UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO

Larissa Feliciana da Silva Britto

Inferência Automática de Nível de Dificuldade de Receitas Culinárias Usando Técnicas de Processamento de Linguagem Natural

Recife

2020

Larissa Feliciana da Silva Britto

Inferência Automática de Nível de Dificuldade de Receitas Culinárias Usando Técnicas de Processamento de Linguagem Natural

Monografia apresentada ao Curso de Bacharelado em Ciências da Computação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Ciências da Computação.

Universidade Federal Rural de Pernambuco – UFRPE

Departamento de Computação

Curso de Bacharelado em Ciências da Computação

Orientador: Luciano Demétrio Santos Pacífico

Coorientador: Teresa Bernarda Ludermir

Recife

2020

Dados Internacionais de Catalogação na Publicação
Universidade Federal Rural de Pernambuco
Sistema Integrado de Bibliotecas
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

B862i Britto, Larissa
Inferência Automática do Nível de Dificuldade em Receitas Culinárias usando Técnicas de Processamento de Linguagem Natural / Larissa Britto. - 2020.
23 f. : il.

Orientador: Luciano Demetrio Santos Pacifico.
Coorientadora: Teresa Bernarda Ludermir.
Inclui referências.

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal Rural de Pernambuco, Bacharelado em Ciência da Computação, Recife, 2020.

1. Nível de Dificuldade em Receitas Culinárias. 2. Processamento de Linguagem Natural. 3. Aprendizagem de Máquina. I. Pacifico, Luciano Demetrio Santos, orient. II. Ludermir, Teresa Bernarda, coorient. III. Título

CDD 004

MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO (UFRPE)
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

http://www.bcc.ufrpe.br

FICHA DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO

Trabalho defendido por Larissa Feliciana da Silva Britto às 14 horas do dia 21 de dezembro de 2020, no link meet.google.com/tcg-qdwb-jnq, como requisito para conclusão do curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, intitulado Inferência Automática do Nível de Dificuldade em Receitas Culinárias usando Técnicas de Processamento de Linguagem Natural, orientado por Luciano Demétrio Santos Pacífico e aprovado pela seguinte banca examinadora:

Luciano Demétrio Santos Pacífico
DC/UFRPE

João Fausto Lorenzato de Oliveira
POLI/UPE

Resumo

Neste trabalho, será proposta uma ferramenta de inferência do nível de dificuldade de receitas culinárias. A inferência será feita através da classificação textual dos modos de preparo das receita. A ferramenta será parte fundamental no desenvolvimento de um sistema de recomendação de receitas culinárias sensível ao contexto baseado em conteúdo. Serão adotados alguns dos principais classificadores da literatura de Classificação de Texto, além de diferentes métodos de extração de características. Uma avaliação experimental é executada, no intuito de selecionar as melhores abordagens para compor o sistema.

Palavras-chave: Nível de Dificuldade em Receitas Culinárias, Processamento de Linguagem Natural, Aprendizagem de Máquina.

Abstract

In this work, a tool for inferring the degree of difficulty of cooking recipes will be proposed. The inference will be made by the textual classification of the recipe preparation methods. The tool will be a fundamental piece to the development of a context-aware content-based cooking recipe recommendation system. Some of the main classifiers in Text Classification literature will be adopted, in addition to different feature extraction methods. An experimental evaluation is performed, in order to select the best approaches to compose the system.

Keywords: Recipe Difficulty Level Inference, Natural Language Processing, Machine Learning.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

BoW	Bag-of-Words
IDF	Inverse Document Frequency
LR	Logistic Regression
NB	Naive Bayes
RF	Random Forest
TC	Text Classification
TF	Term Frequency

Sumário

1 Introdução

Com a popularização da internet, a quantidade de dados disponibilizados todos os dias por usuários cresce cada vez mais. Processos que antes eram feitos de forma artesanal, hoje em dia acontecem *on-line*, de forma automática, rápida e em larga escala. Com a culinária não foi diferente, a prática se torna cada vez mais popular, sendo um dos tópicos mais discutidos em serviços de compartilhamento de vídeos, blogs e redes sociais. A aquisição de receitas culinárias também se torna mais fácil, tendo em vista que a partir de um computador é possível acessar milhões de receitas. Qualquer pessoa pode aprender a cozinhar pratos diversos sem precisar sair de casa.

Apesar da facilidade de acesso, encontrar a receita ideal se torna uma tarefa árdua, devido ao grande volume de dados e à complexidade do processo de busca, que envolve diversos fatores. Para auxiliar os usuários, sistemas de recomendação têm sido aplicados a esse problema de busca (YANG et al., 2016; SCHäFER et al., 2017; MAIA; FERREIRA, 2018). Esses sistemas auxiliam na filtragem dos dados retornados pela busca, exibindo os mais relevantes e diminuindo significativamente o número de resultados. Como a escolha do alimento a ser consumido leva em consideração diversos fatores, o ideal é que a recomendação gerada seja o mais personalizada possível. Entre os fatores levados em consideração para escolha do prato a ser consumido estão as necessidades nutricionais do usuário, restrições alimentares, dieta, tempo disponível, habilidades culinárias do usuário e companhia. Para lidar com parte desses fatores, alguns trabalhos propõem sistemas de recomendação sensíveis ao contexto do usuário (MAIA; FERREIRA, 2018; PRATIBHA; KAUR, 2019), que tem por objetivo gerar recomendações mais relevantes e adequadas à situação atual do usuário (ADOMAVICIUS et al., 2011), diferente dos modelos usuais que ignoram o fato de que quando o usuário interage com o sistema, o contexto em que ele está inserido influencia na sua escolha (LAMCHE et al., 2015). Por exemplo, durante o fim de semana, um usuário pode dispor de mais tempo e desejar fazer um prato mais complexo e elaborado, porém, essas mesmas características em um prato podem não ser adequadas para o usuário numa segunda feira.

