

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

**Avaliação automática da qualidade de recursos educacionais
abertos usando métodos de Processamento de Línguas
Naturais**

Murilo Gleyson Gazzola

Tese de Doutorado do Programa de Pós-Graduação em Ciências de
Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Murilo Gleyson Gazzola

**Avaliação automática da qualidade de recursos
educacionais abertos usando métodos de Processamento
de Línguas Naturais**

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Sandra Maria Aluísio

USP – São Carlos
Abril de 2021

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

G291a Gazzola, Murilo Gleyson
 Avaliação automática da qualidade de recursos
educacionais abertos usando métodos de
Processamento de Línguas Naturais / Murilo Gleyson
Gazzola; orientadora Sandra Maria Aluísio. -- São
Carlos, 2021.
 145 p.

 Tese (Doutorado - Programa de Pós-Graduação em
Ciências de Computação e Matemática Computacional) --
 Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2021.

 1. Processamento de Línguas Naturais. 2.
Complexidade Textual. 3. Deep Learning. 4. Multi-
task Learning. I. Aluísio, Sandra Maria, orient. II.
 Título.

Murilo Gleyson Gazzola

Automatically assessing open educational resources using
methods based on Natural Language Processing

Doctoral dissertation submitted to the Institute of
Mathematics and Computer Sciences – ICMC-USP, in
partial fulfillment of the requirements for the degree of
the Doctorate Program in Computer Science and
Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Sandra Maria Aluísio

USP – São Carlos
April 2021

Dedico este trabalho:

Aos meus pais Paulo Cesar Gazzola e Elenice Aparecida Ferrari Gazzola.

Aos meus familiares da família Ferrari e Gazzola.

A todos que confiaram e acreditaram no meu trabalho e na minha pesquisa.

A ciência e a educação brasileira.

Aos pesquisadores de Processamento de Línguas Naturais e Inteligência Artificial.

Aos professores e pesquisadores que acreditam que a educação é um meio para o crescimento da sociedade, do indivíduo e de uma nação.

AGRADECIMENTOS

Agradeço inicialmente a Deus, à toda minha família pelo amor, confiança, apoio em minhas decisões, pela educação, carinho e por entenderem minha ausência para dedicação aos estudos.

À minha orientadora, Profa. Dra. Sandra Maria Aluísio, que aceitou me orientar e confiou no meu trabalho; sempre me manteve super motivado, trazendo novas ideias para desenvolver nesse doutorado. Confiou e apoiou esta tese de doutorado até o fim. Sempre demonstrando paciência, dedicação e ensinamentos. Ela me deu a oportunidade de se tornar um pesquisador na área de Processamento de Línguas Naturais (PLN) e tenho a honra de ter ela como minha orientadora.

A Capes pelo auxílio a bolsa de doutorado nº PROEX-8436630/D¹.

Aos integrantes do Núcleo Interinstitucional de Linguística Computacional (NILC), pelas conversas, reuniões, *happy hour*, hora do café, e, em geral, por todos os momentos de convivência que continuaram por toda minha vida. Pois, o doutorado pode concluir, mas o NILC sempre estará comigo.

Agradeço também a Profa. Dra. Sabine Pompéia da Universidade Federal de São Paulo (UNIFESP) e aos alunos de mestrado e pós-doutorado por ter participação neste projeto.

Agradeço ao Prof. Dr. Edson Amaro Junior, por apoiar minha participação no *Symposium in Information and Human Language Technology* (STIL) de 2019 e me dar a oportunidade de ser um cientista e pesquisador na área de Big Data & Analytics do Hospital Albert Einstein.

Agradeço ainda ao Hospital Albert Einstein por me conceder flexibilidade de tempo, para que eu pudesse finalizar esta tese de doutorado. Como também, possibilitar a continuar desenvolvendo pesquisas na área de Processamento de Línguas Naturais.

¹ O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil - (CAPES) - Código de Financiamento 001.

*“A tarefa essencial do professor é despertar
a alegria de trabalhar e de conhecer.”
(Albert Einstein)*

RESUMO

GAZZOLA, M. G. **Avaliação automática da qualidade de recursos educacionais abertos usando métodos de Processamento de Línguas Naturais**. 2021. 145 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

Contexto: A expansão do acesso à banda larga no mundo está abrindo novos caminhos para o ensino, os quais podemos destacar o ensino aberto. A UNESCO em seu relatório anual de 2016 destacou que existem fortes interesses em incentivar a utilização de recursos educacionais abertos (REA) no campo de aprendizagem. Porém, a falta de qualidade dos REA é um desafio a ser superado. Em uma recolha na Plataforma Integrada do Ministério da Educação (MEC-RED) de maio de 2018, 41% dos recursos não possuíam classificação da etapa de ensino, dificultando sua busca, uso e edição. A literatura mais recente em avaliação automática da complexidade textual traz três grandes abordagens para essa tarefa: (i) uso de técnicas estatísticas como a Análise de Componentes Principais e a Análise Fatorial para reduzir um grande número de métricas linguísticas em poucas dimensões explicativas da complexidade textual; (ii) a abordagem de aprendizado de máquina, via construção de classificadores ou regressores, muitas vezes com o uso de métodos de seleção de *features* ou com avaliação da contribuição de conjuntos de *features* para a tarefa; e (iii) a abordagem de *Deep Learning*, que usa arquiteturas neurais com diversas características para evitar o uso de engenharia de *features*. **Lacunas:** Na avaliação automática da complexidade textual, até onde sabemos, não há trabalhos que aplicaram a abordagem *Multi-task Learning* com o propósito de construir um preditor de complexidade textual, e na avaliação da qualidade de metadados de REA a abordagem *Deep Learning* não havia sido explorada antes desta pesquisa. **Proposta:** A tarefa Complexidade Textual pode ajudar a identificar textos que tem complexidade linguística adequada a séries específicas, permitindo completar a etapa de ensino na MEC-RED. Avaliamos o impacto do gênero textual na avaliação da complexidade textual, com modelos treinados em um grande corpus de textos didáticos e testados em 2 datasets de REA da MEC-RED. Para REA que não possuem material textual, como vídeos e áudios, propusemos a avaliação de sua qualidade via título e resumo. **Métodos de Avaliação:** Para a avaliação em geral, foram usadas métricas comuns da área de Processamento de Línguas Naturais (PLN). **Resultados:** Como resultados, propusemos dois métodos de classificação da qualidade dos **metadados de REA**, usando *Deep Learning* (DeepMD e DeepMDX); além de uma larga avaliação da **complexidade textual de REA** no Português Brasileiro. Nesta última, usamos várias abordagens, incluindo *Multi-task Learning*, que não havia sido testada anteriormente, em duas arquiteturas (MTC-DTG Simplex e MTC-DTG). O melhor resultado para a classificação da qualidade de metadados, utilizando um dataset com 1.179 título/resumo de REA, foi de 0.98% de medida F, tanto com o modelo DeepMD como com o DeepMDX. Na avaliação da complexidade textual, podemos destacar o desempenho do modelo *multi-task learning* com duas

tarefas auxiliares (MTC-DTG) que obteve 0.95% de medida F, o que trouxe uma melhoria de 0.15 pontos no melhor modelo de aprendizado de máquina tradicional (SVM).

Palavras-chave: Processamento de Língua Natural, *Deep Learning*, Complexidade Textual, *Multi-task Learning*.

ABSTRACT

GAZZOLA, M. G. **Automatically assessing open educational resources using methods based on Natural Language Processing**. 2021. 145 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

Background: The expansion of broadband access in the world is opening new paths for education, which we can highlight open education. UNESCO in its 2016 annual report highlighted that there are strong interests in encouraging the use of open educational resources (OER) for learning. However, the lack of quality of OER is a challenge to be overcome. The analysis of a corpus of resources taken from a dump of May 2018 of the Integrated Platform of the Ministry of Education (MEC-RED), 41% of the resources did not have classification of the teaching stage, making it difficult to search, use and edit them. The literature on automatic assessment of textual complexity (or readability) brings three major approaches to this task: (i) use of statistical techniques such as Principal Component Analysis and Factor Analysis to reduce a large number of linguistic metrics in a few explanatory dimensions of textual complexity; (ii) the machine learning approach, through the construction of classifiers or regressors, often with the use of features selection methods or with the evaluation of the contribution of feature sets to the task; and (iii) the Deep Learning approach, which uses neural architectures with different characteristics to avoid the use of feature engineering. **Gaps:** In the automatic evaluation of textual complexity, as far as we know, there are no studies that applied the Multi-task Learning (MTL) approach in order to build a predictor of textual complexity and in the evaluation of the quality of OER metadata, Deep Learning approaches had not been explored before this research. **Objectives:** The Textual Complexity task can help to identify texts that have linguistic complexity appropriate to specific grades, allowing to complete the teaching stage of MEC-RED. We evaluated the impact of the textual genre in the assessment of textual complexity, with models trained in a large corpus of textbooks and tested in two datasets of OER from MEC-RED. For OER that do not have textual material, such as videos and audios, we proposed to assess their quality via title and summary. **Evaluation:** For evaluation, common metrics from the Natural Language Processing (PLN) area were used to assess the performance of the developed approaches. **Results:** As for results, we have developed two methods for classifying the quality of OER metadata, using Deep Learning (DeepMD and DeepMDX) and carried out a broad assessment of the textual complexity of OER in Brazilian Portuguese. In the latter, we used several approaches, including *Multi-task Learning*, which had not been previously tested, on two architectures (MTC-DTG Simplex and MTC-DTG). The best result for the classification of metadata quality, using a dataset with 1,179 OER title/abstract, was 0.98 % of F-measure, both with the DeepMD model and with the DeepMDX. In the assessment of textual complexity, we can highlight the performance of the *multi-task learning* model with two auxiliary tasks

(MTC-DTG) that obtained 0.95% of F-measure, an improvement of 0.15 points over the best model using traditional machine learning (SVM).

Keywords: Natural Language Processing, Deep Learning, Readability, Multitask Learnig.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de material com etapa de ensino informada para uso por estudantes e professores.	25
Figura 2 – Exemplo de material sem etapa de ensino informada	25
Figura 3 – Arquitetura de avaliação da qualidade dos metadados de REA proposta nesta pesquisa	27
Figura 4 – Repositório de REA para saúde	35
Figura 5 – Formulário de Qualidade para um recurso educacional on-line para inteligência artificial	38
Figura 6 – Comparativo entre o número de recurso em cada Etapa de Ensino em dois datasets da MEC-RED (2018 e 2020)	40
Figura 7 – Porcentagens dos Tipos de Recursos mais representativos no dataset compilado em 2018	41
Figura 8 – Componentes curriculares do dataset compilado	42
Figura 9 – Conjuntos das áreas de IA, Aprendizado de Máquina e Processamento de Línguas Naturais	43
Figura 10 – Uma rede neural profunda com suas camadas e módulos	45
Figura 11 – Arquitetura de uma RNN	45
Figura 12 – Arquitetura de uma LSTM	46
Figura 13 – Compartilhamento <i>hard</i> de parâmetros para <i>multi-task learning</i> usando redes neurais profundas	49
Figura 14 – Função não linear usando ConvNetJs	50
Figura 15 – Classificação usando SVR	53
Figura 16 – Metodologia para geração de modelos de avaliação automática de qualidade para OAs	55
Figura 17 – Escala da quantidade de comentários por ranqueamento do REA	56
Figura 18 – Arquitetura do MASECO	57
Figura 19 – Modelo REA do repositório MASECO	57
Figura 20 – Funcionamento da classificação de REA usando MASECO	58
Figura 21 – Extensão para navegador desenvolvido para anotação da qualidade	60
Figura 22 – Avaliação da qualidade por meio de perfis de usuários	61
Figura 23 – Regressão linear usada para avaliação da qualidade de OA	63
Figura 24 – Interface do Edu-AREA para avaliação de REA	64
Figura 25 – REA armazenados no repositório já avaliados pelo algoritmo HITS adaptado	65

Figura 26 – Excerto do dataset de metadados de REA	76
Figura 27 – Histograma do comprimento do texto (<i>small data</i>)	77
Figura 28 – Histograma do comprimento do texto (conjunto completo de dados)	78
Figura 29 – Histograma do comprimento do texto por classe (<i>small data</i>)	78
Figura 30 – Histograma da quantidade de palavras (conjunto de dados completo)	78
Figura 31 – Histograma da quantidade de palavras por classe (<i>small data</i>)	79
Figura 32 – Número de Bigramas do Córpus (<i>small data</i>)	79
Figura 33 – Comprimento de textos após a fase de pré-processamento (<i>small data</i>)	80
Figura 34 – Distribuição dos 10 tokens mais frequentes no córpus, após o pré-processamento do córpus (<i>small data</i>)	81
Figura 35 – Nuvem de palavras antes e depois da limpeza do córpus (<i>small data</i>)	81
Figura 36 – Exemplo de valores TF-IDF	82
Figura 37 – t-SNE do córpus - (A) <i>small data</i> e (B) conjunto completo de dados	83
Figura 38 – SVM: Precision, Recall e F1 - experimentos com (A) <i>small data</i> e (B) conjunto de dados completo	84
Figura 39 – Arquitetura DeepMD - LSTM para classificação dos metadados de REA	85
Figura 40 – Arquitetura DeepMDX - LSTM para classificação dos metadados de REA	85
Figura 41 – Resultados com as features selecionadas pelo Lasso, em quatro níveis escolares	95
Figura 42 – Classificação usando todas as features do trabalho	96
Figura 43 – Gráfico de Declividade dos PC e Variância	98
Figura 44 – Exemplo do PC1 e PC2 em 2 dimensões	99
Figura 45 – Comparativo de classificadores, variando o k dos Componentes Principais (F-Measure)	100
Figura 46 – Conjunto estratificado de treino e teste	102
Figura 47 – Arquitetura usando BiLSTM com Word Embeddings e Mecanismo de Atenção	103
Figura 48 – Matrizes de confusão de 3 modelos da Tabela 33 (linhas 1, 2 e 4)	122
Figura 49 – Arquitetura do MTC-DTG Simplex	123
Figura 50 – Arquitetura do MTC-DTG	124

LISTA DE TABELAS

Tabela 1 – Repositórios REA com diferentes padrões de metadados	33
Tabela 2 – Heterogeneidade dos padrões de metadados encontrado na Web	34
Tabela 3 – Algumas sementes utilizadas pelo SeeOER	34
Tabela 4 – Tabela com 6 critérios que indicam qualidade de um recurso educacional on-line	37
Tabela 5 – Guia definitivo dos critérios de qualidade	39
Tabela 6 – Indicadores automatizados de qualidade	52
Tabela 7 – Indicadores de qualidades fornecidos para classificação dos REA	54
Tabela 8 – Dimensões para indicadores de qualidade	59
Tabela 9 – Indicadores automatizados de qualidade	59
Tabela 10 – Dimensões de qualidade usadas	62
Tabela 11 – Análise crítica dos trabalhos correlatos	67
Tabela 12 – Comparação entre os trabalhos relacionados e a proposta desenvolvida no doutorado	68
Tabela 13 – Extrato dos tipos de materiais nos REA Qualificados e Não Qualificados	77
Tabela 14 – Comparação entre os classificadores e os métodos utilizados	87
Tabela 15 – Córpus de livros-textos da Língua Portuguesa compilado: Marcha criança, Tudo É Linguagem, Projeto Porta Aberta, Projeto Ápis, Português, Buriti, Porta Aberta, Mundo Amigo, Nos Dias de Hoje, Projeto Teláris, CNEC Educação.	91
Tabela 16 – Descrição do Córpus de Textos dos Recursos da MEC-RED	92
Tabela 17 – 79 métricas usadas nesta pesquisa	93
Tabela 18 – Continuação ... 79 métricas usadas nesta pesquisa	94
Tabela 19 – Features selecionadas pelo método CFS	95
Tabela 20 – Resultados da classificação com as features selecionadas pelo CFS	95
Tabela 21 – Resultado do PCA com autovalor, proporção de variância, variância acumulativa e as cargas representativas	97
Tabela 22 – Resultado do PCA com autovalor, proporção de variância, variância acumulativa e as cargas representativas	98
Tabela 23 – Resultados da Classificação do SVM com todas as features	100
Tabela 24 – Resultados dos modelos na avaliação intrínseca	102
Tabela 25 – Resultados por classe dos modelos Deep Learning na avaliação intrínseca	103
Tabela 26 – Resultados dos modelos de Deep Learning na avaliação extrínseca	104

Tabela 27 – C3rpus de Textos Did3ticos da L3ngua Portuguesa	110
Tabela 28 – Descri3o do C3rpus de Textos dos Recursos da MEC-RED	111
Tabela 29 – Corpus Adole-sendo	111
Tabela 30 – <i>Features</i> de Palavras	116
Tabela 31 – As 55 features usados no corpus Adole-sendo.	119
Tabela 32 – Resultados dos Modelos na Avalia3o Intr3nseca	120
Tabela 33 – Resultados dos Modelos na Avalia3o Extr3nseca com datasets da MEC-RED	121
Tabela 34 – PCA aplicado usando a Linguagem R, normaliza3o Z-Score e Varimax para 75 features selecionadas - Parte I	144
Tabela 35 – PCA aplicado usando a Linguagem R, normaliza3o Z-Score e Varimax para 75 features selecionadas - Parte II	145

SUMÁRIO

1	INTRODUÇÃO	23
1.1	Cenário da Pesquisa	23
1.2	Tarefas da Pesquisa	26
1.3	Questões de pesquisa, Hipóteses e Objetivos	28
1.4	Organização da Tese	29
2	FUNDAMENTAÇÃO TEÓRICA	31
2.1	Recursos Educacionais Abertos	31
2.1.1	<i>Caracterização</i>	31
2.1.2	<i>Metadados</i>	32
2.1.3	<i>Indicadores de Qualidade</i>	35
2.2	Plataforma MEC-RED	36
2.2.1	<i>Análise quantitativa dos recursos da Plataforma do MEC</i>	36
2.3	Abordagens de Aprendizado de Máquina na Pesquisa	41
2.3.1	<i>Aprendizado de Máquina Clássico via Engenharia de Features</i>	42
2.3.2	<i>Deep Learning</i>	43
2.3.3	<i>Multi-task Learning</i>	48
2.3.4	<i>Softmax e Cross Entropy Loss</i>	49
3	REVISÃO DA LITERATURA	51
3.1	Considerações Iniciais	51
3.2	Qualidade de REA	51
3.2.1	<i>Trabalho de Bethard et al.(2009)</i>	51
3.2.2	<i>Trabalho de Dalip et al.(2011)</i>	53
3.2.3	<i>Trabalho de Leary et al.(2011)</i>	54
3.2.4	<i>Trabalho de Cechinel et al.(2011)</i>	55
3.2.5	<i>Trabalho de Moise et al.(2011)</i>	56
3.2.6	<i>Trabalho de Wetzler et al.(2013)</i>	59
3.2.7	<i>Trabalho de Sawadogo et al.(2014)</i>	60
3.2.8	<i>Trabalho de Pons et al. (2015)</i>	62
3.2.9	<i>Trabalho de Rodrigues et al. (2016)</i>	63
3.2.10	<i>Trabalho de Ahmed e Fuge (2017)</i>	64
3.3	Avaliação Automática da Complexidade Textual	69

3.3.1	<i>Complexidade Textual via Análise de Componentes Principais</i>	69
3.3.2	<i>Complexidade Textual com ajuda da Engenharia de features</i>	70
3.3.3	<i>Complexidade Textual baseada em modelos Deep Learning</i>	71
3.4	Considerações Finais	73
4	CLASSIFICAÇÃO DA QUALIDADE DOS METADADOS DE REA USANDO DEEP LEARNING	75
4.1	Considerações Iniciais	75
4.2	Análise descritiva do <i>córpus</i> de metadados de REA	75
4.3	Classificador Baseline	82
4.4	Classificador SVM	83
4.5	Classificador LSTM (DeepMD e DeepMDX)	83
4.6	Critérios de Avaliação da Qualidade dos REA	84
4.7	Conclusões e Discussões	86
4.8	Considerações Finais	86
5	PREDIÇÃO DA COMPLEXIDADE TEXTUAL DE REA NO POR- TUGUÊS BRASILEIRO	89
5.1	Considerações Iniciais	89
5.2	<i>Córpus</i> dos Quatro Estágios Escolares do Sistema Educacional Bra- sileiro	90
5.3	Experimentos usando Aprendizado de Máquina Tradicional e Méto- dos de Seleção de Features	91
5.4	Métodos Tradicionais de Aprendizagem de Máquina: Avaliação In- trínseca e Extrínseca	99
5.5	Métodos de Deep Learning: Avaliação Intrínseca e Extrínseca	101
5.6	Conclusões e Discussões	104
5.7	Considerações Finais	105
6	UMA ABORDAGEM PARA COMPLEXIDADE TEXTUAL DE REA USANDO MULTI-TASK LEARNING	107
6.1	Considerações Iniciais	107
6.2	Avaliação da complexidade textual usando MTL com textos trans- critos	108
6.3	Datasets	109
6.3.1	<i>Córpus</i> de Textos Didáticos dos Estágios Escolares do Sistema Edu- cacional Brasileiro	110
6.3.2	<i>Córpus</i> de REA	110
6.3.3	<i>Córpus</i> do Projeto Adole-sendo	111
6.3.3.1	Transcrição de Áudios de Recontos de Histórias Infantis	112

6.3.3.2	<i>Anotação Linguística de Disfluências</i>	113
6.3.3.3	<i>Geração Automática da narrativa sem disfluências</i>	114
6.4	Features	114
6.4.1	<i>Features usadas no cópús de textos didáticos e nos conjuntos de dados da MEC-RED</i>	114
6.4.1.1	<i>Palavras</i>	114
6.4.1.2	<i>Sentenças</i>	115
6.4.1.3	<i>Conexões entre sentenças</i>	117
6.4.2	<i>Features usadas no corpus Adole-sendo</i>	118
6.5	Experimentos e Resultados	118
6.5.1	<i>Matriz de Confusão da Avaliação Extrínseca da Complexidade Textual</i>	121
6.6	Conclusões e Discussões	122
6.7	Considerações Finais	125
7	CONCLUSÕES	127
7.1	Contribuições	128
7.2	Limitações	129
7.3	Trabalhos Futuros	130
7.4	Lista de Publicações	130
	REFERÊNCIAS	133
	APÊNDICE A PCA USANDO VARIMAX	143

INTRODUÇÃO

1.1 Cenário da Pesquisa

Recursos Educacionais Abertos (REA) foram definidos por diversos trabalhos na literatura e são objeto de estudo de uma ampla diversidade de trabalhos (WILEY; BLISS; MCEWEN, 2014; BARANIUK; BURRUS, 2008; MARTÍN *et al.*, 2008; PLOTKIN, 2010; WILEY, 2002). O termo foi cunhado em 2002 pela Unesco (UNESCO., 2002) que convocou o fórum sobre o impacto dos cursos abertos do ensino superior em países em desenvolvimento. Foi definido neste fórum que o termo “aberto”, relacionado a recursos educacionais, implica na possibilidade do conhecimento ser consultado, usado e adaptado por todos sem fins comerciais. Os REA podem ser caracterizados como materiais de ensino, aprendizagem e pesquisa, em qualquer meio de armazenamento, que estão amplamente disponíveis por meio de uma licença aberta que permite quatro liberdades mínimas conhecidas como 4R: Aprimorar (*Revise*), Usar (*Reuse*), Recombinar (*Remix*) e Distribuir (*Redistribute*) (ATKINS; BROWN; HAMMOND, 2007; KANWAR; UVALIĆ-TRUMBIĆ; BUTCHER, 2011; GIMENES; BARROCA; FELTRIM, 2012). Neste trabalho, usamos esta definição para REA e apresentamos mais detalhes sobre eles no Capítulo 2.

REA têm como objetivo tornar o conhecimento do mundo um bem público tendo a tecnologia em geral e, especificamente a Web como uma fonte de acesso a esse conhecimento. Entende-se, assim, que os REA devem permitir seu compartilhamento, uso e reuso. O movimento de REA foi financiado inicialmente pela Fundação Hewlett que considerou o movimento uma filosofia de democratização do conhecimento por meio da Web (SMITH; CASSERLY, 2006; WILEY; BLISS; MCEWEN, 2014). De acordo com Mike Smith, diretor do programa de educação da Fundação Hewlett,

“O coração dos recursos educacionais abertos é uma ideia simples e poderosa de transformar o conhecimento do mundo em um bem público e a Web é uma

oportunidade extraordinária para que todos possam compartilhar, usar e reutilizar esse conhecimento" (SMITH; CASSERLY, 2006, p. 2).

Além das diversas definições de REA, existem diferentes modelos de compartilhamento, diversos modelos de produção, como também muitos desafios para os recursos abertos. Devido à importância do tema, a Organização das Nações Unidas (ONU) definiu os principais problemas relativos ao desenvolvimento e ao uso de REA (MIAO; MISHRA; MCGREAL, 2016; WILEY; BLISS; MCEWEN, 2014) como: i) *o problema da avaliação da qualidade*; ii) *o problema da descoberta*, isto é, como encontrar; iii) *o problema da sustentabilidade*, isto é, como financiá-los; iv) *o problema da localização e re-contextualização*; e v) *o problema do remix*, isto é, a dificuldade de identificar a granularidade de alteração do conteúdo por outras pessoas e o nível de alteração. De acordo com o documento da ONU, estes problemas devem ser enfrentados para que esses recursos possam cumprir seu potencial de contribuir para o desenvolvimento humano.

No Brasil, a plataforma de Recursos Educacionais do Ministério da Educação (MEC), conhecida como Plataforma Integrada MEC¹ (MEC-RED), faz parte de um dos compromissos do Brasil na *Open Government Partnership* para fortalecer práticas que envolvem a transparência dos atos governamentais e promovem a participação social e o acesso à informação pública (MEC, 2019). Escolhemos esta plataforma para servir de cenário de pesquisa das duas tarefas abordadas nesta tese e apresentadas na próxima seção.

Na MEC-RED, há cinco filtros de busca por recursos: (i) componentes curriculares/disciplinas, (ii) tipos de recurso, (iii) etapas de ensino, (iv) idiomas e (v) palavra-chave, embora a plataforma tenha outros metadados. Em maio de 2019, a MEC-RED contava com 31.488 recursos e passou por uma grande expansão, sendo que em setembro de 2020, o volume da MEC-RED era de 321.328 recursos. Entretanto, destes últimos somente 43.742 recursos possuíam a informação sobre etapas de ensino preenchida; 86% dos recursos não indicam a etapa de ensino, dificultando a sua busca, uso e expansão destes recursos.

Um exemplo de REA pode ser visto na Figura 1, em que é possível observar o título, o componente curricular, uma descrição sobre o recurso e a etapa de ensino para a qual o REA se destina. Com esta informação, o REA pode ser usado adequadamente pelo público para o qual foi criado e trabalhado em salas de aula por professores e alunos. A etapa de ensino nos materiais é fundamental dentro de repositórios abertos, principalmente por ela ser um *proxy* ou representante da complexidade textual e conceitual dos materiais e por estar relacionada com a evolução do aprendizado e com a compreensão desses materiais.

A Figura 2 traz um exemplo de REA em que a etapa de ensino não está indicada e, além de prejudicar a própria busca do material dentro do repositório, também inviabiliza seu uso em salas de aulas, ou força o professor a avaliar a sua adequação e inteligibilidade textual antes de seu uso. Esta falta prejudica toda uma cadeia de uso — a ineficiência em buscar o material, a

¹ <<https://plataformaintegrada.mec.gov.br/>>

Figura 1 – Exemplo de material **com etapa de ensino informada** para uso por estudantes e professores.

A literatura infantil ocidental : Chapeuzinho vermelho, de Charles Perrault, França, séc. XVII

Componentes Curriculares: Língua Portuguesa

Outras Temáticas:

Etapas de Ensino: Ensino Fundamental I (1º até o 5º ano)

REPORTAR ABUSO OU ERRO COMPARTILHAR GUARDAR ABRIR RECURSO

Sobre o Recurso

Charles Perrault foi um escritor e poeta francês do século XVII, que estabeleceu bases para um novo gênero literário, o conto de fadas, além de ter sido o primeiro a dar acabamento literário a esse tipo de literatura, feito que lhe conferiu o título de Pai da Literatura Infantil. Suas histórias mais conhecidas são Chapeuzinho Vermelho, A Bela Adormecida, O Gato de Botas, Cinderella, Barba Azul e O Pequeno Polegar.

Autoria: Universidade de São Paulo

Informações adicionais

Idioma: Português

Formato:

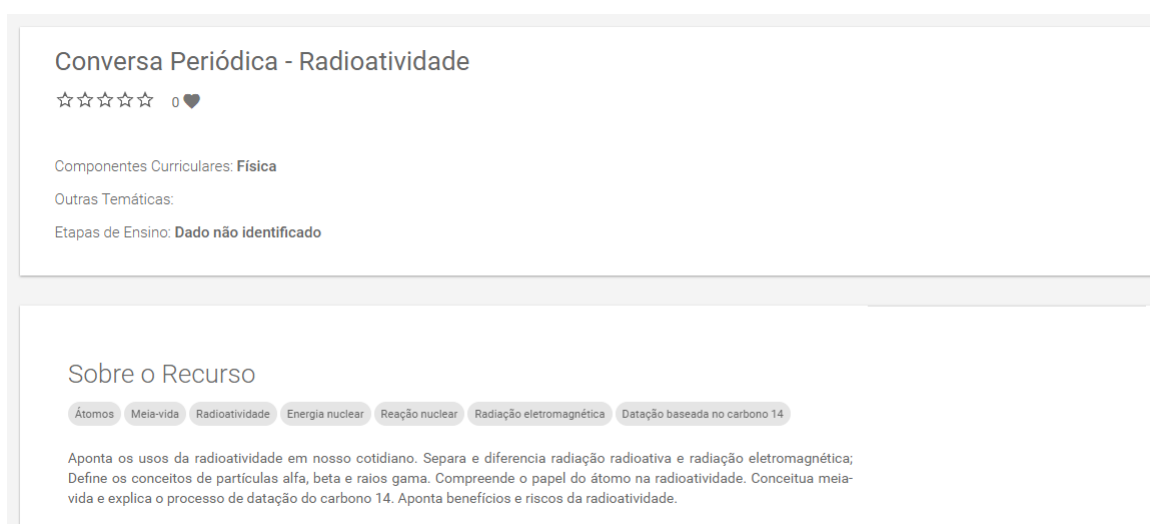
Data de Envio: 10/01/2019

Modificado em: 19/02/2020

Tipo de Licença: CC BY

Fonte: MEC (2019)

inviabilidade em aplicar o material e o mau uso pelo aluno que resgata um material não adequado a sua série escolar.

Figura 2 – Exemplo de material **sem etapa de ensino informada**.

Conversa Periódica - Radioatividade

☆☆☆☆☆ 0 ♥

Componentes Curriculares: Física

Outras Temáticas:

Etapas de Ensino: Dado não identificado

Sobre o Recurso

Átomos Meia-vida Radioatividade Energia nuclear Reação nuclear Radiação eletromagnética Datação baseada no carbono 14

Aponta os usos da radioatividade em nosso cotidiano. Separa e diferencia radiação radioativa e radiação eletromagnética; Define os conceitos de partículas alfa, beta e raios gama. Compreende o papel do átomo na radioatividade. Conceitua meia-vida e explica o processo de datação do carbono 14. Aponta benefícios e riscos da radioatividade.

Fonte: MEC (2019)

1.2 Tarefas da Pesquisa

As tarefas de pesquisa deste trabalho são Avaliação da Qualidade dos Metadados de REA e Avaliação da Qualidade do Conteúdo de REA via Complexidade Textual e são detalhadas abaixo.

