabalho apresenta um novo método de transformação de problema com uma estratégia de segmentação de textos para a classificação multirrótulo de reações em notícias. A principal vantagem do método apresentado é a necessidade menos classificadores sem a necessidade de criação de novas classes, possibilitando assim o uso de algoritmos de classificação mais robustos, como os algoritmos de Deep Learning. Outra vantagem do método é a possibilidade de utilizar diversos idiomas sem a necessidade de alterações, como ocorre com a utilização de recursos léxicos.

Para permitir a avaliação do método, dois novos corpora de notícias nos idiomas inglês e português rotulados com reações foram desenvolvidos. O corpus em português foi testado com diversos métodos consolidados em trabalhos existentes na literatura e no método proposto neste trabalho com os algoritmos LSTM e SVM. O uso do método proposto neste trabalho com o corpus em português e o algoritmo LSTM possibilitou um micro F1 superior a todos os métodos utilizados para a comparação. O uso do mesmo método com o algoritmo SVM gerou um resultado inferior aos outros métodos de ensem-

ble existentes na literatura, mas possibilitou que a tarefa fosse realizada com bases de treinamento menores e com menos classes, necessitando assim de menos processamento. A necessidade de menos processamento também foi observada com o uso do algoritmo LSTM. O método proposto com o algoritmo LSTM também gerou melhores resultados que o BR com o mesmo algoritmo para as métricas micro F1 e jaccard index.

Todas as hipóteses definidas no Capítulo 1 foram validadas integralmente. A hipótese H1 (é possível realizar a classificação multirrótulo de textos utilizando Deep Learning) foi validada pelos trabalhos analisados na revisão bibliográfica apresentada no Capítulo 3 e nos testes realizados, apresentados no capítulo 6. As hipóteses H2 (o uso de Deep Learning para a classificação multirrótulo de notícias com a utilização de métodos de transformação de problema possibilita um resultado superior ao obtido com a utilização de algoritmos tradicionais) e H3 (o método de transformação de problema apresentado neste trabalho possibilita um resultado superior aos métodos existentes na literatura para a classificação multirrótulo de notícias) foram validadas através dos experimentos apresentados no Capítulo 6.

Os experimentos que possibilitaram validar as hipóteses foram realizados com o corpus em português, os testes com o corpus em inglês serão realizados em uma etapa futura. Esses testes serão realizados seguindo o mesmo protocolo utilizado para o corpus em português e utilizando os mesmos equipamentos. Além da execução dos ajustes de todos os parâmetros do algoritmo LSTM para o corpus em inglês, diferentes formas de seleção de grupos de classe para a divisão do ensemble serão realizadas futuramente. Os novos métodos de seleção também podem ser avaliados com outros métodos com ensemble existentes na literatura, como o RAKEL e o HOMER. Até o presente momento, não foram encontradas bases em português ou inglês adequadas para a utilização no método proposto neste trabalho sem a necessidade de alguma das 3 transformações propostas (S2, S3 e S4). Futuramente, uma nova base com as características necessárias pode ser construída para ampliar a avaliação deste método. Por fim, no futuro é possível estudar uma forma de divisão dos dados para outros tipos de entrada, como áudio e informações genéticas.

Anexos

ANEXO A – Exemplos de textos do corpus em português extraído do site BuzzFeed.com

Exemplo 1¹: 10 maneiras de ajudar alguém neste fim de ano. Dividir um pouquinho do que você tem com alguém que precisa bastante pode ser mais fácil do que você pensa. 1. Ao fazer suas compras de Natal, use o “Risü”, um serviço que converte uma porcentagem de suas compras em doação para quem precisa. Você escolhe a loja e a causa que pretende ajudar, eles te geram um link e você faz as compras normalmente, parte do valor da compra é convertido em doação para a causa que você escolheu e você não paga a mais por isso. E dá fazer compras em grandes lojas como Submarino, Ricardo Eletro e outras grandes que vendem online, de vários segmentos. Saiba mais. 2. Você pode comprar os cartões de Natal dos “Pintores com a boca”, da APBP. A Associação dos Pintores com a Boca e os Pés (APBP), disponibiliza para venda cartões de Natal feitos por quem não tem mãos (ou sofre de algum tipo de paralisia) para a prática da pintura. Saiba mais. 3. Você também pode ajudar um animal de abrigo a receber um mimo nesta época do ano sendo um padrinho através do “Natal Animal”. O projeto surgiu em 2010, inspirado pelas crianças carentes, que geralmente arrecadam uma roupa, um calçado e um brinquedo nesta época do ano. Os animais que participam do projeto recebem presentinhos de Natal comprados com o dinheiro arrecadado pela campanha e são utilizados na manutenção e cuidados básicos dos bichinhos nos abrigos/santuários onde vivem. A própria equipe do Natal Animal faz as entregas dos presentes e posta fotos e documentações para prestar contas. Saiba mais. 4. A Fundação José Carlos da Rocha possui uma campanha especial para os idosos durante o mês de dezembro. A instituição que ficou conhecida nas redes sociais pela criativa campanha de pedidos simples de idosos para o Natal, tem um trabalho permanente de ajuda e em dezembro também tem uma campanha para doação de fraldas para os idosos. Você pode saber mais informações e como auxiliar aqui. 5. O projeto “Transvest”, que presta apoio pedagógico e social para ajudar a combater a transfobia, sempre precisa de voluntários e você pode se inscrever agora. O projeto é de Belo Horizonte e dá apoio para a inserção de travestis, transexuais e transgêneros no curso pré-vestibular. Veja como colaborar aqui. 6. Embora não tenha uma campanha específica de Natal, você pode colaborar com o trabalho do projeto global “Médicos Sem Fronteiras”. O “Médicos Sem Fronteiras” é um projeto de atuação mundial que leva cuidados de saúde a pessoas

¹ <https://www.buzzfeed.com/raphaelevangelista/10-instituicoes-que-voce-pode-ajudar-neste-fim-de-ano>

em necessidade de ajuda médico-humanitária em meio a conflitos armados, epidemias, desastres naturais, desnutrição e exclusão do acesso a cuidados de saúde. Saiba mais e como contribuir por aqui. 7. Você pode adquirir um calendário de 2017 da “ActionAid” e ajudar a promover os direitos humanos e superar a pobreza no mundo. A ActionAid é um movimento global que trabalha em prol de pessoas em situação de pobreza, principalmente mulheres e crianças, para que eles sejam sujeitos de seus direitos humanos, econômicos, sociais e culturais e estão vendendo um calendário com fotos do trabalho da organização pelo mundo e parte da venda é revertida para as ações da instituição. Saiba mais sobre eles. 8. Você pode adotar uma das cartinhas da campanha “Adote uma Cartinha de Natal”, dos Correios. A campanha “Papai Noel dos Correios” possibilita que você se transforme em mamães e papais da família Noel adotando uma carta enviada por uma criança carente e enviando a ela o que ela pediu para o “Bom Velhinho”. Realizada há mais de 25 anos, a ação basicamente interliga crianças que mandaram uma carta pelo correio para o “polo norte” com pedidos ao papai Noel e qualquer pessoa pode adotar uma dessas cartinhas e presentear as crianças como se fosse o próprio Papai Noel. Saiba mais aqui. 9. A ONG “Brinquedo Para Todos” quer presenter 300 crianças da Aldeia Trevo, no Paraná. São 300 crianças (120 meninas e 180 meninos) de até 15 anos que fizeram um pedido para Papai Noel. A ONG vai representar o “Bom Velhinho” e vai levar os presentes pessoalmente, saiba mais e como colaborar aqui. 10. O “Exército da Salvação” está com uma campanha especial para receber doações dos mais diversos tipos, chamada “Árvore da Doação”. A ideia é que você monte uma “árvore” de Natal, com coisas que pretende doar e pedir para que eles retirem todo este conjunto em sua casa. Saiba mais.

A notícia apresentada no Exemplo 1 possui a seguinte distribuição de reações: Amável:8, Encantador:0, Engraçado:0, Falho:0, Repulsivo:0, Vencedor:1, Odiável:0 e Surpreendente:0.

Exemplo 2²: 19 perguntas que eu tenho sobre "A Bela e a Fera" agora que eu cresci. Por que Lumière é o único personagem com sotaque francês se a história se passa na França? Começando do início: se o filme se passa em um vilarejo francês, por que Lumière é o único personagem com sotaque francês? OK. Esta dado que a Fera era um tirano babaca. Mas... será que a fada realmente precisava amaldiçoar não só ele, mas TODO MUNDO DO PALÁCIO? Não é um pouco demais? Em outras palavras: se ele era um déspota e um malcriado, provavelmente tratava muito mal todas as pessoas que trabalhavam com ele. Então, por que eles estão sendo punidos? Que mal o pequeno Chip poderia ter feito para merecer isso? A propósito... Que tipo de príncipe abre PESSOALMENTE as portas do seu palácio toda vez que alguém chama nas horas mais impróprias pedindo abrigo? Não há nenhum mordomo no seu reino, Fera? E este príncipe não tem nenhuma família? Tipo, pais ou alguém mais? Vamos fazer as contas: o príncipe, assim