Neste trabalho, será desenvolvido uma ferramenta para inferência do nível de dificuldade de preparo de receitas culinárias. A ferramenta será parte fundamental de um sistema, ainda em desenvolvimento, de recomendação de receitas sensível ao contexto e baseado em conhecimentos extraídos das receitas. O sistema usará dados contextuais e informações obtidas das receitas, fazendo um mapeamento das necessidades do usuário com o alimento final. A inferência será tratada como um problema de Classificação de Texto (*Text Classification* - TC), onde os dados textuais serão os

modos de preparo das receitas e os documentos serão classificados para duas classes que representarão os níveis de dificuldade. Para escolher o melhor classificador para compor nosso módulo de inferência, alguns dos principais modelos adotados na literatura de TC serão experimentados: Floresta Aleatória, Naive Bayes e Regressão Logística. Além disso, métodos de extração de características comumente adotados na literatura serão adotados, tanto para a extração, quanto para a seleção das características, sendo eles *Bag-of-Words*, *Term Frequency–Inverse Document Frequency* e a Bag-of-Words binária. A avaliação experimental está de acordo com a literatura de TC (KOWSARI et al., 2019) e através dela será decidida como será feita a implementação do módulo que irá compor o sistema de recomendação a ser desenvolvido.

Este trabalho foi originalmente publicado nos anais do XVII Encontro Nacional de Inteligência Artificial e Computacional de 2020 (ENIAC 2020) (BRITTO; PACíFICO; LUDERMIR, 2020) [1].

O trabalho está dividido da seguinte forma. Na próxima seção (Seção 2) alguns trabalhos de TC (Seção 2.1) e recomendação de receitas (Seção 2.2) serão brevemente discutidos. Em seguida, a base de dados (Seção 3.1), os métodos de extração de características (Seção 3.2) e os classificadores selecionados (Seção 3.3) são descritos. A Seção 4 apresenta os resultados experimentais. Por fim, as conclusões e linhas para possíveis trabalhos futuros serão apresentadas na Seção 5.

[1] <https://sol.sbc.org.br/index.php/eniac/article/view/12121>

2 Trabalhos Relacionados

Nesta seção, uma breve análise da literatura de Classificação de Texto (Seção 2.1) e Recomendação de Receitas (Seção 2.2) é apresentada.

2.1 Classificação de Texto

A Classificação de Texto é uma das tarefas mais populares do Processamento de Linguagem Natural, e tem crescido cada vez mais nos últimos anos com a popularização da internet e disponibilização em massa de dados textuais *on-line*. Através da TC, é possível, a partir do conhecimento obtido das características de um documento textual, associar este documento a uma categoria pré-definida. O principal ponto da TC é criar um modelo que seja capaz de associar, corretamente, o maior número de documentos às suas respectivas classes, e para isso, algumas das etapas vistas na Figura 1 são comumente aplicadas (DALAL; ZAVERI, 2011).

O pré-processamento é uma etapa fundamental, que inclui a limpeza dos dados textuais, onde são removidos erros, informações irrelevantes e qualquer outro ruído que possa prejudicar a performance dos classificadores. Além disso, os documentos também são padronizados. Tarefas como correção ortográfica, remoção de termos que na maioria das vezes possuem pouco significado, como preposições, artigos e conjunções (*stop words*), e stemização, fazem parte do pré-processamento (KOWSARI et al., 2019).

Com os dados pré-processados, é necessário extrair dos documentos textuais as características que serão empregadas nas etapas seguintes da classificação. Esse processo acontece a partir da conversão dos textos em matrizes numéricas, que serão suportadas pelos classificadores. Além da extração, a seleção de características pode ser utilizada para construção do espaço de características. Nessa etapa, algumas abordagens são adotadas para selecionar características que mais podem contribuir positivamente com a saída dos classificadores. Essa seleção pode ser realizada por diversos motivos, como diminuir o tempo de execução e até melhorar o desempenho

Figura 1 – Etapas da Classificação de Texto.

dos classificadores.

Após a prepração dos dados, finalmente é executado o processo de classificação. A partir de uma coleção de documentos *D* = {$X_1, ..., X_N$}, onde todos os elementos são categorizados com uma classe de *C* = {1,...,*k*}, é possível através de um modelo de classificação associar documentos não categorizados a uma classe de *C*, através do conhecimento obtido na coleção *D*.