Avaliação da Qualidade dos Metadados com Métodos de PLN

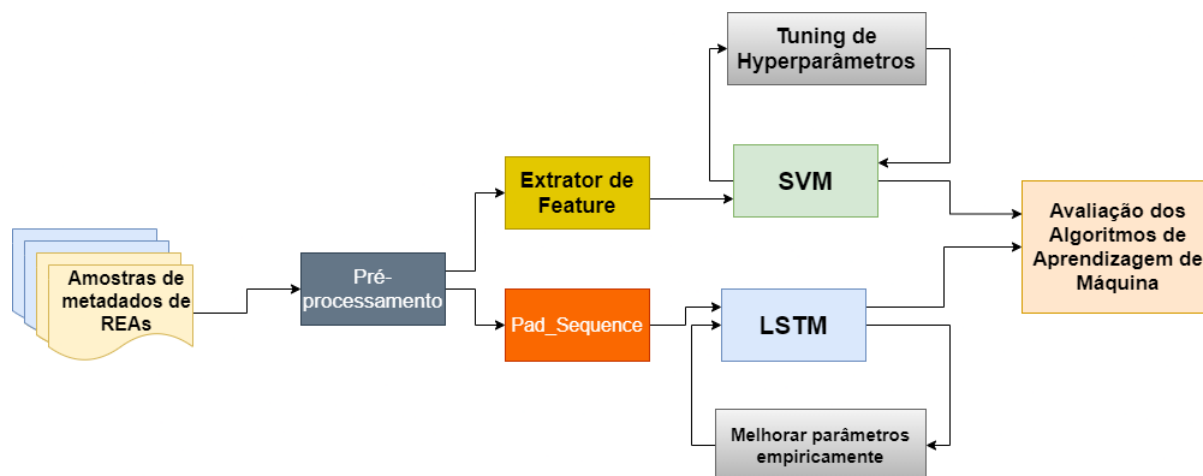
Para propor uma nova abordagem de avaliação da qualidade de metadados de REA, foram analisados os trabalhos da literatura que tratam da qualidade de REA, tais como: (BETHARD *et al.*, 2009), (DALIP *et al.*, 2011), (LEARY *et al.*, 2011a), (CECHINEL; SANCHEZ-ALONSO; GARCIA-BARRIOCANAL, 2011), (MOISE; VLADOIU; CONSTANTINESCU, 2011), (WETZLER *et al.*, 2013), (SAWADOGO; CHAMPAGNAT; ESTRAILLIER, 2014), (PONS *et al.*, 2015), (CAEIRO-RODRÍGUEZ *et al.*, 2016), e (AHMED; FUGE, 2017).

Os trabalhos analisados possuem uma semelhança: quase todos utilizaram métodos tradicionais de aprendizagem de máquina. Além disso, o uso de SVM se mostrou superior às redes Perceptron. Nenhum trabalho utilizou técnicas de *Deep Learning* para avaliar a qualidade automática de metadados de REA.

Os trabalhos de (BETHARD *et al.*, 2009) (LEARY *et al.*, 2011a) (MOISE; VLADOIU; CONSTANTINESCU, 2011) (WETZLER *et al.*, 2013) apresentam avaliadores de qualidade de REA usando SVM como classificador e arquiteturas com diferentes indicadores de qualidade. Enquanto (CECHINEL; SANCHEZ-ALONSO; GARCIA-BARRIOCANAL, 2011) apresenta um classificador de qualidade automática para Objeto de Aprendizagem (OA) e (AHMED; FUGE, 2017) cria uma forma de avaliar desafios on-line abertos, também, com uso do SVM como melhor resultado. Porém, nenhum deles trata especificamente de redes neurais profundas para avaliação da qualidade.

Na Figura 3, é apresentada a arquitetura de avaliação da qualidade dos metadados de REA desenvolvida neste trabalho e que é detalhada no Capítulo 4. As amostras de metadados de REA (título e resumo) são pré-avaliadas para o treinamento. Na arquitetura apresentada (Figura 3), as amostras de REA são avaliadas com algoritmos de aprendizado de máquina supervisionado, e passam por alguns processos de pré-processamento, extração de features (TF-IDF) ou Pad-Sequence (completar a sequência de entrada para garantir que todas elas tenham o mesmo comprimento) e são treinados os modelos de aprendizado de máquina. Além disso, são usados métodos de otimização (*Tuning*) de hiperparâmetros (*Grid Search*) para se encontrar os melhores parâmetros de uma Long Short-Term Memory (LSTM) de forma empírica (testando diversos parâmetros e avaliando seus resultado), trazendo assim uma nova abordagem de avaliação da qualidade de metadados para a literatura.

Figura 3 – Arquitetura de avaliação da qualidade dos metadados de REA proposta nesta pesquisa



Fonte: Elaborada pelo autor.

Avaliação da Qualidade do Conteúdo via Complexidade Textual

A avaliação automática da complexidade textual (em inglês, *readability*) estuda o que faz alguns textos serem mais fáceis de serem lidos do que outros. Do ponto de vista pragmático, fornecer a cada estudante textos que sejam adequados a sua capacidade de leitura é um grande desafio para educadores de todas as disciplinas escolares, pois tomaria o tempo já escasso desse profissional. Mas, ao mesmo tempo, é um tema de grande importância, pois textos são a base do aprendizado e veiculam o conhecimento sobre todas as disciplinas. No cenário da MEC-RED, mostrado acima, vimos como a falta de indicação da etapa de ensino impacta negativamente no cenário de reuso dos materiais.

A literatura mais recente em avaliação automática da complexidade textual traz três grandes abordagens para essa tarefa: (i) uso de técnicas estatísticas como a Análise de Componentes Principais (*Principal Component Analysis*, PCA, em inglês) (GRAESSER; MCNAMARA, 2011), (GRAESSER; MCNAMARA; KULIKOWICH, 2011); (ii) a abordagem de aprendizado de máquina com o uso de métodos de seleção de features e avaliação da contribuição de conjuntos de features para a tarefa (SCARTON; ALUÍSIO, 2010), (HARTMANN *et al.*, 2016), (FILHO *et al.*, 2016); e (iii) a abordagem de *Deep Learning*, que usa arquiteturas neurais para evitar o uso de engenharia de *features* (NADEEM; OSTENDORF, 2018), (DEUTSCH; JASBI; SHIEBER, 2020). Entretanto, até onde sabemos, não encontramos trabalhos que aplicaram a abordagem Multi-task Learning (MTL) com o propósito de construir um preditor de complexidade textual, embora a abordagem tenha já sido avaliada para a complexidade sentencial ((GONZALEZ-GARDUÑO; SØGAARD, 2017) e (LEAL; RODRIGUES; ALUÍSIO, 2020)).

Dado o cenário das 3 abordagens acima, com relação à abordagem (i) implementamos e aplicamos a PCA e avaliamos as *features* mais representativas em cada componente principal. Na abordagem (ii) implementamos e aplicamos o PCA, Lasso e o *Correlation-based Feature*

Selection (CFS) e comparamos com os algoritmos de aprendizado tradicional, avaliando o uso desses métodos de seleção de *features* na complexidade textual para a predição da etapa escolar da MEC-RED. Por fim, na abordagem (iii), em síntese, desenvolvemos seis modelos de *Deep Learning* e dois modelos usando MTL para avaliação da complexidade textual para predição da etapa escolar.

1.3 Questões de pesquisa, Hipóteses e Objetivos

Questões de pesquisa

Dado o cenário acima, selecionamos quatro questões de pesquisa para responder na pesquisa doutorado, sendo elas:

1. É possível avaliar a qualidade dos REA, usando apenas os metadados título e resumo, nos casos em que o conteúdo do recurso não é apresentado textualmente? Por exemplo, em vídeos e áudios ?
2. O uso de métodos de seleção de *features* com os métodos clássicos de aprendizado de máquina têm um impacto positivo na predição da complexidade textual ?
3. O uso de algoritmos de Deep Learning, com Redes Neurais Recorrentes, Word Embeddings e Mecanismo de Atenção, possuem melhor desempenho do que os algoritmos de aprendizado de máquina tradicional, que usam engenharia de features ? O uso de *features* linguísticas nas arquiteturas Deep Learning trazem algum benefício ?
4. Um classificador desenvolvido com dois registros de língua (fala e escrita) via MTL tem melhor desempenho nos REA em Português do que um modelo de tarefa única treinado apenas com textos escritos de várias etapas escolares ?
5. Há impacto positivo no desempenho dos modelos se textos do domínio da MEC-RED forem incluídos no treinamento de classificadores usando aprendizado MTL?

Objetivos

O objetivo geral deste trabalho é alavancar a área de pesquisa de Complexidade Textual no Português Brasileiro e a Qualidade de REA para esta língua, avaliando a qualidade dos metadados de REA e avaliando a complexidade textual de REA para predição da etapa escolar, um dos metadados da MEC-RED. Os objetivos específicos são:

1. Com objetivo exploratório, avaliar as abordagens de seleção de *features* (CFS, Lasso e PCA) no uso da complexidade textual para predição da etapa escolar.

2. Propor arquiteturas de *Deep Learning* para estudar e avaliar a complexidade textual para predição da etapa escolar.
3. Propor arquiteturas *Single e Multi-task Learning* para estudar e avaliar a complexidade textual para predição da etapa escolar.
4. Usar e implementar métodos de aprendizado de máquina tradicional para estudar e avaliar a complexidade textual para predição da etapa escolar.
5. Implementar métodos de aprendizado de máquina tradicional e de *Deep Learning* para estudar e avaliar a predição automática da qualidade dos metadados de REA.
6. Comparar o desempenho dos métodos criados de aprendizado de máquina tradicional, single e MTL para predição da etapa escolar.

Hipóteses

1. O uso de métodos de *Deep Learning* e MTL apresentam melhor desempenho na avaliação da complexidade textual para predição da etapa escolar em relação aos métodos tradicionais de aprendizado de máquina, quando há um grande volume de dados para o aprendizado.
2. O uso de metadados para avaliação da qualidade de REA, em materiais que não possuem conteúdo textual, como vídeos e áudios, pode ser uma alternativa de avaliação da qualidade do REA.

1.4 Organização da Tese

Os capítulos que seguem são organizados da seguinte maneira. No Capítulo 2 é apresentada a primeira parte da Fundamentação Teórica necessária para o entendimento dessa monografia, especificamente, o foco são os REA, os padrões de metadados envolvidos nesses recursos, os indicadores de qualidade que avaliam os REA nos repositórios Web. No Capítulo de Fundamentação Teórica também apresentamos a descrição da Plataforma MEC-RED, uma análise da complexidade textual dos recursos da plataforma integrada do MEC e os dados armazenados nessa plataforma. Por fim, trazemos abordagens de aprendizado de máquinas usadas na pesquisa, envolvendo aprendizado de máquina tradicional via engenharia de *features*, *Deep Learning*, *Multi-task learning* e os conceitos de *Softmax* e *Cross Entropy Loss*. No Capítulo 3, apresentamos os trabalhos correlatos envolvendo qualidade de REA e avaliação automática da complexidade textual, divididos em análise de componentes principais, engenharia de *features* e modelos de *Deep Learning*. No Capítulo 4, é apresentado o trabalho que desenvolvemos para classificação da qualidade dos metadados de REA usando *Deep Learning*, divididos em uma análise descritiva do córpus anotado, um classificador *baseline*, um classificador SVM e uma arquitetura LSTM. No Capítulo 5, é apresentado nosso segundo trabalho desenvolvido para

predição da complexidade textual de REA no Português Brasileiro, com a descrição do *cópus* compilado, os experimentos usando aprendizado de máquina tradicional e métodos de seleção de *features*. Incluindo, uma avaliação intrínseca e extrínseca para os métodos tradicionais de aprendizagem de máquina e os métodos de *Deep Learning*. No Capítulo 6 é apresentado nosso terceiro trabalho desenvolvido considerando uma abordagem para complexidade textual de REA usando *Multi-Task Learning*, onde incluímos uma avaliação da complexidade textual com textos transcritos, análise descritiva dos *dataset* e das *features* linguísticas e os experimentos e resultados envolvidos. Por fim, no Capítulo 7, são apresentadas as conclusões desta pesquisa de doutorado com as contribuições, limitações, trabalhos futuros e uma lista de publicações.

FUNDAMENTAÇÃO TEÓRICA

2.1 Recursos Educacionais Abertos

Nesta seção são apresentados os conceitos teóricos sobre REA e seus principais padrões de metadados. Na Subseção 2.1.1, são descritos os repositórios de REA. Na Subseção 2.1.2, apresentamos os principais padrões de metadados utilizados por esses repositórios e são mostrados exemplos de repositórios de REA em diversas áreas de aplicação. Na Subseção 2.1.3, são apresentados os indicadores de qualidade usados na literatura para avaliação de recursos educacionais.

2.1.1 Caracterização

O interesse por Recursos Educacionais Abertos (REA) está crescendo fortemente. Eles são um dos principais elementos da educação aberta que visa permitir o acesso gratuito e reutilização, com ou sem modificação, ao conteúdo educacional em todo o mundo (OPENCONTENT, 2020b; ABERTA, 2020; GAZZOLA, 2015). Os REA podem ser caracterizados como materiais de ensino, aprendizagem e pesquisa, em qualquer meio de armazenamento, que estão amplamente disponíveis por meio de uma licença aberta que permite quatro liberdades mínimas conhecidas como 4R: Aprimorar (*Revise*), Usar (*Reuse*), Recombinar (*Remix*) e Distribuir (*Redistribute*) (ATKINS; BROWN; HAMMOND, 2007) (KANWAR; UVALIĆ-TRUMBIC; BUTCHER, 2011) (GIMENES; BARROCA; FELTRIM, 2012) (OPENCONTENT, 2020b)(ABERTA, 2020). Alguns autores ainda afirmam que o REA deve possuir 5R, sendo este quinto “R” de *Reter (Retain)* o REA, isto é, controlar uma cópia do recurso; fazer download e manter uma cópia do REA (COMMONS, 2020) (OPENCONTENT, 2020a).

Atualmente, diversas instituições de ensino e pesquisa têm investido no uso de REA para a disponibilização de conteúdo relacionado à educação, como The Open University¹ (OKADA,

¹ <<http://openlearn.open.ac.uk/>>

2007) (LITTLE *et al.*, 2011), MIT², FGV³, UNESP⁴ e USP⁵. Outra vertente que evidencia a propagação de REA são os *Massive Open Online Courses* (MOOCs). Porém, nem todos os MOOCs são REA. Quando os MOOCs atendem a definição de REA acima, eles podem oferecer uma educação alternativa e de qualidade tanto na complementação da educação tradicional quanto na formação continuada (MATKIN, 2013). REA estão relacionados com liberdades em relação a direitos autorais, mas não quer dizer que são obrigatoriamente gratuitos ou de acesso gratuito. A FAQ (*Frequently Asked Questions*) da comunidade “Iniciativa Educação Aberta”⁶ responde esta questão, reforçando que em alguns casos, “valores – geralmente mais baixos que de recursos não REA – são cobrados para acesso à unidade do REA, ou por serviços associados como para sua impressão ou *download* em um celular ou tablet”. Mesmo nestes casos, quase sempre uma versão on-line será gratuita em outra fonte.

REA podem incluir cursos completos, partes de cursos, módulos, guias para estudantes, anotações, livros didáticos, artigos de pesquisa, vídeos, instrumentos de avaliação, recursos interativos como simulações e jogos de interpretação, bancos de dados e aplicativos, dentre outros recursos.

O reconhecimento da necessidade de reutilização de materiais educativos gerou o desenvolvimento de padrões de metadados para compartilhamento e armazenamento de documentos (MCCLELLAND, 2003). Muitos repositórios de REA armazenam diversos formatos de arquivos, como: imagens, animações, arquivos de áudio, vídeos e outros. Para isso alguns padrões de metadados são usados nesses repositórios, como: DCMES, IMS/IEE-LOM, CNX, OGP e o vídeo sitemaps (GAZZOLA; CIFERRI; GIMENES, 2014; ABERTA, 2020). Para um melhor entendimento desses padrões de metadados que podem ser úteis e já foram descritas por outras pessoas, trazemos na próxima seção uma breve descrição de alguns desses padrões.

2.1.2 Metadados

Nesta seção são descritos os principais metadados obtidos dos repositórios de REA na Web. Foram descritos a sua forma de instanciação, seu formato e alguns exemplos. Além disso, é exibido um mapeamento das iniciativas brasileiras através de um mapa do Brasil e a localização estadual para cada iniciativa.

Padrões LOM, DCMES, OGP e Video SiteMap

O padrão IEEE/LOM foi desenvolvido em um esforço conjunto do Comitê de Padrões de Tecnologia de Aprendizagem em colaboração com DCMI e outras organizações. A *Instructional*

² <<http://ocw.mit.edu>>

³ <<http://www5.fgv.br/fgvonline/Cursos/Gratuitos>>

⁴ <<http://www.unesp.br/unespaberta>>

⁵ <<http://eaulas.usp.br/portal/home>>

⁶ <<http://aberta.org.br/faq/>>

Management System (IMS) continuou os trabalhos da IEEE/LOM (KOUTSOMITROPOULOS *et al.*, 2010) e criou seu próprio padrão, o qual é chamado IMS Learning Resource Metadata. Esse padrão é equivalente ao IEEE/LOM, sendo que a principal diferença refere-se à taxonomia utilizada nos metadados (GIMENES; BARROCA; FELTRIM, 2012). O DCMES (*Dublin Core Metadata Element Set*) (ou Dublin Core) está presente também em repositórios REA e pode representar seus metadados para o público com formatos diversos, isto é, o formato como os metadados são instanciados, mas não é sempre o mesmo (POWELL *et al.*, 2007). Por exemplo, o Connexions (DAVIS *et al.*, 2010) instancia seus metadados por meio do formato XML, enquanto o repositório Teses USP instancia seus metadados por meio do formato HTML/XHTML. Outros formatos incluem: HTML/XML, RDF/XML, XML e DC-DS-XML (POWELL *et al.*, 2007). O Protocolo *Open Graph* (OGP) foi criado pelo Facebook e foi inspirado no Dublin Core, Microformatos e RDFa (RDF em HTML com adição de atributos). O OGP requer apenas 4 informações essenciais: o título, o tipo, uma imagem miniatura (*thumbnail*) e uma URL (GRAHAM, 2012). Entretanto, ele possui muitos outros metadados opcionais como metadados para localização (latitude, longitude) e para vídeos (resolução e o tipo), dentre outros. O Vídeo sitemaps representa uma forma de se incluir metadados em vídeos e áudios. Ele é um arquivo XML que lista os URLs de um site junto com metadados adicionais sobre cada URL. No entanto, esses metadados não são suficientes para descrever vídeos e áudios. Como resultado, o Google estendeu os atributos tradicionais do Sitemap e adicionou novos metadados para descrição de vídeos e áudios.

Na Tabela 1, são apresentados os repositórios estudados em Gazzola (2015), a partir dos quais foi possível evidenciar a diversidade entre eles, desde os formatos de arquivos até os tipos de metadados utilizados. O tipo de metadados *Personalizado* refere-se aos tipos de metadados não catalogados.

Tabela 1 – Repositórios REA com diferentes padrões de metadados

Nome do repositório	Principal formato de arquivos	Padrões de metadados
Banco de Imagens Geográficas	imagens	ISO 19115:2003
Flickr	imagens	EXIF e Personalizado
Biblioteca Digital de Ciência	animações	Não encontrado
Banco Internacional de Objetos Educacionais	imagens e arquivos de áudio	DCMES
Connexions	textos	DCMES, IMS, CNX, MathML e OGP
OCW-MIT	vídeos e textos	OGP e Personalizado
WebEduc-MEC	textos, imagens e vídeos	Não encontrado
Khan Academy	vídeos	OGP e MathML
e-Aulas USP	vídeos	Vídeo Sitemaps

Fonte: Adaptada de Gazzola (2015).

A Tabela 2 sintetiza os principais padrões dos repositórios REA encontrados na Web. Mostra também a instanciação, isto é, como pode ser escrito o padrão de metadados. Além disso, são mostrados os elementos descritores que fazem referência aos campos que possibilitam descrever os REA. Por fim, a possibilidade de extensão do padrão que é um campo binário (Sim/Não) indicando se é possível adicionar mais elementos descritores.

Tabela 2 – Heterogeneidade dos padrões de metadados encontrado na Web

Padrão de metadados	Instanciação	Elementos descritores	Possibilidade de estender?
OGP	1 formato	4 elementos	Sim
Vídeo Sitemaps	1 formato	6 elementos	Sim
Dublin Core	5 formatos	15 elementos	Sim
IEEE/LOM	Não descrevem formatos	60 elementos divididos em 9 categorias	Sim

Fonte: Adaptada de Gazzola (2015).

A Tabela 3 exhibe 19 URLs de repositórios REA de vários países. Com essas URLs é possível ilustrar a extensão de propagação dos REA pelo mundo. A tabela mostra as URLs de diferentes países (16 países), com diferentes nacionalidades, línguas e culturas.

Tabela 3 – Algumas sementes utilizadas pelo SeeOER

Título do URL	País	URL
Banco Internacional de Objetos Educacionais	Brasil	objetoseducacionais2.mec.gov.br
Matemática Multimídia	Brasil	m3.ime.unicamp.br
RRU Open Educational Resources	Canadá	oer.royalroads.ca/moodle/
Open Educational Resources for Typography	Argentina	www.oert.org
The Le@rning Federation	Austrália	www.ndlrn.edu.au
Educar Chile	Chile	www.educarchile.cl
Banco de Objetos de Aprendizaje	Colômbia	aplicaciones.virtual.unal.edu.co
Eduteka	Colômbia	www.eduteka.org
RVP Metodicky Portal	Rep. Checa	dum.rvp.cz
Materialeplatformen	Dinamarca	materialeplatform.emu.dk
Open Science Resources	União Europeia	www.osrportal.eu
Edu Fi	Finlândia	www.edu.fi
FREIburger Multimedia Object Repository	Alemanha	freimore.uni-freiburg.de
Open Educational Resources (OER) Africa	Quênia	www.oerafrica.org
VCILT	Rep. Maurícia	vcampus.uom.ac.mu
Aljazeera creative commons repository	Catar	cc.aljazeera.net
Maknaz	Arábia Saudita	maknaz.elc.edu.sa
University of Leicester OER Repository	Reino Unido	www2.le.ac.uk/projects/oer
Connexions	Estados Unidos	cnx.org

Fonte: Adaptada de Gazzola (2015).

Exemplos de repositórios de REA são mostrados na Figura 4. A Figura 4 mostra a página principal de um repositório de REA para saúde pública⁷ criado por uma rede descentralizada de

⁷ <<http://brasil.campusvirtualsp.org/node/3638>>

peessoas, instituições e organizações. Neste repositório intitulado CVSP são compartilhados REA com o objetivo de fortalecer as competências dos trabalhadores da saúde pública.

Figura 4 – Repositório de REA para saúde



Fonte: *print screen* do site do repositório Campus Virtual de Saúde Pública⁷.

2.1.3 Indicadores de Qualidade

Nesta seção, são apresentados os trabalhos relacionados a indicadores de qualidade de REA, considerando diversos contextos na área educacional. Para cada trabalho, é detalhada a sua abordagem de critérios desenvolvidos para indicar a qualidade de REA.

O trabalho de [Park e Tosaka \(2009\)](#) faz uma avaliação da qualidade de metadados, especificamente em repositórios on-line. Os repositórios on-line armazenam os REA na Web. [Park e Tosaka \(2009\)](#) avalia práticas de controle de qualidade de metadados nesses repositórios e para isso usa um levantamento on-line de profissionais de catalogação e metadados nos Estados Unidos. O estudo examina: (1) a importância percebida da qualidade dos metadados, (2) os critérios e questões de avaliação da qualidade dos metadados, e (3) os mecanismos para construir a garantia de qualidade no processo de criação de metadados. [Park e Tosaka \(2009\)](#) conclui que a semântica de metadados afeta muito o preenchimento realizado pelos usuários da aplicação e isso implica futuramente em consistência e precisão. A forte conscientização da qualidade dos metadados está correlacionada com a adoção generalizada de vários mecanismos de controle de qualidade, tais como treinamento do pessoal, revisão manual, diretrizes de metadados e ferramentas de geração de metadados. No entanto, as diretrizes de metadados são usadas com menos frequência como um mecanismo de garantia de qualidade nesses repositórios de REA que muitas vezes envolvem várias instituições.

Leary *et al.* (2011b) criaram um guia para avaliar e definir a qualidade para recursos educacionais. Eles afirmam que definir o que é “qualidade” é uma tarefa muito difícil e depende de diversas percepções sobre qualidade; e colocam que é de suma importância a revisão por um outro usuário e deste usuário dependerá qual a finalidade desta revisão e seus conhecimentos prévios sobre o assunto que esteja revisando ou qualificando. Os autores analisaram também de forma detalhada os principais trabalhos da literatura sobre indicadores de qualidade em recursos educacionais on-line e criaram 6 grupos qualitativos; esses critérios foram descritos e agrupados na Tabela 4. Após as definições estipuladas, os autores criaram uma forma de pontuação baseada em 5 estrelas, sendo que 1 estrela representa a pior pontuação e 5 estrelas representam a maior pontuação para um recurso educacional on-line. Além disso, usaram as definições gerais estipuladas na Tabela 4 para fazer uma tabela de pontuação para um projeto instrucional voltado para Inteligência Artificial, como mostra a Figura 5.

A fim de criar um guia mais geral para os critérios de qualidade sobre recursos educacionais on-line, os autores definiram uma versão final para ser usada de forma geral. Essa versão consta de 6 grupos qualitativos divididos em: Precisão, Clareza Textual, Links do Recurso, Completude do Recurso, o Recurso em si e no Geral. Além disso, para cada grupo existe uma linha para comentários para cada revisor poder opinar sobre tal critério. A Tabela 5 mostra o guia final de indicadores criados pelos autores Leary *et al.* (2011b). Porém, a linha de comentários para cada grupo qualitativo foi subtraída.

De forma geral, o trabalho de Leary *et al.* (2011b) criou critérios objetivos e claros para definição de qualidade sobre recursos educacionais voltados para Web. Além disso, deixaram clara a fundamentação que usaram para criar tais critérios. Mostraram 3 formas de modelagem para serem usadas em cada caso específico. Por fim, criaram um modelo generalizado que pode ser usado em qualquer recurso educacional.

2.2 Plataforma MEC-RED

Nesta seção, apresentamos a plataforma de Recursos Educacionais do Ministério da Educação (MEC) que serviu de estudo para os métodos de Complexidade Textual avaliados nesta tese. Na Subseção 2.2.1, são apresentados dados quantitativos dos filtros de busca “Etapas de Ensino”, “Tipos de Recursos” e “Componentes Curriculares” da MEC-RED.

2.2.1 Análise quantitativa dos recursos da Plataforma do MEC

No Brasil, a plataforma de Recursos Educacionais do Ministério da Educação (MEC), conhecida como Plataforma Integrada MEC⁸ (MEC-RED), é um exemplo de REA e faz parte de um dos compromissos do Brasil na *Open Government Partnership* para fortalecer práticas que envolvem a transparência dos atos governamentais e promovem a participação social e o

⁸ <<https://plataformaintegrada.mec.gov.br/>>

Tabela 4 – Tabela com 6 critérios que indicam qualidade de um recurso educacional on-line

Critério de Qualidade	Definições
Interface	<p>O recurso é atraente.</p> <p>O recurso é fácil de navegar.</p> <p>O recurso não contém publicidade.</p> <p>O recurso contém links que funcionam.</p> <p>O recurso foi projetado para acomodar dispositivos móveis</p>
Confiabilidade Técnica	<p>O recurso usa multimídia (flash, cliques de som, vídeos ou applets em java) que funcionam.</p> <p>O recurso identifica claramente os requisitos necessários para utilização.</p> <p>O recurso contém manual de instrução para utilização.</p>
Conteúdo	<p>O conteúdo dos recursos possui coerência e coesão textual.</p> <p>O conteúdo do recurso está completo.</p> <p>O conteúdo do recurso é imparcial, ou seja, não se vale apenas da opinião do autor.</p> <p>O conteúdo do recurso possui objetivos claros.</p>
Pedagogia	<p>O recurso é envolvente.</p> <p>A fonte é motivadora.</p> <p>O recurso é interativo.</p> <p>Os recursos incluem avaliação(ões).</p> <p>O recurso fornece feedback.</p> <p>O recurso apoia os alunos a prosseguirem ao seu próprio ritmo.</p> <p>O recurso suporta recursos introdutórios, de reforço ou atividades que se completam.</p>
Administrativo	<p>O recurso contém links diretos e explícitos para o país e estado que foi criado e seus padrões de ensino.</p> <p>O recurso contém informações sobre o autor do recurso, como e-mail ou site do autor.</p> <p>O recurso é descrito por metadados.</p>
Outros	<p>O recurso requer um valor monetário para ser de acessado.</p> <p>O nível geral de avaliação ou confiança do recurso por meio de outra fonte.</p>

Fonte: Adaptada de [Leary et al. \(2011b\)](#).

Figura 5 – Formulário de Qualidade para um recurso educacional on-line para inteligência artificial

Number	Criteria	Rating
1	Content accuracy	<p>☆ ☆ ☆ ☆ ☆</p> <p>Very Somewhat Not enough Somewhat Very Inaccurate Inaccurate Information Accurate Accurate</p> <p>Comments:</p>
2	Text clarity	<p>☆ ☆ ☆ ☆ ☆</p> <p>Very Somewhat Not enough Somewhat Very Unclear Unclear Information Clear Clear</p> <p>Comments:</p>
3.	Links in project	<p>☆ ☆ ☆ ☆ ☆</p> <p>No links Some links Not enough Some links All links work don't work Information work work</p> <p>Comments:</p>
4	Project completeness: includes the state standard learning goal, assessment, etc.	<p>☆ ☆ ☆ ☆ ☆</p> <p>Very Somewhat Not enough Somewhat Very Incomplete incomplete Information complete complete</p> <p>Comments:</p>
5	The project: (circle one)	<p>a. Provides a resource list b. Teaches a concept c. Reinforces a concept d. Provides a summary of content with a learning activity e. Don't Know f. Does something else (describe below)</p>
6	Overall rating of the project	<p>☆ ☆ ☆ ☆ ☆</p> <p>Meets no Meets very Meets some Meets Exceed criteria little criteria criteria criteria criteria</p> <p>Comments:</p>

Fonte: Adaptada de Leary *et al.* (2011b).

Tabela 5 – Guia definitivo dos critérios de qualidade

Precisão			
A informação é confiável, verdadeira e atual			
Muito impreciso	Impreciso	Preciso	Muito preciso

Clareza Textual			
O nível escolar está apropriado e o comprimento textual do recurso está adequado			
Muito pouco claro	Pouco claro	Claro	Muito Claro

Links do Recurso			
Os links estão funcionais e associados corretamente			
Todos os links não funcionam	Alguns links não funcionam	Alguns links funcionam	Todos os links funcionam

Completeness do Recurso			
O recurso inclui uma padronização, objetivos, metas de aprendizagem, avaliação, e pelo menos um exemplo			
Muito incompleto	Incompleto	Completo	Muito Completo

O Recurso			
Fornecer uma lista de outros recursos	Reforça um conceito	Ensina um conceito	Ensina mais que um conceito

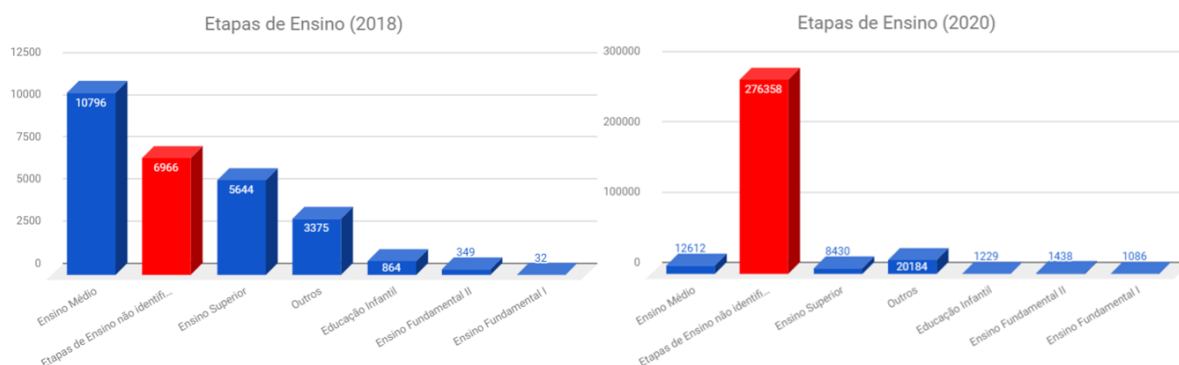
No geral			
Não satisfaz nenhum critério	Satisfaz alguns critérios	Satisfaz todos os critérios	Satisfaz todos os critérios e excede

Fonte: Adaptada de [Leary et al. \(2011b\)](#).

acesso à informação pública ([MEC, 2019](#)). Devido à importância do tema ([MIAO; MISHRA; MCGREAL, 2016](#)) ([WILEY; BLISS; MCEWEN, 2014](#)), a Organização das Nações Unidas (ONU) definiu os principais problemas relacionados ao desenvolvimento e uso de REA: i) *o problema de qualidade dos REA*; ii) *o problema da busca, ou seja, como encontrar REA*; iii) *o problema da sustentabilidade*, isto é, como financiá-los; iv) *o problema de localização e recontextualização dos REA*; e v) *o problema do remix*, isto é, a dificuldade de identificar a granularidade da mudança de conteúdo por outras pessoas e o nível de mudança.