² <https://www.buzzfeed.com/beatrizserranomolina/19-perguntas-que-eu-tenho-sobre-a-bela-e-a-fera>

que se tornou a Fera, teoricamente tinha até os 21 anos de idade para quebrar o feitiço que recebeu. Quando a Bela aparece, Lumière diz que já “se passaram 10 anos” desde que a velha senhora veio até a porta do castelo. Então ele foi amaldiçoado quando tinha 11 anos de idade? Onde estavam os pais dele quando tudo isso aconteceu? De férias na praia? Esse cara não está feliz demais considerando o dia que ele está tendo? E quem deu aquele chapéu para ele? Como é possível este cara manter sua livraria aberta se todos na cidade dizem que a Bela é a única pessoa que lê livros? Bela é, afinal, sua única cliente. Outra coisa: O livro favorito da Bela (aquele que ela já leu 12 milhões de vezes) diz: “O príncipe encantado, e ela só descobre quem ele é quase no fim...!” SERÁ QUE ELA NÃO SE DEU CONTA QUE ISSO ERA UM SPOILER? Você está lendo “A Bela e a Fera”, Bela? Se a Fera tinha um temperamento difícil e colapsos diante da menor provocação, não era meio perigoso chamá-lo na lata de “Fera”? Por quê alguns móveis falam e cantam enquanto outros permanecem em silêncio? Nunca vimos os banheiros do castelo. Será que eles eram pessoas amaldiçoadas também? Quantos filhos a Sra. Potts teve antes da maldição? Duzentos? E cadê o pai dessas crianças? E com todas as xícaras perfeitamente refinadas que eles tinham na despensa, por quê eles sempre serviam os hóspedes com a única xícara quebrada? São hóspedes que você está tentando impressionar. Mostre a melhor porcelana que você tem! O Gastón é o dono do bar? Será que é por isso que existe um retrato dele ali e ele pode ficar atirando e quebrando todos os móveis? O que acontece se a Bela tocar na rosa? “Não, Bela, não vá para aquela parte do castelo, isso fará com que você descubra tudo, comece a simpatizar com a Fera e termine o filme antes da hora. ”. Por quê a Fera não sabe usar uma colher? Ele nasceu humano. E tem mais, ELE NASCEU PRÍNCIPE. Por que o Lumière nunca derrete? Seu rosto é feito de cera. E ele está sempre aceso! O que isso poderia ser? Espere um pouco... A Sra. Potts não é meio velha pra ter uma criança tão pequena? Ela parece ter sessenta anos. E, finalmente: por que quando o príncipe se transforma... a Bela faz ESSA CARA como se tudo não tivesse ocorrido na sua frente e ela não soubesse ao certo quem é esse homem? “Oh, que mistério... quem poderia ser esse cara tão atraente?” É sério Bela? Sério? Esse post foi escrito originalmente em espanhol.

A notícia apresentada no Exemplo 2 possui a seguinte distribuição de reações: Amável:4, Encantador:0, Engraçado:57, Falho:6, Repulsivo:2, Vencedor:1, Odiável:3 e Surpreendente:2.

ANEXO B – Exemplos de textos do corpus em inglês extraído do site BuzzFeed.com

Exemplo 1¹: 12 Stories Of People Experiencing A Glitch In The Matrix That'll Freak You Out. These are stories of unexplainable happenings. The following stories are from multiple AskReddit threads found [here](#), [here](#), and [here](#). 1. The wife who was seen in two different spots. One evening, a man and his wife returned home from dinner. When he opened the door, he saw his wife sitting at the computer doing work, which is a normal sight when he gets off of work, except for the fact that SHE WAS ENTERING THE HOUSE WITH HIM. When they went to bed, he told her that he'd unexplainably seen her sitting at the computer earlier, and she got very serious, then told him that when he opened the door, she saw herself sitting at the computer too. Creepy. Read the full story [here](#). 2. The creepy driver and his unconscious passengers. An off-duty EMT was driving home through a storm when he spotted a Mercedes parked past a barricade with its hazard lights on. He stopped and walked over to it, shining his light in the backseat, where he saw a man slumped over, as if he were sleeping. Initially, he thought it might be a car full of drunks, but in the driver's seat was a man sitting upright, staring straight ahead. He didn't blink, move, or respond to taps on the window, and the other passenger was slumped forward on the dashboard. Creeped out, he stepped away to call the sheriff's station. Dispatch asked him for a license plate number, but when he went to check, a large truck was coming down the road, so he waited for it to pass. When it did, the car was suddenly gone. There were trees in one direction and barricades in the other, leaving the man wondering how in the world this car and its mysterious occupants could have simply vanished. Read the full story [here](#). 3. The couple who dreamed something very specific. This is a simple story. A guy has a girlfriend of two years named Susan. One night he has a dream that he cheated on Susan with his ex-girlfriend. In the dream, while having sex with his ex, he looked over and standing outside of the window is his mortified girlfriend Susan, crying. He forgot about the dream, but then a few days later when he was hanging out with Susan, she told him that she had had a weird dream a few days earlier. She then described the exact same dream he'd had, but from her point of view, looking through the window. Read the full story [here](#). 4. The dream that came true. As a 15-year-old, someone remembers having a dream that was all blackness and talking. In

¹ <https://www.buzzfeed.com/christopherhudspeth/stories-of-people-experiencing-a-glitch-in-the-matrix-that-a>

the dream, she could hear her mother saying comforting things like “You’re doing great” and “It’s gone.” She didn’t think anything of it, but a few months later, her earrings were embedded inside her earlobes, and she had to go to the hospital to have them removed. They applied some type of freeze spray, but it accidentally went into her ear, touching the eardrum. She was knocked out, but could still hear, and her mother was saying the exact same comforting things to her that she’d heard in the dream a few months earlier. Read the full story here.

5. The mom who was in two places. A boy and his mom were in a woody area near their backyard taking pictures of old cars to sell. Suddenly, there was a rustling in the bushes and his mom told him to run, so he booked it. As he sprinted, he heard a clanking sound, as if someone had jumped on the cars. He arrived at his house in a panic, only to find his mom sitting at the table reading the paper. He’s asked his mother about it as an adult, and while she remembers having cars in the backyard that she sold, she doesn’t recall this incident ever happening. Read the full story here.

6. The guys who somehow lost an hour of life. A dude was in the car with a couple of friends, heading back from a road trip. His parents called to see how far away he was, and he told them it’d be about 25 minutes. Shortly thereafter, he came around a bend and saw a full moon, which was reflected in the lake below. The road ahead was completely empty, with not a single car in sight, then suddenly everything went pitch black for a moment. No dashboard lighting, no headlights on the road, the music even stopped playing. Then, the music was back, playing the CD from the beginning, and there was now a vehicle being pulled over by a cop a quarter mile ahead, though it hadn’t been there a split second earlier. He assumed that he’d fallen asleep momentarily, but after a few moments, the driver of the car exclaimed that he thought he’d just fallen asleep as well. They shared the experience, and more concerning was the clock, which was reading an hour later than it had a bit earlier. To avoid freaking out, they convinced themselves that the car may’ve had an electrical failure that reset the clock and whatnot, but when they arrived at his house 25 minutes later, he was an hour late. To this day, he has no idea what the explanation for that brief blackout and missing hour is. Read the full story here.

7. The man who briefly had a second life. A man recalls working as a vendor at a fish market. One morning he woke up, got dressed, got tea, went to the docks, bought some fish, and did his entire routine before selling fish all day. He spoke with friends, smoked cigarettes, counted his money, paid his stall rent, went home, ate dinner, relaxed, and went to bed. The next morning, he woke up ready to go back to the docks for fresh fish, only things were different. He was in bed next to his wife and it was a Saturday, so he didn’t have work. It was all a dream. He’s a married man who has never worked in a fish market, yet somehow he experienced life as some single fish vendor in an entirely different life, with no idea how it happened. Read the full story here.

8. The orange doll that refused to go away. Someone recalls their dad having a little toy monkey that he used to teasingly say was his “favorite child.” It was an ongoing joke, and this person and their siblings would

always try to steal the monkey, playing into it. One day they got ahold of the monkey and drew all over it with a Sharpie, then threw it in the garbage. They laughed when their dad searched for it, but he gave up when he figured they'd thrown it out. A few years later, now 17 years old, this person was walking down the street in Toronto, when they saw an orange object on the side of the road. They walked up to it, and it was the exact same toy monkey, black Sharpie lines and all. This person is still stunned, unable to figure out how this is possible when their garbage is sent to a local dump that's nowhere near Toronto. Read the full story here. 9. The woman with the roses who must've like teleported or something. A guy was waiting to catch a train when a woman with glazed eyes asked him for money. She said that her brother was in the hospital and she wanted to buy him some flowers. Despite thinking that she might be a drug addict, he grudgingly gave her \$5, certain that it would be wasted on her habit. Eventually his train pulls up and he gets on. As it's pulling away he sees the woman through the glass asking other people for money to fund her "roses." His train arrived at the station 15 minutes later, he walked to his bus stop, and right on time, his bus was arriving. He hopped on and read the paper for about 5 to 10 minutes, then the bus made a routine stop. The doors opened, and can you guess who walked on? None other than the woman from earlier, with a dozen roses in hand. She looked him right in the eye as she passed by to take a seat. He was stunned, unsure how it was possible that she had time to get flowers, and arrive there, when he caught the train before her, and saw her as he pulled away, then caught the first and only bus going in that particular direction. The mystery went unsolved, with no practical explanation. Read the full story here. 10. The magical, refilling dozen boiled eggs. A man described an occasion in which he went to grab breakfast at a small buffet his work had every morning. The kitchen was deserted, as he went to a ceramic egg tray that contained 12 warm, fresh boiled eggs. He took one and went to a nearby garbage bin to peel off the shell, then he turned around, stunned. There were 12 eggs in the tray again. Nobody had entered the room while he was peeling, yet this quantum egg had inexplicably reappeared. Read the full story here. 11. The dreams that warned of disaster. A guy had two intense dreams in the same night. The first was about one of his online friends calling him to discuss her breakup, in which he sang some lines from "Don't Cry" by Seal to her. His second dream was about finding one of his real-life friends dead, floating in the bathtub. The next evening, he logged online and his online friend messaged him saying that she and her boyfriend broke up. He immediately wondered if this meant something. Then, he got a call from his real-life friend, and though they spoke on a regular basis, he was freaked out. They chatted about school and whatnot, then he heard a splash in the background and asked if she was in the bathtub. When she said yes, he was stunned. He asked her, "What did you do?" and she hesitated, before revealing that she'd taken pills, eaten mushrooms, and drank vodka. She'd gotten scared, waiting for it to hit her, so she'd called him for the comfort of hearing someone's voice. He called

911 on her behalf and when help arrived, she was unconscious, but alive. According to the man, she's doing great today. Read the full story here. 12. The mystery man seen sitting on the bench every day. Every morning on his way to work, a man would pass through a shopping mall, and on one of the benches he saw an elderly man who appeared to be waiting for someone. He was a fairly normal-looking guy, though he sometimes wore a piece of attire that wasn't particularly stylish, like a golfer's cap, logger boots, or a vest. Every morning for four years he'd see this man, but he never actually witnessed him meeting anyone. Then he got a new job located on a completely different part of town. Only a few days into working, he walked out of the subway and to his surprise, he saw THE EXACT SAME GUY sitting on a bench. Read the full story here.