Apesar dos processos serem semelhantes, diversas técnicas e métodos podem ser usados em cada etapa da TC. Em (PRANCKEVICIUS; MARCINKEVIčIUS, 2016), diferentes algoritmos foram utilizados para fazer a classificação de sentimentos: Naive Bayes, Regressão Logística, Máquina de Vetores de Suporte, Árvore de Decisão e Floresta Aleatória. Uma base de dados de comentários de aplicativos móveis foi adotada, sendo a extração das características realizada pelo método *Bag-of-Words*. Os melhores resultados foram obtidos pelos classificadores Regressão Logística, Floresta Aleatória e Naive Bayes, nessa ordem.

Em (BAHRAWI, 2019), a classificação de sentimentos também é feita, através do uso do classificador Floresta Aleatória, e do método *Term Frequency–Inverse Document Frequency*, para extração de características. Os dados textuais foram extraídos de redes sociais, o que requer um pré-processamento diferenciado, devido às características próprias da linguagem utilizada nessas redes, sendo feita a remoção de menções, *hashtags* e *emoticons*.

Os classificadores Regressão Logística, Máquina de Vetores de Suporte e Redes Neurais Convolucionais foram empregados para uma aplicação médica em (YAO; MAO; LUO, 2018), onde é feita a classificação de doenças em textos clínicos. Em (ZHANG et al., 2019), textos científicos e tecnológicos são classificados em diferentes tópicos, utilizando o algoritmo Naive Bayes e o método de extração *Bag-of-Words*.

2.2 Recomendação de Receitas Culinárias

Diversos trabalhos foram propostos, nos últimos anos, com o objetivo de automatizar a tarefa de escolha de alimentos e criação de dietas para o usuário. Diferentes abordagens para recomendação de receitas culinárias têm sido empregadas na literatura. Algumas dessas abordagens são voltadas para uma recomendação mais personalizada, tentando atender características pessoais de cada indivíduo.

Para atender as necessidades nutricionais de cada usuário, (SCHäFER et al., 2017) propõem uma modelagem do estado nutricional do usuário para compor um sistema de recomendação. Em (YANG et al., 2016), um sistema de recomendação de refeições, baseado em informações nutricionais, é proposto. O sistema é personali-

zado para o usuário através de um formulário preenchido pelo mesmo. Uma base de dados de receitas culinárias contendo 10 mil receitas para diferentes tipos de dietas foi adotada. O sistema, além de personalizar as receitas para o perfil nutricional do usuário, ainda leva em consideração restrições e preferências alimentares.

Uma abordagem muito comum nas técnicas tradicionais de recuperação e recomendação, que também é focada em direcionar ainda mais a recomendação ao perfil do usuário, é a baseada no contexto, onde é gerada uma recomendação personalizada para a situação atual do usuário. Em (MAIA; FERREIRA, 2018), a localização do usuário foi aplicada como contexto para recomendar receitas culinárias, adotando matrizes de fatorização como método de modelagem de contexto, para adquirir a informação contextual, um módulo de mapeamento de localização precisa, usando a tecnologia *Bluetooth*, foi implementado para ser empregado no dispositivo móvel do usuário.

Em (PRATIBHA; KAUR, 2019), um sistema de recomendação de receitas culinárias, integrado com uma geladeira inteligente, foi proposto. Além de informações contextuais do usuário, como tempo disponível para o preparo, informações sobre o ambiente também são utilizadas, gerando recomendações de receitas com os ingredientes disponíveis na geladeira.

3 Metodologia

Nesta seção, a base de dados adotada neste trabalho é descrita (Seção 3.1), assim como os métodos de extração de características (Seção 3.2) e os classificadores selecionados (Seção 3.3).

3.1 Base de Dados

A base de dados utilizada neste trabalho foi proposta em (MAJUMDER et al., 2019) e extraída da rede social culinária Food.com[1]. A base é composta por mais de 230 mil receitas culinárias em inglês, e 1 milhão de interações de usuários, feitas entre os anos de 2000 e 2018. Dentre as informações encotradas na base de dados estão: lista de ingredientes, modos de preparo, informações nutricionais e *tags*. Nessas *tags* são encontradas informações a respeito de diferentes aspectos das receitas, como tipo de prato, tipo de culinária, dietas, tempo de preparo e dificuldade. Para este trabalho, os dados adotados são os modos de preparo, contendo textos instruncionais sobre o preparo da receita, e as *tags* informativas. Na Figura 2 é possível visualizar exemplos dos dados utilizados.