Na MEC-RED, há cinco filtros de busca por recursos: (i) componentes curriculares/disciplinas, (ii) tipos de recurso, (iii) etapas de ensino, (iv) idiomas e (v) palavra-chave, embora a plataforma tenha outros metadados como título, pessoas que favoritam o recurso, URL para download associado ao material, descrição do recurso, autor do envio, autor do material, *tags* associadas ao material e tipo de recurso.

Figura 6 – Comparativo entre o número de recurso em cada Etapa de Ensino em dois datasets da MEC-RED (2018 e 2020)



Fonte: Elaborada pelo autor.

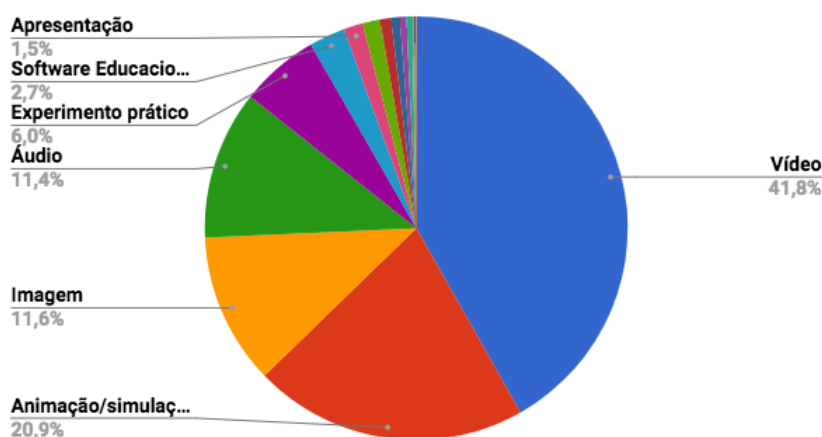
Em maio de 2019, a MEC-RED contava com 31.488 recursos e passou por uma grande expansão, sendo que em setembro de 2020, o volume da MEC-RED era de 321.337 recursos. Entretanto, destes últimos somente 44.979 recursos possuíam a informação sobre etapas de ensino preenchida; 86% dos recursos não indicam a etapa de ensino, dificultando a sua busca, uso e expansão destes recursos.

Na Figura 6, é apresentado um comparativo entre as etapas de ensino dos recursos que estavam presentes na recolha de abril de 2018, realizada por este doutorando (mostrado à esquerda), e as etapas dos recursos de uma análise realizada em maio de 2020 (mostrado à direita); a barra vermelha exhibe o número de recursos com etapa de ensino não identificada. Este metadado, particularmente, permite realizar a busca por material relacionado com a sua complexidade textual e conceitual, para a recuperação de material adequado a uma das quatro etapas do Sistema Educacional Brasileiro — Ensino Fundamental I e II, Ensino Médio e Ensino Superior. Assim, espera-se o seu preenchimento correto, sendo um item importante para se avaliar a qualidade de um recurso e para facilitar sua busca — dois dos problemas de REA elencados pela ONU e citados acima.

Como a MEC-REC possui 15 tipos de recursos (apresentação, software educacional, experimento prático, áudio, imagem, animação, vídeo, aplicativo móvel, infográfico, jogo, livro digital, mapa, plano de aula, website externo, outros) (Figura 7), excluimos a análise dos áudios, imagens, infográficos, mapas e vídeos nos estudos de complexidade textual, por não se apresentarem no formato textual. Apresentações, livros digitais, *website externo*, planos de aula e textos são recursos que se aproximam do gênero de textos didático. Os tipos de recursos animação/simulação, aplicativo móvel, jogos, experimento prático e software educacional são de interesse especial para os estudos de complexidade textual, porque são bem diferentes linguisticamente do dataset compilado para treinamento dos modelos de predição automática de complexidade textual, descrito no Capítulo 5.

Porém, na avaliação dos recursos da MEC-RED em 2019, 100% dos aplicativos móveis e

Figura 7 – Porcentagens dos Tipos de Recursos mais representativos no dataset compilado em 2018



Fonte: Elaborada pelo autor.

99% dos jogos não apresentavam informação sobre a etapa de ensino, inviabilizando a compilação de um grande corpúsculo balanceado de REA para o treinamento para um preditor de complexidade textual. Este cenário da MEC-RED nos levou a propor uma outra abordagem para a avaliação da complexidade textual, via Multi-task learning, descrita no Capítulo 6.

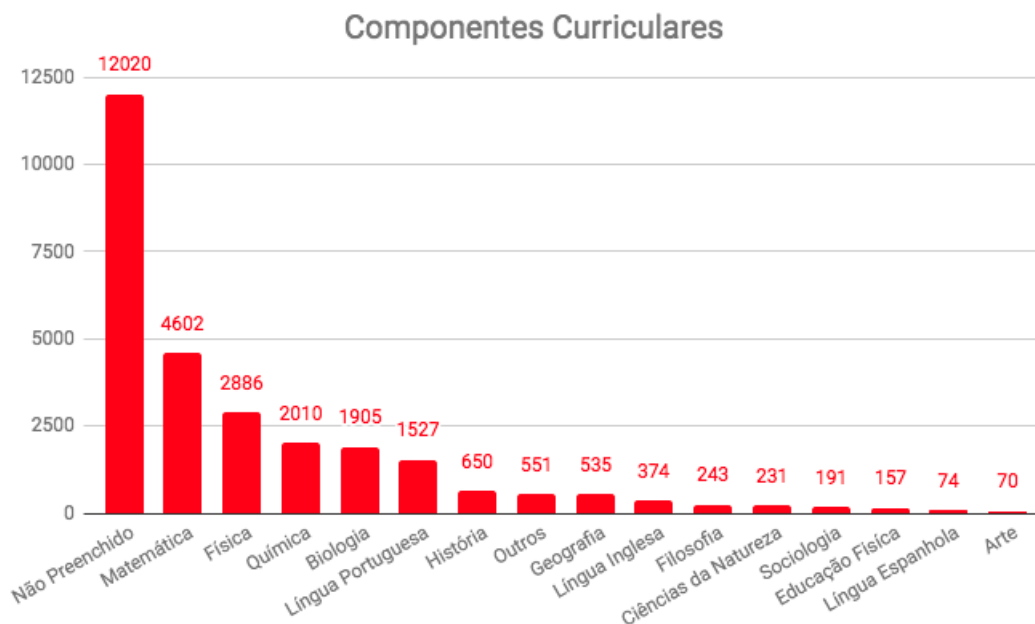
Finalmente, trouxemos também uma análise quantitativa dos dados de “Componentes Curriculares” da plataforma MEC-RED (Figura 8), divididos em “não preenchidos”, Matemática, Física, Química, Biologia, Língua Portuguesa, História, Geografia, Língua Inglesa, Filosofia, Ciências da Natureza, Sociologia, Educação Física, Língua Espanhola, Arte e Outros. Podemos observar que 12.020 recursos envolvendo componentes curriculares não foram preenchidos e 550 recursos foram preenchidos como “Outros”, que impactam negativamente na busca por um recurso na MEC-RED.

Em resumo, embora tenhamos escolhido estudar e propor modelos de avaliação automática de complexidade textual para completar o metadado “Etapas de Ensino”, “Componentes Curriculares” também é um forte candidato a um estudo e proposta de preditores automáticos.

2.3 Abordagens de Aprendizado de Máquina na Pesquisa

Nesta seção, é apresentada a fundamentação teórica sobre aprendizado de máquina supervisionado que embasa a tarefa de avaliação da complexidade textual. Na subseção 2.3.2, são apresentadas as principais diferenças entre aprendizagem de máquina tradicional e *Deep Learning*, incluindo uma abordagem do uso da aprendizagem supervisionada. Além disso, são apresentadas as Redes Neurais Recorrentes e *Long Short-Term Memory* (LSTM). Na seção 2.3.3 são abordados os conceitos e aplicação do Multi-task Learning. Por fim, na seção 2.3.4 são abordados os conceitos da função de ativação *Softmax* e da função de perda *Cross Entropy Loss* usados neste trabalho.

Figura 8 – Componentes curriculares do dataset compilado

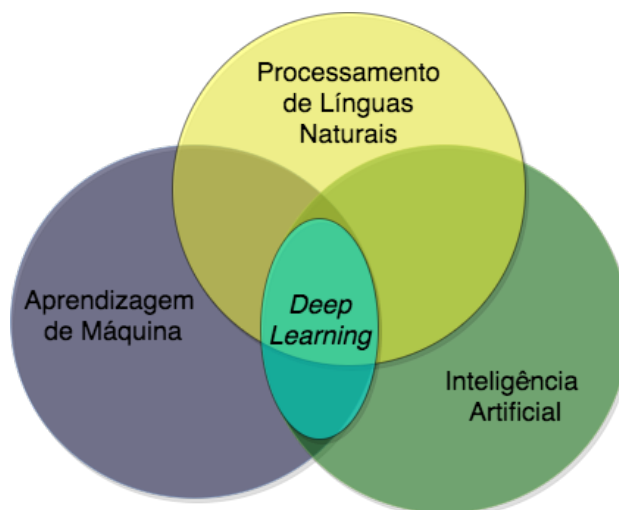


Fonte: Elaborada pelo autor.

2.3.1 Aprendizagem de Máquina Clássico via Engenharia de Features

A Inteligência Artificial (IA) é um campo de pesquisa bastante extenso. Esse campo de pesquisa trabalha com tarefas que usam dados para modelar as capacidades cognitivas semelhantes aos seres humanos como: inferência, dedução, visão, reconhecimento de fala, tradução e outras (RUSSELL; NORVIG, 2005), ou mesmo para fazer a própria área evoluir, como é o caso do uso de redes neurais artificiais já estudadas no passado e atualmente revisitadas com o emprego em larga escala da abordagem de *Deep Learning* (MIKOLOV *et al.*, 2013b). Já a Aprendizagem de Máquina tradicional é uma subárea da IA na qual se estudam os algoritmos de aprendizagem para os quais não é necessário programar todas as tarefas específicas que o computador precisa aprender. Com isso, é possível criar algoritmos que aprendam e façam previsões no conjunto de dados. Em geral, a Aprendizagem de Máquina é dividida em 3 categorias (LONG; SHELHAMER; DARRELL, 2015): i) aprendizagem supervisionada; ii) aprendizagem sem supervisão; e iii) aprendizagem por reforço. Na aprendizagem supervisionada é apresentada para a máquina dados de entrada e saída desejados. E, o objetivo é aprender com esses exemplos de treinamento de tal forma que possam ser feitas previsões significativas para novos dados não vistos. Na aprendizagem sem supervisão são apresentados apenas os dados de entrada e a máquina deve encontrar alguma estrutura significativa por si mesma sem supervisão externa; ainda existe a possibilidade de usar um método híbrido entre aprendizagem supervisionada e sem supervisão, chamado aprendizado semi-supervisionado. Na aprendizagem por reforço, são apresentados os dados de entrada, porém, a máquina atua como um agente interagindo com o meio ambiente e aprendendo quais são os comportamentos que geram recompensas ou punições.

Figura 9 – Conjuntos das áreas de IA, Aprendizado de Máquina e Processamento de Línguas Naturais



Fonte: Adaptada de Kochmar (2017).

2.3.2 Deep Learning

Em *Deep Learning* ou Aprendizagem Profunda, a palavra profunda se refere à presença de muitas camadas na rede neural artificial. Mas, esse significado mudou ao longo do tempo (BENGIO *et al.*, 2009; LECUN; BENGIO; HINTON, 2015; DRAELOS *et al.*, 2017). Com o advento do trabalho de Mikolov *et al.* (2013b) criou-se uma forma para representar vetores de características conhecidos como *word embeddings* e voltados especificamente para área de Processamento de Línguas Naturais (PLN) (Seção 2.3.2). Após isso, houve uma conglomeração de novos recursos linguísticos associadas a *word embeddings* e *Deep Learning* criando métodos mais eficientes e eficazes para área de PLN. Além disso, o uso de *Deep Learning* foi aplicado em diversos domínios como imagem, texto, vídeo, fala e visão. Essas aplicações melhoraram significativamente os resultados do estado da arte anterior para as tarefas de processamento de imagens, fala e visão, por exemplo. Além disso, o sucesso de *Deep Learning* também se deve à maior disponibilidade de dados para treinamento e o custo baixo para utilização de GPUs para computação numérica eficiente. Podemos concluir que o advento do uso de *Deep Learning* redesenhou a área de PLN (Figura 9), trazendo uma grande variedade de novas abordagens, arquiteturas neurais e novos valores de precisão para métodos do estado da arte, superando os antigos valores da literatura (KOCHMAR, 2017).

Além disso, algoritmos de aprendizagem de máquinas se tornaram muito relevantes com a expansão do uso de *smartphones* e o grande volume de dados na Web. Isso se deve à importância de aplicações como, por exemplo, filtragem de conteúdo em redes sociais, recomendações em sites de comércio eletrônico e o reconhecimento de faces. As técnicas convencionais para aprendizagem de máquina são limitadas na sua capacidade de processar dados naturais em sua forma bruta (LECUN; BENGIO; HINTON, 2015), sendo necessário, para construção de um sistema de aprendizagem de máquina, um conhecimento profundo sobre o do sistema que se

deseja construir. Além disso, é necessária uma engenharia cuidadosa na escolha de *features* para que o sistema funcione adequadamente (SILVER *et al.*, 2016).

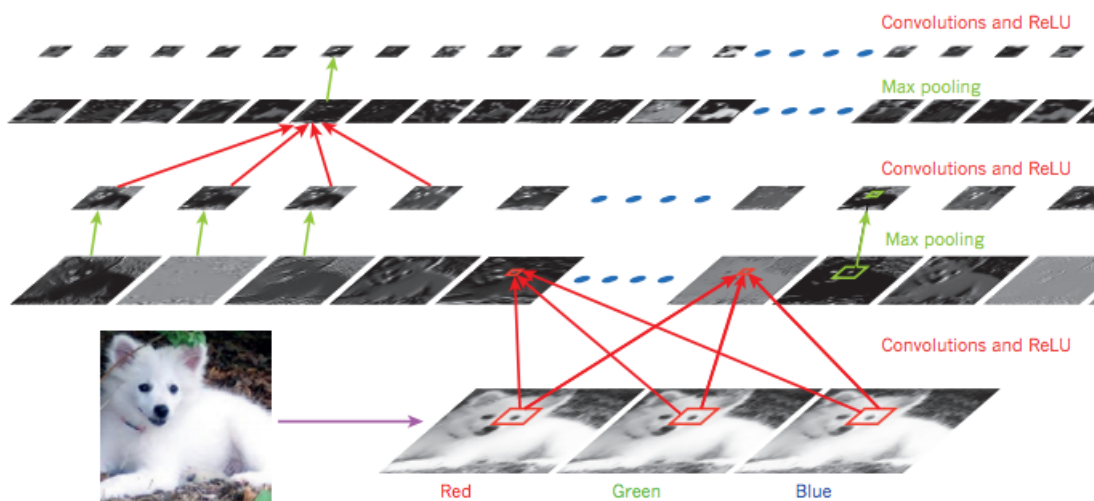
Para a abordagem convencional de aprendizagem de máquina é necessário construir um extrator para transformar os dados brutos (por exemplo, valores de pixel de uma imagem) para uma representação interna adequada como um vetor de características/*features*. E, a partir daí são usados um sistema de aprendizagem, muitas vezes um classificador, para classificar os padrões de entrada.

A área se voltou para o desenvolvimento de algoritmos de aprendizagem de representação (LECUN; BENGIO; HINTON, 2015; ESTEVA *et al.*, 2017). Aprendizagem de representação é um conjunto de métodos que permite que uma máquina seja alimentada com dados brutos e descubra automaticamente as representações necessárias para detecção e classificação. Métodos de *Deep Learning* são métodos de aprendizagem de representação com múltiplos níveis de representação. Esses múltiplos níveis de representação são obtidos pela composição de módulos simples, mas não lineares. Isso quer dizer que, para cada nível, o módulo transforma a representação um outro nível. Por exemplo, o primeiro nível, com dados brutos de entradas, sendo o nível superior o mais abstrato (Figura 10).

Para classificação, usando *Deep Learning*, as camadas mais altas de representação amplificam os aspectos de entrada. Essa amplificação é importante para discriminação e remoção de variações que não são relevantes. Por exemplo, uma imagem tem como entrada uma matriz de valores, e os recursos aprendidos na primeira camada de representação normalmente representam a presença ou ausência de arestas em orientações e locais particulares na imagem. A segunda camada detecta arranjos particulares de bordas, independentemente de pequenas variações nas posições da borda. A terceira camada pode reunir combinações maiores que correspondem às partes de objetos familiares e as camadas subsequentes detectam objetos como combinações dessas partes. Porém, o aspecto principal de *Deep Learning* é que essas camadas não são projetadas por engenheiros humanos (ESTEVA *et al.*, 2017); elas são aprendidas com dados usando um procedimento de aprendizado de uso geral. Portanto, *Deep Learning* está trazendo grandes avanços na resolução de problemas que resistiram às melhores tentativas da comunidade de inteligência artificial.

Uma Rede Neural Recorrente (RNN) é uma classe de redes neurais que exploram a natureza sequencial da sua entrada (WU *et al.*, 2017). As RNN têm sido aplicadas em várias tarefas como: reconhecimento de dígitos (GRAVES; SCHMIDHUBER, 2009), geração de texto (SUTSKEVER; MARTENS; HINTON, 2011), criação de modelos de língua (MIKOLOV, 2012a; GRAVES, 2013; PASCANU; MIKOLOV; BENGIO, 2013), reconhecimento de voz (GRAVES; MOHAMED; HINTON, 2013), tradução automática (BAHDANAU; CHO; BENGIO, 2014; CHO *et al.*, 2014; SUTSKEVER; VINYALS; LE, 2014) e inferência textual (BOWMAN *et al.*, 2015; ROCKTÄSCHEL *et al.*, 2015). Uma RNN pode ser imaginada como um grafo de células (Figura 12), em que cada célula executa a mesma operação em cada elemento da sequência. Na

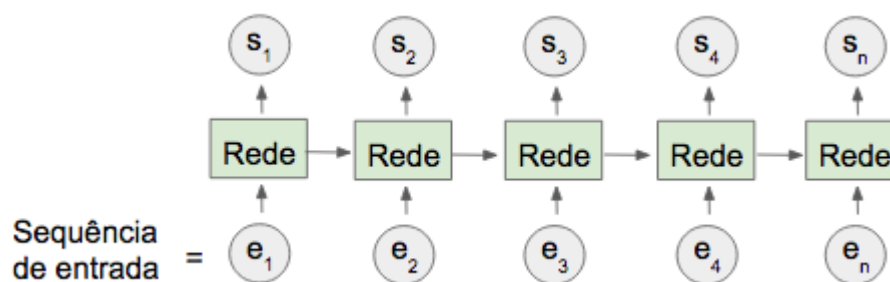
Figura 10 – Uma rede neural profunda com suas camadas e módulos



Fonte: LeCun, Bengio e Hinton (2015).

Figura 12, cada elemento da sequência é representado por e_n , em que n é o tamanho máximo da sequência de entrada. Além disso, o componente Rede representa parte de uma rede. Essa rede pode ser, por exemplo, uma RNN-Simples. O S_n representa cada ocorrência após ter passado por parte da Rede. As principais arquiteturas de RNNs são a arquitetura LSTM, do inglês *Long Short-Term Memory*, e a arquitetura GRU, do inglês *Gated Recurrent Units*. Essas duas arquiteturas não são as únicas. Porém, são as arquiteturas de RNNs mais usadas (JOZEFOWICZ; ZAREMBA; SUTSKEVER, 2015) para os principais problemas de sequência.

Figura 11 – Arquitetura de uma RNN

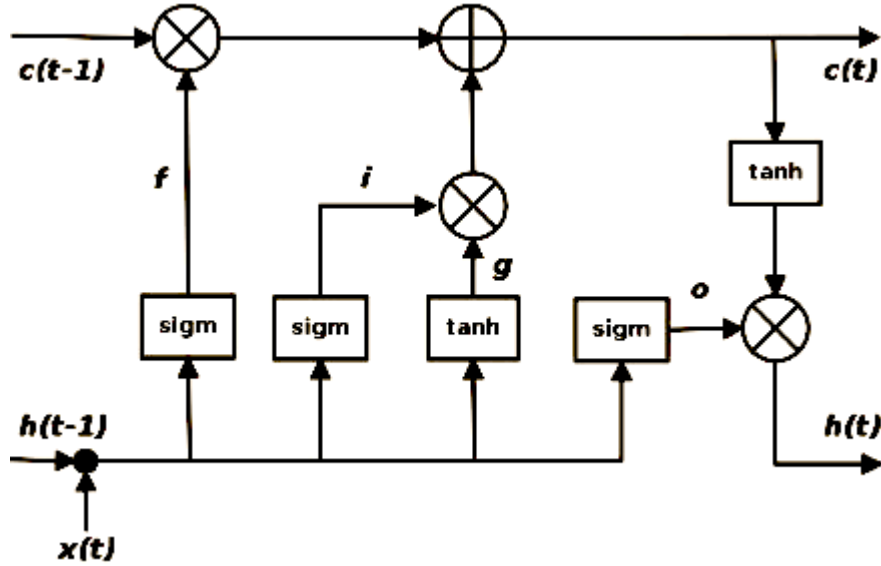


Fonte: Adaptada de Graves (2012).

A LSTM é uma variante da RNN, ela é capaz de aprender dependências de longo prazo. A LSTM foi proposta pela primeira vez por (HOCHREITER; SCHMIDHUBER, 1997) e refinada por outros pesquisadores ao longo do tempo (DONAHUE *et al.*, 2015). Uma LSTM é mostrada na Figura 12. A linha na parte superior do diagrama é o estado da célula c que representa a memória interna da unidade. A linha em baixo é o estado oculto, e os *gates* i , f , g e o são o mecanismo pela qual a LSTM funciona (ANDRYCHOWICZ *et al.*, 2016). Durante o treinamento, a LSTM aprende os parâmetros para esses *gates*. A fim de obter uma compreensão

mais profunda de como esses *gates* modulam o estado oculto da LSTM, seja a entrada \mathbf{x}_t , saída \mathbf{y}_t e a camada escondida \mathbf{h}_t e t um instante no tempo, uma rede *RNN* pode ser definida pelas Equações 2.1 e 2.2:

Figura 12 – Arquitetura de uma LSTM



Fonte: Donahue *et al.* (2015).

$$\mathbf{h}_t = f_h(\mathbf{x}_t, \mathbf{h}_{t-1}) \quad (2.1)$$

$$\mathbf{y}_t = f_o(\mathbf{h}) \quad (2.2)$$

em que f_h e f_o são as equações da camada escondida e de saída respectivamente.

Por exemplo, dado um conjunto de treinamento \mathcal{G} de tamanho N e considerando um par (\mathbf{x}, \mathbf{y}) como entrada e saída, $\mathcal{G} = \{(\mathbf{x}_1^n, \mathbf{y}_1^n), (\mathbf{x}_2^n, \mathbf{y}_2^n), \dots, (\mathbf{x}_{T_n}^n, \mathbf{y}_{T_n}^n)\}_{n=1}^N$ (PASCANU, 2014), uma *RNN* convencional é definida formalmente por uma função de transição e uma função saída.

$$\mathbf{h}_t = f_h(\mathbf{x}_t, \mathbf{h}_{t-1}) = \sigma_h(\mathbf{W}^x \mathbf{x}_t + \mathbf{W}^h \mathbf{h}_{t-1} + \mathbf{b}^h) \quad (2.3)$$

$$\mathbf{y}_t = f_o(\mathbf{h}) = \sigma_o(\mathbf{W}^o \mathbf{h} + \mathbf{b}^y) \quad (2.4)$$

em que \mathbf{W}^h , \mathbf{W}^x e \mathbf{W}^o são as matrizes de peso, \mathbf{b}^h e \mathbf{b}^y são os termos de *bias*. A função de transição σ_h geralmente é a função tangente hiperbólica ou uma função sigmoide (PASCANU, 2014) e σ_o é função da camada de saída.

RNN bidirecionais são simplesmente duas *RNNs* independentes e juntas com ordens de tempo futuro e passado. A sequência de entrada é feita em ordem de tempo para uma rede e em ordem inversa de tempo para outra rede. As saídas das duas redes são normalmente concatenadas

a cada passo de tempo, mas existem outras opções, por exemplo, somar as saídas. Essa estrutura permite que as redes tenham informações para trás (*backward*) e para frente (*forward*) sobre a sequência a cada etapa de tempo. Uma LSTM bidirecional (ou biLSTM) é uma rede RNN bidirecional, com o modelo de rede LSTM.

Em tarefas envolvendo PLN e redes neurais, é necessário mapear as palavras de um texto em um vetor numérico. Este vetor numérico é chamado de *word embeddings* ou *word vectors* (MIKOLOV, 2012b; COLLOBERT *et al.*, 2011). Existem duas famílias de métodos para se obter esses vetores numéricos. A primeira família são os métodos baseados em contagem de palavras dos corpú, por exemplo, Análise de Semântica Latente (LSA). A segunda família são os métodos preditivos, que utilizam um tamanho de janela para prever a palavra em torno desse contexto local, como o *Skip-gram* (MIKOLOV; YIH; ZWEIG, 2013).

Para língua portuguesa, Hartmann *et al.* (2017) criaram 31 modelos⁹ de *word embeddings* para tarefas de PLN envolvendo métodos preditivos e métodos baseados em contagem de palavras. Para isso, eles utilizaram os métodos GloVe, Word2Vec, Wang2Vec e FastText.

O método GloVe (*Global Vectors*) foi proposto por Pennington, Socher e Manning (2014), sendo usado para tarefas envolvendo analogias sintáticas e semânticas. Esse método consiste em uma matriz M de co-ocorrência que é construída observando as palavras do contexto. Para cada elemento, M_{ij} representa a probabilidade da palavra i estar perto da palavra j . Na matriz M , as linhas (ou vetores) são geradas aleatoriamente e treinadas obedecendo a equação $P(w_i, w_j) = \log(M_{ij}) = w_i w_j + b_i + b_j$ onde w_i são palavras do vetor e b_i, b_j são os vieses ou limiares.

O método Word2Vec é um método amplamente utilizado em PLN para gerar *word embeddings*. Esse método possui duas diferentes estratégias de predição: i) *Continuous Bag-of-Words* (CBOW); e ii) *Skip-Gram*. A arquitetura CBOW recebe uma sequência de palavras sem a palavra do meio e tenta prever ela. A arquitetura *Skip-Gram* recebe uma palavra e tenta prever suas palavras vizinhas. Nos dois casos, o Word2Vec usa uma única matriz de peso para predição (MIKOLOV *et al.*, 2013a).

O método Wang2Vec é uma modificação do Word2Vec desenvolvido para resolver o problema da falta de ordem das palavras na arquitetura do Word2Vec (LING *et al.*, 2015). Wang2Vec traz duas modificações, uma na arquitetura *Skip-gram*, em que traz um conjunto diferente de parâmetros para prever cada palavra de contexto, considerando a posição como dependente em relação a palavra alvo. A segunda modificação é na arquitetura em que o *input* passa ser a concatenação das *embeddings* de palavras na ordem em que ocorrem.

O método FastText foi proposto por Bojanowski *et al.* (2017) e seu objetivo foi associar as *embeddings* com n -gramas caracteres. Nesse método, uma representação de palavras é induzida pela soma de vetores n -gramas com os vetores de palavras ao redor. Com isso, Bojanowski *et al.*

⁹ Disponível em <<http://nilc.icmc.usp.br/nilc/index.php/repositorio-de-word-embeddings-do-nilc>>

(2017) tenta capturar informações morfológicas para induzir as *word embeddings*.

2.3.3 Multi-task Learning

De acordo com Collobert e Weston (2008), as tarefas de PLN como Part-of-Speech Tagging (POS), Chunking, Reconhecimento de Entidade (*Named Entity Recognition* (NER)), Anotação de Papéis Semânticos (*Semantic Role Labeling* (SLR)) têm sido abordadas com os métodos tradicionais de engenharia de features. Essa abordagem tradicional é composta por um método de extração de um conjunto de características (*features*) dos textos para serem usadas em algoritmos de aprendizado de máquina tradicional (ou clássico), como Máquina de Vetores de Suporte (*Support Vector Machine* (SVM)). Collobert e Weston (2008) descrevem que a escolha das *features* é um processo empírico, baseado principalmente na tentativa e erro. O uso de métodos de seleção de *features* é algo dependente para cada tarefa de PLN e necessário refazê-la, com uma pesquisa adicional, a cada nova tarefa.

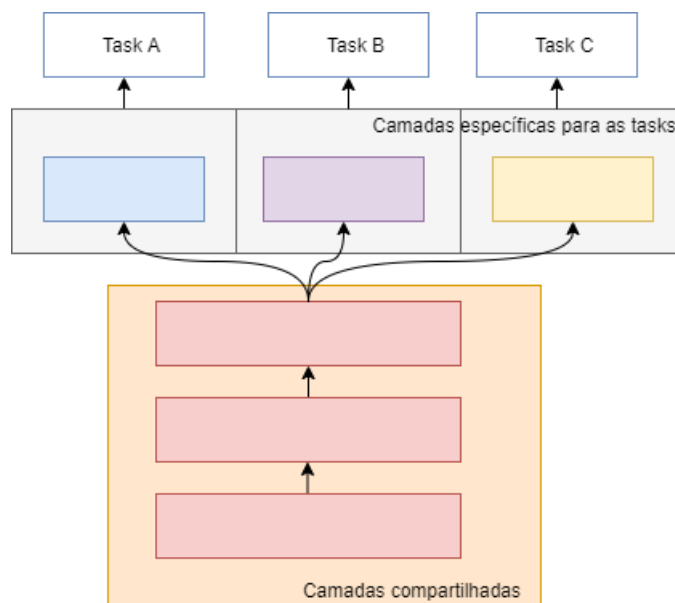
De acordo com Collobert e Weston (2008), as arquiteturas de rede neural profunda (*Deep Learning*) trazem a necessidade de trabalhar na engenharia da arquitetura da rede neural e não mais na engenharia de *features*, pois a arquitetura da rede neural profunda tem como entrada as palavras brutas (*raw words*) e não mais os valores das *features*. Então, a primeira camada da rede deve mapear as palavras em um vetor de números reais para serem processados pelas próximas camadas da rede. Nesse caso, estamos focados em uma única tarefa, ignorando informações que possam ajudar a rede neural aprender com outras informações sobre tarefas relacionadas que poderiam ajudar no aprendizado da rede neural. Quando compartilha-se representações entre tarefas relacionadas, é possível criar uma arquitetura que pode generalizar melhor a tarefa principal. Essa arquitetura é chamada de Multi-Task Learning (MTL).

MTL é uma abordagem de aprendizagem de máquina usada para aprender várias tarefas ao mesmo tempo e para prever o resultado de cada tarefa, usando redes neurais. Seu objetivo é beneficiar as várias tarefas (*tasks*) envolvidas (CARUANA, 1997; COLLOBERT; WESTON, 2008) com o compartilhamento de conhecimento.

Uma das estratégias para implementar MTL é usar o treinamento profundo em conjunto (*deep joint training*) que é, basicamente, a aprendizagem nas camadas mais profundas da rede. Considere que as tarefas a serem aprendidas sejam relacionadas, então os conhecimentos mais úteis para uma tarefa poderão ser úteis para outras tarefas, reforça (CARUANA, 1997). De acordo com Li *et al.* (2020), o uso de MTL pode ser uma abordagem promissora em PLN quando os dados de treinamento são insuficientes para uma determinada tarefa.

No contexto de Deep Learning, o MTL normalmente é implementado com o compartilhamento *soft* ou *hard* de parâmetros de camadas ocultas (RUDER, 2017a). No Capítulo 6, nós trabalhamos com o compartilhamento *hard* de parâmetros. No compartilhamento *hard* de parâmetros (Figura 13) é feito o compartilhamento das camadas ocultas entre todas as tare-

Figura 13 – Compartilhamento *hard* de parâmetros para *multi-task learning* usando redes neurais profundas



Fonte: Adaptada de Ruder (2017a).

fas, sendo as camadas de saída específicas para cada *task*. O compartilhamento *hard* reduz o risco de *overfitting*. De acordo com Ruder (2017a), quanto mais tarefas estão sendo aprendidas simultaneamente, o modelo tem que trabalhar mais (isto é, generalizar) para encontrar uma representação que capture todas as tarefas, sendo menor a chance de ocorrer *overfitting* na tarefa original/principal.