A notícia do corpus em inglês apresentada no Exemplo 1 possui a seguinte distribuição de reações: cute:0, ew:2, fail:2, hate:97, lol:4, love:250, omg:74, trashy:0, win:10, wtf:49 e yaaass:5.

Exemplo 2²: 25 Things For People Who Like Comfort More Than People; Hold on, theres a missed call from my sofa. ; We hope you love the products we recommend! Just so you know, BuzzFeed may collect a share of sales from the links on this page. 1. A monthly stylish sock subscription that means one less trip outside. Subscriptions start at \$11.99 at Say it With a Sock. 2. This ostrich pillow that lets you bury your face in sleep mode no matter where you are. Get it for \$46.80 on Amazon. 3. A t-shirt that reminds people how much you're sacrificing for them right now. Get it for \$12.50 from QuirkyShirty on Etsy. 4. A four-foot-long bear pillow that gives the best snuggles. Also comes in fox, chocolate lab, yellow lab, cow, fawn, and cat. Get it for \$79.99 at Wayfair. 5. This shower curtain that lets other people know when it's "me time." Get it for \$80 from The Red Umbrella Shop on Etsy. 6. Hobbit slippers you can wear for the 34 hours it takes to watch the whole Lord of the Rings trilogy. Get them for \$18.99 on Amazon. 7. This shearling cocoon pet bed because sometimes even dogs can't people. Get it for \$79.99 at Amazon. 8. These Edison bulb string lights that are dim and cozy and cinematic. Get them for \$32 at Urban Outfitters. 9. A travel sleep mask for disconnecting. Get it for \$10.69 on Amazon. 10. These sleek stereo headphones because... sorry I can't hear you, I don't want to. Available in Bordeaux Pink, Charcoal Black, Lime Yellow, Cinnabar Red, and Viridian Blue. Get them for \$198 on Amazon. 11. A wearable blanket for giving up in. Get it for \$29.99 on Amazon. 12. A decorative plate that elegantly states the obvious. Get it for \$24.54 from Lou Brown Vintage on Etsy. 13. The perfect wine glass for those not comfortable with sharing. Get it for \$17.99 on Amazon. 14. A bath mat that tells you what human people might forget to. Get it for \$34 at Urban Outfitters. 15. A split-king sleep number bed, like I know you want to cuddle, but I'm a 46, and you're a 28, soooo... Get it for \$3,949.99 on Amazon. 16. A single-person sauna for when you just want to

² <https://www.buzzfeed.com/betsydickerson/things-for-people-who-value-comfort-more-than-human-relat>

have a good sweat and not risk eye contact with unfamiliar balls. Get it for \$1,259.99 at Wayfair. 17. This tee for people who made their decision and are absolutely okay with it. Get it for \$28 at Greeving Cards. 18. This one-person tent for spectators who are here to watch the game, not make friends. Get it for \$168 on Amazon. 19. A super cozy sleeping bag for when you just gotta get away from people and sleep in grass. Get it for \$49.84 on Amazon. 20. A casual 1,079-page read that'll have you canceling all of next year's plans. Get it for \$11.90 from Amazon. 21. This ultra-foofy down alternative comforter for hibernating. Get it for \$29 on Amazon. 22. These toe alignment socks that you'll wear and literally have to fight off people trying to be friends with you. Get them for \$6.99 on Amazon. 23. A blanket for people who want to be where the people aren't. Get it for \$26.99 on Amazon. 24. A floor-mopping robot that lets you stay in bed and watch Swamp People. Get it for \$39 from Firebox. 25. And an enormous foam bag chair that sucks you into its loving quicksand arms and no, you can't get on it with me. Available in ten colors. Get it for \$110 on Amazon. Want even more great product suggestions? Sign up for BuzzFeed's Gift Guide Newsletter! If you can't see the signup box above, just go here to sign up for the BuzzFeed Gift Guide newsletter!

A notícia do corpus em inglês apresentada no Exemplo 2 possui a seguinte distribuição de reações: cute:3, ew:0, fail:0, hate:3, lol:3, love:24, omg:4, trashy:0, win:4, wtf:1 e yaaass:4.

Referências

- ACKLEY, David H; HINTON, Geoffrey E; SEJNOWSKI, Terrence J. A learning algorithm for boltzmann machines. *Cognitive science*, Wiley Online Library, v. 9, n. 1, p. 147–169, 1985. Citado na página 34.
- AGARWAL, Apoorv; SHARMA, Vivek; SIKKA, Geeta; DHIR, Renu. Opinion mining of news headlines using sentiwordnet. In: IEEE. *Symposium on Colossal Data Analysis and Networking (CDAN)*. [S.l.], 2016. p. 1–5. Citado na página 25.
- ALIPANAHI, Babak; DELONG, Andrew; WEIRAUCH, Matthew T; FREY, Brendan J. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, Nature Research, v. 33, n. 8, p. 831–838, 2015. Citado na página 28.
- ALVARES-CHERMAN, Everton; METZ, Jean; MONARD, Maria Carolina. Incorporating label dependency into the binary relevance framework for multi-label classification. *Expert Systems with Applications*, Elsevier, v. 39, n. 2, p. 1647–1655, 2012. Citado na página 38.
- ARANHA, Christian N. *Uma abordagem de pré-processamento automático para mineração de textos em português: sob o enfoque da inteligência computacional*. 2007. 144 f. Tese (Doutorado) — Tese (Doutorado em Engenharia Elétrica)–Programa de Pós-Graduação em Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2007. Citado 3 vezes nas páginas 6, 25 e 26.
- ASGHAR, Muhammad Zubair. Detection and scoring of internet slangs for sentiment analysis using sentiwordnet. *Life Science Journal*, v. 11, n. 9, 2014. Citado na página 25.
- BACCIANELLA, Stefano; ESULI, Andrea; SEBASTIANI, Fabrizio. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: *Proceedings of Language Resources and Evaluation Conference (LREc)*. [S.l.: s.n.], 2010. v. 10, p. 2200–2204. Citado na página 16.
- BAKLIWAL, Akshat; ARORA, Piyush; MADHAPPAN, Senthil; KAPRE, Nikhil; SINGH, Mukesh; VARMA, Vasudeva. Mining sentiments from tweets. *Proceedings of the WASSA*, v. 12, 2012. Citado na página 26.
- BALAHUR, Alexandra; TURCHI, Marco. Improving sentiment analysis in twitter using multilingual machine translated data. In: *International Conference Recent Advances in Natural Language Processing (RANLP)*. [S.l.: s.n.], 2013. p. 49–55. Citado na página 23.
- BALAZS, Jorge A; VELÁSQUEZ, Juan D. Opinion mining and information fusion: a survey. *Information Fusion*, Elsevier, v. 27, p. 95–110, 2016. Citado na página 15.
- BATISTA, Gustavo EAPA; CARVALHO, Andre CPLF; MONARD, Maria Carolina. Applying one-sided selection to unbalanced datasets. In: SPRINGER. *Mexican International Conference on Artificial Intelligence*. [S.l.], 2000. p. 315–325. Citado na página 60.