[1] <https://www.kaggle.com/shuyangli94/food-com-recipes-and-user-interactions>

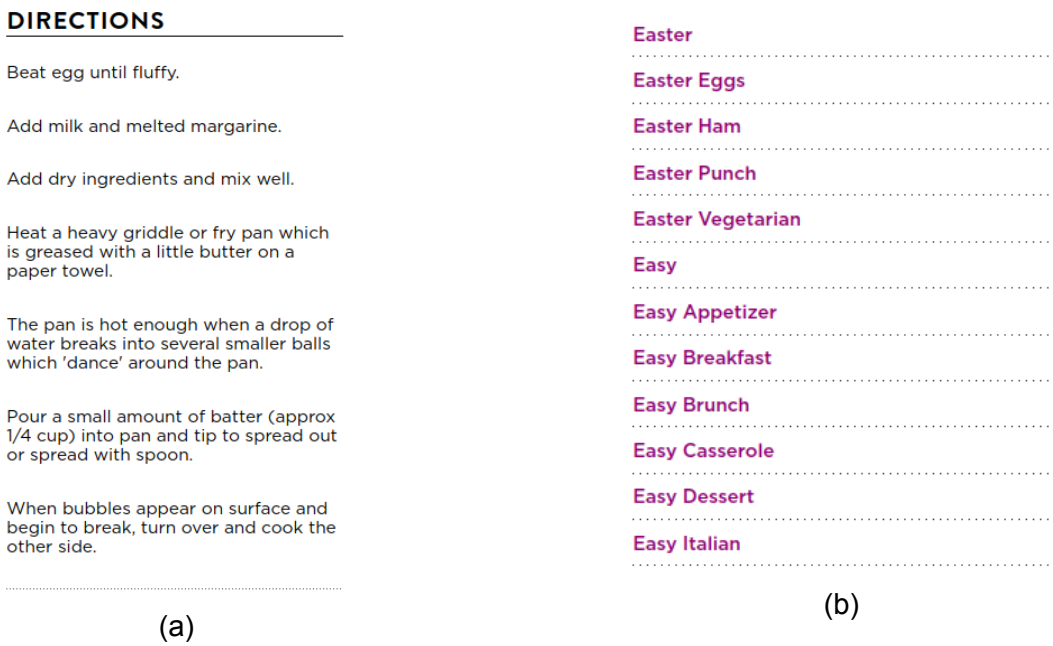

Figura 2 – Exemplos dos dados encontrados na base (MAJUMDER et al., 2019), obtidos no Food.com: (a) Lista de etapas do modo de preparo da receita, (b) algumas *tags* informativas disponíveis no site.

Tabela 1 – Informações sobre os dados utilizados.

	Positivo	Negativo
Quantidade de Receitas	5000	5000
Média Quantidades de Etapas por Receita	7.30	23.37
Média de Quantidade de Termos por Receita	46.56	150.64
Média de Quantidade de Termos por Etapa	6.67	6.48
Tamanho do Vocabulário	6109	8604

Através das *tags* foi possível categorizar essas receitas em duas classes, de acordo com seu nível de dificuldade, sendo um documento categorizado como **positivo** quando faz parte das receitas mais fáceis, e **negativo** quando pertence às receitas difíceis. Foram selecionadas aleatoriamente 5 mil receitas em cada uma dessas categorias. Algumas informações sobre a base de dados selecionada podem ser vistas na Tabela 1.

Os modos de preparo das receitas são encontrados em linguagem natural. Para transformar esses textos em conjuntos mais significativos de dados é necessária uma etapa de pré-processamento. Essa etapa incluiu as seguintes tarefas:

- **Remoção de *Tags* HTML** - *Tags* que formam a estrutura dos documentos originais em HTML, que foram extraídas junto com os dados das receitas no momento da aquisição, e que não possuem nenhuma informação útil, são removidas.
- **Conversão para *Lower Case*** - Todas as letras dos documentos textuais são convertidas para a forma minúscula, fazendo com que termos idênticos representados de formas diferentes, como por exemplo, “Bake”e “bake”, sejam considerados um único termo.
- **Remoção de *Stop Word*s e Pontuações** - Nesta etapa, palavras que são comuns e que não possuem, na maioria das vezes, significado relevante no contexto desse problema, são removidas, assim como pontuações.

Nas Figuras 3 e 4 podemos observar os unigramas e bigramas mais comuns nas receitas positivas e negativas, respectivamente. Através de uma breve análise verificamos alguns termos gerais que aparecem com muita frequência em ambos os conjuntos, alguns ingredientes comuns como “salt”/“sal”, “sugar”/“açúcar”e “water”/“água”. Algumas diferenças também são notáveis, como a existência, nas receitas difíceis (negativas), de termos relacionados a preparos mais complexos e demorados como “oven”/“forno”, “preheat oven”/“pré-aquecer forno”, “set aside”/“reservar”, além de termos relacionados à temperatura e longos períodos de tempo, como “350 degrees”/“350 graus”, “30 minutes”/“30 minutos”, que indicam a necessidade de processos de cozimento ou mais demorados. Já nas receitas fáceis (positivas), se encontram processos

simples, como “combine ingredients”/“misture os ingredientes”, “mix”/”misturar”, e curtos períodos de tempo, como “10 minutes”/“10 minutos”e “30 seconds”/“30 segundos”.