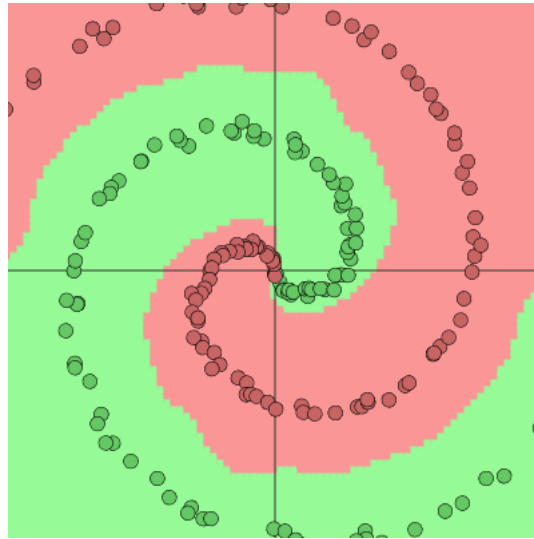
2.3.4 Softmax e Cross Entropy Loss

O conceito de Entropia Cruzada (em inglês, *Cross Entropy*) vem da teoria da informação. Em síntese, é uma função de perda (*loss*) entre duas distribuições de probabilidade que podem ser usadas dentro das redes neurais para mover os parâmetros (ou coeficientes) para os valores ótimos. O q representa a distribuição de probabilidade verdadeira (*true*). O p representa a probabilidade do modelo computado. O c representa a classe. Em nosso trabalho, usamos a entropia cruzada nas redes neurais profundas. Dessa forma a rede pode estimar qual é a probabilidade de uma classe de acordo com o modelo verdadeiro (MANNING, 2019). A entropia cruzada é representada pela fórmula 2.5:

$$H(p, q) = - \sum_{c=1}^C p(c) \log q(c) \quad (2.5)$$

Treinando com *softmax* e *cross-entropy loss* temos a equação 2.6. Em que para cada exemplo de treinamento (x, y) o objetivo é maximizar a probabilidade da classe correta y . Na equação 2.6, o objetivo é minimizar a probabilidade do log negativo para cada classe (MANNING,

Figura 14 – Função não linear usando ConvNetJs



Fonte: Elaborada pelo autor.

2019):

$$-\log p(y|x) = -\log \frac{\exp(f_y)}{\sum_{c=1}^C \exp(f_c)} \quad (2.6)$$

Para a função de entropia cruzada e *softmax* para um conjunto de dados completos $\{x_i, y_i\}_{i=1}^N$ temos a equação 2.7. Em que x_i são os *inputs*, isto é, palavras, sentenças, documentos, etc. Os y_i são os rótulos (*labels*) e N a quantidade de elementos do conjunto (MANNING, 2019).

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N -\log \left(\frac{\exp(f_{y_i})}{\sum_{c=1}^C \exp(f_c)} \right) \quad (2.7)$$

As redes neurais podem aprender funções complexas e tomar decisões não lineares. Criamos a Figura 14 usando ConvNetJs¹⁰. Nessa figura é ilustrado uma simulação de um problema binário com alguns pontos de dados de classe 0 (vermelho) e 1 (verde). A rede ConvNetJs foi configurada com 128 neurônios na primeira camada, 64 neurônios na segunda camada, e usamos a função de ativação *softmax* na camada de saída, o *learning rate* em 0.01 e *bath_size* em 10. O simulador envia todos os pontos para treinar a rede e as predições são pintadas no gráfico sob os pontos de dados para mostrar a generalização.

¹⁰ <<https://cs.stanford.edu/people/karpathy/convnetjs/demo/classify2d.html>>

REVISÃO DA LITERATURA

3.1 Considerações Iniciais

Neste capítulo, são abordados os trabalhos correlatos. Os trabalhos foram divididos em duas seções principais. A Seção 3.2 traz os trabalhos sobre Qualidade de REA, da área da Educação e Inteligência Artificial, com modelos de aprendizado de máquina para avaliar a qualidade dos REA. Na Seção 3.3, trouxemos os trabalhos relacionados com a avaliação da complexidade textual, com foco na grande área de Processamento de Línguas Naturais e Inteligência Artificial.

3.2 Qualidade de REA

Conforme descrito na Seção 1.3, este trabalho visa tratar o problema da avaliação da qualidade de REA. Para isso, são analisados nesta seção os trabalhos correlatos que estudaram esse problema. Os trabalhos correlatos foram organizados cronologicamente e para cada trabalho descrito há uma discussão sobre a relação do trabalho descrito com este trabalho de doutorado, além das críticas do trabalho sendo revisado quanto ao esperado para a eficácia da tarefa de avaliação da qualidade.

3.2.1 Trabalho de Bethard et al.(2009)

Bethard et al. (2009) trata da qualidade em um trabalho dividido em duas partes. A primeira é uma *meta-análise* e explora a forma como os usuários (professores, pesquisadores e especialistas) avaliam a qualidade dos recursos educacionais na Web; dentro desse grupo de usuários foram identificadas características comuns. A segunda parte explora de maneira computacional a anotação automática da qualidade dos recursos educacionais na Web. Para isso, os autores usam algoritmos de aprendizado de máquina supervisionado, usando a base de

Tabela 6 – Indicadores automatizados de qualidade

Indicador	Acurácia
1) Tem instruções	78%
2) Tem patrocinador	96%
3) Tem um grande patrocinador	81%
4) Identifica a faixa etária	87%
5) O conteúdo é apropriado para a faixa etária	99%
6) Identifica objetivos de aprendizagem;	81%
7) Organizado por objetivos de aprendizagem	83%

Fonte: Adaptada de Bethard *et al.* (2009).

treinamento obtida na primeira parte do trabalho. Para a criação da base foram consideradas sete características ou indicadores de qualidade, na forma de perguntas: 1) tem instruções?; 2) tem patrocinador?; 3) tem um grande patrocinador?; 4) identifica objetivos de aprendizagem?; 5) identifica a faixa etária?; 6) organizado por objetivos de aprendizagem?; e 7) o conteúdo é apropriado para a faixa etária?.

A Tabela 6 mostra o resultado da avaliação automatizada feita por Bethard *et al.* (2009) por meio desses indicadores. Como medida de avaliação foi utilizada a acurácia de cada classe, e o método Máquina de Vetores de Suporte (SVM), com parâmetros otimizados por meio do pacote SVMLight¹ e usando o *kernel* linear.

Discussão

O trabalho de Bethard *et al.* (2009) apresentou uma abordagem automática para indicar a qualidade dos recursos educacionais. Porém, seus sete indicadores de qualidades têm uma acurácia baixa quando essas características não são identificadas automaticamente. Embora seu conjunto de teste seja composto por 1.000 recursos educacionais, são considerados apenas os metadados, isto é, o trabalho não aborda o conteúdo dos REA como o trabalho desenvolvido nesta tese. Também, o trabalho não compara os diferentes métodos de aprendizagem de máquina, mostrando qual seria o mais adequado, apenas indica o SVM como o melhor método. Além disso, não mostra os resultados com outros kernels disponíveis para o SVM.

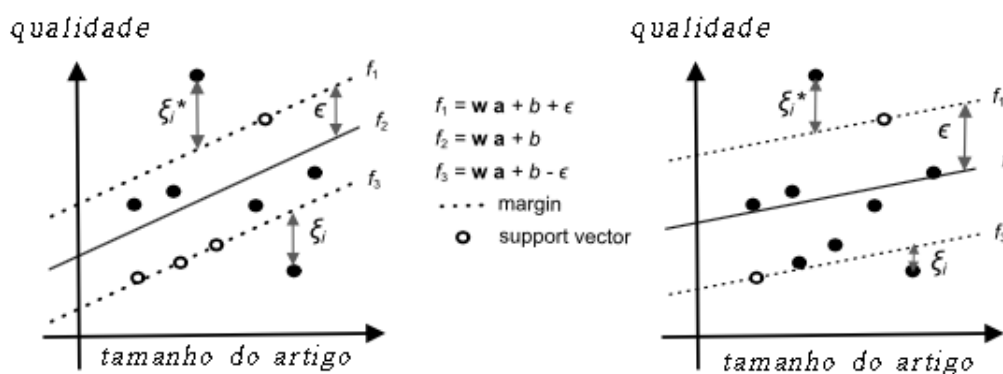
¹ <<http://svmlight.joachims.org/>>

3.2.2 Trabalho de Dalip et al.(2011)

Dalip et al. (2011) usaram uma escala contínua² para avaliação da qualidade e o repositório da Wikipédia de 2010 no idioma Inglês.

Além disso, Dalip et al. (2011) usaram um modelo de regressão numérica para prever os rótulos qualificados dos artigos da Wikipédia. Eles aplicaram Regressão de Vetores de Suporte (SVR) para aprendizagem e avaliação automática da qualidade nos artigos. O SVR é um método que se fundamenta no SVM, sendo utilizado para resolver problemas de regressão não linear (RODRIGUES et al., 2014). Para exemplificar o funcionamento do SVR, imagine um tubo em volta de uma função de aproximação que tenha no máximo um desvio de κ sobre todos os exemplos rotulados que se deseja que o método aprenda. Então, procura-se obter o tubo mais estreito possível da função de generalização em $f(x) = (x_1, y_1), \dots, (x_n, y_n)$ em que os dados de entrada x estejam no conjunto \mathfrak{R} e o objetivo é encontrar o máximo do desvio de κ em que os alvos são y . Considere uma função linear desta função de generalização como $f(x) = \langle \omega, x \rangle + b$ em que $\omega \in x, b \in \mathfrak{R}$ e neste caso deseja-se buscar uma reta mais paralela possível a esta função e que esteja contida no conjunto \mathfrak{R} (RODRIGUES et al., 2014).

Figura 15 – Classificação usando SVR



Fonte: Dalip et al. (2011).

A Figura 15 mostra o problema de regressão, proposto por Dalip et al. (2011), com 10 artigos representados pelos seus tamanhos versus qualidade. Como explicado no parágrafo anterior, ξ_{i1} e ξ_{i2} denotam as margens de erro e quaisquer pontos fora dessas margens são desconsiderados pelo algoritmo SVR.

² Para diferenciar escala contínua de escala discreta, considere os seguintes exemplos. A escala contínua é comumente empregada em diversos repositórios on-line como a média de todas as pontuações feitas pelos usuários a partir dos valores de 1 a 5, sendo que 5 representa o melhor material e 1 o pior material daquele repositório. Enquanto que, na escala discreta, considere as categorias em que se têm artigos em classes como Classe A - foram verificados por diversos especialistas, Classe B - artigos que foram avaliados por um especialista, mas precisam ser verificados por outros especialistas e Classe C que são artigos que foram avaliados por diversos usuários, mas aguardam avaliação de especialistas.

Discussão

O trabalho de Dalip *et al.* (2011) apresentou uma representação para qualificar os artigos baseada nos seguintes critérios: i) integridade, neutralidade, boa organização do documento, precisão, conectividade entre artigos e fornecimento de referências relevantes. No entanto, o trabalho aborda apenas o contexto da Wikipédia. Além disso, não considera a qualidade dos REA e nem mesmo os metadados, e as *features* linguísticas que são abordadas neste trabalho.

3.2.3 Trabalho de Leary *et al.* (2011)

Leary *et al.* (2011a) criaram um algoritmo chamado OPERA para avaliação de REA, sendo ele uma extensão do trabalho de Bethard *et al.* (2009), descrito na Seção 3.2.1. Neste novo projeto, Leary *et al.* (2011a) utilizam 200 projetos de REA e construíram 5 indicadores de qualidade, descritos na Tabela 7. O algoritmo OPERA fornece uma saída binária para cada indicador de qualidade. Essa saída pode ser 0 ou 1 em que 0 representa o valor mais próximo da resposta do indicador de qualidade ser “não” e 1 como sendo “sim”. Leary *et al.* (2011a) consideram isso como uma pseudo-probabilidade para as afirmações dos indicadores de qualidade.

Tabela 7 – Indicadores de qualidades fornecidos para classificação dos REA

Indicador de Qualidade	Definição
Tem instruções	Indica ao usuário como usar ou navegar no projeto
Links para patrocinador conhecido (Por exemplo: Google, Facebook, ...)	Links de patrocinadores ou apoiadores considerados altamente respeitados pela comunidade
Identifica objetivos da aprendizagem	Material identifica os objetivos da aprendizagem e articula os conhecimentos e habilidade que o aluno adquirirá
Organização do material	Organiza o conteúdo adequadamente para facilitar os objetivos da aprendizagem
Conteúdo adequado para faixa etária	Fornece a faixa etária e possibilita uma leitura e atividades que não são nem muito difíceis nem fáceis para a faixa de idade que forneceu

Fonte: Adaptada de Leary *et al.* (2011a).

Além disso, Leary *et al.* (2011a) fizeram o seguinte experimento para criar uma base de treinamento, utilizando três professores experientes da área de Inteligência Artificial (IA), pois todos os 200 REA eram da área de IA. Para cada projeto de IA, os professores tinham os 5 indicadores de qualidade citados na Tabela 7 e poderiam escolher entre 1 a 5 em que 1 era “discordo totalmente” e 5 “concordo totalmente”.

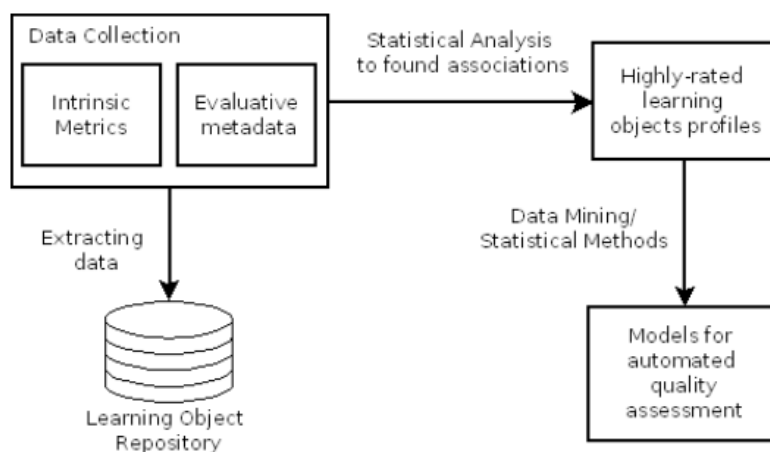
Discussão

O trabalho de [Leary et al. \(2011a\)](#) apresenta uma forma de avaliação automática de qualidade para REA, o que está diretamente relacionado com o presente projeto de doutorado. Porém, [Leary et al. \(2011a\)](#) continua usar uma SVM para classificar a qualidade. Além disso, trabalham com um domínio pré-definido que neste caso é a área de IA. Enquanto que a tese de doutorado abrange diversas áreas do conhecimento e compilou um *dataset* com um número maior de trechos de textos didáticos (2.067 extratos, min = 300 palavras, max = 596 palavras e média = 448), classificados em estágios escolares para treinamento, sendo que utilizou 140 REA para teste.

3.2.4 Trabalho de Cechinel et al.(2011)

[Cechinel, Sanchez-Alonso e Garcia-Barriocanal \(2011\)](#) descreve três modelos automáticos para avaliação da qualidade de Objetos de Aprendizagem (OA), utilizando metadados já preenchidos dentro dos repositórios. Sua abordagem é baseada no uso das avaliações existentes nos repositórios, feitas por diversos usuários, nas quais foram considerados perfis de qualidade *a priori*. Esse trabalho analisou 35 métricas em diversos OA dentro do repositório MERLOT. Foram elaborados, também, perfis para estes OA para diferentes categorias e tipos de materiais.

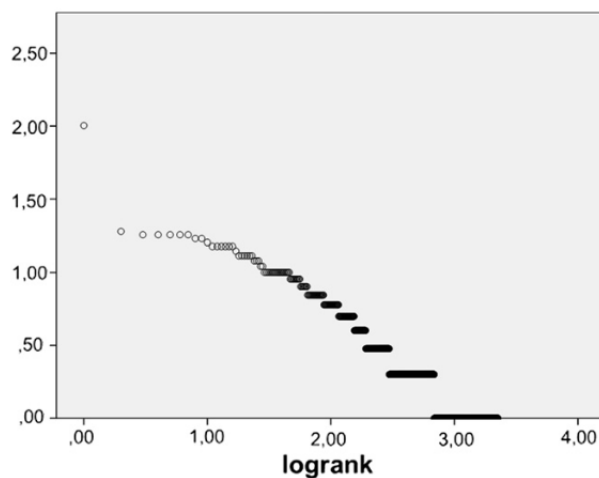
Figura 16 – Metodologia para geração de modelos de avaliação automática de qualidade para OAs



Fonte: [Cechinel, Sanchez-Alonso e Garcia-Barriocanal \(2011\)](#).

A Figura 16 mostra a arquitetura desenvolvida por [Cechinel, Sanchez-Alonso e Garcia-Barriocanal \(2011\)](#) para geração de modelos para avaliação da qualidade de OA. A metodologia desenvolvida pelos autores se baseia na criação de perfis de OA de alta pontuação para intersecção de categorias de disciplinas e tipos de materiais. Os autores descrevem que esses perfis criados foram utilizados posteriormente para criação de modelos de avaliação automatizada como mostra a Figura 16.

Figura 17 – Escala da quantidade de comentários por ranqueamento do REA



Fonte: [Cechinel, Sanchez-Alonso e Garcia-Barriocanal \(2011\)](#).

Foram analisados os comentários dos OA pela ótica da quantidade de comentários e *rating*; a Figura 17 mostra essa proporção na categoria Ciência e Tecnologia em escala logarítmica. Os três modelos para avaliação automática da qualidade de OAs são: Simulação \cap Ciência e Tecnologia, Simulação \cap Matemática e Estatística e Tutorial \cap Ciência e Tecnologia. Esses modelos criados por [Cechinel, Sanchez-Alonso e Garcia-Barriocanal \(2011\)](#) se baseiam na classificação da qualidade por meio de mineração de dados. Para isso, os autores utilizam os algoritmos do Weka ([HALL *et al.*, 2009](#)). Para classificação os autores utilizaram J48, SimpleCart, Part, Perceptron Multilayer e Rede Bayesiana, sendo que os resultados de cada classificador foram diferentes para cada modelo criado. Por exemplo, para Simulação \cap Ciência e Tecnologia os classificadores melhores foram SimpleCart e Perceptron Multilayer.

Discussão

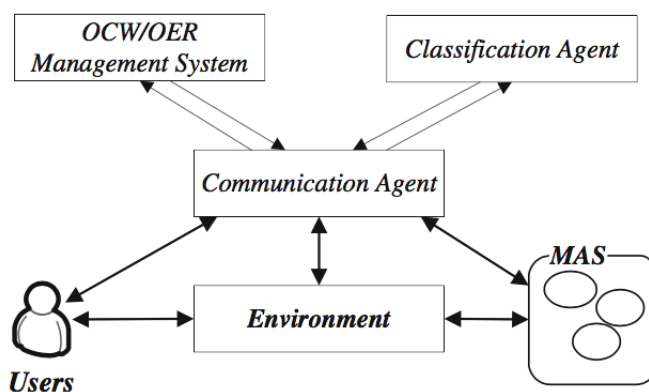
O trabalho de [Cechinel, Sanchez-Alonso e Garcia-Barriocanal \(2011\)](#) apresenta uma abordagem de automatização da avaliação da qualidade por meio da mineração de dados e uso de classificadores preexistentes no Weka. Além disso, os autores consideram os comentários nos repositórios o que define uma característica importante que também poderia ser utilizada para avaliar a qualidade automatizada de REA neste projeto de doutorado. Contudo, é um trabalho bastante diferente do presente doutorado, pois OA não é REA; REA possui uma licença aberta, sendo que OA não necessariamente precisa ter. Além disso, não utilizamos mineração de dados para classificação, em contrapartida, foram usadas redes neurais artificiais profundas não triviais.

3.2.5 Trabalho de Moise *et al.*(2011)

[Moise, Vladiu e Constantinescu \(2011\)](#) desenvolveram um sistema multi-agente para avaliação da qualidade para REA chamado MASECO composto por um repositório próprio. A

Figura 18 mostra a arquitetura do MASECO que possui dois tipos de agentes: um agente de comunicação e um agente de classificação. O agente de comunicação tem o papel de comunicação interna entre os componentes do sistema. O agente de classificação classifica os REA. Por fim, o componente MAS tem a função de habilitar a inclusão de outros sistemas na arquitetura.

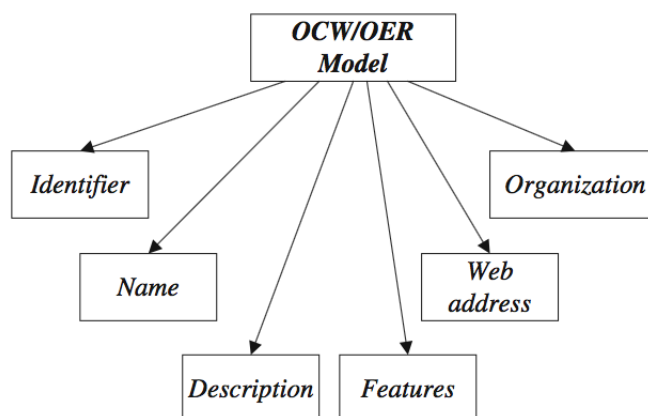
Figura 18 – Arquitetura do MASECO



Fonte: Moise, Vladioiu e Constantinescu (2011).

O agente de classificação é composto por uma SVM, uma rede neural *perceptron* (rede neural simples de apenas uma camada) e uma árvore de decisão. Moise, Vladioiu e Constantinescu (2011) relata que o REA passa por um processo de extração de características e depois é realizada a escolha do algoritmo de classificação.

Figura 19 – Modelo REA do repositório MASECO

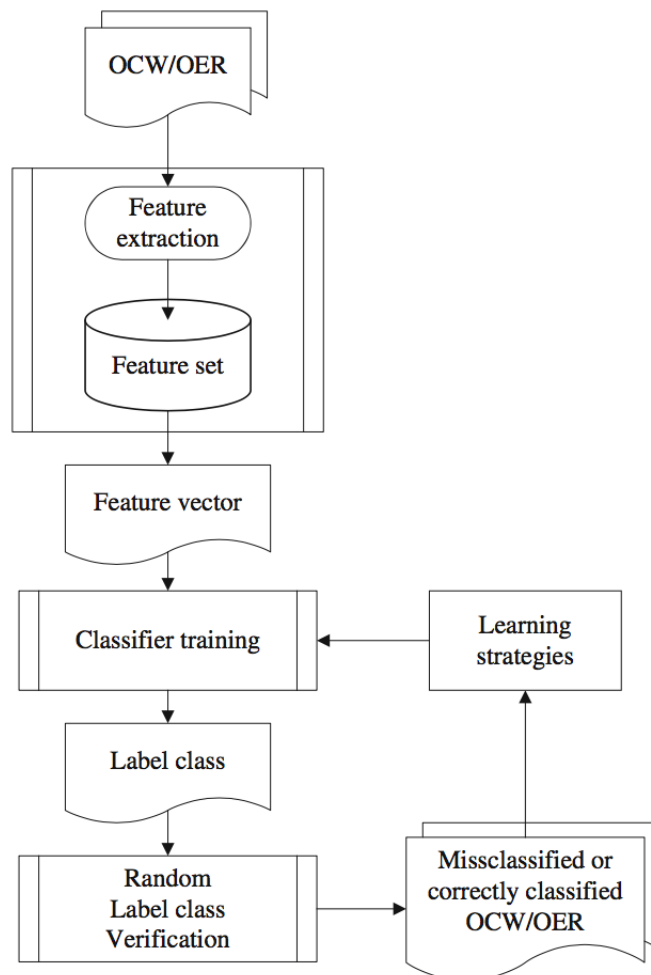


Fonte: Moise, Vladioiu e Constantinescu (2011).

Moise, Vladioiu e Constantinescu (2011) também cria um modelo próprio para inclusão de REA em seu repositório; a Figura 19 mostra esse modelo. Neste modelo, para cada REA incluído no repositório tem os seguintes metadados: identificador, nome, descrição, *features* extraídas, endereço URL de origem e instituição do criador. Com esses metadados, o MASECO

realiza a extração de características dos REA, cria um conjunto dessas *features* e treina os classificadores de qualidade (Figura 20).

Figura 20 – Funcionamento da classificação de REA usando MASECO



Fonte: Moise, Vladioiu e Constantinescu (2011).

Discussão

O trabalho de Moise, Vladioiu e Constantinescu (2011) apresenta uma arquitetura para armazenamento e extração de características de REA para obtenção da qualidade. Para isso, eles utilizam metadados para aprendizagem e multi-agentes para classificação. Esses multi-agentes são compostos por dois algoritmos tradicionais de classificação para automatização da qualidade de REA. Porém, não apresentam justificativa para a escolha desses agentes classificadores. Além disso, utilizam apenas métodos tradicionais. Em contrapartida, neste doutorado em tela, são abordados, além dos métodos tradicionais de classificação, as arquiteturas de redes neurais e a diversidade de contextos e formatos REA na Web. Também, foram criados métodos que consideram metadados e textos dos REA, e não apenas metadados como no trabalho de Moise, Vladioiu e Constantinescu (2011).

Tabela 8 – Dimensões para indicadores de qualidade

Dimensão	Indicadores
Reflete autoridade de origem	i) Tem patrocinador ii) Tem um grande patrocinador iii) Tem instruções
Suporte pedagógico robusto	i) Identifica os objetivos de aprendizagem ii) Organizado por objetivos de aprendizagem e identifica a faixa etária iii) Conteúdo é apropriado para faixa etária identificada

Fonte: Adaptada de [Wetzler et al. \(2013\)](#).

Tabela 9 – Indicadores automatizados de qualidade

Indicador de qualidade	Bethard et al. 2009 Acurácia	Acurácia
1) Tem instruções	78%	88.8%
2) Tem patrocinador	96%	98.8%
3) Tem um grande patrocinador	81%	80.1%
4) Identifica a faixa etária	87%	91.3%
5) O conteúdo é apropriado para a faixa etária	99%	100%
6) Identifica objetivos de aprendizagem	81%	90.1%
7) Organizado por objetivos de aprendizagem	83%	92.1%

Fonte: Adaptada de [Wetzler et al. \(2013\)](#), [Bethard et al. \(2009\)](#).

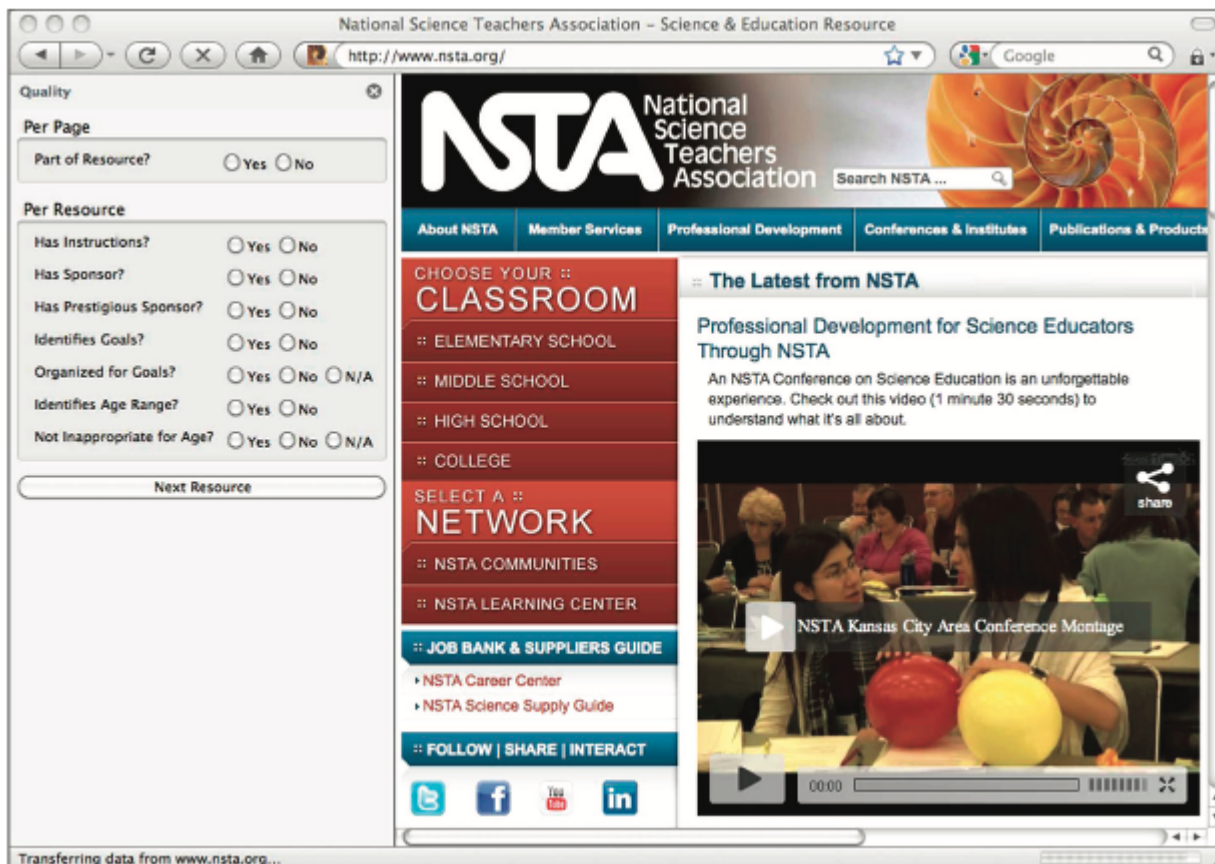
3.2.6 Trabalho de Wetzler et al.(2013)

[Wetzler et al. \(2013\)](#) é uma continuação do projeto publicado por [Bethard et al. \(2009\)](#) (Seção 3.2.1). A diferença é que nesta extensão os autores quebram os indicadores de qualidade em duas dimensões. As dimensões estendidas foram “Reflete autoridade de origem” e “Suporte pedagógico robusto” (Tabela 8) e são apresentadas na Tabela 6 para facilitar o entendimento da extensão do trabalho de [Wetzler et al. \(2013\)](#). Além disso, os autores criaram uma extensão para navegador com o objetivo de facilitar a qualificação de REA pelos usuários.

A Figura 21 mostra o uso da extensão criada pelos autores para avaliar os REA. Além disso, as 2 dimensões ficam ocultas na extensão para comparação com as mesmas características do trabalho anterior, mas sem considerar a quebra de dimensão. As dimensões são usadas apenas para o método SVM.

A precisão indicada pelos autores nesta extensão do trabalho de [Wetzler et al. \(2013\)](#) está indicada na Tabela 9, utilizando os mesmos algoritmos anteriores. É possível observar que apenas um indicador teve perda e todos os outros tiveram ganho em comparação ao trabalho de [Bethard et al. \(2009\)](#). O indicador “3) Tem um grande patrocinador” tinha anteriormente 70% de referência de desempenho e diminuiu para 59.6% e a referência usando aprendizagem de

Figura 21 – Extensão para navegador desenvolvido para anotação da qualidade



Fonte: [Wetzler et al. \(2013\)](#).

máquina era 81% e diminuiu para 80.1%. Ademais, todos os outros indicadores tiveram ganhos superiores quanto ao desempenho e na aprendizagem de máquina. Porém, o trabalho foi testado em apenas um repositório chamado DLESE. DLESE é uma biblioteca digital voltada para o tema Terra e é administrada por uma comunidade de especialistas da área para identificar e realizar a acurácia manual de todo REA incluído no repositório. Além disso, diversos REA são rejeitados pelo DLESE por possuírem baixa qualidade.

Discussão

O trabalho de [Wetzler et al. \(2013\)](#) apresenta uma extensão do trabalho desenvolvido por [Bethard et al. \(2009\)](#). Porém, utiliza os mesmos algoritmos de avaliação de qualidade de REA. Portanto, o trabalho não compara os diferentes métodos de aprendizagem de máquina. Também, não mostra os resultados com outros kernels disponíveis para o SVM utilizado.