- BENGIO, Yoshua; LAMBLIN, Pascal; POPOVICI, Dan; LAROCHELLE, Hugo et al. Greedy layer-wise training of deep networks. *Advances in Neural Information Processing Systems (NIPS)*, MIT; 1998, v. 19, p. 153, 2007. Citado na página 34.
- BERMINGHAM, Adam; SMEATON, Alan F. Classifying sentiment in microblogs: is brevity an advantage? In: ACM. *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*. [S.l.], 2010. p. 1833–1836. Citado na página 23.
- BERTHON, Pierre R; PITT, Leyland F; PLANGGER, Kirk; SHAPIRO, Daniel. Marketing meets web 2.0, social media, and creative consumers: Implications for international marketing strategy. *Business horizons*, Elsevier, v. 55, n. 3, p. 261–271, 2012. Citado na página 14.
- BHOWMICK, Plaban Kumar; BASU, Anupam; MITRA, Pabitra. Reader perspective emotion analysis in text through ensemble based multi-label classification framework. *Computer and Information Science*, v. 2, n. 4, p. 64, 2009. Citado 3 vezes nas páginas 18, 52 e 53.
- BHOWMICK, Plaban Kumar; BASU, Anupam; MITRA, Pabitra; PRASAD, Abhishek. Multi-label text classification approach for sentence level news emotion analysis. In: SPRINGER. *Proceedings of the International Conference on Pattern Recognition and Machine Intelligence (PReMI)*. [S.l.], 2009. p. 261–266. Citado 2 vezes nas páginas 52 e 53.
- BI, Wei; KWOK, James Tin-Yau. Efficient multi-label classification with many labels. In: *International Conference on Machine Learning (ICML)*. [S.l.: s.n.], 2013. p. 405–413. Citado 3 vezes nas páginas 18, 41 e 63.
- BOBICEV, Victoria; SOKOLOVA, Marina. Confused and thankful: Multi-label sentiment classification of health forums. In: SPRINGER. *Proceedings of the Canadian Conference on Artificial Intelligence (Canadian AI)*. [S.l.], 2017. p. 284–289. Citado 3 vezes nas páginas 51, 52 e 53.
- BOJANOWSKI, Piotr; GRAVE, Edouard; JOULIN, Armand; MIKOLOV, Tomas. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016. Citado na página 27.
- BORRA, Simone; CIACCIO, Agostino Di. Measuring the prediction error. a comparison of cross-validation, bootstrap and covariance penalty methods. *Computational statistics & data analysis*, Elsevier, v. 54, n. 12, p. 2976–2989, 2010. Citado na página 60.
- BRAVE, Scott; NASS, Clifford. Emotion in human–computer interaction. *Human-Computer Interaction*, p. 53, 2003. Citado na página 23.
- BRUM, Henrico; ARAUJO, Filipe; KEPLER, Fabio. Sentiment analysis for brazilian portuguese over a skewed class corpora. In: SPRINGER. *Proceedings of the International Conference on Computational Processing of the Portuguese Language (Propor)*. [S.l.], 2016. p. 134–138. Citado na página 17.
- BUZZFEED. *How to Use BuzzFeed*. 2017. Disponível em: <https://www.buzzfeed.com/helphow_to_use-q13>. Acesso em: 30 de Novembro de 2017. Citado na página 70.

- CAKIR, Emre; HEITTOLA, Toni; HUTTUNEN, Heikki; VIRTANEN, Tuomas. Multi-label vs. combined single-label sound event detection with deep neural networks. In: IEEE. *Proceedings of European Signal Processing Conference (EUSIPCO)*. [S.l.], 2015. p. 2551–2555. Citado na página 56.
- CAMBRIA, Erik; SCHULLER, Björn; XIA, Yunqing; HAVASI, Catherine. New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, IEEE, v. 28, n. 2, p. 15–21, 2013. Citado 2 vezes nas páginas 15 e 21.
- CARVALHO, Paula; SARMENTO, Luís; SILVA, Mário J; OLIVEIRA, Eugénio De. Clues for detecting irony in user-generated contents: oh...!! it's so easy;- . In: ACM. *Proceedings of the International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*. [S.l.], 2009. p. 53–56. Citado na página 23.
- CHEN, Xinlei; ZITNICK, C Lawrence. Mind's eye: A recurrent visual representation for image caption generation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. p. 2422–2431. Citado na página 53.
- CHEN, Xue-Wen; LIN, Xiaotong. Big data deep learning: challenges and perspectives. *IEEE Access*, IEEE, v. 2, p. 514–525, 2014. Citado na página 16.
- CHENG, Weiwei; HÜLLERMEIER, Eyke; DEMBCZYNSKI, Krzysztof J. Bayes optimal multilabel classification via probabilistic classifier chains. In: *Proceedings of the International Conference on Machine Learning (ICML)*. [S.l.: s.n.], 2010. p. 279–286. Citado na página 39.
- CHERMAN, Everton Alvares; MONARD, Maria Carolina; METZ, Jean. Multi-label problem transformation methods: a case study. *CLEI Electronic Journal*, Centro Latinoamericano de Estudios en Informática, v. 14, n. 1, p. 4–4, 2011. Citado 3 vezes nas páginas 37, 38 e 41.
- CHO, Youngmin; SAUL, Lawrence K. Kernel methods for deep learning. In: *Advances in Neural Information Processing Systems (NIPS)*. [S.l.: s.n.], 2009. p. 342–350. Citado na página 16.
- CLARE, Amanda; KING, Ross D. Knowledge discovery in multi-label phenotype data. In: SPRINGER. *Proceedings of the European Conference on Principles of Data Mining and Knowledge Discovery*. [S.l.], 2001. p. 42–53. Citado na página 42.
- COLLOBERT, Ronan; WESTON, Jason. A unified architecture for natural language processing: Deep neural networks with multitask learning. In: ACM. *Proceedings of the International Conference on Machine Learning (ICML)*. [S.l.], 2008. p. 160–167. Citado na página 16.
- DAVE, Kushal; LAWRENCE, Steve; PENNOCK, David M. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In: ACM. *Proceedings of the International Conference on World Wide Web (WWW)*. [S.l.], 2003. p. 519–528. Citado na página 22.
- DENG, Li; HINTON, Geoffrey; KINGSBURY, Brian. New types of deep neural network learning for speech recognition and related applications: An overview. In: IEEE. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2013. p. 8599–8603. Citado 3 vezes nas páginas 28, 30 e 53.

DESMET, Pieter. Measuring emotion: Development and application of an instrument to measure emotional responses to products. In: *Funology*. [S.l.]: Springer, 2003. p. 111–123. Citado na página 15.

DIEHL, Peter U; PEDRONI, Bruno U; CASSIDY, Andrew; MEROLLA, Paul; NEFTCI, Emre; ZARRELLA, Guido. Truehappiness: Neuromorphic emotion recognition on truenorth. In: IEEE. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2016. p. 4278–4285. Citado 2 vezes nas páginas 16 e 63.

DING, Xiaowen; LIU, Bing; YU, Philip S. A holistic lexicon-based approach to opinion mining. In: ACM. *Proceedings of the International Conference on Web Search and Data Mining (WSDM)*. [S.l.], 2008. p. 231–240. Citado na página 15.

DIX, Alan; FINLAY, Janet; ABOWD, Gregory D; BEALE, Russell. *Human-computer interaction*. [S.l.]: Springer, 2005. Citado 2 vezes nas páginas 14 e 21.

DOSCIATTI, Mariza Miola. *Um método para a identificação de emoções básicas em textos em português do Brasil usando máquinas de vetores de suporte em solução multiclasse. 2015. 223 f.* Tese (Doutorado) — Tese (Doutorado em Informática) – Programa de Pós Graduação em Informática, Pontifícia Universidade Católica do Paraná, 2015. Citado 2 vezes nas páginas 57 e 61.

DOSCIATTI, Mariza Miola; FERREIRA, Lohann Paterno Coutinho; PARAISO, Emerson Cabrera. Anotando um corpus de notícias para a análise de sentimento: um relato de experiência. In: SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. *Proceedings of Symposium in Information and Human Language Technology (STIL)*. [S.l.], 2015. p. 122–130. Citado na página 26.

DUAN, Lei; OYAMA, Satoshi; SATO, Haruhiko; KURIHARA, Masahito. Separate or joint? estimation of multiple labels from crowdsourced annotations. *Expert Systems with Applications*, Elsevier, v. 41, n. 13, p. 5723–5732, 2014. Citado 3 vezes nas páginas 51, 52 e 53.

DUAN, Lei; OYAMA, Satoshi; SATO, Haruhiko; KURIHARA, Masahito. Multi-emotion estimation in narratives from crowdsourced annotations. In: ACM. *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*. [S.l.], 2015. p. 91–100. Citado 3 vezes nas páginas 51, 52 e 53.

EKMAN, Paul; OSTER, Harriet. Facial expressions of emotion. *Annual review of psychology*, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 30, n. 1, p. 527–554, 1979. Citado na página 23.

ELISSEEFF, André; WESTON, Jason et al. A kernel method for multi-labelled classification. In: *Proceedings of Annual Conference on Neural Information Processing Systems (NIPS)*. [S.l.: s.n.], 2001. v. 14, p. 681–687. Citado na página 42.

FELDMAN, Ronen. Techniques and applications for sentiment analysis. *Communications of the ACM*, ACM, v. 56, n. 4, p. 82–89, 2013. Citado na página 24.

FITRIAWAN, Aries; WASITO, Ito; SYAFIANDINI, Arida Ferti; AMIEN, Mukhlis; YANUAR, Arry. Multi-label classification using deep belief networks for virtual screening of multi-target drug. In: IEEE. *Proceedings of International Conference on Computer*,

Control, Informatics and its Applications (IC3INA). [S.l.], 2016. p. 102–107. Citado na página 56.

FITZ-GIBBON, Carol T; MORRIS, Lynn Lyons. *How to analyze data*. [S.l.]: Sage, 1987. Citado 2 vezes nas páginas 71 e 72.

FUKUSHIMA, Kunihiko; MIYAKE, Sei. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In: *Competition and cooperation in neural nets*. [S.l.]: Springer, 1980. p. 267–285. Citado na página 30.

FÜRNKRANZ, Johannes; HÜLLERMEIER, Eyke; MENCÍA, Eneldo Loza; BRINKER, Klaus. Multilabel classification via calibrated label ranking. *Machine learning*, Springer, v. 73, n. 2, p. 133–153, 2008. Citado na página 40.

GAO, Mingchen; XU, Ziyue; LU, Le; HARRISON, Adam P; SUMMERS, Ronald M; MOLLURA, Daniel J. Holistic interstitial lung disease detection using deep convolutional neural networks: Multi-label learning and unordered pooling. *arXiv preprint arXiv:1701.05616*, 2017. Citado na página 56.

GHAMRAWI, Nadia; MCCALLUM, Andrew. Collective multi-label classification. In: *ACM. Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*. [S.l.], 2005. p. 195–200. Citado na página 43.