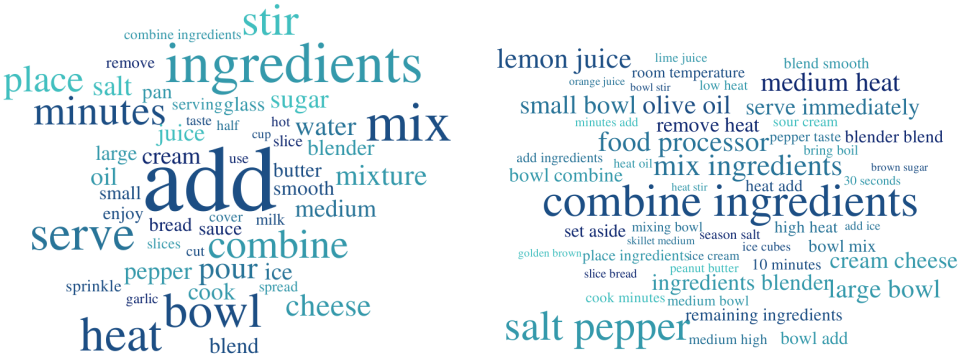

Figura 3 – Unigramas e Bigramas mais frequentes nas receitas fáceis.

Figura 4 – Unigramas e Bigramas mais frequentes nas receitas difíceis.

3.2 Extração de Características

Alguns dos principais métodos de extração de características foram adotados neste trabalho, esses métodos são listados abaixo.

3.2.1 *Bag-of-Words*

Bag-of-Words (BoW) é um dos métodos mais comuns e simples. Nesse modelo, um texto é representado pelo conjunto de suas palavras, sendo esse texto convertido em uma matriz, onde cada palavra do texto é uma coluna nessa matriz, e cada posição representa o número de ocorrência do termo correspondente nesse texto. Nenhuma ordem ou estrutura dos documentos é considerada nessa representação (KOWSARI et al., 2019; BRITTO; PACíFICO, 2019).

3.2.2 Frequência do Termo – Inverso da Frequência nos Documentos

O TF-IDF combina a Frequência do Termo (TF), que tenta mensurar quão importante um termo é em determinado documento, com o Inverso da Frequência nos Documentos (IDF), que mensura a importância do termo em todos os documentos, tentando diminuir assim a influência de termos que ocorrem com uma grande frequência, mas que possuem pouca relevância. A fórmula fo TF-IDF pode ser vista na Equação (3.1).

$$TF - IDF_{t,d} = \frac{f_{t,d}}{\sum_{t_n \in d} f_{t_n,d}} \times \log \frac{N}{df_t} \tag{3.1}$$

onde t representa o termo e d o documento, N é o número total de documentos, df é o número de documentos em que t ocorre e f retorna sua frequência.

3.2.3 Bag-of-Words Binária

Nesse simples modelo, os documentos são convertidos em matrizes binárias, onde cada palavra do texto é uma coluna nessa matriz, e cada posição representa a ausência ou ocorrência do termo nesse texto, resultando em uma matriz de termo de documento binária (BRITTO et al., 2019; ZISER; REICHART, 2016).

3.3 Classificadores Selecionados

Nesta seção, serão descritos os algoritmos selecionados para análise experimental deste trabalho: Naive Bayes, Floresta Aleatória, Regressão Logística.

3.3.1 Naive Bayes

Naive Bayes (NB) é um dos mais populares e simples modelos de classificação de texto. Baseado no teorema de Bayes, o modelo probabílistico utiliza da probabilidade de cada evento ocorrer, assumindo que todas as variáveis são independentes, desconsiderando assim a correlação entre as características. Portanto, na classificação de texto, qualquer ordem ou estrutura do texto é desconsiderada, ignorando completamente o contexto dos termos encontrados no documento.

3.3.2 Floresta Aleatória

Outro classificador muito popular na classificação de texto é a Floresta Aleatória (*Random Forest* - RF) (BREIMAN, 2001). O algoritmo de Floresta Aleatória cria um conjunto de Árvores de Decisão (estrutura hierarquizada que representa uma função

de aprendizagem (BHATNAGAR, 2018)), para subconjuntos selecionados aleatoriamente dos dados de treinamento. Agregando os votos dos diferentes estimadores, o RF é capaz de decidir a classe final do dado de teste (BRITTO; PACíFICO, 2019).

3.3.3 Regressão Logística

O modelo estatístico Regressão Logística (*Logistic Regression* - LR) é usado para prever a probabilidade das possíveis saídas de uma variável dependente, dado um conjunto de variáveis independentes, assumindo que a variável dependente pode ser prevista através da combinação linear das características do problema e parâmetros do modelo. Na classificação de texto, isso siginifica que os termos do documento podem ser combinados linearmente com os parâmetros para determinar a classe à qual determinado documento pertence.

4 Avaliação Experimental

Nesta seção, os resultados experimentais serão apresentados. Três classificadores provenientes da literatura de Classificação de Texto são comparados: Floresta Aleatória, Naive Bayes e Regressão Logística, assim como três métodos de extração de caraceterísticas, sendo eles BoW, TF-IDF e a BoW binária. Além da extração das características, também foi feita a seleção das *n* características mais significativas, de acordo com cada método de extração, para *n* = {1000, 5000}. A avaliação da seleção das características é feita com o objetivo de experimentar como a limitação da quantidade de características disponíveis pode influenciar nos resultados dos classificadores.