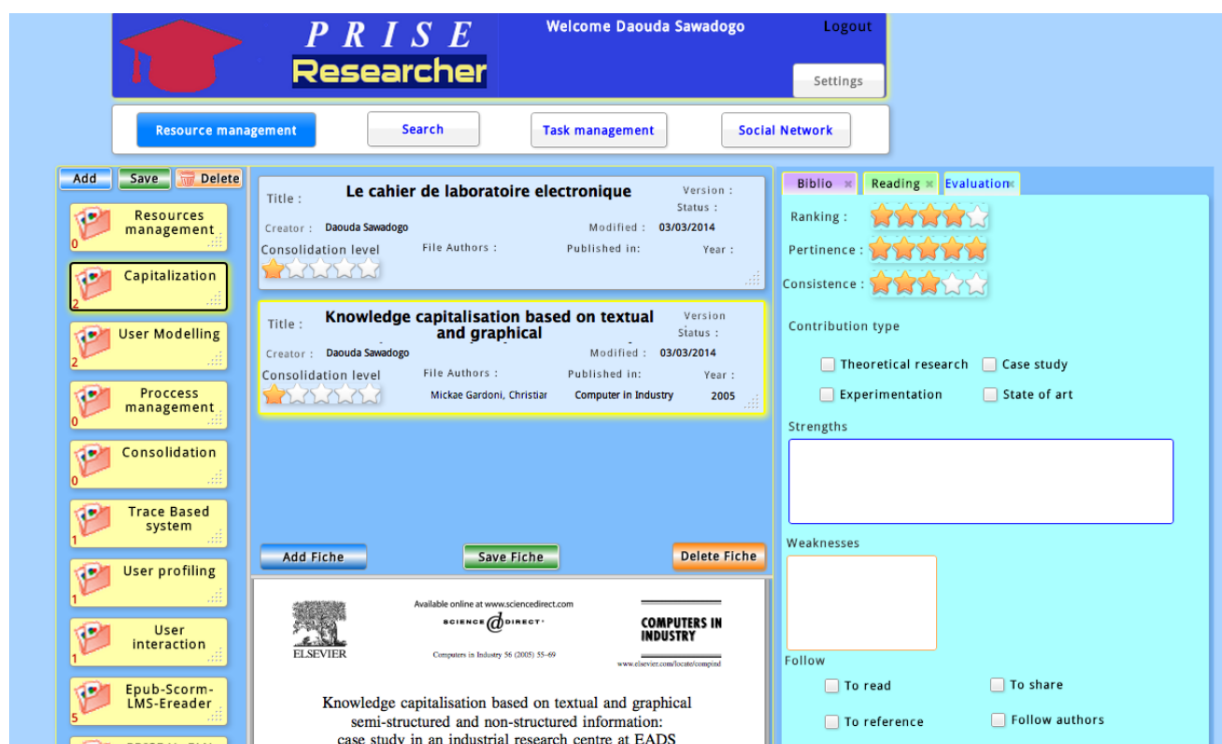
3.2.7 Trabalho de [Sawadogo et al. \(2014\)](#)

[Sawadogo, Champagnat e Estrailier \(2014\)](#) criaram um sistema adaptativo de qualidade baseado em metadados e tratam a gestão de rastreamento do usuário. Eles identificam dois níveis

de uso para perfis de usuários. Nível estático, diretamente relacionado ao uso da informação, e nível dinâmico, relacionado às informações interativas do sistema (SAWADOGO; CHAMPAGNAT; ESTRAILLIER, 2014) como, por exemplo, cálculos e informações dinâmicas alteráveis pelo contexto.

A diferença entre o uso de metadados de Sawadogo, Champagnat e Estrailier (2014) com outros autores, é que ele usa um padrão específico (RAZMERITA; ANGEHRN; MAEDCHE, 2003) conhecido como IMS-LIP.

Figura 22 – Avaliação da qualidade por meio de perfis de usuários



Fonte: Sawadogo, Champagnat e Estrailier (2014).

Além disso, usam TF-IDF para calcular a relevância dos REA com os níveis de uso. A Figura 22 mostra o trabalho desenvolvido por Sawadogo, Champagnat e Estrailier (2014) chamado de PRISE.

Discussão

O trabalho de Sawadogo, Champagnat e Estrailier (2014) apresenta uma forma de avaliar qualidade de recursos educacionais através de perfis. Porém, não é desenvolvido especificamente para REA. Também, não desenvolvem nenhuma maneira automatizada para avaliação da qualidade de REA, e a parte textual fica restrita apenas a uma técnica simples associada ao perfil do usuário.

3.2.8 Trabalho de Pons et al. (2015)

Pons et al. (2015) criaram um método para avaliar a qualidade de OA para o repositório MERLOT (CAFOLLA, 2006). Para isso, usaram os metadados dos OA armazenados no MERLOT. Definiram diversas métricas baseadas em cinco dimensões descritas na Tabela 10. Para a quantificação da qualidade dos OA, criaram uma fórmula 3.2 matemática baseada em conjunto de métricas.

$$I = \sum_{i=1}^n B_i Q_i \quad (3.1)$$

$$I = \sum_{i=1}^n \left(B_i \sum_{j=1}^m \left(b_j \frac{\log_{10}(q_j)}{\max(\log_{10}(q_j))} \right) \right) \quad (3.2)$$

Na fórmula 3.2 apresentada, Q representa as métricas definidas sobre as dimensões de qualidade da Tabela 10. O parâmetro b representa a porcentagem de peso de cada métrica de qualidade definida por Pons et al. (2015). O parâmetro q corresponde aos metadados utilizados para compor a métrica qualitativa. Um ajuste logaritmo é feito para diminuir a dispersão de valores mais elevados, dividindo o máximo dos valores observados, variando de 0 a 1.

Os autores, Pons et al. (2015), também relatam que seu estudo para estimar a qualidade de OA é voltado para o repositório MERLOT e especificamente para área de engenharia.

Tabela 10 – Dimensões de qualidade usadas

Métricas	Informações disponíveis no MERLOT
Aptidão pedagógica	Eficácia, número de exercícios, número de coleções, número de categorias
Usabilidade	Fácil de usar
Portabilidade	Exigência técnica
Distribuição	Custo envolvido, disponibilidade de código fonte e direitos autorais
Acessibilidade	Acessibilidade disponível

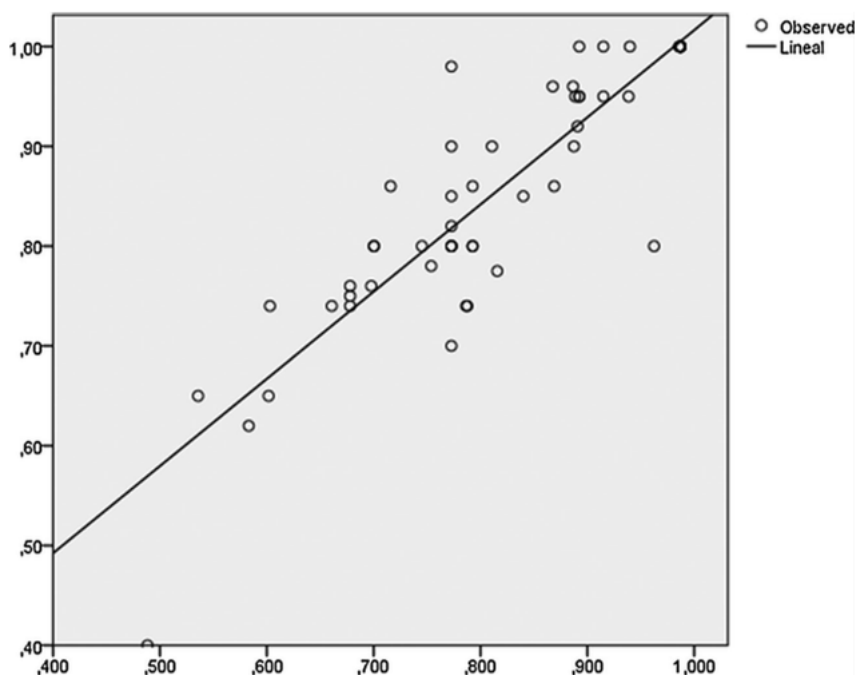
Fonte: Adaptada de Pons et al. (2015).

Após o uso da fórmula para estimar a qualidade, os autores usaram métodos estatísticos, especificamente regressão linear múltipla, para avaliação da qualidade dos OA. A Figura 23 mostra os resultados obtidos pelos autores, usando a regressão linear múltipla para o repositório MERLOT na avaliar a qualidade dos OA.

Discussão

O trabalho de Pons et al. (2015) apresenta um método para avaliação da qualidade de OA. Como já mencionado anteriormente, OA é diferente de REA devido a sua licença. Além disso,

Figura 23 – Regressão linear usada para avaliação da qualidade de OA



Fonte: Pons *et al.* (2015).

neste projeto de doutorado utilizamos redes neurais artificiais profundas não triviais, seguindo a abordagem de *Deep Learning*.

Também consideramos diversas áreas de REA e não apenas Engenharia, além da realização de experimentos comparativos com métodos tradicionais.

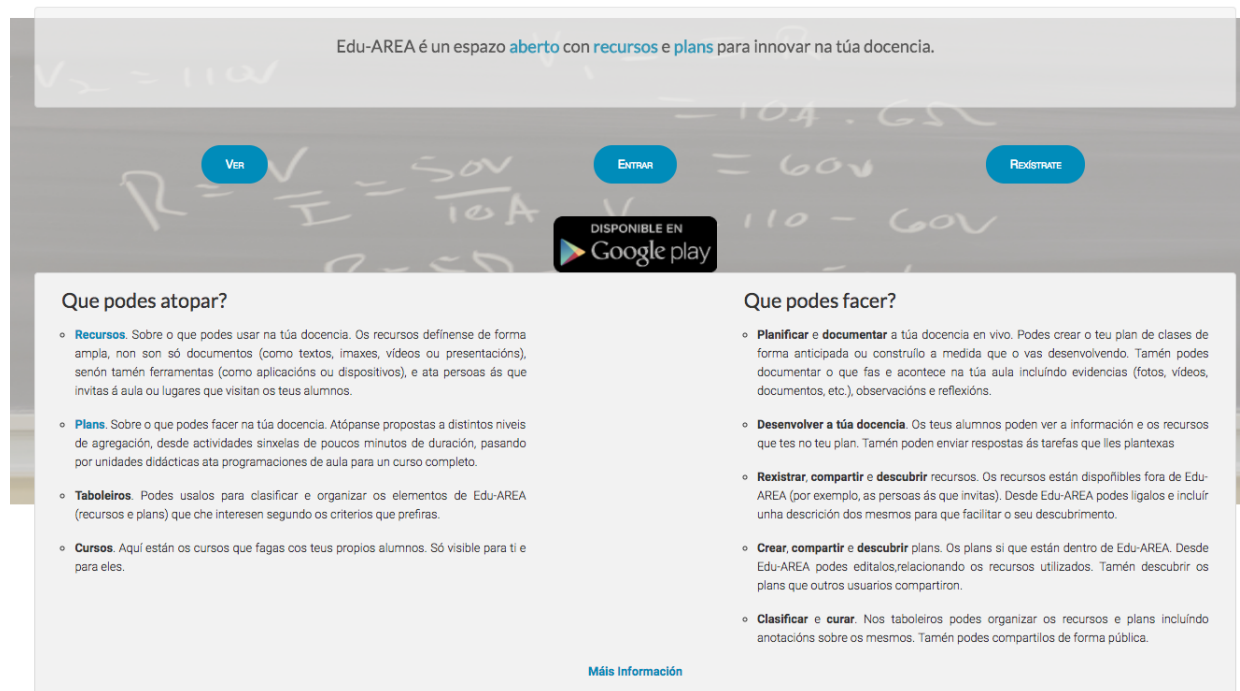
3.2.9 Trabalho de Rodríguez *et al.* (2016)

Caeiro-Rodríguez *et al.* (2016) são membros seniores da *IEEE* e curadores de um repositório de REA com nome Edu-AREA. O Edu-REA é dedicado ao compartilhamento e gerenciamento de REA na Web (Figura 24) e utiliza-se de recursos da Web 2.0 (Figura 25). Os autores desenvolveram uma solução para cálculo automático de indicadores de relevância e reputação para REA baseada no algoritmo de recuperação de informação (HITS).

Os autores, Caeiro-Rodríguez *et al.* (2016), criaram uma adaptação do algoritmo HITS. O algoritmo HITS classifica as páginas em autoridades e também criam ponteiros para elas chamados de *hubs*. Porém, são dependentes de uma consulta. Com isso, as páginas de autoridade são as mais relevantes para uma determinada consulta. No caso dos autores, eles criaram uma adaptação do HITS para calcular a relevância dos REA a partir das palavras-chave associadas aos REA ao invés da consulta.

A partir do trabalho realizado para classificação desses REA, eles analisaram os dados obtidos com o algoritmo HITS. Porém, não mostram esses dados em seu trabalho. Além disso, os

Figura 24 – Interface do Edu-AREA para avaliação de REA



Fonte: Adaptada de [Caeiro-Rodríguez et al. \(2016\)](#).

autores comentam que realizaram uma análise manual dos resultados e disseram que foi avaliado com êxito, mas não realizam nem um experimento controlado para provar a eficácia do uso do algoritmo adaptado por eles.

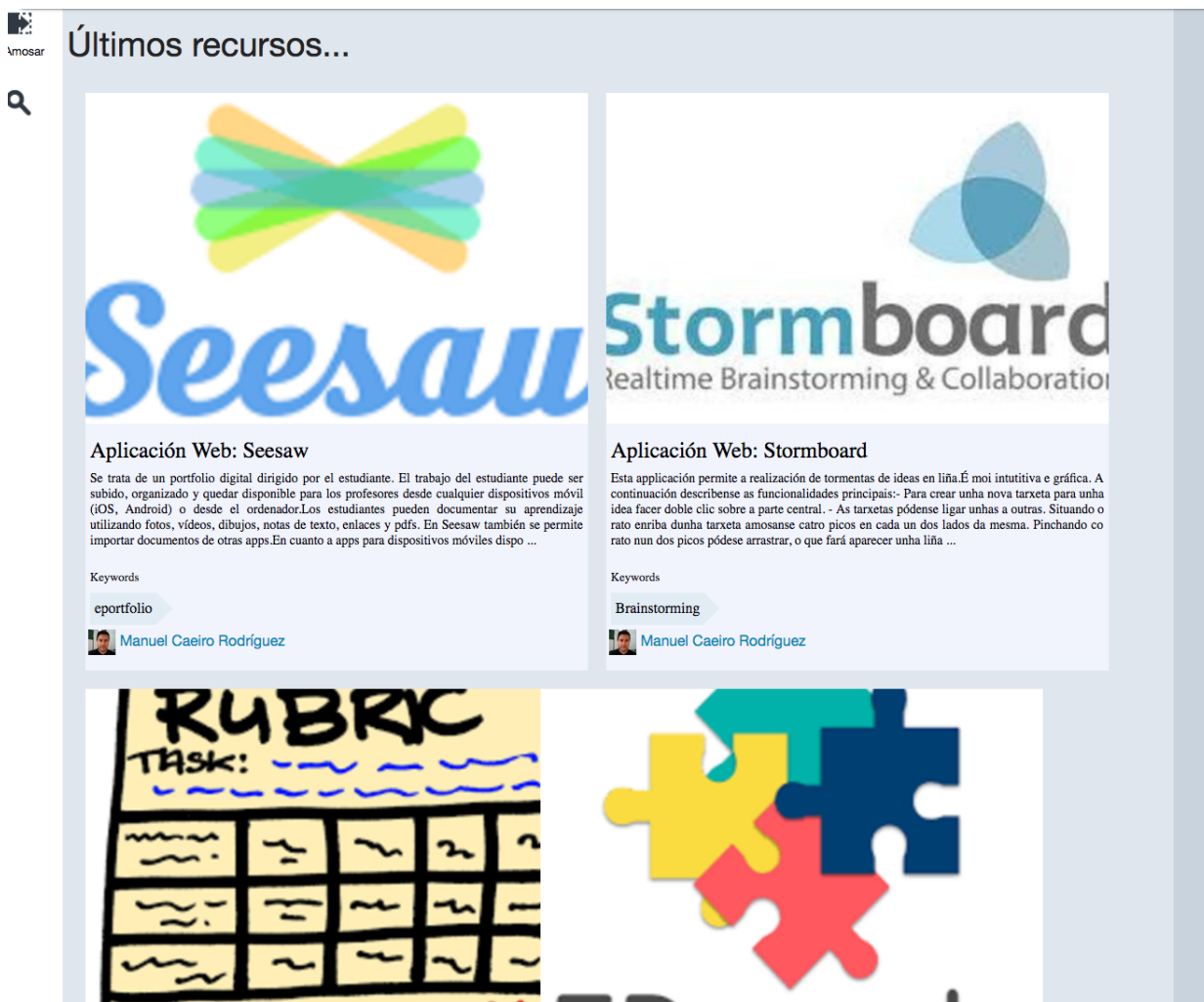
Discussão

O trabalho de [Caeiro-Rodríguez et al. \(2016\)](#) apresenta um algoritmo adaptado para relevância de páginas usado para classificação de REA por meio de palavras-chave associadas aos metadados dos REA em um repositório próprio. É uma abordagem nova e diferente de todos os autores anteriores. Porém, não mostram a eficácia comparativa entre diversos algoritmos de classificação existentes e mencionados nos outros trabalhos já descritos neste projeto. Além disso, não descrevem a possibilidade dos metadados não terem sido preenchidos (palavras-chave em branco) e, neste caso, como o REA seria classificado. Portanto, a abordagem adotada neste doutorado vai além de se restringir às palavras-chaves como meio para avaliação automática de REA.

3.2.10 Trabalho de Ahmed e Fuge (2017)

O trabalho de [Ahmed e Fuge \(2017\)](#) faz avaliação da qualidade dentro do repositório OpenIDEO. Porém, esse repositório não é um repositório para REA e sim para desafios on-

Figura 25 – REA armazenados no repositório já avaliados pelo algoritmo HITS adaptado



Fonte: Adaptada de Caeiro-Rodríguez *et al.* (2016).

line escritos por diversos usuários. Esses desafios possuem a descrição do problema e seus estágios para serem resolvidos. Os autores, em seu trabalho, utilizaram 3918 desafios para classificá-los quanto à qualidade. Para isso, utilizaram algoritmos tradicionais de classificação como árvore de decisão, regressão logística, SVM e vizinhos mais próximos (KNN). Porém, nenhum dos classificadores usados conseguiu obter resultados consistentes. Com isso, utilizaram um algoritmo chamado RUSBoost. Mas, os autores não citam as comparações realizadas em seu trabalho, apenas mostram o resultado obtido pelo algoritmo RUSBoost, desenvolvido em MatLab. O RUSBoost é um algoritmo criado em 2010 para lidar com desequilíbrio de classes com rótulos discretos — é considerado por alguns autores como um algoritmo híbrido de variações entre algoritmos que tratam de subamostras aleatórias de classificação.

Ahmed e Fuge (2017) usaram um conjunto menor do que os desafios capturados, com 14 desafios, para validação dos resultados. Para isso, o conjunto de dados foi dividido em 11 desafios para treinamento e 3 desafios para teste e realizaram 5 vezes a validação cruzada.

Os critérios usados para classificação das ideias subordinadas aos desafios foram obtidos em 3 etapas e depois o uso do RUSBoost. Calcularam o conjunto de características que representava os diferentes aspectos da ideia, tais como: singularidade, capacidade de leitura, coerência e semântica. A segunda etapa consistiu da análise do desafio em si e não apenas da ideia. Nessa segunda etapa, eles analisavam se os autores do desafio já adiantaram algumas etapas do desafio como Avaliação e Vencedor do Desafio. Os autores descrevem também que a terceira etapa foi uma classificação realizada por meio do XGBoost ou *Gradient Boosting* usando os vetores de característica da segunda etapa. O XGBoost é um algoritmo otimizado para árvore de decisão, usando gradiente descendente. Após isso, [Ahmed e Fuge \(2017\)](#) usam o RUSBoost para encontrar ideias vencedoras — aquelas que possuem altíssima qualidade nos temas desenvolvidos de forma discreta.

Discussão

O trabalho de [Ahmed e Fuge \(2017\)](#) é um trabalho recente, de 2017. O trabalho enfatiza que gerentes de repositórios on-line, especificamente de desafios, possuam maior flexibilidade e eficiência na avaliação automática da qualidade em grandes volumes de dados. Esse grande volume de dados se torna inviável para ser analisado manualmente, pois, são escritos pelos usuários de forma discreta e voltadas para resolver os problemas obtidos nos desafios. Porém, o trabalho não considera o uso do tratamento da qualidade de forma híbrida, ou seja, a avaliação automática de textos e metadados, e tampouco tratam sobre REA.

A Tabela 11 resume os trabalhos revisados sobre avaliação da qualidade de REA.

Tabela 11 – Análise crítica dos trabalhos correlatos

	Descrição	Possui indicadores de qualidade	Repositório de REA	Material não REA	Comparação com outros métodos de aprendizagem	Algoritmos comparados
Bethard et al. (2009)	Avaliação dos REA de repositório geográfico	Sim	Sim	-	Não	-
Dalip et al. (2011)	Uso de SVR para avaliar qualidade	Sim	Não	Wikipédia	Sim	Rassbach e Dondio
Leary et al. (2011)	Aumento da quantidade de REA	Sim	Sim	-	Não	-
Cechinel et al. (2011)	Criação de perfis estatísticos e suas intersecções para avaliação dos REA	Sim	Não	OA	Sim	Algoritmos de Clusterização, Uma Rede Neural Perceptron e Rede Bayesiana
Moise et al. (2011)	MASECO - Criação de mecanismo de extração de metadados e repositório próprio para REA	Sim	Sim	-	Sim	SVM e Uma Rede Neural Perceptron
Wetzler et al. (2011)	Dimensões de qualidade	Sim	Sim	-	Não	-
Sawadogo et al. (2015)	TF-IDF para calcular a relevância dos REA por meio do uso	Não	Não	OA	Não	-
Pons et al. (2015)	Métrica própria e usando indicadores criados	Sim	Não	OA	Não	-
Rodríguez et al. (2016)	Edu-REA e usaram as palavras chaves para calcular relevância	Sim	Sim	-	Não	-
Ahmed e Fuge (2017)	Qualidade de desafios educacionais	Sim	Não	Desafios educacionais	Sim	SVM, Regressão Logística e KNN

Fonte: Elaborada pelo autor.

Tabela 12 – Comparação entre os trabalhos relacionados e a proposta desenvolvida no doutorado

Proposta	Avaliação da qualidade em textos	Avaliação da qualidade no idioma Português	Avaliação da qualidade usando <i>embeddings</i>	Avaliação da qualidade dos metadados	Método de classificação	Considera diversos repositórios de REA	Utiliza redes neurais	Utiliza redes neurais profundas
Bethard et al. (2009)	✓				SVM			
Dalip et al. (2011)	✓				SVR			
Leary et al. (2011)	✓				SVM			
Cechinel et al. (2011)	✓				Perceptron Multilayer		✓	
Moise et al. (2011)	✓				Perceptron e SVM		✓	
Wetzler et al. (2011)	✓				SVM			
Pons et al. (2015)	✓				Método próprio e Regressão Linear			
Rodrigues et al. (2016)	✓				Algoritmo HITs personalizado SVM, KNN e			
Ahmed e Fuge (2017)	✓				regressão logística			
Nosso trabalho	✓	✓	✓	✓	Diversos	✓	✓	✓

Fonte: Elaborada pelo autor.

3.3 Avaliação Automática da Complexidade Textual

A avaliação automática da complexidade textual (*readability*, em inglês) estuda o que faz alguns textos serem mais fáceis de serem lidos do que outros. Esse tema é pesquisado há longa data por várias áreas de pesquisa como a Educação, Linguística Cognitiva, Psicolinguística, Processamento de Línguas Naturais.

Do ponto de vista pragmático, fornecer a cada estudante textos que sejam adequados a sua capacidade leitora é um grande desafio para educadores de todas as disciplinas escolares, pois tomaria o tempo já escasso desse profissional. Mas, ao mesmo tempo, é um tema de grande importância, pois textos são a base do aprendizado e veiculam o conhecimento sobre todas as disciplinas. A literatura mais recente em avaliação automática da complexidade textual traz três grandes abordagens para essa tarefa: (i) uso de técnicas estatísticas como a Análise de Componentes Principais e a Análise Fatorial para reduzir um grande número de métricas linguísticas em poucas dimensões explicativas da complexidade textual; (ii) a abordagem de aprendizado de máquina, via construção de classificadores ou regressores, muitas vezes com o uso de métodos de seleção de features ou com avaliação da contribuição de conjuntos de features para a tarefa; e (iii) a abordagem de *Deep Learning*, que usa arquiteturas neurais com diversas características para evitar o uso de engenharia de *features*. Entretanto, até onde sabemos, não há trabalhos que aplicaram a abordagem *multi-task learning* com o propósito de construir um preditor de complexidade textual, embora a abordagem tenha já sido avaliada para a complexidade sentencial (GONZALEZ-GARDUÑO; SØGAARD, 2017; LEAL; RODRIGUES; ALUÍSIO, 2020). Nas próximas seções trazemos uma revisão dos trabalhos das três abordagens citadas acima.

3.3.1 Complexidade Textual via Análise de Componentes Principais

Graesser e McNamara (2011) e Graesser, McNamara e Kulikowich (2011) apresentam a ferramenta Coh-Metrix³, projetada para a para língua inglesa. Coh-Metrix analisa textos escritos e falados usando métricas dos vários níveis da língua e que estão alinhadas com um arcabouço teórico de análise multinível, envolvendo métricas que tratam de: (i) palavras, (ii) sintaxe, (iii) da base textual (que se refere às ideias apresentadas no texto), (iv) modelo situacional (conteúdo sendo tratado; textos narrativos, por exemplo, incluem personagens, objetos, ações, eventos, emoções dos personagens), e (v) gênero e estrutura retórica. Graesser, McNamara e Kulikowich (2011) usaram um grande cópulo da língua inglesa com 37.520 extratos fornecidos pelo *Touchstone Applied Science Associates* (TASA), classificados por um escore chamado de Degrees of Reading Power (DRP)⁴ em uma escola de 1-100 e alinhados com uma série escolar. O DRP indica a habilidade de um estudante de compreender e absorver o conhecimento contido em trechos de um texto. Tendo este grande cópulo classificado em séries, utilizaram 53

³ <<http://cohmetrix.com/>>

⁴ <http://www.questarai.com/assessments/district-literacy-assessments/degrees-of-reading-power/>

métricas da ferramenta Coh-Metrix para mostrar que um aparato de análise multi-nível, como o Coh-Metrix, é necessário para indicar as características de textos para diferentes séries e as características dos vários gêneros textuais que as crianças leem o que não seria possível com o uso de fórmulas simples de *readability* como Índice Flesch. Os autores utilizaram a Análise de Componentes Principais para reduzir o grande número de métricas em 8 dimensões funcionais, ou componentes principais, que foram nomeados de: (1) narratividade, (2) coesão referencial, (3) simplicidade sintática, (4) concretude de palavras, (5) coesão causal, (6) coesão verbal, (7) coesão lógica, e (8) coesão temporal. Esses 8 componentes principais conseguiram explicar 67.3% da variação entre os textos analisados.

Discussão

Nossa proposta utiliza 182 métricas linguísticas, disponibilizadas na plataforma NILC Metrix⁵ e seleciona 55 métricas para transcrições e 79 métricas para textos didáticos com objetivos pedagógicos (livros, experimentos práticos, jogos, etc). Também, tratamos a língua portuguesa usando abordagens de *Multi Task Learning* (MTL) para avaliação intrínseca e extrínseca de um preditor de complexidade textual, além da abordagem clássica de aprendizado de máquina.

3.3.2 Complexidade Textual com ajuda da Engenharia de features

Scarton e Aluísio (2010) classificaram de forma binária (simples versus complexos) os textos de seu estudo, usando o Coh-Metrix-Port versão 1.0⁶ com 40 métricas de complexidade textual. Usaram 4 corpúscos para treinamento e teste: textos jornalísticos do jornal Zero Hora (ZH) dos anos de 2006 e 2007, textos reescritos para crianças da seção “Para o seu filho ler” (PSFL) do ZH e textos do gênero científico do Ciência Hoje (CH) e Ciência Hoje das Crianças (CHC). O trabalho apresenta resultados da classificação binária, sendo o SVM do Weka o melhor classificador treinado, com 97% de precisão. Porém sabemos que classificadores binários que medem uma grande distância de idade (crianças versus adultos) são mais simples do que classificadores com mais classes.

Discussão

Neste trabalho, diferente de Scarton e Aluísio (2010), usamos a classificação multiclasse em tarefas similares para avaliarmos abordagens de aprendizado de máquina clássico e de MTL na predição da complexidade textual para 4 etapas escolares como a tarefa principal de classificação, treinando juntamente duas outras tarefas: classificação de 7 séries escolares (5^o, 6^o, 7^o, 8^o, 9^o e Ensino Médio) e classificação de textos do domínio da MEC-REC para, também

⁵ <https://simpligo.sidle.al/>

⁶ A Versão 3.0 do CohMetrix-Port — <http://fw.nilc.icmc.usp.br:23380/cohmetrixport> — disponibiliza 46 métricas

em 4 etapas escolares. Além disso, avaliamos os resultados de forma extrínseca com textos da MEC-RED.

Hartmann *et al.* (2016) reportam a classificação da complexidade de textos do gênero didático em português para cinco séries consecutivas do Ensino Fundamental (3º, 4º, 5º, 6º e 7º séries). O corpus compilado é formado por textos de diversas fontes (Livros Didáticos, NILC Corpus, Testes do SARESP, Ciência Hoje das Crianças, Folhinha da Folha de São Paulo, Seção *Para Seu Filho Ler* do jornal Zero Hora e Mundo Estranho) e possui 7.645 textos, sem indicação da quantidade mínima de palavras que esses textos possuem.

Discussão

Nossa proposta, além de usar 4 etapas do sistema escolar brasileiro, utiliza transcrições em uma abordagem de MTL, diferente da proposta Hartmann *et al.* (2016) que utilizou o classificador SVM para predição da complexidade textual.

Filho *et al.* (2016) reportam a predição automática do nível escolar da Wikilivros, considerando 3 níveis escolares (nível 1, 2 e 3). O corpus possui 77 textos e usaram 7 *features* para avaliação da complexidade textual, via Regressão Logística⁷.

Discussão

Nossa proposta se diferencia, trazendo também o treinamento de um classificador de complexidade para 7 anos escolares (5º, 6º, 7º, 8º, 9º e Ensino Médio) com um conjunto de 55 métricas selecionadas com o método Recursive Feature Elimination and Cross-Validated (RFECV), além do classificador de 4 etapas escolares, treinado com 79 métricas usadas no Gazzola, Leal e Aluísio (2019) em uma nova abordagem usando MTL.

3.3.3 Complexidade Textual baseada em modelos Deep Learning

Nadeem e Ostendorf (2018) reporta modelos de deep learning usando Redes Neurais Recorrentes (RNR) com *self attention* e RNR hierárquicas bidirecionais para classificação da complexidade textual de 6 níveis escolares do sistema de ensino americano (K-1, 2-3, 4-5, 6-8, 9-10 e 11-12). Usaram amostras de treinamento de dois corpus: um com livros didáticos e outro foi o corpus WeeBit, compilado por Vajjala e Meurers (2012), sendo composto de documentos da revista WeeklyReader e do site educacional BBC Bitesize. Nadeem e Ostendorf (2018) trabalham com amostras de texto pequenos, do tamanho de um parágrafo. Também, fizeram uma avaliação extrínseca usando o dataset de questões de avaliação científica chamada Ai2 Science Questions Mercury⁸ com o tamanho médio de cada pergunta sendo 49 palavras e textos abertos da iniciativa

⁷ Modelo SimpleLogistic da ferramenta Weka.

⁸ <http://data.allenai.org/ai2-science-questions-mercury/>

Common Core State Standards ⁹.

Discussão

Nossa proposta envolve uma arquitetura ainda não explorada na classificação da complexidade textual que usa MTL envolvendo duas tarefas similares, para classificar a complexidade textual das transcrições em séries escolares (7 séries escolares) e classificar a complexidade textual de amostras de textos educacionais de diversos gêneros textuais em 4 grandes etapas escolares. A última abordagem de MTL avaliada usa três tarefas, trazendo também textos do domínio da MEC-RED.

Além disso, abordamos a tarefa com uma arquitetura de deep learning, usando métodos de word embeddings presentes no NILC Embeddings dentre eles o Word2Vec, Wang2Vec, FastText e GloVe.

Também, contamos com a utilização do Elmo em Português. Cobrimos assim uma lacuna na literatura, que não havia avaliado nenhuma arquitetura de deep learning para a tarefa de complexidade textual.

Deutsch, Jasbi e Shieber (2020) reportam um trabalho de classificação da complexidade textual usando algoritmos de aprendizado de máquina tradicional e aprendizado máquina profundo para dois corpus: Newsela (LU, 2010), e WeeBit (VAJJALA; MEURERS, 2012). WeeBit possui documentos separados por idades de 7 a 16 anos (10 classes) com 625 documentos em seu corpus *downsampled*. Newsela é um corpus com 1.911 artigos de notícias que consiste na reescrita em até 4 níveis de simplificação para leitores em diferentes níveis escolares; o nível escolar varia de 2 a 12 níveis (10 classes). Deutsch, Jasbi e Shieber (2020) usaram SVM, uma rede do tipo Transformer, uma rede neural convolucional (CNN - Convolutional Neural network), Redes Neurais Hierárquicas com Atenção, e combinaram as arquiteturas para avaliar o impacto das features linguísticas. Por exemplo, SVM com Transformer e features linguísticas e sem as features linguísticas. Por fim, concluíram que aprendizado de máquina profundo com as features textuais não obteve o melhor desempenho com significância estatística.