GHARROUDI, Ouadie; ELGHAZEL, Haytham; AUSSEM, Alex. Ensemble multi-label classification: a comparative study on threshold selection and voting methods. In: *IEEE. International Conference on Tools with Artificial Intelligence (ICTAI)*. [S.l.], 2015. p. 377–384. Citado na página 66.

GO, Alec; BHAYANI, Richa; HUANG, Lei. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, v. 1, n. 12, 2009. Citado na página 26.

GODBOLE, Shantanu; SARAWAGI, Sunita. Discriminative methods for multi-labeled classification. In: *SPRINGER. Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*. [S.l.], 2004. p. 22–30. Citado na página 43.

GONCALVES, Eduardo Corrêa; PLASTINO, Alexandre; FREITAS, Alex A. A genetic algorithm for optimizing the label ordering in multi-label classifier chains. In: *IEEE. Proceedings of the International Conference on Tools with Artificial Intelligence (ICTAI)*. [S.l.], 2013. p. 469–476. Citado na página 40.

GORDON, S. L. The sociology of sentiments and emotion. *Social Psychology: Sociological Perspectives*, M. Rosenberg and R. H. Turner, eds., New York, NY, USA: Basic Books, p. 562–592, 1981. Citado na página 15.

GOUK, Henry; PFAHRINGER, Bernhard; CREE, Michael J. Learning distance metrics for multi-label classification. In: *Proceedings of Asian conference on machine learning*. [S.l.: s.n.], 2016. v. 63, p. 318–333. Citado na página 44.

GRAVES, Alex. Supervised sequence labelling with recurrent neural networks. *ISBN 9783642212703*. URL <http://books.google.com/books>, 2012. Citado na página 17.

GRAVES, Alex; SCHMIDHUBER, Jürgen. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, Elsevier, v. 18, n. 5, p. 602–610, 2005. Citado na página 17.

GUAN, Renchu; WANG, Xu; YANG, Mary Qu; ZHANG, Yu; ZHOU, Fengfeng; YANG, Chen; LIANG, Yanchun. Multi-label deep learning for gene function annotation in cancer pathways. *Scientific reports*, Nature Publishing Group, v. 8, n. 1, p. 267, 2018. Citado na página 56.

GUTHRIE, David; ALLISON, Ben; LIU, Wei; GUTHRIE, Louise; WILKS, Yorick. A closer look at skip-gram modelling. In: SN. *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*. [S.l.], 2006. p. 1–4. Citado na página 27.

HARTMANN, Nathan; FONSECA, Erick; SHULBY, Christopher; TREVISIO, Marcos; RODRIGUES, Jessica; ALUISIO, Sandra. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *arXiv preprint arXiv:1708.06025*, 2017. Citado 2 vezes nas páginas 65 e 74.

HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778. Citado 2 vezes nas páginas 28 e 53.

HE, Xiaodong; DENG, Li. Deep learning for image-to-text generation: A technical overview. *IEEE Signal Processing Magazine*, IEEE, v. 34, n. 6, p. 109–116, 2017. Citado na página 53.

HERRERA, Francisco; CHARTE, Francisco; RIVERA, Antonio J; JESUS, María J Del. *Multilabel Classification: Problem Analysis, Metrics and Techniques*. [S.l.]: Springer, 2016. Citado 2 vezes nas páginas 43 e 61.

HERZIG, Jonathan; FEIGENBLAT, Guy; SHMUELI-SCHEUER, Michal; KONOP-NICKI, David; RAFAELI, Anat; ALTMAN, Daniel; SPIVAK, David. Classifying emotions in customer support dialogues in social media. In: *Proceedings of the SIGDIAL Conference*. [S.l.: s.n.], 2016. p. 64–73. Citado 3 vezes nas páginas 51, 52 e 53.

HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long short-term memory. *Neural Computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997. Citado 2 vezes nas páginas 17 e 31.

HU, Mingqing; LIU, Bing. Mining and summarizing customer reviews. In: ACM. *Proceedings of the International Conference on Knowledge Discovery and Data Mining (SIGKDD)*. [S.l.], 2004. p. 168–177. Citado na página 23.

HUANG, Shu; PENG, Wei; LI, Jingxuan; LEE, Dongwon. Sentiment and topic analysis on social media: a multi-task multi-label classification approach. In: ACM. *Proceedings of the Annual ACM Web Science Conference (WebSci)*. [S.l.], 2013. p. 172–181. Citado na página 50.

HUANG, Yan; WANG, Wei; WANG, Liang; TAN, Tieniu. Multi-task deep neural network for multi-label learning. In: IEEE. *Proceedings of International Conference on Image Processing (ICIP)*. [S.l.], 2013. p. 2897–2900. Citado na página 55.

HUBEL, David H; WIESEL, Torsten N. Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*, Wiley Online Library, v. 148, n. 3, p. 574–591, 1959. Citado na página 30.

- HUBEL, David H; WIESEL, Torsten N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, Wiley Online Library, v. 160, n. 1, p. 106–154, 1962. Citado na página 30.
- IRSOY, Ozan; CARDIE, Claire. Opinion mining with deep recurrent neural networks. In: *Proceedings of Conference on Empirical Methods on Natural Language Processing (EMNLP)*. [S.l.: s.n.], 2014. p. 720–728. Citado 3 vezes nas páginas 28, 31 e 53.
- IVAKHNENKO, A. G.; LAPA, V. G. *Cybernetic predicting devices*. [S.l.], 1966. Citado na página 30.
- JAIN, Anil K; MAO, Jianchang; MOHIUDDIN, K Moidin. Artificial neural networks: A tutorial. *Computer*, IEEE, v. 29, n. 3, p. 31–44, 1996. Citado 3 vezes nas páginas 6, 29 e 30.
- JOHNSON, Melvin; SCHUSTER, Mike; LE, Quoc V; KRIKUN, Maxim; WU, Yonghui; CHEN, Zhifeng; THORAT, Nikhil; VIÉGAS, Fernanda; WATTENBERG, Martin; CORRADO, Greg et al. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, v. 5, p. 339–351, 2017. Citado na página 53.
- KARALAS, Konstantinos; TSAGKATAKIS, Grigorios; ZERVAKIS, Michalis; TSAKALIDES, Panagiotis. Deep learning for multi-label land cover classification. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *SPIE Remote Sensing*. [S.l.], 2015. p. 96430Q–96430Q. Citado na página 55.
- KARLIK, Bekir; OLGAC, A Vehbi. Performance analysis of various activation functions in generalized mlp architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems*, v. 1, n. 4, p. 111–122, 2011. Citado na página 29.
- KHAN, Asad Ullah Rafiq; KHAN, Madiha; KHAN, Mohammad Badruddin. Naïve multi-label classification of youtube comments using comparative opinion mining. *Procedia Computer Science*, Elsevier, v. 82, p. 57–64, 2016. Citado na página 50.
- KIM, Ji-Hyun. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational statistics & data analysis*, Elsevier, v. 53, n. 11, p. 3735–3745, 2009. Citado na página 60.
- KIM, Soo-Min; HOVY, Eduard. Determining the sentiment of opinions. In: ACL. *Proceedings of the International Conference on Computational Linguistics (COLING)*. [S.l.], 2004. p. 1367. Citado na página 25.
- KIM, Yoon. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014. Citado na página 59.
- KINGHORN, Philip; ZHANG, Li; SHAO, Ling. A region-based image caption generator with refined descriptions. *Neurocomputing*, Elsevier, v. 272, p. 416–424, 2018. Citado 2 vezes nas páginas 18 e 55.
- KOHAVI, Ron. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: MONTREAL, CANADA. *Ijcai*. [S.l.], 1995. v. 14, n. 2, p. 1137–1145. Citado na página 60.

- KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems (NIPS)*. [S.l.: s.n.], 2012. p. 1097–1105. Citado na página 31.
- KUKANOV, Ivan; HAUTAMÄKI, Ville; SINISCALCHI, Sabato Marco; LI, Kehuang. Deep learning with maximal figure-of-merit cost to advance multi-label speech attribute detection. In: IEEE. *Proceedings of Spoken Language Technology Workshop (SLT)*. [S.l.], 2016. p. 489–495. Citado na página 56.
- LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. *Nature*, Nature Research, v. 521, n. 7553, p. 436–444, 2015. Citado 4 vezes nas páginas 16, 28, 57 e 63.
- LECUN, Yann; BOSER, Bernhard E; DENKER, John S; HENDERSON, Donnie; HOWARD, RE; HUBBARD, Wayne E; JACKEL, Lawrence D. Handwritten digit recognition with a back-propagation network. In: *Advances in Neural Information Processing Systems (NIPS)*. [S.l.: s.n.], 1990. p. 396–404. Citado na página 31.
- LENC, Ladislav; KRÁL, Pavel. Deep neural networks for czech multi-label document classification. In: *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*. [S.l.: s.n.], 2016. Citado na página 56.
- LENC, Ladislav; KRÁL, Pavel. Deep neural networks for czech multi-label document classification. *arXiv preprint arXiv:1701.03849*, 2017. Citado na página 56.
- LI, Jun; RAO, Yanghui; JIN, Fengmei; CHEN, Huijun; XIANG, Xiyun. Multi-label maximum entropy model for social emotion classification over short text. *Neurocomputing*, Elsevier, v. 210, p. 247–256, 2016. Citado 3 vezes nas páginas 50, 52 e 53.
- LI, Xin; XIE, Haoran; RAO, Yanghui; CHEN, Yanjia; LIU, Xuebo; HUANG, Huan; WANG, Fu Lee. Weighted multi-label classification model for sentiment analysis of online news. In: IEEE. *Proceedings of International Conference on Big Data and Smart Computing (BigComp)*. [S.l.], 2016. p. 215–222. Citado 3 vezes nas páginas 50, 52 e 53.
- LIM, Kar Wai; BUNTINE, Wray. Twitter opinion topic model: Extracting product opinions from tweets by leveraging hashtags and sentiment lexicon. In: ACM. *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*. [S.l.], 2014. p. 1319–1328. Citado na página 50.
- LIN, Shuying; XIE, Huosheng; YU, Liang-Chih; LAI, K Robert. Sentinlp at ijcnlp-2017 task 4: Customer feedback analysis using a bi-lstm-cnn model. *Proceedings of the 8th International Joint Conference on Natural Language Processing (IJCNLP)*, p. 149–154, 2017. Citado na página 56.
- LIU, Bing. Sentiment analysis and subjectivity. In: *Handbook of Natural Language Processing, Second Edition*. [S.l.]: Chapman and Hall/CRC, 2010. p. 627–666. Citado na página 24.
- LIU, Bing. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, Morgan & Claypool Publishers, v. 5, n. 1, p. 1–167, 2012. Citado 3 vezes nas páginas 15, 22 e 52.