Todos os experimentos foram realizados através de um *framework* do tipo 10-*folds*, onde a base de dados é dividida em dez partes aleatórias sem reposição, e em cada rodada de repetição dos experimentos, o conjunto de teste se alterna entre essas 10 partes, enquanto as outras nove partes são usadas como conjunto de treinamento dos modelos. Com o objetivo da obtenção de um conjunto maior de amostras, visando evitar resultados obtidos por sorte, o processo de validação cruzada 10-*fold* foi repetido 10 vezes, cada uma das vezes com os dados redestribuídos aleatoriamente, entre os *folds*. Quatro métricas comumente aplicadas na literatura de Classificação de Texto foram adotadas: a Acurácia (Equação (4.1)), a Precisão (Equação (4.2)), Revocação (Equação (4.3)) e *F-Measure* (Equação (4.4)).

$$Acurcia = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

$$Preciso = \frac{TP}{TP + FP} \tag{4.2}$$

$$Revocao = \frac{TP}{TP + FN} \tag{4.3}$$

$$F - Measure = \frac{2 \times Preciso \times Revocao}{Preciso + Revocao} \tag{4.4}$$

onde, TP e TN equivalem à qantidade de documentos positivos e negativos classificados corretamente, e FN e FP a quantidade de documentos positivos e negativos, respectivamente, categorizados de forma errônea.

Tabela 2 – Resultados Experimentais.

1000						
Class.	Ext.	Acurácia	Precisão	Revocação	F-Measure	Tempo
3*LR	*Bag-of-Words*	0.9436 ± 0.0067	0.9387 ± 0.0100	0.9493 ± 0.0101	0.9439 ± 0.0068	1.1101 ± 0.0383
	TF-IDF	0.9387 ± 0.0070	0.9402 ± 0.0087	0.9371 ± 0.0116	0.9386 ± 0.0071	1.2119 ± 0.2141
	BoW Binária	**0.9456 ± 0.0062**	0.9419 ± 0.0099	**0.9498 ± 0.0092**	**0.9458 ± 0.0064**	1.1155 ± 0.0208
3*NB	*Bag-of-Words*	0.8841 ± 0.0105	0.9103 ± 0.0118	0.8523 ± 0.0173	0.8802 ± 0.0111	**0.7425 ± 0.0110**
	TF-IDF	0.8770 ± 0.0117	0.9227 ± 0.0121	0.8230 ± 0.0189	0.8699 ± 0.0125	1.0483 ± 0.3528
	BoW Binária	0.8973 ± 0.0100	0.9168 ± 0.0110	0.8740 ± 0.0170	0.8948 ± 0.0103	0.7523 ± 0.008
3*RF	*Bag-of-Words*	0.9329 ± 0.0074	**0.9429 ± 0.0094**	0.9216 ± 0.0120	0.9321 ± 0.0076	13.3885 ± 0.2938
	TF-IDF	0.9316 ± 0.0073	**0.9429 ± 0.0093**	0.9189 ± 0.0116	0.9307 ± 0.0075	16.8788 ± 1.9068
	BoW Binária	0.9335 ± 0.0067	0.9416 ± 0.0096	0.9244 ± 0.0115	0.9329 ± 0.0068	19.8144 ± 2.4499
5000						
Class.	Ext.	Acurácia	Precisão	Revocação	F-Measure	Tempo
3*LR	*Bag-of-Words*	0.9459 ± 0.0070	0.9408 ± 0.0093	0.9517 ± 0.0107	0.9462 ± 0.0071	1.1664 ± 0.0229
	TF-IDF	0.9343 ± 0.0074	0.9358 ± 0.0093	0.9327 ± 0.0119	0.9342 ± 0.0075	1.1447 ± 0.2894
	BoW Binária	**0.9487 ± 0.0065**	**0.9444 ± 0.0099**	**0.9535 ± 0.0093**	**0.9489 ± 0.0066**	1.1682 ± 0.0303
3*NB	*Bag-of-Words*	0.8854 ± 0.0108	0.9160 ± 0.0122	0.8487 ± 0.0174	0.8810 ± 0.0112	0.7465 ± 0.0141
	TF-IDF	0.8719 ± 0.0115	0.9304 ± 0.0113	0.8041 ± 0.0194	0.8625 ± 0.0125	0.9053 ± 0.2395
	BoW Binária	0.8970 ± 0.0107	0.9235 ± 0.0116	0.8660 ± 0.0171	0.8937 ± 0.0110	**0.7386 ± 0.0298**
3*RF	*Bag-of-Words*	0.9318 ± 0.0072	0.9401 ± 0.0095	0.9224 ± 0.0121	0.9311 ± 0.0074	13.1342 ± 1.7499
	TF-IDF	0.9301 ± 0.0072	0.9406 ± 0.0093	0.9183 ± 0.0123	0.9292 ± 0.0074	12.9690 ± 0.9847
	BoW Binária	0.9326 ± 0.0076	0.9387 ± 0.0103	0.9258 ± 0.0123	0.9321 ± 0.0079	13.3377 ± 1.9991
Sem Seleção (Todas as Características)						
Class.	Ext.	Acurácia	Precisão	Revocação	F-Measure	Tempo
3*LR	*Bag-of-Words*	0.9462 ± 0.0070	0.9408 ± 0.0094	0.9523 ± 0.0107	0.9464 ± 0.0071	1.4709 ± 0.1506
	TF-IDF	0.9332 ± 0.0075	0.9352 ± 0.0093	0.9310 ± 0.0119	0.9330 ± 0.0077	1.3255 ± 0.2582
	BoW Binária	**0.9490 ± 0.0065**	0.9445 ± 0.0099	**0.9542 ± 0.0089**	**0.9493 ± 0.0066**	1.7480 ± 0.3628
3*NB	*Bag-of-Words*	0.8818 ± 0.0112	0.9232 ± 0.0116	0.8329 ± 0.0186	0.8756 ± 0.0120	**0.7905 ± 0.0667**
	TF-IDF	0.8581 ± 0.0121	**0.9479 ± 0.0106**	0.7580 ± 0.0209	0.8422 ± 0.0136	0.8554 ± 0.0854
	BoW Binária	0.8923 ± 0.0104	0.9299 ± 0.0112	0.8487 ± 0.0176	0.8873 ± 0.0111	0.9192 ± 0.4304
3*RF	*Bag-of-Words*	0.9318 ± 0.0071	0.9386 ± 0.0098	0.9241 ± 0.0121	0.9312 ± 0.0074	13.3522 ± 1.2536
	TF-IDF	0.9304 ± 0.0072	0.9400 ± 0.0093	0.9196 ± 0.0123	0.9296 ± 0.0073	14.3733 ± 2.3357
	BoW Binária	0.9325 ± 0.0072	0.9367 ± 0.0101	0.9278 ± 0.0116	0.9322 ± 0.0074	14.0640 ± 1.1751