Discussão

Nossa proposta de MTL envolve uma arquitetura ainda não avaliada na tarefa de complexidade textual, no melhor do nosso conhecimento (houve uma avaliação da tarefa de complexidade sentencial (LEAL; RODRIGUES; ALUÍSIO, 2020)), usando MTL. Além disso, testamos os modelos com abordagem *undersampling* e também *oversampling* e avaliamos os resultados da nossa arquitetura.

Por fim, no Capítulo 5, apresentamos nossa segunda proposta de uso de métodos de aprendizado de máquina que envolve uma arquitetura de deep learning, com uso dos métodos

⁹ <https://corestandards.org>

de word embeddings dentre eles Word2Vec, Wang2Vec, FastText e GloVe com 300 dimensões. Além disso, usamos o Elmo em nossa arquitetura, onde treinamos mais de 100 milhões de parâmetros para avaliar a complexidade textual nos níveis escolares do Sistema Educacional Brasileiro para o Português.

3.4 Considerações Finais

Neste capítulo foram apresentados os trabalhos mais relevantes da área de Avaliação da Qualidade de REA e da Avaliação Automática da Complexidade Textual. Também, fizemos uma comparação entre o que desenvolvemos neste trabalho e as características de cada trabalho correlato. Além disso, para cada trabalho revisado foi feita uma discussão crítica na tentativa de contextualizar as técnicas desenvolvidas. Após a leitura de cada um dos trabalhos e reflexão sobre o que foi feito, percebemos que nos trabalhos que avaliam modelos de aprendizado de máquina tradicional para a tarefa de complexidade textual os autores trabalham com a engenharia de *features* linguísticas e seleção de *features* para explicar o impacto das *features* em seus modelos. Enquanto que os trabalhos mais recentes, que trazem uma abordagem de *deep learning*, fazem uma comparação com os métodos tradicionais para a tarefa. Sendo que, muito deles demonstram diversos experimentos envolvendo arquiteturas de *deep learning* (engenharia da arquitetura) e comparam seus resultados com modelos tradicionais e com suas próprias variações de arquiteturas de *deep learning*.

CLASSIFICAÇÃO DA QUALIDADE DOS METADADOS DE REA USANDO *DEEP LEARNING*

4.1 Considerações Iniciais

Neste capítulo, são descritos os resultados da avaliação automática de metadados dos REA, usando uma arquitetura de rede neural profunda, especificamente a *long short-term memory* (LSTM). A avaliação inicial foi publicada no Simpósio Brasileiro de Informática na Educação (SBIE)¹ e após a publicação implementamos diversas melhorias na arquitetura da rede neural e no *dataset* usado para treinamento.

Na Seção 4.2, é apresentada uma análise descritiva do córpus. Nas Seções 4.3, 4.4, e 4.5 são apresentados os modelos desenvolvidos usando métodos tradicionais de aprendizado de máquina e de *Deep Learning*. Na Seção 4.6, é descrita a avaliação manual realizada na base de dados. Na Seção 4.7, são apresentados os resultados obtidos e as discussões sobre os modelos. Por fim, na Seção 4.8, são apresentadas as considerações finais do capítulo.

4.2 Análise descritiva do córpus de metadados de REA

O *dataset* de REA utilizado é composto de duas partes: pequeno conjunto de dados, que possui 80 REA anotados manualmente seguindo os critérios apresentados na Seção 4.6, e conjunto de dados completo, que possui 1.179 REA anotados manualmente, seguindo os mesmos critérios apresentados na Seção 4.6. O *dataset* de REA está disponibilizado publicamente em <<https://github.com/gazzola/MetaQualityNLP>>. O *dataset* completo possui título, tipo de

¹ <<https://goo.gl/1Bg7NX>>

Figura 26 – Excerto do dataset de metadados de REA

titulo	tipo	etapa_ensino	resumo	formato_comp
"Bracatinga"	Imagem	Etapas de ensino nao identificado	"Acrescentar ao aluno informações acerca de uma espécie de árvore bastante utilizada no Brasil"	image/jpeg
"Inscrição e Circunscrição"	Animação/simulação	Etapas de ensino nao identificado	"Explorar a inscrição e a circunscrição de polígonos em uma circunferência"	application/zip
"Protocolo: experimentos com água - experiência 2"	Experimento prático	Etapas de ensino nao identificado	"Demonstrar como ocorre o processo de ebulição"	application/msword
"Mudança de fase"	Experimento prático	Etapas de ensino nao identificado	"É necessário ter instalado no computador o programa Adobe Reader"	application/pdf
"Prisma de água"	Experimento prático	Etapas de ensino nao identificado	"Experimento prático que permite ao aluno entender a decomposição da luz utilizando como prisma uma bacia de água. A luz passa através da água é ..."	application/pdf
"Ressonância na garrafa: parte 1: experimento prático"	Experimento prático	Etapas de ensino nao identificado	"Mostrar como são formadas ondas estacionárias no interior de um tubo fechado"	application/pdf
"Podemos construir - atividade 1: Podemos construir um ludião?"	Experimento prático	Etapas de ensino nao identificado	"Experimento simples <VIRGULA> com a utilização de materiais de baixo custo e de fácil obtenção"	application/pdf
"Formando um precipitado: parte 2: vídeo"	Vídeo	Etapas de ensino nao identificado	"Duração de 2:15 minutos. Este vídeo complementa o experimento prático \"Formando um precipitado: parte 1: experimento prático\""	video/mp4
"Respostas curtas"	Software Educacional	Etapas de ensino nao identificado	"Aprender como dar respostas curtas em inglês"	application/zip

Fonte: Elaborada pelo autor.

material, etapa de ensino, url, resumo e formato do arquivo. Porém, na avaliação com métodos de PLN usamos apenas os metadados título e resumo.

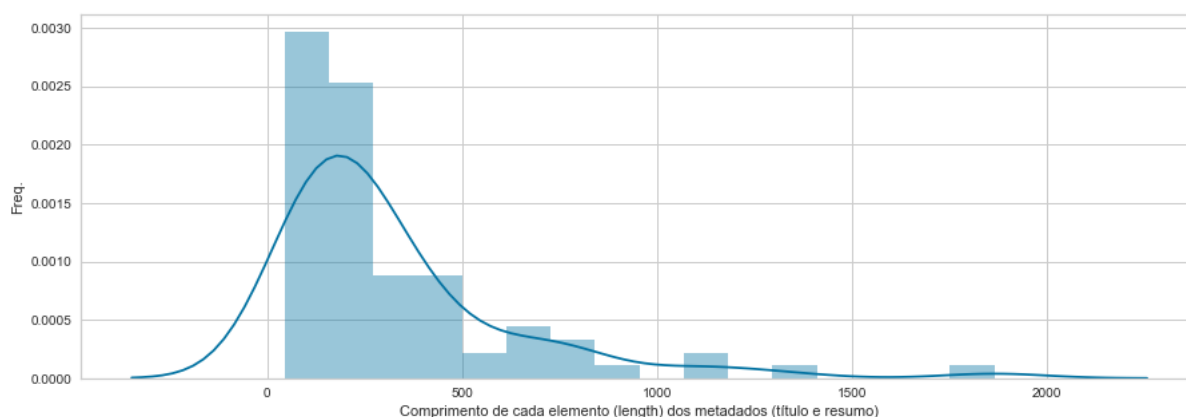
A Figura 26 mostra uma amostra do *dataset* de REA utilizado neste trabalho com 5 metadados. A Tabela 13 apresenta um resumo do cópulus anotado em 2 classes (não qualificado e qualificado) do pequeno conjunto de dados (*small data*) e do conjunto de dados completo. Também, informamos o tipo de material para cada classe anotada, embora o tipo de material não tenha sido levado em consideração nas regras de anotação para avaliação da qualidade. Mesmo assim, é possível observar que nosso cópulus possui uma gama de recursos do gênero instrucional, por exemplo, experimentos práticos, animações/simulações, software educacional, que são do interesse desta tese.

Foi feita uma concatenação do título e resumo para cada REA para podermos analisar e usar o conjunto de dados. Em seguida, avaliamos o comprimento dos títulos e resumos do nosso conjunto com um histograma. O gráfico da Figura 27 (pequeno conjunto de dados) e a Figura 28 (conjunto completo de dados) mostram a somatória do tamanho do título e resumo para cada REA (comprimento do vetor) no eixo x e frequência observada dos comprimentos no eixo y. Observamos na Figura 27 o comprimento máximo de 1864 caracteres, mínimo de 44 e desvio padrão de 315,46. Na Figura 28 observamos o comprimento máximo de 4504 caracteres,

Tabela 13 – Extrato dos tipos de materiais nos REA Qualificados e Não Qualificados

Pequeno conjunto de dados (<i>small data</i>)				Conjunto de dados completo			
Classe dos Não Qualificados		Classe dos Qualificados		Classe dos Não Qualificados		Classe dos Qualificados	
Vídeo	14	Vídeo	14	Vídeo	289	Vídeo	272
Experimento prático	10	Áudio	8	Animação/simulação	130	Animação/simulação	98
Animação/simulação	9	Animação/simulação	7	Experimento prático	75	Apresentação	87
Áudio	3	Imagem	6	Áudio	51	Imagem	32
Imagem	2	Experimento prático	5	Imagem	22	Áudio	31
Outros	1			Livro digital	13	Experimento prático	21
Software Educacional	1			Software Educacional	10	Website externo	9
TOTAL	40	TOTAL	40	Infográfico	6	Livro digital	8
				Jogo	5	Jogo	4
				Outros	4	Software Educacional	4
				Apresentação	2	Outros	3
				Mapa	1	Aplicativo móvel	1
						Infográfico	1
				TOTAL	608	TOTAL	571

Fonte: Elaborada pelo autor.

Figura 27 – Histograma do comprimento do texto (*small data*)

Fonte: Elaborada pelo autor.

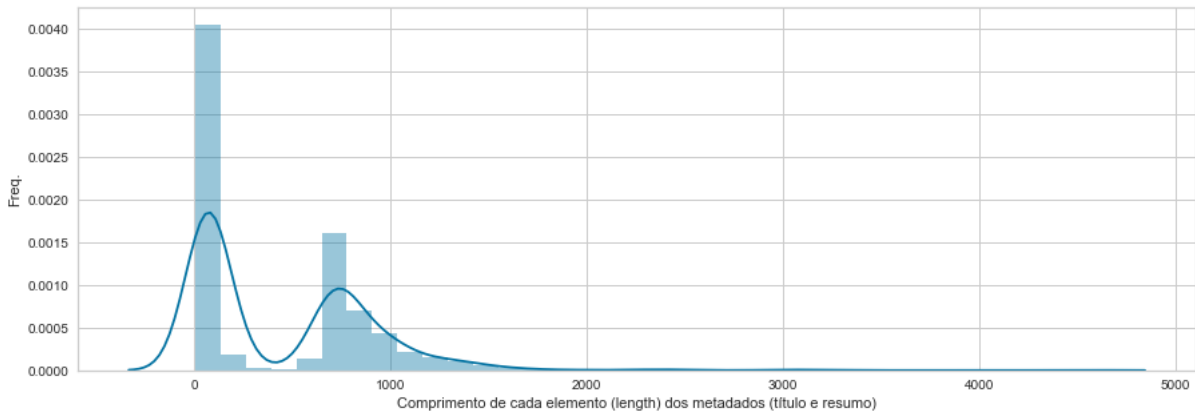
mínimo de 4 e desvio padrão de 466,77.

Um gráfico comparativo entre as classes qualificados e não qualificados e seus respectivos tamanho de textos (título + resumo) são apresentados na Figura 29 (*small data*) e na Figura 30 (conjunto completo de dados). Observamos que os REA qualificados possuem um comprimento de texto maior comparado com os não qualificados que se destacam com valores pequenos de comprimentos.

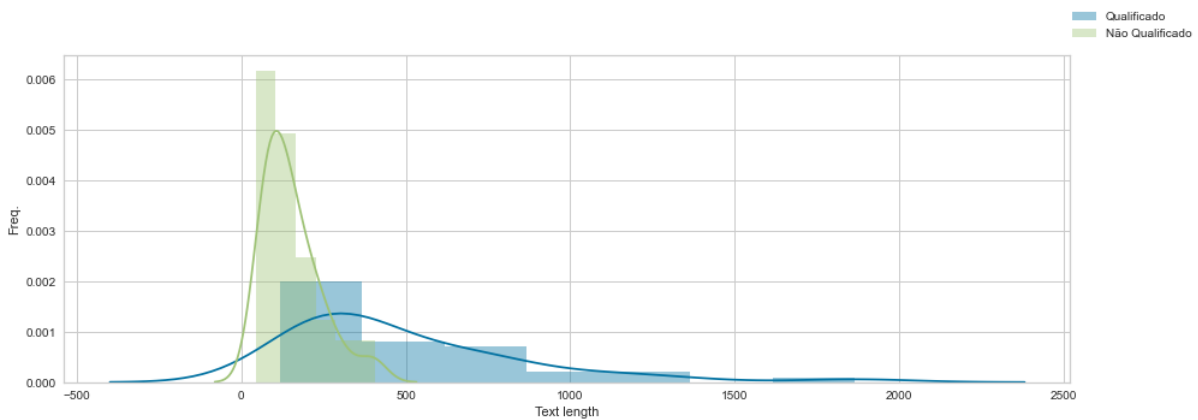
A Figura 31 apresenta um gráfico do pequeno conjunto de dados e ilustra as classes de REA anotadas (qualificados e não qualificados), com o histograma das contagens de palavras entre as classes. A classe não qualificados apresenta o número de no máximo 64 palavras, mínimo de 6 e o desvio padrão de 12,41. Enquanto que a classe de Qualificados apresenta a quantidade máxima de 187 palavras, mínimo de 22 e desvio padrão de 47,72.

Com o objetivo de avaliar os tokens em nosso *córpus*, trouxemos um gráfico com os

Figura 28 – Histograma do comprimento do texto (conjunto completo de dados)

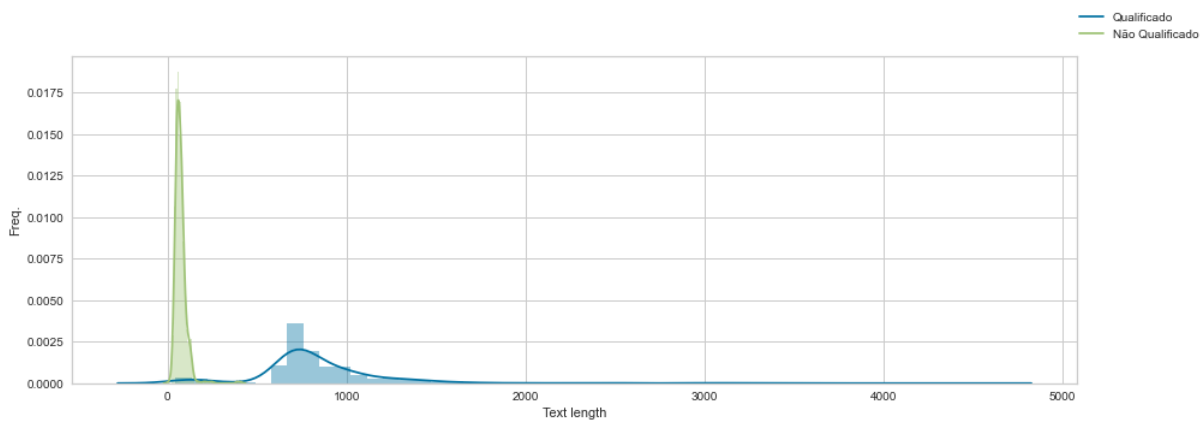


Fonte: Elaborada pelo autor.

Figura 29 – Histograma do comprimento do texto por classe (*small data*)

Fonte: Elaborada pelo autor.

Figura 30 – Histograma da quantidade de palavras (conjunto de dados completo)



Fonte: Elaborada pelo autor.

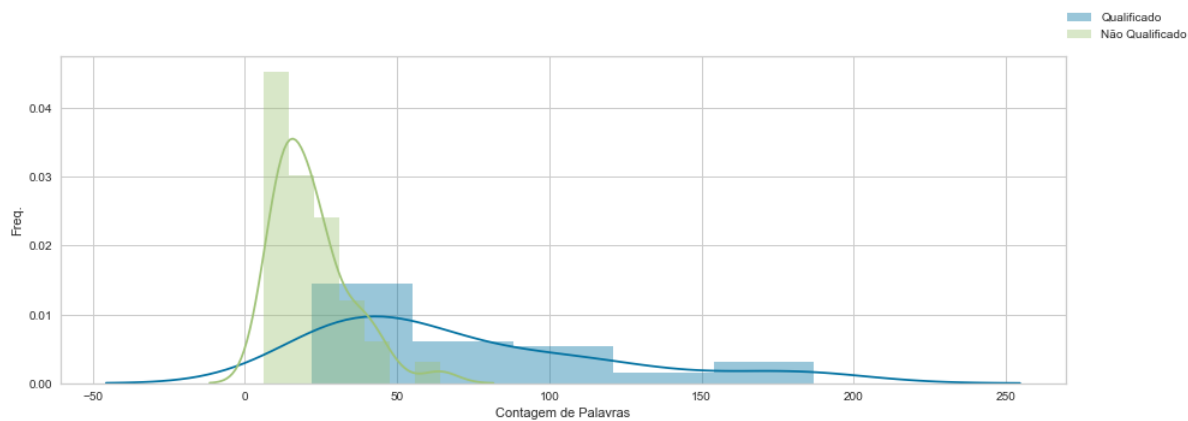
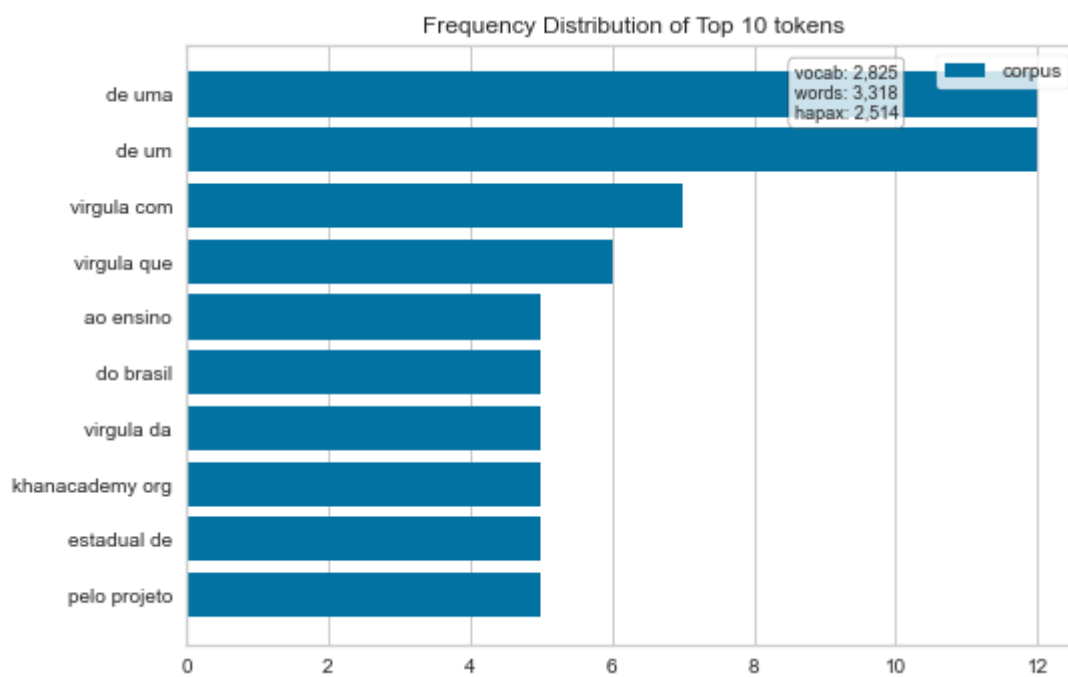
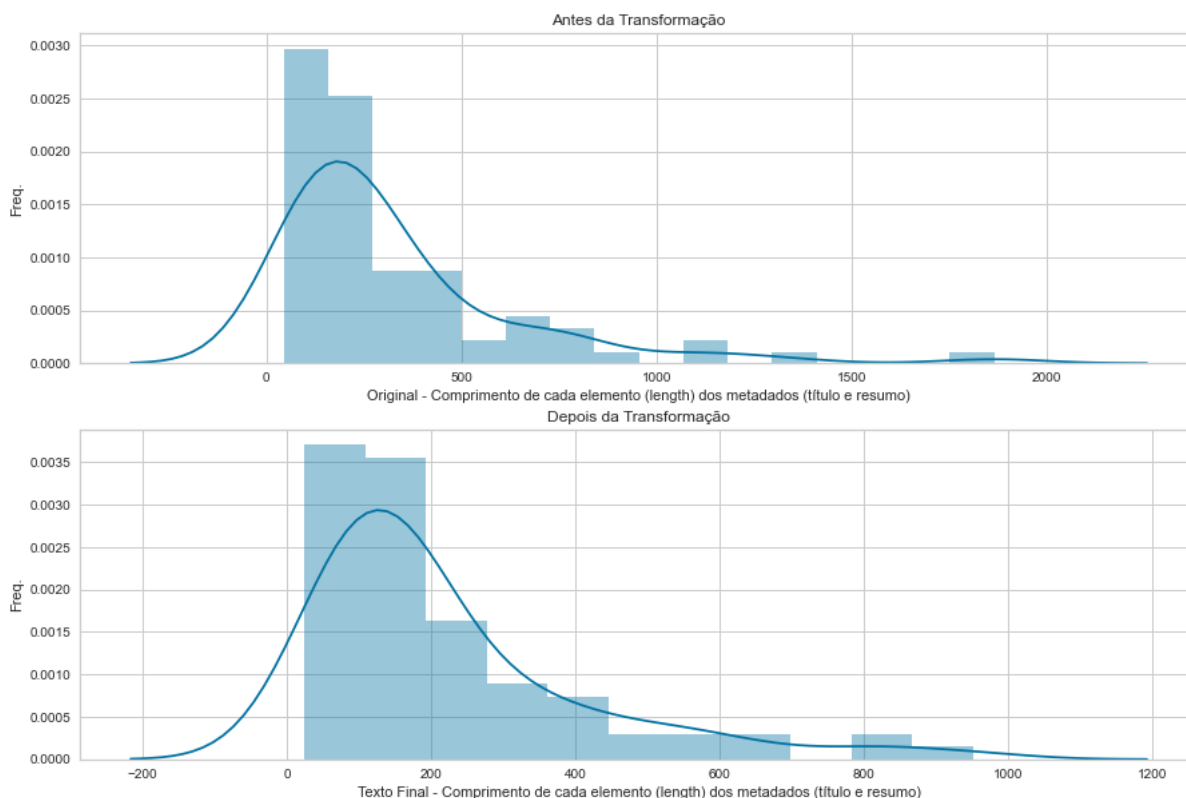
Figura 31 – Histograma da quantidade de palavras por classe (*small data*)Figura 32 – Número de Bigramas do *Córpus* (*small data*)

Figura 33 – Comprimento de textos após a fase de pré-processamento (*small data*)

Fonte: Elaborada pelo autor.

bigramas ($n\text{-gram}=2$) na Figura 32, que apresenta os 10 bigramas mais frequentes do corpus. Observamos a prevalência dos bigramas: “de uma”, “de um”, “virgula com”, “virgula que”. Realizamos um pré-processamento em nosso corpus para remoção de tags HTML, links/urls, pontuações, números e stop words².

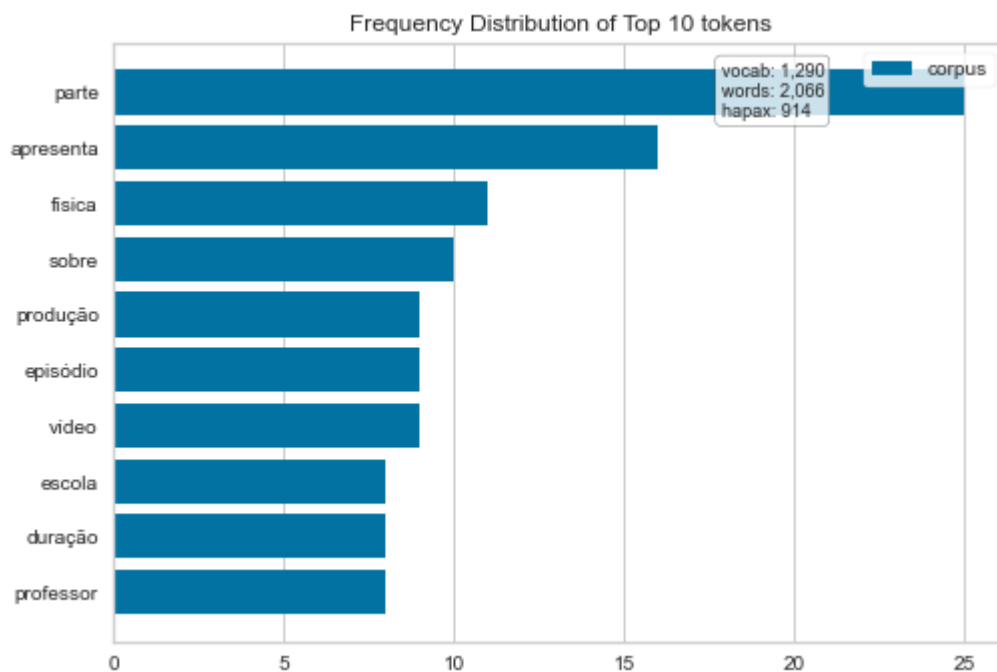
Após o pré-processamento do corpus, comparamos os dois histogramas de comprimento de texto (antes e depois) na Figura 33. Na Figura 34, é possível observar a distribuição dos 10 *tokens* mais frequentes em nosso corpus após o pré-processamento.

Outra forma de visualizarmos a frequência de palavras no corpus é via nuvem de palavras. Na Figura 35, mostramos na nuvem à esquerda, realizada antes do pré-processamento, a alta frequência de *stop words* no dataset, por exemplo: “para”, “de”, “da”, “na”, “um”, “os”. Na segunda nuvem, à direita, é possível observar as palavras de conteúdo mais frequentes após o processamento, por exemplo: “escola”, “apresenta”, “episódio”, “produção”, “matemático”, “explica”, “professor”.

Após a fase de pré-processamento (ou limpeza do corpus), extraímos a *feature* TF-IDF do nosso corpus para utilização nos algoritmos de aprendizado de máquina tradicional. Para isso, escolhemos o modelo TF-IDF (*Term Frequency - Inverse Document Frequency*) (RAMOS

² Utilizamos a biblioteca NLTK(<<https://www.nltk.org/>>) e stopwords.words(“portuguese”).

Figura 34 – Distribuição dos 10 tokens mais frequentes no corpú, após o pré-processamento do corpú (*small data*)



Fonte: Elaborada pelo autor.

Figura 35 – Nuvem de palavras antes e depois da limpeza do corpú (*small data*)



Fonte: Elaborada pelo autor.

Figura 36 – Exemplo de valores TF-IDF

	episódio	escola	mostrar
13	0.000000	0.000000	0.108290
16	0.000000	0.000000	0.228957
20	0.069451	0.000000	0.000000
26	0.000000	0.111699	0.000000
28	0.093020	0.093020	0.000000
32	0.000000	0.000000	0.213348
36	0.133091	0.000000	0.000000
44	0.000000	0.146089	0.000000
45	0.000000	0.000000	0.109602
56	0.000000	0.000000	0.237165
57	0.131845	0.000000	0.000000
58	0.000000	0.140705	0.000000
63	0.186912	0.186912	0.000000

Fonte: Elaborada pelo autor.

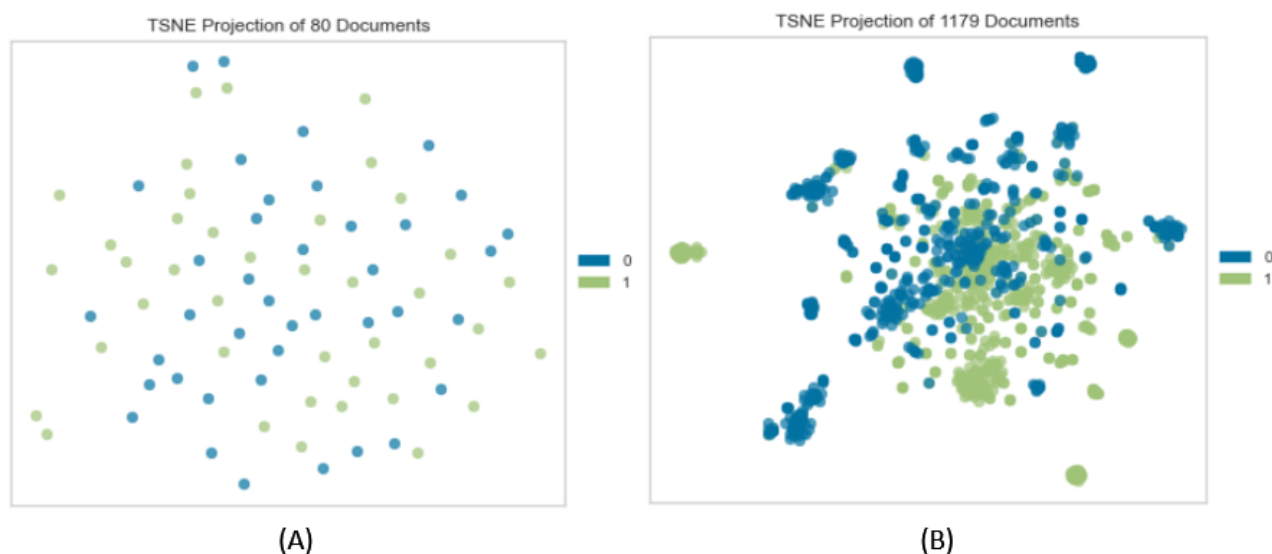
et al., 2003). A ideia é simples, quanto maior a frequência de um termo em um documento, mais representativo ele é para o conteúdo e quanto mais documentos contiverem um termo, menos discriminante ele é para o conteúdo. Para demonstrar um exemplo do funcionamento do TF-IDF, a Figura 36 traz três palavras (“episódio”, “escola”, “mostrar”). Observamos que a palavra “episódio” possui um valor TF-IDF maior no documento 63; a palavra “escola” no 63 e a palavra “mostrar” no documento 56.

Na Figura 37 mostramos o resultado do algoritmo t-Distributed Stochastic Neighbour Embedding (t-SNE) para visualização dos pontos no espaço para cada classe, em que 0 representa os REA da classe de não qualificado (cor azul) e 1 representa da classe de qualificados (cor verde). É possível observar pela visualização com t-SNE que não é um problema simples a separação em 2 classes usando TF-IDF somente. Na Figura 37 usamos o t-SNE com pequeno conjunto de dados (A) e (B) conjunto completo de dados.

4.3 Classificador Baseline

Como baseline foi usado o classificador DummyClassifier da biblioteca scikit-learn³, normalmente usado como baseline para comparar o desempenho de outros classificadores. Os parâmetros usados foram strategy=default (estratificado) que gera as predições respeitando a distribuição das classes do conjunto de treino; a divisão entre conjunto de treino e de teste foi de

³ <<https://scikit-learn.org/stable/modules/generated/sklearn.dummy.DummyClassifier.html>>

Figura 37 – t-SNE do córpus - (A) *small data* e (B) conjunto completo de dados

Fonte: Elaborada pelo autor.

80%, 20%, respectivamente. O resultado do baseline foi F1 (média) de 0.375 usando *small data*. Enquanto, com conjunto de dados completo o resultado foi F1 (média) de 0.341.