- LIU, Shuhua Monica; CHEN, Jiun-Hung. A multi-label classification based approach for sentiment classification. *Expert Systems with Applications*, Elsevier, v. 42, n. 3, p. 1083–1093, 2015. Citado 7 vezes nas páginas 18, 19, 42, 49, 52, 53 e 60.
- LUYCKX, Kim; VAASSEN, Frederik; PEERSMAN, Claudia; DAELEMANS, Walter. Fine-grained emotion detection in suicide notes: A thresholding approach to multi-label classification. *Biomedical informatics insights*, Libertas Academica, v. 5, n. Suppl 1, p. 61, 2012. Citado 3 vezes nas páginas 50, 52 e 53.
- MADJAROV, Gjorgji; GJORGJEVIKJ, Dejan; DELEV, Tomche. Efficient two stage voting architecture for pairwise multi-label classification. In: SPRINGER. *Proceedings of Australasian Joint Conference on Artificial Intelligence (AI)*. [S.l.], 2010. p. 164–173. Citado na página 40.
- MADJAROV, Gjorgji; GJORGJEVIKJ, Dejan; DŽEROSKI, Sašo. Two stage architecture for multi-label learning. *Pattern Recognition*, Elsevier, v. 45, n. 3, p. 1019–1034, 2012. Citado na página 40.
- MAHDAVI-SHAHRI, Amirreza; HOUSHMAND, Mahboobeh; YAGHOUBI, Mahdi; JALALI, Mehrdad. Applying an ensemble learning method for improving multi-label classification performance. In: IEEE. *International Conference of Signal Processing and Intelligent Systems (ICSPIS)*. [S.l.], 2016. p. 1–6. Citado na página 41.
- MARTINAZZO, Barbara. *Um método de identificação de emoções em textos curtos para o português do Brasil. 2010. 68 f.* Tese (Doutorado) — Dissertação (Mestrado em Informática) – Programa de Pós Graduação em Informática, Pontifícia Universidade Católica do Paraná, 2010. Citado na página 57.
- MAYNARD, Diana; GREENWOOD, Mark A. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In: *Proceedings of Language Resources and Evaluation Conference (LREc)*. [S.l.: s.n.], 2014. p. 4238–4243. Citado na página 23.
- MCCULLOCH, Warren S; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, n. 4, p. 115–133, 1943. Citado na página 29.
- MEDHAT, Walaa; HASSAN, Ahmed; KORASHY, Hoda. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, Elsevier, v. 5, n. 4, p. 1093–1113, 2014. Citado 5 vezes nas páginas 6, 16, 24, 25 e 28.
- MELAMUD, Oren; GOLDBERGER, Jacob; DAGAN, Ido. context2vec: Learning generic context embedding with bidirectional lstm. In: *Proceedings of The SIGNLL Conference on Computational Natural Language Learning (CoNLL)*. [S.l.: s.n.], 2016. p. 51–61. Citado na página 65.
- MENCÍA, Eneldo Loza; PARK, Sang-Hyeun; FÜRNKRANZ, Johannes. Efficient voting prediction for pairwise multilabel classification. *Neurocomputing*, Elsevier, v. 73, n. 7, p. 1164–1176, 2010. Citado na página 40.
- MIKOLOV, Tomas; CHEN, Kai; CORRADO, Greg; DEAN, Jeffrey. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. Citado 2 vezes nas páginas 27 e 65.

- MIKOLOV, Tomas; SUTSKEVER, Ilya; CHEN, Kai; CORRADO, Greg S; DEAN, Jeff. Distributed representations of words and phrases and their compositionality. In: *Proceedings of the Neural Information Processing Systems Conference (NIPS)*. [S.l.: s.n.], 2013. p. 3111–3119. Citado 2 vezes nas páginas 16 e 27.
- MUNEZERO, Myriam D; MONTERO, Calkin Suero; SUTINEN, Erkki; PAJUNEN, John. Are they different? affect, feeling, emotion, sentiment, and opinion detection in text. *IEEE transactions on affective computing*, IEEE, v. 5, n. 2, p. 101–111, 2014. Citado 2 vezes nas páginas 15 e 23.
- MUNKHDALAI, Tsendsuren; YU, Hong. Neural tree indexers for text understanding. *arXiv preprint arXiv:1607.04492*, 2016. Citado na página 17.
- NASUKAWA, Tetsuya; YI, Jeonghee. Sentiment analysis: Capturing favorability using natural language processing. In: ACM. *Proceedings of the International Conference on Knowledge Capture (K-CAP)*. [S.l.], 2003. p. 70–77. Citado na página 22.
- O'REILLY, Tim. What is web 2.0: Design patterns and business models for the next generation of software. communications & strategies, 2007. Citado na página 22.
- OUYANG, Wanli; WANG, Xiaogang. Joint deep learning for pedestrian detection. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2013. p. 2056–2063. Citado na página 28.
- PAK, Alexander; PAROUBEK, Patrick. Twitter as a corpus for sentiment analysis and opinion mining. In: *Proceedings of Language Resources and Evaluation Conference (LREc)*. [S.l.: s.n.], 2010. v. 10, n. 2010. Citado 2 vezes nas páginas 15 e 22.
- PANG, Bo; LEE, Lillian. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In: ACL. *Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL)*. [S.l.], 2005. p. 115–124. Citado na página 24.
- PANG, Bo; LEE, Lillian et al. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, Now Publishers, Inc., v. 2, n. 1–2, p. 1–135, 2008. Citado na página 22.
- PANG, Bo; LEE, Lillian; VAITHYANATHAN, Shivakumar. Thumbs up?: sentiment classification using machine learning techniques. In: ACL. *Proceedings of the ACL Conference on Empirical Methods in Natural Language Processing (EMNLP)*. [S.l.], 2002. p. 79–86. Citado na página 26.
- PENNINGTON, Jeffrey; SOCHER, Richard; MANNING, Christopher. Glove: Global vectors for word representation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. [S.l.: s.n.], 2014. p. 1532–1543. Citado na página 27.
- PESTIAN, John P; MATYKIEWICZ, Pawel; LINN-GUST, Michelle; SOUTH, Brett; UZUNER, Ozlem; WIEBE, Jan; COHEN, K Bretonnel; HURDLE, John; BREW, Christopher. Sentiment analysis of suicide notes: A shared task. *Biomedical informatics insights*, Libertas Academica, v. 5, n. Suppl 1, p. 3, 2012. Citado na página 50.

- PHAM, Thi-Ngan; PHAN, Thi-Thom; NGUYEN, Phuoc-Thao; HA, Quang-Thuy. Hidden topic models for multi-label review classification: An experimental study. In: SPRINGER. *Proceedings of International Conference on Computational Collective Intelligence (ICCCI)*. [S.l.], 2013. p. 603–611. Citado na página 50.
- PHAN, Duc-Anh; SHINDO, Hiroyuki; MATSUMOTO, Yuji. Multiple emotions detection in conversation transcripts. *Proceedings of the Pacific Asia Conference on Language, Information and Computation (PACLIC)*, p. 85, 2016. Citado 3 vezes nas páginas 51, 52 e 53.
- PICARD, Rosalind W. *Affective computing*. [S.l.: s.n.], 1997. Citado 2 vezes nas páginas 14 e 21.
- PICARD, Rosalind W. Affective computing for hci. In: *Proceedings of International Conference on Human-Computer Interaction (HCI)*. [S.l.: s.n.], 1999. p. 829–833. Citado na página 21.
- PLUTCHIK, Robert. Emotions: A general psychoevolutionary theory. *Approaches to emotion*, v. 1984, p. 197–219, 1984. Citado na página 23.
- PORIA, Soujanya; CAMBRIA, Erik; HOWARD, Newton; HUANG, Guang-Bin; HUSSAIN, Amir. Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing*, Elsevier, v. 174, p. 50–59, 2016. Citado na página 21.
- PURVER, Matthew; BATTERSBY, Stuart. Experimenting with distant supervision for emotion classification. In: ACL. *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. [S.l.], 2012. p. 482–491. Citado na página 26.
- QUAN, Changqin; REN, Fuji. Construction of a blog emotion corpus for chinese emotional expression analysis. In: ACL. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. [S.l.], 2009. p. 1446–1454. Citado na página 48.
- QUAN, Changqin; REN, Fuji. A blog emotion corpus for emotional expression analysis in chinese. *Computer Speech & Language*, Elsevier, v. 24, n. 4, p. 726–749, 2010. Citado 5 vezes nas páginas 6, 19, 49, 50 e 52.
- RAVI, Kumar; RAVI, Vadlamani. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, Elsevier, v. 89, p. 14–46, 2015. Citado na página 15.
- READ, Jesse; MARTINO, Luca; LUENGO, David. Efficient monte carlo methods for multi-dimensional learning with classifier chains. *Pattern Recognition*, Elsevier, v. 47, n. 3, p. 1535–1546, 2014. Citado na página 40.
- READ, Jesse; PEREZ-CRUZ, Fernando. Deep learning for multi-label classification. *arXiv preprint arXiv:1502.05988*, 2014. Citado na página 56.
- READ, Jesse; PFAHRINGER, Bernhard; HOLMES, Geoff; FRANK, Eibe. Classifier chains for multi-label classification. In: SPRINGER. *Proceedings of Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*. [S.l.], 2009. p. 254–269. Citado na página 39.