Avaliação é feita através de uma análise empírica dos resultados obtidos para o conjunto de teste dos experimentos e uma análise do tempo médio de execução dos classificadores. Os resultados obtidos podem ser vistos na Tabela 2.

O algoritmo Regressão Logística obteve a melhor perfomance média em relação aos classificadores testados, tendo o maior desempenho na maioria dos cenários avaliados, chegando a alcançar aproximadamente 95% de acurácia. Em seguida, encontra-se a Floresta Aleatória, que obteve 93.3% de acurácia, ao ser testada com o método Bag-of-Words binária, selecionando 1000 características. Por último, com um desempenho inferior aos demais, o Naive Bayes alcançou 89.7% de acurácia em seu melhor cenário (com 1000 características e método de extração binária). Em compensação, o Naive Bayes foi o classificador com melhor desempenho em relação ao tempo médio de execução, não passando de 0.80 segundo, seguido pelo Regressão Logística, que levou até 1.7 segundo. A Floresta Aleatória obteve tempo médio de execução superior a 10 segundos, o que representa uma grande diferença em relação aos demais classificadores avaliados.

Em relação aos métodos de extração de características, a Bag-of-Words binária foi responsável pelos melhores resultados médios dos classificadores, para a maioria das métricas, com pouca diferença para o segundo colocado, a *Bag-of-Words*. Na se-

leção das características, todos os experimentos obtiveram resultados muito próximos. Em alguns casos, a seleção das 1000 melhores características sobressaiu em relação à classificação com todas as características, como no classificador Naive Bayes. A pequena variação no desempenho médio dos classificadores, em relação à quantidade de características, pode ser usada a favor do sistema de recomendação a ser desenvolvido, tendo em vista que um menor número de características poderia reduzir consideravelmente o custo computacional médio do sistema, sem afetar o desempenho dos classificadores.

5 Conclusões

Nesse trabalho, o desempenho de três dos principais algoritmos adotados na literatura de Classificação de Texto foi comparado na tarefa de classificação de modos de preparo de receitas culinárias. Foi comparado ainda o desempenho de três diferentes métodos de extração de característica, e analisada a performance desses métodos ao serem empregados para seleção de características. Para os experimentos, uma base de dados contendo 10 mil receitas culinárias, categorizadas entre fáceis e difíceis, foi utilizada.

Uma análise dos resultados experimentais sugere que o classificador Regressão Logística seria a melhor escolha para compor o módulo de inferência de dificuldade em um sistema de recomendação a ser desenvolvido, tendo o mesmo obtido o melhor desempenho (aproximadamente 95% de acurácia), além de um tempo médio de execução de até 2 segundos. A seleção de características demonstrou ser uma boa opção para diminuir o custo computacional médio de execução, tendo em vista que a a variação no desempenho dos classificadores em relação às quantidades de características avaliadas é baixa. De todos os métodos de extração de características, o modelo Bag-of-Words binária obteve o melhor resultado na maioria das métricas, independentemente do classificador selecionado.

Como trabalhos futuros, pretendemos analisar como diferentes informações podem ser inferidas a partir dos modos de preparo, como, por exemplo, tempo de preparo. Pretendemos também integrar o módulo de inferência de dificuldade no nosso sistema de recomendação sensível ao contexto e baseado em conteúdo a ser desenvolvido, e avaliar como a informação da dificuldade das receitas pode contribuir com o desempenho do sistema e qualidade das recomendações feitas. Por fim, o sistema de recomendação em desenvolvimento será disponibilizado ao público, como ferramenta de auxílio à elaboração de dietas saudáveis, nutritivas e adequadas à situação atual do usuário.