4.4 Classificador SVM

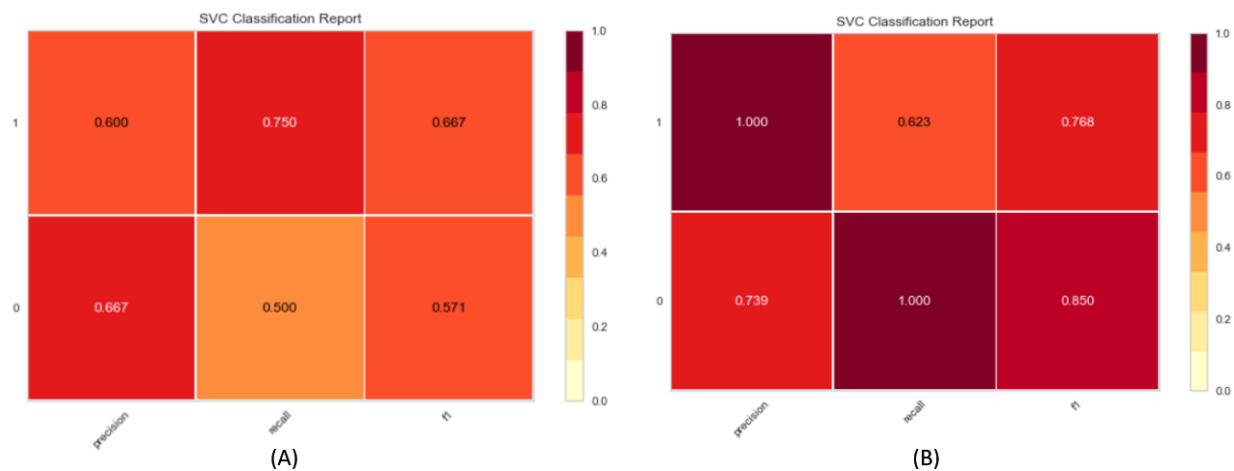
Para efeito de comparação entre os classificadores, foi usado o método SVM, pois diversos trabalhos de avaliação automática de REA utilizam esse método (BETHARD *et al.*, 2009; DALIP *et al.*, 2011; CECHINEL; SANCHEZ-ALONSO; GARCIA-BARRIOCANAL, 2011; AHMED; FUGE, 2017). Os parâmetros do SVM foram escolhidos por meio do GridSearchCV⁴ que realiza uma busca exaustiva para encontrar os melhores hiperparâmetros do SVM. Neste caso, os melhores hiperparâmetros foram $C=0.01$ e $\text{kernel}=\text{linear}$. O córpus foi dividido em 80% dos dados para treino e 20% para teste. O resultado com (A) *small data* do SVM superou o baseline e teve F1 (média) de 0.619 (Figura 38). Enquanto, o (B) conjunto de dados completo o SVM teve F1 (média) de 0.8089 superando o baseline e o SVM com *small data*.

4.5 Classificador LSTM (DeepMD e DeepMDX)

A arquitetura da rede LSTM é uma variante da Rede Neural Recorrente (RNN), sendo capaz de aprender dependências de longa distância. A LSTM foi proposta por Hochreiter e Schmidhuber (1997) e refinada por outros pesquisadores ao longo do tempo Donahue *et al.* (2015). Neste trabalho propomos duas arquiteturas baseadas em LSTM. A primeira é a DeepMD desenvolvida usando TensorFlow e Keras, composta por uma camada de *embeddings* que pode

⁴ <https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html>

Figura 38 – SVM: Precision, Recall e F1 - experimentos com (A) *small data* e (B) conjunto de dados completo



Fonte: Elaborada pelo autor.

variar o tamanho do vocabulário (1607), tamanho da embeddings (128) e o tamanho máximo da sentença (255). A camada de *embeddings* possui 205.696 parâmetros treináveis. Essa camada é conectada na camada *Dropout* (SpatialDropout1D) com o parâmetro 0.2. A camada dropout pode minimizar o *overfitting*. Em seguida, é conectada na camada da LSTM, que possui o tamanho da camada oculta (*hidden layer size*) de 64, sendo configurado os parâmetros em *dropout*=0.2 e *recurrent_dropout*=0.2. Por fim, essa camada é conectada em uma camada densa de saída com 1 neurônio, conectada na camada de ativação configurada para a função Sigmoid. Na camada de ativação, sairá o resultado 0 que representa não qualificado ou 1 que representa qualificado. Foram utilizados a função de custo *binary_crossentropy* e o otimizador Adam.

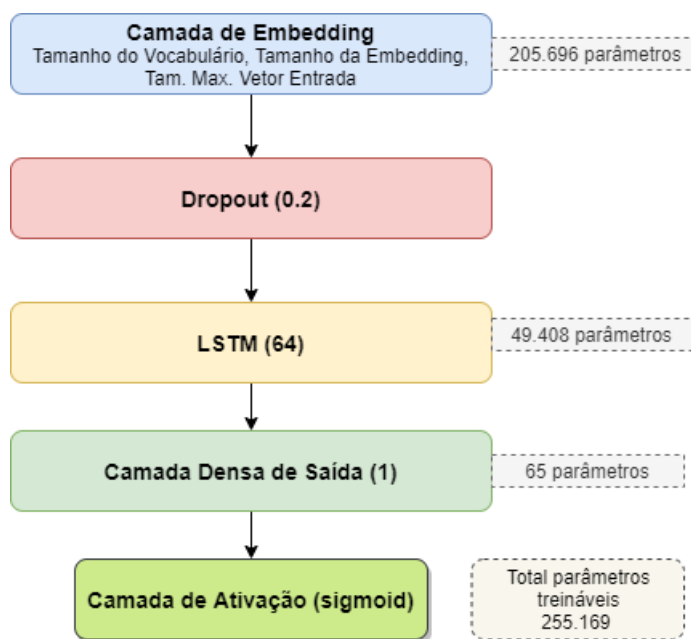
A segunda arquitetura desenvolvida é a DeepMDX, usando as mesmas bibliotecas da DeepMD. A diferença entre elas está na adição, na DeepMDX, de mais uma camada densa com 32 neurônios entre a camada da LSTM e a camada densa de saída; contabilizando 2048 parâmetros treináveis a mais, comparado com a DeepMD. As redes neurais DeepMD e DeepMDX foram treinadas usando k-fold=10; a DeepMD obteve F1 de 0.9250 e DeepMDX obteve F1 de 0.9857 com o conjunto *small data*. Enquanto, DeepMD e DeepMDX com o conjunto de dados completo obteve F1 de 0.98 (Tabela 14).

4.6 Critérios de Avaliação da Qualidade dos REA

Os metadados foram avaliados manualmente (o anotador é o autor desta tese) considerando **Título, URL e Resumo** dos REA.

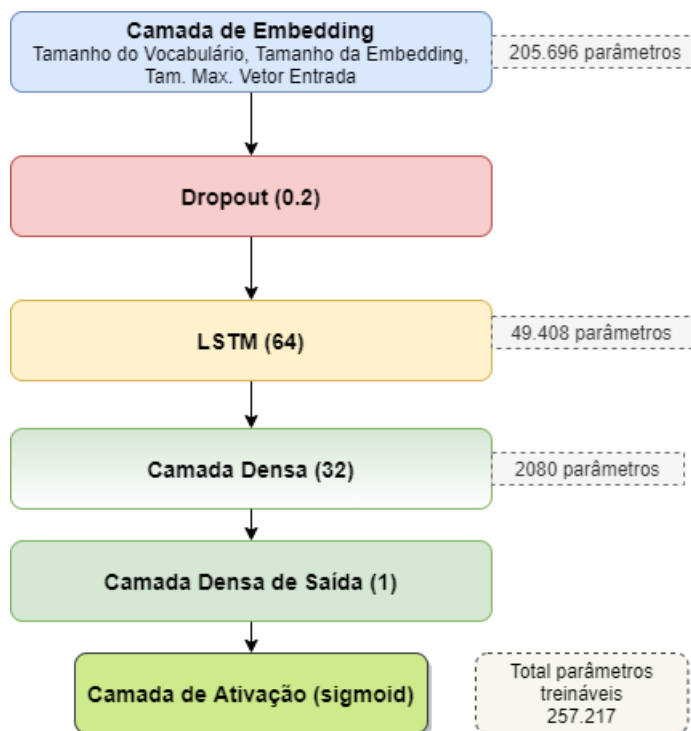
Avaliação manual é composta de duas etapas: (1) ler todos os metadados e (2) verificar os seguintes critérios apresentados a seguir:

Figura 39 – Arquitetura DeepMD - LSTM para classificação dos metadados de REA



Fonte: Elaborada pelo autor.

Figura 40 – Arquitetura DeepMDX - LSTM para classificação dos metadados de REA



Fonte: Elaborada pelo autor.

- O avaliador deve verificar se o título possui reconhecibilidade (MEDELYAN *et al.*, 2009) (MATSUNE, 2007). A reconhecibilidade é uma descrição do assunto que trata o REA; mesmo uma pessoa que não é especialista da área do material consegue reconhecer o assunto. Utilizamos o conceito de reconhecibilidade, pois facilita a classificação manual para diversos assuntos e não necessita de especialistas para cada área. Por exemplo: “Introdução à Ecologia e Biologia Evolutiva” é um título reconhecível, “ECE 420 Fall 2013” é um título não reconhecível.
- O avaliador deve identificar 5 elementos retóricos na estrutura do resumo. Os elementos a serem identificados são: a fundamentação teórica sobre o assunto do REA, propósito do REA, metodologia usada, resultados e conclusão. Se forem observados pelo menos 2 elementos, o resumo deve ser considerado qualificado.

Portanto, o REA será considerado não qualificado se algum critério de qualidade for desrespeitado. Caso contrário, o REA deve ser considerado qualificado. Por exemplo, se não possuir um título legível é considerado desqualificado ou se não possuir no mínimo 2 elementos retóricos do resumo, dentre os 5 elementos citados anteriormente, ele é considerado não qualificado.

4.7 Conclusões e Discussões

O principal objetivo deste estudo foi criar um classificador de metadados para a área da Educação usando técnicas de PLN. Na área da Educação, foram encontrados poucos trabalhos em 2017 envolvendo a avaliação da qualidade de metadados de REA com o uso de técnicas de Deep Learning e PLN. Além disso, é um desafio avaliar os REA dos tipos de arquivos como vídeos, imagens, áudio e outros e que são predominantes em diversos repositórios de REA. Neste caso, procuramos por dados textuais para aplicar técnicas de PLN, como os metadados título e resumo, presente em grande parte deles em repositórios de REA.

Na avaliação das arquiteturas DeepMD e DeepMDX, usamos cross-validation e configuração k-fold=10 com 10 épocas. A Tabela 14 apresenta os valores de F1 para os métodos avaliados. O melhor resultado foi do modelo DeepMDX, que obteve F1 de 0.9857, no conjunto *small data*. No conjunto de dados completo o modelo DeepMD e DeepMDX obteve resultados semelhantes com F1 de 0.98.

4.8 Considerações Finais

Com base nesses experimentos, pode-se concluir que o uso de *Deep Learning* é uma alternativa viável que não exige a necessidade de se usar um vetor de features como a TF-IDF ou features mais linguísticas. Para esta tarefa de avaliação da qualidade de metadados, métodos de *Deep Learning* podem desempenhar melhor do que os métodos tradicionais da área de

Tabela 14 – Comparação entre os classificadores e os métodos utilizados

Classificador	<i>Small data</i>	Conjunto de dados completo	Aprendizado de Máquina
	F1 (média)	F1 (média)	
Dummy	0,37	0,34	Tradicional
SVM	0,61	0,80	Tradicional
DeepMD	0,92	0,98	Deep Learning (LSTM)
DeepMDX	0,98	0,98	Deep Learning (LSTM)

Fonte: Elaborada pelo autor.

aprendizagem de máquina. Entretanto, o estudo ainda é exploratório dado o tamanho do cópús de treinamento. Assim, o mérito deste estudo foi apresentar e motivar o uso de *Deep Learning* no contexto de classificação automática de metadados (título e resumo) para avaliação de REA.

PREDIÇÃO DA COMPLEXIDADE TEXTUAL DE REA NO PORTUGUÊS BRASILEIRO

5.1 Considerações Iniciais

Em 2016, a UNESCO escreveu em seu relatório as prioridades para o uso de REA, destacando os principais desafios de pesquisas. A falta de qualidade dos REA é um desafio a ser superado. Em uma recolha na Plataforma Integrada do Ministério da Educação (MEC-RED) de **maio de 2018**, 41% dos recursos não possuíam classificação da etapa de ensino, dificultando sua busca, uso e edição. A tarefa Complexidade Textual pode ajudar a identificar textos que tem complexidade linguística adequada a séries específicas, permitindo completar a etapa de ensino.

Neste capítulo, avaliamos o impacto do gênero textual na avaliação da complexidade textual, com um modelo treinado em um grande cópús de textos didáticos e testado em 2 conjuntos de REA da MEC-RED.

As seções a seguir são organizadas da seguinte maneira. Na Seção 5.2 apresentamos o cópús que compilamos e que abrange textos de 4 etapas de ensino do Sistema Educacional Brasileiro. Na Seção 5.3 trouxemos os experimentos com os métodos tradicionais de aprendizagem de máquina e os métodos de seleção de *features*. Na Seção 5.4 apresentamos a avaliação intrínseca e extrínseca dos métodos tradicionais de aprendizagem de máquina. Na Seção 5.5 trouxemos a avaliação intrínseca e extrínseca dos métodos de *Deep Learning*. Por fim, nas Seções 5.6 e 5.7 descrevemos as conclusões e discussões sobre o trabalho desenvolvido e relatado no capítulo e as considerações finais.

5.2 Córpus dos Quatro Estágios Escolares do Sistema Educacional Brasileiro

Compilamos um grande córpus que contém textos utilizados em diferentes etapas de ensino (ou níveis escolares) do Sistema Educacional Brasileiro, organizado nas seguintes etapas: Ensino Fundamental I (1º ao 5º ano), Ensino Fundamental II (6º ao 9º ano), Ensino Médio e Ensino Superior. Essas quatro etapas de ensino são as mesmas utilizadas na MEC-RED para classificar os REA nos Estágios Escolares. Embora a plataforma também permita pesquisar os REA pela etapa Educação Infantil, não estamos usando esse estágio em nosso trabalho. O córpus inclui:

- livros-texto,
- notícias da Seção *Para Seu Filho Ler* (PSFL) do jornal Zero Hora¹ que apresenta algumas notícias sobre os mesmos tópicos do Zero Hora, mas que são escritas para crianças de 8 a 11 anos de idade²,
- Exames do SAEB³,
- Livros Digitais do Wikilivros em Português⁴, por exemplo, *Marcha Criança*, *Tudo É Linguagem*, *Projeto Porta Aberta*, *Projeto Ápis*, *Português*, *Buriti*, *Porta Aberta*, *Mundo Amigo*, *Nos Dias de Hoje*, *Projeto Teláris*, *CNEC Educação*, e
- Exames do Enem dos anos 2015, 2016 e 2017⁵.

Nosso córpus de trabalho compreende 2.067 extratos (min = 300 palavras, max = 596 palavras, média = 448, Total *Type-Token Ratio* (TTR) = 0.571, Total de palavras únicas (*Types*) = 450.621) dos textos do córpus compilado. Como pode ser visto na Tabela 15, nosso córpus não é balanceado, pois o número de textos do Ensino Médio possui aproximadamente o dobro da quantidade do Ensino Fundamental I e do Ensino Fundamental II, por exemplo. Para resolver esse problema, foi utilizado o método *ClassBalancer* do Weka⁶ antes da execução dos métodos de aprendizado de máquina tradicional (cf. mais detalhes na Seção 5.4). Weka foi uma ferramenta que usamos apenas nos experimentos envolvendo aprendizado de máquina tradicional e para esse capítulo da tese. Além disso, para utilização do Weka foi necessário fazer um pré-processamento das métricas obtidas nas amostras e transformá-las no formato *Attribute-Relation File Format* (ARFF), que é o formato de dados de entrada para utilizar os algoritmos do Weka.

¹ <zh.clicrbs.com.br/rs>

² Vários textos foram agrupados para se obter um tamanho razoável.

³ <<http://portal.inep.gov.br/educacao-basica/saeb>>

⁴ <<https://pt.wikibooks.org/wiki/Wikilivros>>

⁵ <<http://portal.inep.gov.br/provas-e-gabaritos>>

⁶ <<https://www.cs.waikato.ac.nz/~ml/weka/>>

Tabela 15 – Córpus de livros-textos da Língua Portuguesa compilado: Marcha Criança, Tudo É Linguagem, Projeto Porta Aberta, Projeto Ápis, Português, Buriti, Porta Aberta, Mundo Amigo, Nos Dias de Hoje, Projeto Teláris, CNEC Educação.

	Ensino Fund I	Ensino Fund II	Ensino Médio	Ensino Superior	Total
Fontes de textos	Livros-texto + PSFL	Livros-texto, SAEB, Ebooks	Wikilivros + ENEM 2015, 2016, 2017	Wikilivros	
# Docs	296	325	627	819	2.067
# Sents	5.258	5.598	9.316	10.416	30.588
ASL*	20.58	24.31	29.81	39.15	28.463
Types	63.081	75.698	134.788	177.054	450.621
Tokens	101.911	127.705	241.267	342.534	813.417
TTR	0.618	0.592	0.558	0.516	0.571

*ASL = Average Sentence Length (in words)

Fonte: Elaborada pelo autor.

Neste trabalho, nós implementamos um crawler usando Python e a biblioteca Selenium⁷ que reconhece os dados em páginas AngularJS⁸/JavaScript para capturar as páginas dos recursos do MEC-RED, incluindo os metadados principais de forma automática e recursiva. Também, passamos esses dados para um banco de dados relacional. Após isso, usamos os links obtidos na captura que aponta para um material armazenado externamente ao repositório MEC-RED para extrair uma amostra textual dos REA. Os REA estão armazenados em diferentes extensões como .zip, html, flash e pdf. Assim, separamos amostras destes REA e dividimos em 2 datasets (Tabela 16) que possuem conjuntos de textos de recursos da MEC-RED para o treinamento: um dataset chamado de Dataset 1 (40 REA) de recursos da MEC-RED do tipo apresentações, livros digitais, website externo, planos de aula e textos que são considerados do gênero de textos didáticos, e outro chamado de Dataset 2 (60 REA) composto por textos de recursos do tipo animação/simulação, experimento prático e software educacional, considerados do gênero de texto instrucional, que indica o que fazer, isto é, fornece mais frequentemente os passos para se executar uma tarefa/ação.

5.3 Experimentos usando Aprendizado de Máquina Tradicional e Métodos de Seleção de Features

A seleção inicial das métricas para a avaliação da complexidade textual baseou-se no estudo de Graesser e McNamara (2011) que utilizou 53 métricas do Coh-Metrix agrupadas naquelas relacionadas às palavras, sentenças e conexões entre sentenças. Para selecionar métricas

⁷ <<https://www.selenium.dev/documentation/en/webdriver/>>

⁸ <<https://angularjs.org/>>

Tabela 16 – Descrição do Córpus de Textos dos Recursos da MEC-RED

	Dataset 1 (D1)	Dataset 2 (D2)
# Docs	40	60
# Sent	675	1382
ASL	27.43	18.77
Type	8511	11572
Token	15527	22110
TTR	0.548	0.523

Fonte: Elaborada pelo autor.

similares para o português, escolhemos duas ferramentas disponíveis publicamente: Coh-Matrix-Port (SCARTON; GASPERIN; ALUÍSIO, 2010) e Coh-Matrix-Dementia (CUNHA, 2015) e também as métricas sobre concretude, imageabilidade, Idade de Aquisição e Familiaridade resultantes do trabalho de Santos *et al.* (2017a).

Na Tabela 17 e Tabela 18 estão presentes as 79 métricas usadas nesta pesquisa. Entretanto, 17 métricas do estudo de Graesser e McNamara (2011) não foram adaptadas para o português, seja por falta de recursos linguísticos ou ferramentas precisas de PLN. Elas são listadas aqui para futuras pesquisas: conectivos adversativos, *meaningfulness*, verbos causais, ações intencionais, eventos e partículas, similaridade sintática (sentenças no parágrafo), sobreposição de palavras de conteúdo em todas as sentenças, dissimilaridade de PoS entre sentenças e dissimilaridade de palavras entre sentenças, coesão causal, temporal e intencional, repetição de tempo e de aspecto verbal, log da frequência de palavras, sobreposição de verbo adjacente e sobreposição de verbo no modelo LSA em sentenças adjacentes. Para suprir essa falta, novas foram anexadas, como, por exemplo, Complexidade de Yngve e de Frasier, Distância de Dependência, dentre outras.

Foram avaliados o *Correlation-based Feature Selection* (CFS) e o *Least Absolute Shrinkage and Selection Operator* (Lasso); o CFS resultou em 34 *features* (Tabela 19) e o Lasso resultou em 8 *features*. Foram realizados experimentos de predição usando as 34 *features* selecionadas pelo CFS e 8 *features* selecionadas pelo Lasso com os classificadores Logistic Regression (LR), *Random Forest* (RF), *Support Vector Machines* (SVM) e *Multilayer Perceptron* (MLP) (Tabela 20). Os melhores resultados com 34 *features* foi para o classificador *Random Forest* com média de *F-Measure* de 0.798. Também, foram realizados experimentos de predição usando as 8 *features* selecionadas pelo Lasso. Nos experimentos, os melhores resultados foram com o classificador *Random Forest*, também, com média de *F-Measure* de 69.6. A configuração do experimento com o Lasso teve o parâmetro alpha ajustado em 0.2. O método selecionou 8 *features* consideradas mais representativas para o conjunto de dados, são elas: proporção de pronomes, pronomes de primeira pessoa, proporção de palavras de conteúdo do texto com familiaridade entre 4 e 5.5, desvio padrão da imageabilidade, proporção de palavras de conteúdo do texto com imageabilidade entre 4 e 5.5, proporção de palavras de conteúdo do texto com imageabilidade entre 5.5 e 7, desvio padrão do comprimento da sentença e densidade de conteúdo. Na Figura

Tabela 17 – 79 métricas usadas nesta pesquisa

Nº	Nome da Métrica	Descrição
1	adjective_ratio	Proporção de Adjetivos em relação à quantidade de palavras do texto
2	adverbs	Proporção de Advérbios em relação à quantidade de palavras do texto
3	syllables_per_content_word	Quantidade média de sílabas por palavra no texto
4	words_per_sentence	Número de palavras dividido pelo número de sentenças
5	noun_ratio	Proporção de Substantivos em relação à quantidade de palavras do texto
6	pronoun_ratio	Proporção de pronomes em relação à quantidade de palavras do texto
7	verbs	Proporção de Verbos em relação à quantidade de palavras do texto
8	negation_ratio	Proporção de palavras que denotam negação em relação à quantidade de palavras do texto
9	cw_freq	Média das frequências absolutas das palavras de conteúdo do texto
10	min_cw_freq	Média das frequências das palavras de conteúdo mais raras das sentenças
11	first_person_pronouns	Proporção de pronomes pessoais nas primeiras pessoas em relação a todos os pronomes pessoais do texto.
12	ttr	Proporção de palavras sem repetições (types) em relação ao total de palavras com repetições (tokens).
13	conn_ratio	Proporção de Conectivos em relação à quantidade de palavras do texto
14	add_neg_conn_ratio	Proporção de conectivos aditivos negativos em relação à quantidade de palavras do texto
15	add_pos_conn_ratio	Proporção de conectivos aditivos positivos em relação à quantidade de palavras do texto
16	cau_neg_conn_ratio	Proporção de conectivos causais negativos em relação à quantidade de palavras do texto
17	cau_pos_conn_ratio	Proporção de conectivos causais positivo em relação à quantidade de palavras do texto
18	log_neg_conn_ratio	Proporção de Conectivos Lógicos Negativos em relação à quantidade de palavras do texto
19	log_pos_conn_ratio	Proporção de Conectivos Lógicos Positivos em relação à quantidade de palavras do texto
20	tmp_neg_conn_ratio	Proporção de conectivos temporais negativos em relação à quantidade de palavras do texto
21	tmp_pos_conn_ratio	Proporção de conectivos temporais positivo sem relação à quantidade de palavras do texto
22	adjectives_ambiguity	Ambiguidade de Adjetivos
23	adverbs_ambiguity	Ambiguidade de Advérbios
24	nouns_ambiguity	Ambiguidade de Substantivos
25	verbs_ambiguity	Ambiguidade de Verbos
26	yngeve	Árvores sintáticas das sentenças tendem a se ramificar para a direita e desvios em relação a esse padrão correspondem a uma maior complexidade na linguagem. Abordagem bottom-up para o cálculo da complexidade sintática de uma sentença, que parteda palavra e sobe na árvore sintática até encontrar um nó que não seja o filho mais à esquerda de seu pai. Cada nó na árvore recebe uma pontuação 1, e nós filhos de nós do tipo sentença, 1.5.A pontuação de cada palavra é dada pela soma das pontuações dos nós pertencentes a seu ramo.
27	frazier	A distância de dependência utiliza uma árvore de dependências para REALizar o cálculo. A cada relação de dependência está associada uma distância entre as palavras na superfícies textual.
28	dep_distance	Quantidade Média de palavras antes dos verbos principais das orações principais das sentenças
29	words_before_main_verb	Média dos tamanhos médios dos sintagmas nominais nas sentenças
30	mean_noun_phrase	Mínimo entre os sintagmas nominais do texto
31	min_noun_phrase	Máximo entre os sintagmas nominais do texto
32	max_noun_phrase	Desvio-padrão do tamanho dos sintagmas nominais do texto
33	std_noun_phrase	Proporção de orações na voz passiva analítica em relação à quantidade de orações do texto.
34	passive_ratio	Quantidade média de referentes que se repetem nos pares de sentenças adjacentes do texto
35	adj_arg_ovl	Quantidade média de referentes que se repetem nos pares de sentenças do texto
36	arg_ovl	Quantidade média de radicais de palavras de conteúdo que se repetem nos pares de sentenças adjacentes do texto
37	adj_stem_ovl	Quantidade média de radicais de palavras de conteúdo que se repetem nos pares de sentenças do texto.
38	stem_ovl	Quantidade média de radicais de palavras de conteúdo que se repetem nos pares de sentenças adjacentes do texto
39	adj_cw_ovl	Taxa de pronomes pessoais nas terceiras pessoas.
40	third_person_pronouns	

Fonte: Elaborada pelo autor.

Tabela 18 – Continuação ... 79 métricas usadas nesta pesquisa

Nº	Nome da Métrica	Descrição
41	concretude_mean	Média dos valores de concretude das palavras de conteúdo do texto
42	concretude_std	Desvio padrão dos valores de concretude das palavras de conteúdo do texto
43	concretude_1_25_ratio	Proporção de palavras com valor de concretude entre 1 e 2.5 em relação a todas as palavras de conteúdo do texto
44	concretude_25_4_ratio	Proporção de palavras com valor de concretude entre 2.5 e 4 em relação a todas as palavras de conteúdo do texto
45	concretude_4_55_ratio	Proporção de palavras com valor de concretude entre 4 e 5.5 em relação a todas as palavras de conteúdo do texto
46	concretude_55_7_ratio	Proporção de palavras com valor de concretude entre 5.5 e 7 em relação a todas as palavras de conteúdo do texto
47	content_word diversity	Proporção de types de palavras de conteúdo em relação à quantidade de tokens de palavras de conteúdo no texto
48	familiaridade_mean	Média dos valores de familiaridade das palavras de conteúdo do texto
49	familiaridade_std	Desvio padrão dos valores de familiaridade das palavras de conteúdo do texto
50	familiaridade_1_25_ratio	Proporção de palavras com valor de familiaridade entre 1 e 2.5 em relação a todas as palavras de conteúdo do texto
51	familiaridade_25_4_ratio	Proporção de palavras com valor de familiaridade entre 2.5 e 4 em relação a todas as palavras de conteúdo do texto
52	familiaridade_4_55_ratio	Proporção de palavras com valor de familiaridade entre 4 e 5.5 em relação a todas as palavras de conteúdo do texto
53	familiaridade_55_7_ratio	Proporção de palavras com valor de familiaridade entre 5.5 e 7 em relação a todas as palavras de conteúdo do texto
54	idade_aquisicao_mean	Média dos valores de idade de aquisição das palavras de conteúdo do texto
55	idade_aquisicao_std	Desvio padrão dos valores relativos à idade de aquisição das palavras de conteúdo do texto
56	idade_aquisicao_1_25_ratio	Proporção de palavras com valor de idade de aquisição entre 1 e 2.5 em relação a todas as palavras de conteúdo do texto
57	idade_aquisicao_4_55_ratio	Proporção de palavras com valor de idade de aquisição entre 2.5 e 4 em relação a todas as palavras de conteúdo do texto
58	idade_aquisicao_55_7_ratio	Proporção de palavras com valor de idade de aquisição entre 4 e 5.5 em relação a todas as palavras de conteúdo do texto
59	idade_aquisicao_25_4_ratio	Proporção de palavras com valor de idade de aquisição entre 5.5 e 7 em relação a todas as palavras de conteúdo do texto
60	imageabilidade_mean	Média dos valores de imageabilidade das palavras de conteúdo do texto
61	imageabilidade_std	Desvio padrão dos valores de imageabilidade das palavras de conteúdo do texto
62	imageabilidade_1_25_ratio	Proporção de palavras com valor de imageabilidade entre 1 e 2.5 em relação a todas as palavras de conteúdo do texto
63	imageabilidade_25_4_ratio	Proporção de palavras com valor de imageabilidade entre 2.5 e 4 em relação a todas as palavras de conteúdo do texto
64	imageabilidade_4_55_ratio	Proporção de palavras com valor de imageabilidade entre 4 e 5.5 em relação a todas as palavras de conteúdo do texto
65	imageabilidade_55_7_ratio	Proporção de palavras com valor de imageabilidade entre 5.5 e 7 em relação a todas as palavras de conteúdo do texto
66	sentence_length_max	Quantidade Máxima de palavras por sentença
67	sentence_length_min	Quantidade Mínima de palavras por sentença
68	sentence_length_standard_deviation	Desvio Padrão da quantidade de palavras por sentença
69	verb diversity	Proporção de types de verbos em relação à quantidade de tokens de verbos no texto
70	lsa_adj_mean	similaridade média entre pares de sentenças adjacentes.
71	lsa_adj_std	desvio padrão de similaridade entre pares de sentenças adjacentes.
72	lsa_all_mean	média de similaridade entre todos os pares de frases.
73	lsa_all_std	Desvio padrão de similaridade entre palavras
74	lsa_givennes_mean	média de givenness de cada sentença do texto, a partir da segunda sentença
75	lsa_givennes_std	Desvio padrão de givenness de cada frase do texto, a partir da segunda sentença
76	lsa_span_mean	O desvio padrão do período de cada sentença do texto.
77	lsa_span_std	Desvio padrão do span da cada sentença do texto, a partir da segunda
78	content density	A densidade de conteúdo de um texto é calculada como o número de palavras de classe aberta (também denominadas palavras de conteúdo) dividido pelo número de palavras de classe fechada (ou palavras funcionais)
79	ratio_function_to_content_words	Proporção de palavras funcionais por palavras de conteúdo

Fonte: Elaborada pelo autor.

Tabela 19 – Features selecionadas pelo método CFS

1	noun_ratio	12	verbs_ambiguity	23	idade_aquisicao_1_25_ratio
2	pronoun_ratio	13	yngve	24	idade_aquisicao_55_7_ratio
3	verbs	14	std_noun_phrase	25	idade_aquisicao_25_4_ratio
4	negation_ratio	15	passive_ratio	26	imageabilidade_std
5	min_cw_freq	16	concretude_25_4_ratio	29	imageabilidade_55_7_ratio
6	first_person_pronouns	17	concretude_4_55_ratio	30	sentence_length_std_deviation
7	conn_ratio	18	familiaridade_std	31	verb_diversity
8	tmp_neg_conn_ratio	19	familiaridade_4_55_ratio	32	adj_mean
9	tmp_pos_conn_ratio	20	familiaridade_55_7_ratio	33	span_mean
10	adjectives_ambiguity	21	idade_aquisicao_mean	34	content_density
11	adverbs_ambiguity	22	idade_aquisicao_std		

Fonte: Elaborada pelo autor.

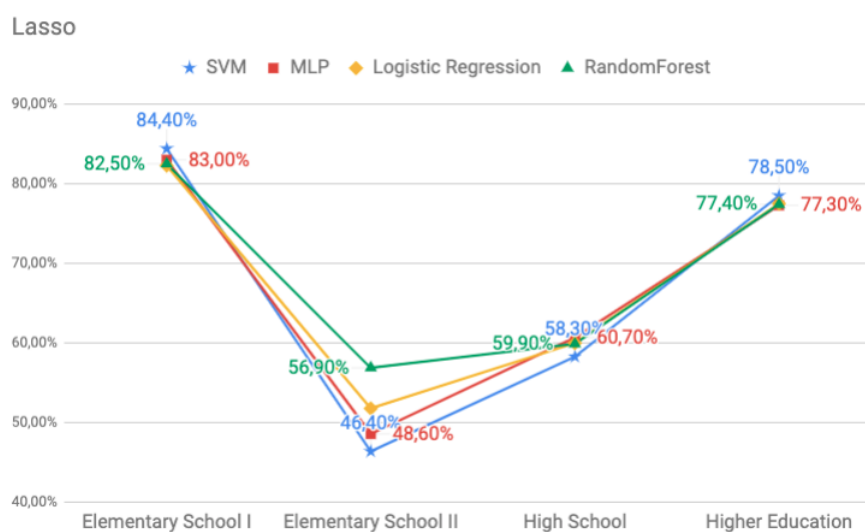
41 trouxemos os resultados divididos nas etapas escolares com seus F1 respectivos, usando os classificadores LR, RF, SVM e MLP para as 8 *features* estimadas pelo Lasso. Além dos experimentos envolvendo a seleção de features, também, realizamos a classificação com todas as features.