- READ, Jesse; PFAHRINGER, Bernhard; HOLMES, Geoff; FRANK, Eibe. Classifier chains for multi-label classification. *Machine learning*, Springer, v. 85, n. 3, p. 333, 2011. Citado na página 39.
- READ, Jesse; PUURULA, Antti; BIFET, Albert. Multi-label classification with meta-labels. In: IEEE. *Proceedings of IEEE International Conference on Data Mining (ICDM)*. [S.l.], 2014. p. 941–946. Citado na página 44.
- READ, Jesse; REUTEMANN, Peter; PFAHRINGER, Bernhard; HOLMES, Geoff. Meka: a multi-label/multi-target extension to weka. *The Journal of Machine Learning Research*, JMLR. org, v. 17, n. 1, p. 667–671, 2016. Citado na página 61.
- REED, Scott; AKATA, Zeynep; YAN, Xinchun; LOGESWARAN, Lajanugen; SCHIELE, Bernt; LEE, Honglak. Generative adversarial text to image synthesis. In: *Proceedings of The International Conference on Machine Learning (ICML)*. [S.l.: s.n.], 2016. v. 3. Citado na página 28.
- REFAEILZADEH, Payam; TANG, Lei; LIU, Huan. Cross-validation. In: *Encyclopedia of database systems*. [S.l.]: Springer, 2009. p. 532–538. Citado na página 60.
- REIS, Willian Augusto Dias dos. *Um Método de Identificação de Emoções Baseado na Mineração de Padrões Frequentes. 2017*. Tese (Doutorado) — Dissertação (Mestrado em Informática) – Programa de Pós Graduação em Informática, Pontifícia Universidade Católica do Paraná, 2017. Citado na página 57.
- ROKACH, Lior; SCHCLAR, Alon; ITACH, Ehud. Ensemble methods for multi-label classification. *Expert Systems with Applications*, Elsevier, v. 41, n. 16, p. 7507–7523, 2014. Citado na página 66.
- ROSA, Renata L; RODRIGUEZ, Demsteneso Z; BRESSAN, Graça. Music recommendation system based on user’s sentiments extracted from social networks. *IEEE Transactions on Consumer Electronics*, IEEE, v. 61, n. 3, p. 359–367, 2015. Citado na página 21.
- ROSENBLATT, Frank. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958. Citado na página 29.
- RUDER, Sebastian; GHAFARI, Parsa; BRESLIN, John G. Insight-1 at semeval-2016 task 5: Deep learning for multilingual aspect-based sentiment analysis. *arXiv preprint arXiv:1609.02748*, 2016. Citado na página 50.
- SALAKHUTDINOV, Ruslan; HINTON, Geoffrey E. Deep boltzmann machines. In: *Proceedings of International Conference on Artificial Intelligence and Statistics (AISTATS)*. [S.l.: s.n.], 2009. v. 1, p. 3. Citado na página 34.
- SALTON, Gerard; BUCKLEY, Christopher. Term-weighting approaches in automatic text retrieval. *Information processing & management*, Elsevier, v. 24, n. 5, p. 513–523, 1988. Citado na página 27.
- SANTOS, Cícero Nogueira Dos; GATTI, Maira. Deep convolutional neural networks for sentiment analysis of short texts. In: *Proceedings of International Conference on Computational Linguistics (COLING)*. [S.l.: s.n.], 2014. p. 69–78. Citado 2 vezes nas páginas 17 e 59.

- SCHAPIRE, Robert E; SINGER, Yoram. Boostexter: A boosting-based system for text categorization. *Machine learning*, Springer, v. 39, n. 2-3, p. 135–168, 2000. Citado na página 43.
- SCHMIDHUBER, Jürgen. Deep learning in neural networks: An overview. *Neural networks*, Elsevier, v. 61, p. 85–117, 2015. Citado 2 vezes nas páginas 28 e 30.
- SCHOUTEN, Kim; FRASINCAR, Flavius. Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 28, n. 3, p. 813–830, 2016. Citado na página 24.
- SHEN, Dinggang; WU, Guorong; SUK, Heung-II. Deep learning in medical image analysis. *Annual review of biomedical engineering*, Annual Reviews, v. 19, p. 221–248, 2017. Citado na página 53.
- SHI, Shaohuai; WANG, Qiang; XU, Pengfei; CHU, Xiaowen. Benchmarking state-of-the-art deep learning software tools. *arXiv preprint arXiv:1608.07249*, 2016. Citado na página 36.
- SOCHER, Richard; HUVAL, Brody; MANNING, Christopher D; NG, Andrew Y. Semantic compositionality through recursive matrix-vector spaces. In: *ACL. Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. [S.l.], 2012. p. 1201–1211. Citado na página 17.
- SOCHER, Richard; PERELYGIN, Alex; WU, Jean Y; CHUANG, Jason; MANNING, Christopher D; NG, Andrew Y; POTTS, Christopher et al. Recursive deep models for semantic compositionality over a sentiment treebank. In: *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*. [S.l.: s.n.], 2013. v. 1631, p. 1642. Citado 2 vezes nas páginas 17 e 24.
- SONG, Kaisong; FENG, Shi; GAO, Wei; WANG, Daling; CHEN, Ling; ZHANG, Chengqi. Build emotion lexicon from microblogs by combining effects of seed words and emoticons in a heterogeneous graph. In: *ACM. Proceedings of the ACM Conference on Hypertext & Social Media*. [S.l.], 2015. p. 283–292. Citado 3 vezes nas páginas 49, 52 e 53.
- SORIANO, Domingo Ribeiro; GARRIGOS-SIMON, Fernando J; ALCAMÍ, Rafael Lapedra; RIBERA, Teresa Barberá. Social networks and web 3.0: their impact on the management and marketing of organizations. *Management Decision*, Emerald Group Publishing Limited, v. 50, n. 10, p. 1880–1890, 2012. Citado na página 14.
- SUTSKEVER, Ilya; MARTENS, James; HINTON, Geoffrey E. Generating text with recurrent neural networks. In: *Proceedings of the International Conference on Machine Learning (ICML)*. [S.l.: s.n.], 2011. p. 1017–1024. Citado na página 16.
- SUTSKEVER, Ilya; VINYALS, Oriol; LE, Quoc V. Sequence to sequence learning with neural networks. In: *Advances in Neural Information Processing Systems (NIPS)*. [S.l.: s.n.], 2014. p. 3104–3112. Citado na página 53.
- TANG, Duyu; QIN, Bing; LIU, Ting. Deep learning for sentiment analysis: successful approaches and future challenges. *Wiley Interdisciplinary Reviews: Data Mining and*

Knowledge Discovery, Wiley Online Library, v. 5, n. 6, p. 292–303, 2015. Citado 4 vezes nas páginas 16, 53, 57 e 63.

TANG, Duyu; WEI, Furu; YANG, Nan; ZHOU, Ming; LIU, Ting; QIN, Bing. Learning sentiment-specific word embedding for twitter sentiment classification. In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. [S.l.: s.n.], 2014. p. 1555–1565. Citado na página 65.

TAO, Jianhua; TAN, Tieniu. Affective computing: A review. In: SPRINGER. *Proceedings of International Conference on Affective computing and intelligent interaction (ACII)*. [S.l.], 2005. p. 981–995. Citado na página 14.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 2006. Citado 3 vezes nas páginas 18, 37 e 67.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis; VLAHAVAS, Ioannis. Effective and efficient multilabel classification in domains with large number of labels. In: *Proceedings of Workshop on Mining Multidimensional Data (MMD)*. [S.l.: s.n.], 2008. p. 30–44. Citado na página 39.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis; VLAHAVAS, Ioannis. Mining multi-label data. In: *Data mining and Knowledge Discovery handbook*. [S.l.]: Springer, 2009. p. 667–685. Citado na página 37.

TSOUMAKAS, Grigorios; KATAKIS, Ioannis; VLAHAVAS, Ioannis. Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 23, n. 7, p. 1079–1089, 2011. Citado 2 vezes nas páginas 41 e 66.

TSOUMAKAS, Grigorios; SPYROMITROS-XIOUFIS, Eleftherios; VILCEK, Jozef; VLAHAVAS, Ioannis. Mulan: A java library for multi-label learning. *Journal of Machine Learning Research*, v. 12, n. Jul, p. 2411–2414, 2011. Citado na página 61.

TSOUMAKAS, Grigorios; VLAHAVAS, Ioannis. Random k-labelsets: An ensemble method for multilabel classification. In: SPRINGER. *European Conference on Machine Learning (ECML)*. [S.l.], 2007. p. 406–417. Citado 3 vezes nas páginas 40, 43 e 66.

TURNER, Matthew D; CHAKRABARTI, Chayan; JONES, Thomas B; XU, Jiawei F; FOX, Peter T; LUGER, George F; LAIRD, Angela R; TURNER, Jessica A. Automated annotation of functional imaging experiments via multi-label classification. *Frontiers in neuroscience*, Frontiers, v. 7, p. 240, 2013. Citado na página 43.