Referências

ADOMAVICIUS, G. et al. Context-aware recommender systems. *AI Magazine*, v. 32, p. 67–80, 09 2011. Citado na página 8.

BAHRAWI, B. Sentiment analysis using random forest algorithm online social media based. v. 2, p. h.29–33, 12 2019. Citado na página 11.

BHATNAGAR, R. Machine learning and big data processing: A technological perspective and review. In: ____. [S.l.: s.n.], 2018. p. 468–478. ISBN 978-3-319-74689-0. Citado na página 17.

BREIMAN, L. Random forests. *Machine Learning*, v. 45, n. 1, p. 5–32, Oct 2001. ISSN 1573-0565. Disponível em: <https://doi.org/10.1023/A:1010933404324>. Citado na página 16.

BRITTO, L.; PACíFICO, L.; LUDERMIR, T. Inferência automática do nível de dificuldade em receitas culinárias usando técnicas de processamento de linguagem natural. In: *Anais do XVII Encontro Nacional de Inteligência Artificial e Computacional*. Porto Alegre, RS, Brasil: SBC, 2020. p. 104–115. Disponível em: <https://sol.sbc.org.br/index.php/eniac/article/view/12121>. Citado na página 9.

BRITTO, L. F. S. et al. Uma abordagem de análise de textos para a classificação de receitas culinárias baseadas em documentos em português brasileiro. In: *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*. Porto Alegre, RS, Brasil: SBC, 2019. p. 436–447. Disponível em: <https://sol.sbc.org.br/index.php/eniac/article/view/9304>. Citado na página 16.

BRITTO, L. F. S.; PACíFICO, L. D. S. Análise de sentimentos para revisões de aplicativos mobile em português brasileiro. In: *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*. Porto Alegre, RS, Brasil: SBC, 2019. p. 1080–1090. Disponível em: <https://sol.sbc.org.br/index.php/eniac/article/view/9359>. Citado 2 vezes nas páginas 15 e 17.

DALAL, M. K.; ZAVERI, M. A. Article: Todv: Automatic text classification: A technical review. *International Journal of Computer Applications*, v. 28, n. 2, p. 37–40, August 2011. Full text available. Citado na página 10.

KOWSARI, K. et al. Text classification algorithms: A survey. *CoRR*, abs/1904.08067, 2019. Disponível em: <http://arxiv.org/abs/1904.08067>. Citado 3 vezes nas páginas 9, 10 e 15.

LAMCHE, B. et al. Context-aware recommendations for mobile shopping. *CEUR Workshop Proceedings*, v. 1405, p. 21–27, 01 2015. Citado na página 8.

MAIA, R. L.; FERREIRA, J. C. Context-aware food recommendation system. In: . [S.l.: s.n.], 2018. Citado 2 vezes nas páginas 8 e 12.

MAJUMDER, B. P. et al. Generating personalized recipes from historical user preferences. In: *Proceedings of the 2019 Conference on Empirical Methods in*

Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, China: Association for Computational Linguistics, 2019. p. 5976–5982. Disponível em: <https://www.aclweb.org/anthology/D19-1613>. Citado 2 vezes nas páginas 4 e 13.

PRANCKEVICIUS, T.; MARCINKEVIčIUS, V. Application of logistic regression with part-of-the-speech tagging for multi-class text classification. In: . [S.l.: s.n.], 2016. p. 1–5. Citado na página 11.

PRATIBHA; KAUR, P. D. A context-aware recommender engine for smart kitchen. In: PANIGRAHI, B. K. et al. (Ed.). *Smart Innovations in Communication and Computational Sciences*. Singapore: Springer Singapore, 2019. p. 161–170. Citado 2 vezes nas páginas 8 e 12.

SCHäFER, H. et al. User nutrition modelling and recommendation: Balancing simplicity and complexity. In: . [S.l.: s.n.], 2017. p. 93–96. Citado 2 vezes nas páginas 8 e 11.

YANG, L. et al. Yum-me: Personalized healthy meal recommender system. *CoRR*, abs/1605.07722, 2016. Disponível em: <http://arxiv.org/abs/1605.07722>. Citado 2 vezes nas páginas 8 e 11.

YAO, L.; MAO, C.; LUO, Y. Clinical text classification with rule-based features and knowledge-guided convolutional neural networks. *CoRR*, abs/1807.07425, 2018. Disponível em: <http://arxiv.org/abs/1807.07425>. Citado na página 11.

ZHANG, H. et al. Research on classification of scientific and technological documents based on naive bayes. In: *Proceedings of the 2019 11th International Conference on Machine Learning and Computing*. New York, NY, USA: Association for Computing Machinery, 2019. ISBN 9781450366007. Disponível em: <https://doi.org/10.1145/3318299.3318330>. Citado na página 11.

ZISER, Y.; REICHART, R. Neural structural correspondence learning for domain adaptation. *CoRR*, abs/1610.01588, 2016. Disponível em: <http://arxiv.org/abs/1610.01588>. Citado na página 16.