VEMBU, Shankar; GÄRTNER, Thomas. Label ranking algorithms: A survey. In: *Preference Learning*. [S.l.]: Springer, 2010. p. 45–64. Citado na página 40.

VINCENT, Pascal; LAROCHELLE, Hugo; BENGIO, Yoshua; MANZAGOL, Pierre-Antoine. Extracting and composing robust features with denoising autoencoders. In: ACM. *Proceedings of the International Conference on Machine Learning (ICML)*. [S.l.], 2008. p. 1096–1103. Citado na página 34.

WANG, Chu; FENG, Shi; WANG, Daling; ZHANG, Yifei. Fuzzy-rough set based multi-labeled emotion intensity analysis for sentence, paragraph and document. In: SPRINGER. *Proceedings of the National CCF Conference on Natural Language*

Processing and Chinese Computing. [S.l.], 2015. p. 444–452. Citado 3 vezes nas páginas 49, 52 e 53.

WANG, Jiang; YANG, Yi; MAO, Junhua; HUANG, Zhiheng; HUANG, Chang; XU, Wei. Cnn-rnn: A unified framework for multi-label image classification. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 2285–2294. Citado na página 55.

WANG, Lei; MIAO, Duoqian; ZHAO, Cairong. Chinese emotion recognition based on three-way decisions. In: SPRINGER. *Proceedings of the International Conference on Rough Sets and Knowledge Technology (RSKT)*. [S.l.], 2015. p. 299–308. Citado 3 vezes nas páginas 48, 52 e 53.

WANG, Lei; REN, Fuji; MIAO, Duoqian. A novel method for recognizing emotions of weblog sentences. In: IEEE/SICE. *Proceedings of the International Symposium on System Integration (SII)*. [S.l.], 2013. p. 358–363. Citado 3 vezes nas páginas 48, 52 e 53.

WANG, Lei; REN, Fuji; MIAO, Duoqian. Multi-label emotion recognition of weblog sentence based on bayesian networks. *IEEJ Transactions on Electrical and Electronic Engineering*, Wiley Online Library, v. 11, n. 2, p. 178–184, 2016. Citado 3 vezes nas páginas 50, 52 e 53.

WANG, Mingqiang; LIU, Mengting; FENG, Shi; WANG, Daling; ZHANG, Yifei. A novel calibrated label ranking based method for multiple emotions detection in chinese microblogs. In: *Natural Language Processing and Chinese Computing (NLPCC)*. [S.l.]: Springer, 2014. p. 238–250. Citado 3 vezes nas páginas 49, 52 e 53.

WANG, Yaqi; FENG, Shi; WANG, Daling; YU, Ge; ZHANG, Yifei. Multi-label chinese microblog emotion classification via convolutional neural network. In: SPRINGER. *Asia-Pacific Web Conference*. [S.l.], 2016. p. 567–580. Citado 4 vezes nas páginas 48, 52, 53 e 56.

WEI, Yunchao; XIA, Wei; HUANG, Junshi; NI, Bingbing; DONG, Jian; ZHAO, Yao; YAN, Shuicheng. Cnn: Single-label to multi-label. *arXiv preprint arXiv:1406.5726*, 2014. Citado na página 19.

WEI, Yunchao; XIA, Wei; LIN, Min; HUANG, Junshi; NI, Bingbing; DONG, Jian; ZHAO, Yao; YAN, Shuicheng. Hcp: A flexible cnn framework for multi-label image classification. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 38, n. 9, p. 1901–1907, 2016. Citado 6 vezes nas páginas 6, 18, 42, 55, 56 e 63.

WEISZFLOG, Walter. *Dicionário Brasileiro da Língua Portuguesa*. [S.l.]: Editora Melhoramentos Ltda., 2017. Citado na página 16.

WESTON, Jason; CHOPRA, Sumit; BORDES, Antoine. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014. Citado na página 17.

WU, Yunong; KITA, Kenji; MATSUMOTO, Kazuyuki; KANG, Xin. A joint prediction model for multiple emotions analysis in sentences. In: SPRINGER. *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*. [S.l.], 2013. p. 149–160. Citado 3 vezes nas páginas 48, 52 e 53.

- WU, Ye; REN, Fuji. A corpus-based multi-label emotion classification using maximum entropy. 2009. Citado 3 vezes nas páginas 48, 52 e 53.
- WU, Ye; REN, Fuji. Simple linguistic processing effect on multi-label emotion classification. In: IEEE. *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE)*. [S.l.], 2009. p. 1–5. Citado 3 vezes nas páginas 48, 52 e 53.
- XU, Ruifeng; CHEN, Tao; XIA, Yunqing; LU, Qin; LIU, Bin; WANG, Xuan. Word embedding composition for data imbalances in sentiment and emotion classification. *Cognitive Computation*, Springer, v. 7, n. 2, p. 226–240, 2015. Citado 3 vezes nas páginas 49, 52 e 53.
- YAMAMOTO, Shuhei; SATOH, Tetsuji. Two phase extraction method for multi-label classification of real life tweets. In: ACM. *Proceedings of Proceedings of International Conference on Information Integration and Web-based Applications & Services (iiWAS)*. [S.l.], 2013. p. 16. Citado na página 50.
- YANG, Dingqi; ZHANG, Daqing; YU, Zhiyong; WANG, Zhu. A sentiment-enhanced personalized location recommendation system. In: ACM. *Proceedings of the ACM Conference on Hypertext and Social Media (HT)*. [S.l.], 2013. p. 119–128. Citado na página 21.
- YANG, Hao; ZHOU, Joey Tianyi; CAI, Jianfei; ONG, Yew Soon. Mimi-fcn+: Multi-instance multi-label learning via fully convolutional networks with privileged information. *arXiv preprint arXiv:1702.08681*, 2017. Citado 2 vezes nas páginas 55 e 63.
- YE, Lu; XU, Rui-Feng; XU, Jun. Emotion prediction of news articles from reader's perspective based on multi-label classification. In: IEEE. *Proceedings of International Conference on Machine Learning and Cybernetics (ICMLC)*. [S.l.], 2012. v. 5, p. 2019–2024. Citado 3 vezes nas páginas 50, 52 e 53.
- YIM, Soobin; WARSCHAUER, Mark. Web-based collaborative writing in l2 contexts: Methodological insights from text mining. *Language Learning Technology*, v. 21, n. 1, p. 146—165, 2017. Citado na página 24.
- YU, Qinghua; WANG, Jinjun; ZHANG, Shizhou; GONG, Yihong; ZHAO, Jizhong. Combining local and global hypotheses in deep neural network for multi-label image classification. *Neurocomputing*, Elsevier, v. 235, p. 38–45, 2017. Citado 2 vezes nas páginas 55 e 63.
- ZABIN, Jeff; JEFFERIES, Alex. Social media monitoring and analysis: Generating consumer insights from online conversation. *Aberdeen Group Benchmark Report*, v. 37, n. 9, 2008. Citado na página 22.
- ZENG, Xinchuan; MARTINEZ, Tony R. Distribution-balanced stratified cross-validation for accuracy estimation. *Journal of Experimental & Theoretical Artificial Intelligence*, Taylor & Francis, v. 12, n. 1, p. 1–12, 2000. Citado na página 60.
- ZHANG, Min-Ling; ZHOU, Zhi-Hua. A k-nearest neighbor based algorithm for multi-label classification. In: IEEE. *Proceedings of the IEEE International Conference on Granular Computing*. [S.l.], 2005. v. 2, p. 718–721. Citado na página 42.

- ZHANG, Min-Ling; ZHOU, Zhi-Hua. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE transactions on Knowledge and Data Engineering*, IEEE, v. 18, n. 10, p. 1338–1351, 2006. Citado na página 43.
- 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. Citado na página 42.
- ZHANG, Min-Ling; ZHOU, Zhi-Hua. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, IEEE, v. 26, n. 8, p. 1819–1837, 2014. Citado 7 vezes nas páginas 6, 18, 37, 38, 42, 43 e 44.
- ZHANG, Xiang; LECUN, Yann. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*, 2015. Citado na página 59.
- ZHANG, Xiao; LI, Wenzhong; LU, Sanglu. Emotion detection in online social network based on multi-label learning. In: SPRINGER. *Proceedings of the International Conference on Database Systems for Advanced Applications (DASFAA)*. [S.l.], 2017. p. 659–674. Citado 3 vezes nas páginas 50, 52 e 53.
- ZHANG, Zixing; GEIGER, Jürgen; POHJALAINEN, Jouni; MOUSA, Amr El-Desoky; SCHULLER, Björn. Deep learning for environmentally robust speech recognition: An overview of recent developments. *arXiv preprint arXiv:1705.10874*, 2017. Citado na página 53.
- ZHAO, Fang; HUANG, Yongzhen; WANG, Liang; TAN, Tieniu. Deep semantic ranking based hashing for multi-label image retrieval. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. p. 1556–1564. Citado na página 55.
- ZHOU, Tianyi; TAO, Dacheng. Multi-label subspace ensemble. In: *Artificial Intelligence and Statistics*. [S.l.: s.n.], 2012. p. 1444–1452. Citado na página 66.
- ZWAAN, Janneke M van der; LEEMANS, Inger; KUIJPERS, Erika; MAKS, Isa. Heem, a complex model for mining emotions in historical text. In: IEEE. *Proceedings of the International Conference on e-Science (e-Science)*. [S.l.], 2015. p. 22–30. Citado 3 vezes nas páginas 51, 52 e 53.