Tabela 20 – Resultados da classificação com as features selecionadas pelo CFS

Classificador	Ensino Fundamental I	Ensino Fundamental II	Ensino Médio	Ensino Superior	F-Measure (Weighted Avg.)
SVM	85,30%	60,90%	74,80%	83,90%	0.777
MLP	80,30%	63,10%	73,30%	83,30%	0.767
Logistic Regression	86,00%	63,10%	75,10%	84,10%	0.783
RandomForest	87,10%	67,60%	76,00%	85,00%	0.798

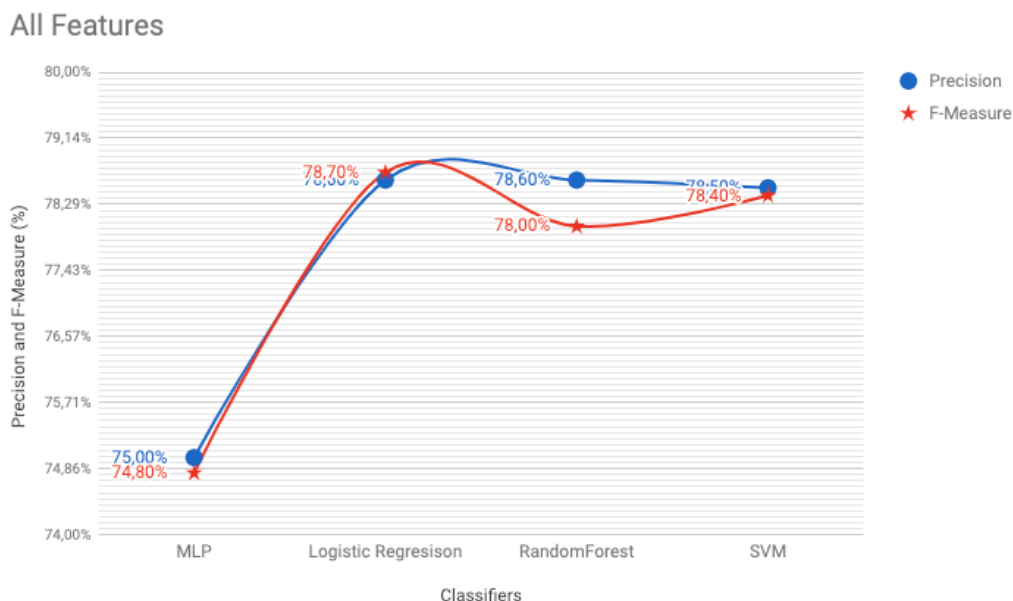
Fonte: Elaborada pelo autor.

Figura 41 – Resultados com as features selecionadas pelo Lasso, em quatro níveis escolares



Fonte: Elaborada pelo autor.

Figura 42 – Classificação usando todas as features do trabalho



Fonte: Elaborada pelo autor.

Além do CFS e do Lasso, foi aplicado o algoritmo Análise de Componentes Principais (em inglês, *Principal Component Analysis* (PCA)), implementado em Python, usando o Scikit-learn⁹. O PCA é uma técnica de extração de *features* que gera novas features com a combinação linear sobre as features de entrada. O PCA mapeia cada instância do conjunto de dados fornecido em um espaço dimensional D para um subespaço dimensional d tal que $d < D$. O conjunto de d novas dimensões geradas é chamado de Componentes Principais (ou em inglês, *Principal Components* (PC)) e cada componente principal é direcionado para a variância máxima excluindo a variância já contabilizada em todos os seus componentes anteriores (VASAN; SURENDIRAN, 2016).

Nas Tabela 21 e Tabela 22 trouxemos os 11 componentes principais, sendo que o PC1 possui o maior autovalor e o menor para o PC 11. Além disso, trouxemos o valor da proporção de variância e a variância acumulativa em cada PC, com suas respectivas cargas representativas sendo que foi aplicado um corte do módulo de $|0, 2|$ para as features com suas cargas.

Na Figura 43, observamos a diminuição da variância dos dados para cada componente principal respectivo (PC1, PC2, ... PC11). Além disso, para observar o PC1 e o PC2 em 2 dimensões, trouxemos a Figura 44 em 4 cores que representam as etapas de ensino e os pontos no gráfico com seus respectivos nomes de *features*. É possível observar que as classes mais visíveis neste gráfico são do Ensino Médio e do Ensino Superior. Porém, os pontos do Ensino Fundamental I e II podem estar sobrepostos sobre as outras etapas de ensino em uma visualização 2D de PCs.

⁹ <<https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>>

Tabela 21 – Resultado do PCA com autovalor, proporção de variância, variância acumulativa e as cargas representativas

	Autovalor	Proporção de Variância	Variância Acumulativa	Cargas Representativas
PC1	14,59863	0,17284	0,17284	-0,223sentence_length_standard_deviation +0,222imageabilidade_55_7_ratio +0,203verb_diversity
PC2	9,92897	0,11756	0,2904	+0,288verbs_ambiguity +0,247words_per_sentence +0,22 frazier +0,209all_mean +0,208yngve
PC3	7,57615	0,0897	0,3801	0,267noun_ratio -0,227trr -0,221adverbs -0,22familiaridade_55_7_ratio -0,215pronoun_ratio -0,213idade_aquisicao_55_7_ratio -0,209verbs -0,207yngve
PC4	6,27003	0,07423	0,45433	-0,267syllables_per_content_word +0,252passive_ratio +0,24mean_noun_phrase +0,228min_noun_phrase -0,222familiaridade_4_55_ratio +0,217max_noun_phrase +0,204std_noun_phrase -0,203first_person_pronouns
PC5	4,06223	0,0481	0,50243	-0,335idade_aquisicao_mean +0,311idade_aquisicao_4_55_ratio -0,271imageabilidade_mean -0,268idade_aquisicao_1_25_ratio -0,255adj_stem_ovl -0,246concretude_mean -0,24adj_arg_ovl -0,235adj_cw_ovl -0,221idade_aquisicao_std +0,213imageabilidade_1_25_ratio
PC6	3,3214	0,03932	0,54175	+0,275idade_aquisicao_mean-0, 262idade_aquisicao_4_55_ratio +0,252span_mean +0,246idade_aquisicao_1_25_ratio -0,229dep_distance +0,225givenness_mean -0,218concretude_mean -0,212adj_arg_ovl +0,212adj_std
PC7	2,78072	0,03292	0,57468	-0,379dep_distance -0,274imageabilidade_mean +0,249stem_ovl +0,238imageabilidade_1_25_ratio +0,226arg_ovl +0,203ratio_function_to_content_words
PC8	2,41166	0,02855	0,60323	-0,325adj_std -0,318span_std -0,308span_mean -0,29givenness_mean -0,243dep_distance -0,218stem_ovl

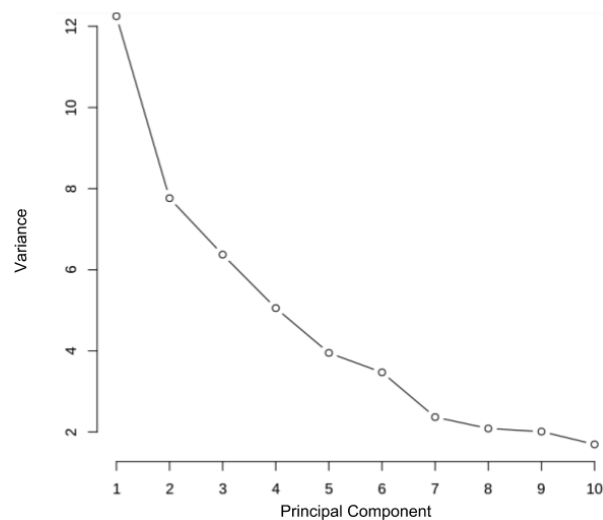
Fonte: Elaborada pelo autor.

Tabela 22 – Resultado do PCA com autovalor, proporção de variância, variância acumulativa e as cargas representativas

	Autovalor	Proporção de Variância	Variância Acumulativa	Cargas Representativas
PC9	2,04118	0,02417	0,6274	-0,312nouns_ambiguity -0,306concretude_25_4_ratio -0,286cw_freq +0,274sentence_length_max +0,248content_word_diversity +0,201familiaridade_25_4_ratio
PC10	1,94801	0,02306	0,65046	+0,378cau_neg_conn_ratio +0,365log_neg_conn_ratio +0,333trr +0,301add_neg_conn_ratio +0,26dep_distance -0,229cau_pos_conn_ratio -0,201add_pos_conn_ratio
PC11	1,70545	0,02019	0,67065	+0,363sentence_length_min +0,35concretude_25_4_ratio +0,283sentence_length_max +0,274familiaridade_mean +0,255content_word_diversity -0,252content_density +0,208familiaridade_25_4_ratio -0,202familiaridade_1_25_ratio

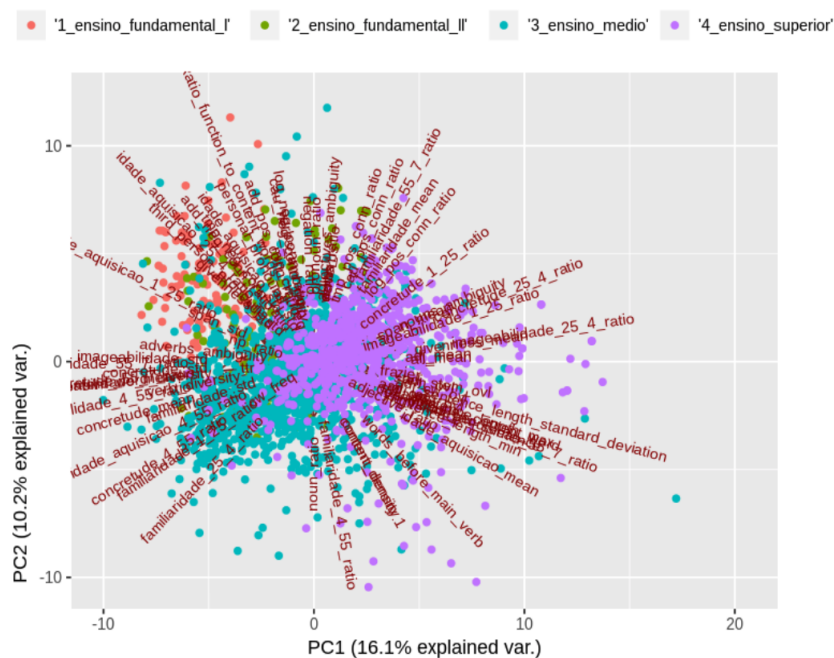
Fonte: Elaborada pelo autor.

Figura 43 – Gráfico de Declividade dos PC e Variância



Fonte: Elaborada pelo autor.

Figura 44 – Exemplo do PC1 e PC2 em 2 dimensões



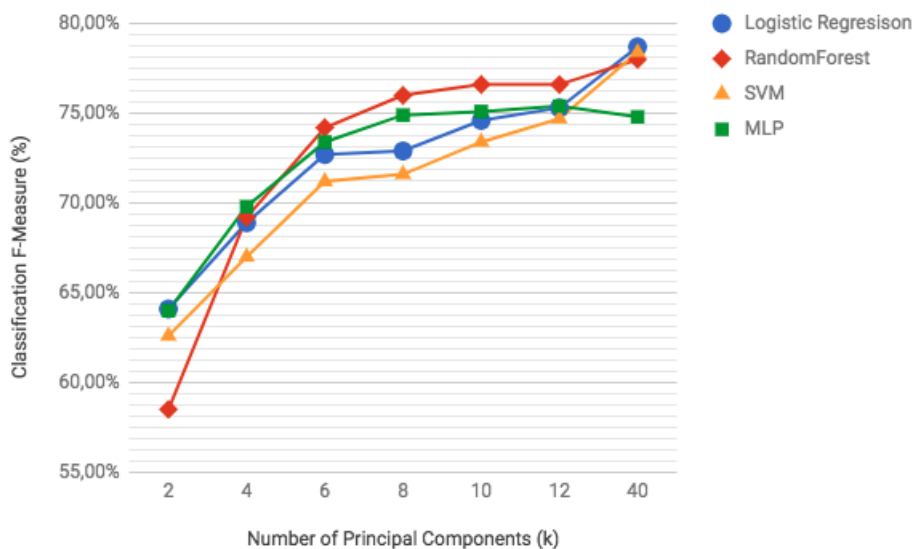
Fonte: Elaborada pelo autor.

Fizemos alguns experimentos variando o número de componentes principais de forma incremental de 2 em 2; o número de PCs é o valor de k e trouxemos os resultados dos classificadores Logistic Regression (LR), Random Forest (RF), SVM e MLP na Figura 45. Observamos no intervalo $2 < k \leq 6$ que todos os classificadores ficaram com resultados de F1 abaixo de 0.75. Enquanto que a partir do $k=8$, o Random Forest (RF) obteve o melhor resultado de F1, sendo superior a 0.75.

5.4 Métodos Tradicionais de Aprendizagem de Máquina: Avaliação Intrínseca e Extrínseca

Para avaliar a tarefa de classificação da complexidade textual em nosso cópuz, anotado com quatro etapas de ensino e com 79 métricas, quatro classificadores foram escolhidos, com base nos trabalhos relacionados, que foram revisados no Capítulo 3. Os algoritmos selecionados foram: SVM, MLP, LR e RF. Para a avaliação do melhor modelo, foi usada a validação cruzada com valor *10-folds*. Os melhores resultados foram do SVM, que alcançou uma média ponderada de F-Measure de **0.804**; o resultado do RF foi 0.794, da MLP foi de 0.698 e da LR foi de 0.802, caracterizando um empate técnico com o SVM.

Comparando as predições dos modelos com seleção de features via Lasso e CFS e treinado com todas as 79 *features*, foi possível observar que o desempenho dos modelos com seleção de features é inferior ao do modelo com todas as features.

Figura 45 – Comparativo de classificadores, variando o k dos Componentes Principais (F-Measure)

Fonte: Elaborada pelo autor.

Tabela 23 – Resultados da Classificação do SVM com todas as features

	Precisão	Precisão c/ Balanceamento	Recall	Recall c/ Balanceamento	F-Measure	F-Measure c/ Balanceamento
Ensino Fund I	81.60%	85.0%	91.2%	92.6%	0.861	0.886
Ensino Fund II	69.8%	75.0%	65.50%	75.7%	0.676	0.754
Ensino Médio	80.10%	80.5%	71.80%	69.1%	0.757	0.743
Ensino Superior	83.40%	81.6%	88.50%	85.2%	0.859	0.834

Fonte: Elaborada pelo autor.

Os resultados por nível escolar do classificador SVM com todas as *features* pode ser visto na Tabela 23. Como havia desbalanceamento da classe nível escolar, usamos o ClassBalancer (JAIN; KOTSAMPASAKOU; ECKER, 2018). Esse método reutiliza instâncias para que a soma total de pesos em todas as instâncias seja equilibrada. Desta forma, ficamos com 516,8 instâncias em cada classe.

Para avaliar a utilidade e robustez do melhor modelo treinado no grande corpus de textos didáticos, descrito na Seção 5.2, em prever a complexidade textual de REA, executamos uma avaliação extrínseca em dois conjuntos de REA: com gêneros diferentes do modelo treinado e com mesmo gênero textual (cf. Seção 5.2). Utilizamos o melhor modelo de classificação que foi o SVM com 79 *features*.

O *dataset 1* ou *D40* é composto por 40 textos do gênero textual didático; na avaliação de robustez, a média ponderada da *F-Measure* foi de 0.518. O *dataset 2* ou *D60* é composto por 60 REA de experimentos práticos e animações/simulações. Na avaliação de robustez, a média ponderada da *F-Measure* foi de 0.389.

Em uma análise detalhada dos textos disponíveis, verificamos que eles possuem muitos erros ortográficos e a anotação da Etapa Escolar estipulada pelos autores dos materiais que disponibilizaram na plataforma MEC-RED parecia equivocada para alguns REA. Sendo este o primeiro trabalho a avaliar a Etapa de Ensino de REA, antevemos novas pesquisas para validar a predição automática. Por exemplo, fazer uma correção gramatical nos textos e utilizar REA para os quais os metadados *peessoas que favoritam o recurso, e escala de estrelas sejam usados* nos indique que os recursos são usados nas escolas.

Há também algumas melhorias para a tarefa como inclusão de novas métricas de complexidade, como as 17 métricas do trabalho de Graesser e McNamara (2011), citadas na Seção 5.3, que não foram incluídas neste estudo atual, pois não foram implementadas para o Português do Brasil. Esta melhoria, em especial será deixada para trabalhos futuros.

5.5 Métodos de Deep Learning: Avaliação Intrínseca e Extrínseca

Para avaliar a tarefa de classificação da complexidade textual em nosso córpus (Seção 5.2) usando Deep Learning, selecionamos os modelos de BiLSTM com *Self Attention*, com base nos trabalhos relacionados, que foram revisados no Capítulo 3.3.

Para a avaliação do melhor modelo foram separados um conjunto de treino e teste de 80% e 20%, respectivamente. Os conjuntos foram estratificados de forma proporcional para cada uma das 4 classes, como é mostrado na Figura 46.

Para implementação da BiLSTM, usamos as bibliotecas PyTorch¹⁰ (versão GPU) e AllenNLP¹¹. Também, foram usadas a GPU NVIDIA GeForce e a biblioteca Cuda¹² para os experimentos. Em nossos experimentos, usamos o conjunto de embeddings do NILC Embeddings¹³ na camada *Token Embedder* (Figura 47), com 300 dimensões e modelo Skip-gram: Word2Vec¹⁴, Wang2Vec¹⁵, FastText¹⁶ e GloVe¹⁷. Também, utilizamos o modelo Elmo (PETERS *et al.*, 2018) em Português, treinado com a Wikipédia, disponível em <<https://allennlp.org/elmo>> e que foi desenvolvido pela Universidade Federal de Goiás (UFG).

Na Figura 47, ilustramos a arquitetura usando BiLSTM com Word Embeddings e o Mecanismo de Atenção. Os parâmetros envolvidos nos modelos foram *bath_size=4*, *learning rate (lr)=0.001*, 25 épocas, vocabulário máximo de 55.430 (*max_vocab_size*) e sequência máxima

¹⁰ <<https://pytorch.org/>>

¹¹ <<https://allennlp.org/>>

¹² <<https://developer.nvidia.com/cuda-downloads>>

¹³ <<http://www.nilc.icmc.usp.br/embeddings>>

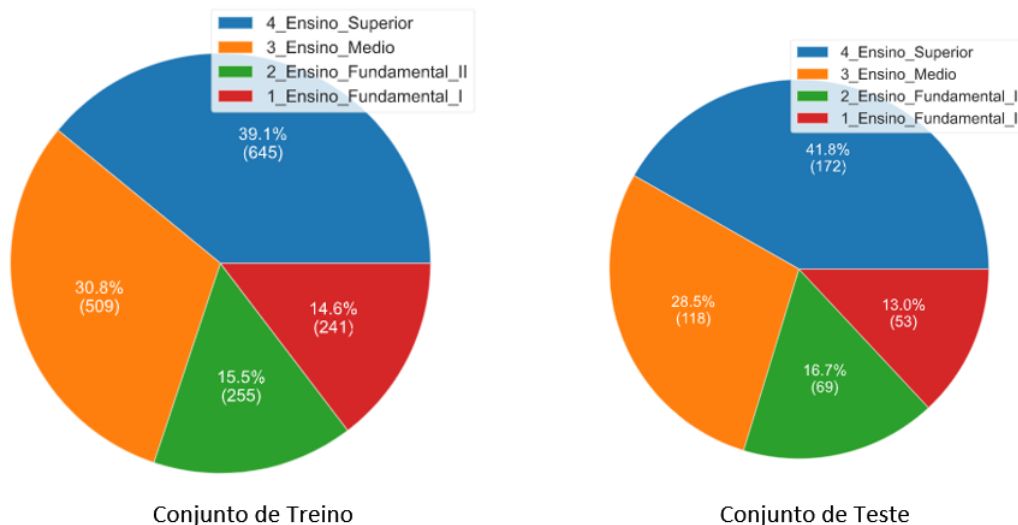
¹⁴ <http://143.107.183.175:22980/download.php?file=embeddings/word2vec/skip_s300.zip>

¹⁵ <http://143.107.183.175:22980/download.php?file=embeddings/wang2vec/skip_s300.zip>

¹⁶ <http://143.107.183.175:22980/download.php?file=embeddings/fasttext/skip_s300.zip>

¹⁷ <http://143.107.183.175:22980/download.php?file=embeddings/glove/glove_s300.zip>

Figura 46 – Conjunto estratificado de treino e teste



Fonte: Elaborada pelo autor.

Tabela 24 – Resultados dos modelos na avaliação intrínseca

Avaliação Intrínseca	
Modelo	F1 (weighted)
BiLSTM+Word2Vec+ Self.Attention	0,9375
BiLSTM+Wang2Vec+ Self.Attention	0,9441
BiLSTM+Wang2Vec	0,9220
BiLSTM+Wang2Vec+Self.Attention+Elmo	0,9443
BiLSTM+GloVe+Self.Attention	0,9472
BiLSTM+FastText+Self.Attention	0,9468

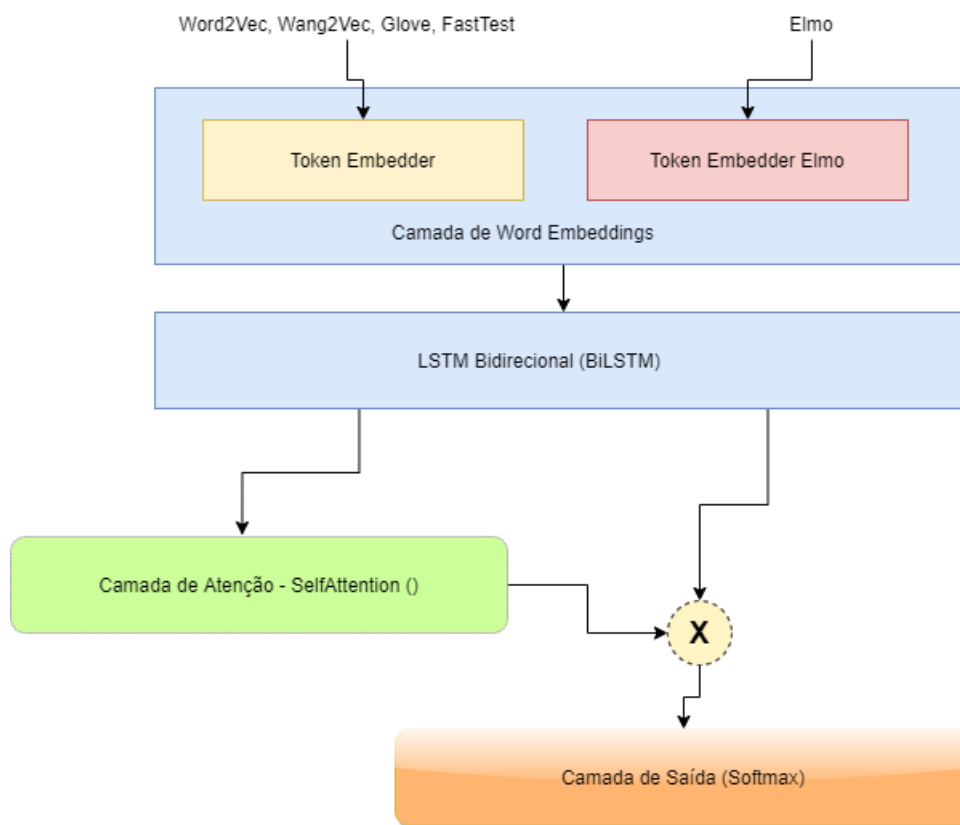
Fonte: Elaborada pelo autor.

de 1281 (`max_seq_len`). Utilizamos os módulos *Dataset Reader*, *Token Embedders*, *Token Indexer* (`elmo_indexer`) do AllenNLP. A função de custo (*loss*) usada foi *CrossEntropyLoss* e otimizador *Adam*. A quantidade de parâmetros treináveis variam de acordo com os modelos. Avaliamos o modelo BiLSTM Wang2Vec Self.Attention Elmo que resultou em 109.546.298 milhões de parâmetros treinados.

Os melhores resultados na avaliação intrínseca (Tabela 24) foram do BiLSTM+Wang2Vec+Self.Attention, BiLSTM+Wang2Vec+Self.Attention+Elmo, BiLSTM+GloVe+Self.Attention, BiLSTM+FastText+Self.Attention, que alcançaram uma média ponderada de F-Measure de **0,94**; sendo eles os modelos selecionados para próxima avaliação (avaliação extrínseca) e na avaliação intrínseca individualizada por predição das classes das etapas de ensino.

Na Tabela 25 trouxemos os melhores resultados dos modelos da avaliação intrínseca para avaliarmos a classificação individual dos modelos por Etapa Escolar. Nesse caso, todos os mode-

Figura 47 – Arquitetura usando BiLSTM com Word Embeddings e Mecanismo de Atenção



Fonte: Elaborada pelo autor.

Tabela 25 – Resultados por classe dos modelos Deep Learning na avaliação intrínseca

Avaliação Intrínseca - Resultado por Classe					
	BiLSTM Self.Attention Wang2Vec Elmo	BiLSTM Wang2Vec	BiLSTM Wang2Vec Self.Attention	BiLSTM FastText Self.Attention	BiLSTM GloVe Self.Attention
Etapa	F1	F1	F1	F1	F1
Escolar	(weighted)	(weighted)	(weighted)	(weighted)	(weighted)
EFI	0,99	0,92	0,94	0,93	0,96
EFII	0,94	0,88	0,92	0,94	0,96
EM	0,91	0,91	0,93	0,94	0,92
ES	0,95	0,95	0,97	0,97	0,96

Fonte: Elaborada pelo autor.

los que usam um mecanismo de atenção (*Self Attention*) se sobressaíram em relação aos demais. Sendo que o modelo BiLSTM+FastText+Self.Attention obteve um resultado de F1 de 0,94 e 0,97 no Ensino Médio e Ensino Superior respectivamente, mostrando ser o melhor classificador comparado com todos os outros modelos. Embora o modelo BiLSTM+Wang2Vec+Self.Attention+Elmo teve um resultado de F1 de 0,99 no Ensino Fundamental I, sendo o maior F1 de classificação das etapas de ensino entre os modelos. Por fim, o modelo BiLSTM+GloVe+Self.Attention teve um resultado superior entre os demais com F1 de 0,96 no Ensino Fundamental II.

Tabela 26 – Resultados dos modelos de Deep Learning na avaliação extrínseca

Avaliação Extrínseca		
	D60	D40
Modelo	F1 (weighted)	F1 (weighted)
BiLSTM+Wang2Vec+ Self.Attention	0,35	0,58
BiLSTM+Wang2Vec+Self.Attention+Elmo	0,37	0,59
BiLSTM+GloVe+Self.Attention	0,33	0,56
BiLSTM+FastText+Self.Attention	0,25	0,54

Fonte: Elaborada pelo autor.

Também, realizamos uma avaliação extrínseca usando os datasets D1 e D2 (Seção 5.2), com os modelos Deep Learning melhor avaliados na avaliação intrínseca. Nesse caso, os melhores resultados foram do modelo BiLSTM+Wang2Vec+Self.Attention+Elmo com F1 de 0,37 no D60 e F1 de 0,59 no D40.

5.6 Conclusões e Discussões

O principal objetivo destes estudos foi explorar os métodos de aprendizagem de máquina tradicional, Deep Learning e Seleção de Features para as tarefas de complexidade textual para o Português do Brasil, no cenário de REA. Ainda, realizamos uma avaliação intrínseca e extrínseca para prever 4 etapas de ensino (Ensino Fundamental I, Ensino Fundamental II, Ensino Médio e Ensino Superior). Foi necessário a construção de um grande corpus para modelar a tarefa de complexidade textual e assim avaliar os modelos com textos de gêneros textuais didáticos e outros como animação/simulação e experimento prático, categorizados como gênero instrucional.

Neste estudo, respondemos duas questões relacionadas ao aprendizado de máquina e seleção de features:

- Qual o impacto da predição da complexidade textual com o uso de métodos de seleção de features ?
- O uso de algoritmos de Deep Learning, com Redes Neurais Recorrentes, Word Embeddings e Mecanismo de Atenção, com alterações na arquitetura e sem o uso de *features* linguísticas teriam melhor desempenho do que os algoritmos de aprendizado de máquina tradicional, que usam engenharia de features ?

O uso do métodos de seleção de *features* demonstrou que a escolha das *features* e a remoção delas pode ter um impacto negativo na avaliação da F1 nos modelos preditivos.

Em relação aos modelos de aprendizado de máquina tradicional, os resultados da avaliação extrínseca do Dataset com 60 REA de experimentos práticos e animações/simulações demonstrou um resultado da classificação de REA com uma média de F1 próximo dos algoritmos

de Deep Learning. Assim, o melhor modelo de aprendizado de máquina tradicional foi o SVM com 79 features linguísticas que teve o resultado de F1 de 0.38 e o melhor modelo de Deep Learning foi BiLSTM+Wang2Vec+Self.Attention+Elmo com F1 de 0,37. Enquanto no Dataset com 40 textos do gênero textual didático os resultados dos modelos de Deep Learning mostraram um resultado de F1 de 0,59 e os modelos tradicionais tiveram um resultado de F1 de 0,51. Na avaliação intrínseca entre os modelos o melhor classificador foi o modelo de Deep Learning BiLSTM+GloVe+Self.Attention que obteve F1 de 0,9472 e para os modelos de aprendizado de máquina tradicional o melhor foi o SVM com todas as 79 features com F1 de 0,804.

5.7 Considerações Finais

Em resumo, este capítulo explorou métodos automáticos para prever a etapa de ensino da plataforma MEC-RED, embora este trabalho possa ser utilizado para outras plataformas, por exemplo, a plataforma de Educação em Saúde e Diversidade da USP¹⁸ que possui diversos materiais educativos (planos de aula, pesquisas, artigos científicos, dentre outros). Foi criado um grande corpus para modelar a tarefa de complexidade textual e assim avaliar o modelo com textos do gênero textual didático e outros como animação/simulação e experimento prático, do gênero instrucional. A avaliação intrínseca mostrou um ótimo desempenho para o modelo treinado (F-measure de 0.9472) com métodos Deep Learning e modelo de aprendizado de máquina tradicional com F-measure de 0.804). Foi feita uma seleção de *features* usando 3 métodos de redução de dimensionalidade que foram comparados com o modelo treinado com todas as 79 *features*, além de usarmos um método de balanceamento de classes para os métodos tradicionais de aprendizado de máquina. Os recursos estão disponibilizados no GitHub do projeto (<https://github.com/gazzola/corpus_readability_nlp_portuguese>). Por fim, foi possível observar o impacto dos gêneros textuais na complexidade textual para prever a Etapa Escolar. A avaliação extrínseca, usando recursos da MEC-RED mostrou que a tarefa é difícil e que merece ser revisitada trazendo mais REA para serem incorporados no modelo. A incorporação de dados do domínio no modelo de predição foi explorada no Capítulo 6.

¹⁸ <<https://prceu.usp.br/repositorio-uspdiversidade>>

UMA ABORDAGEM PARA COMPLEXIDADE TEXTUAL DE REA USANDO MULTI-TASK LEARNING

6.1 Considerações Iniciais

Este capítulo apresenta um estudo da tarefa de complexidade textual usando *Multi-task Learning* (MTL), com o objetivo de construir um modelo com melhor generalização para avaliar a complexidade dos textos da MEC-RED. Como tarefas auxiliares, usamos textos transcritos de áudios gravados de adolescentes e textos da própria MEC-RED (domínio alvo); como tarefa principal, usamos a predição da complexidade textual treinada com o corpus de textos de Estágios Escolares descrito no Capítulo 4 e retomado aqui neste capítulo.

Este estudo foi motivado por um corpus recém-criado de narrativas faladas e transcritas que foi coletado para avaliar o desenvolvimento da autorregulação comportamental de adolescentes. O corpus de narrativas, denominado corpus do projeto Adole-sendo¹, contém 237 narrativas transcritas de fala espontânea da tarefa de recontar histórias infantis a partir de sua própria memória. Adolescentes de sete séries consecutivas (da quarta à nona série do Ensino Fundamental e primeira série do Ensino Médio) realizaram a tarefa.

Participamos do projeto Adole-sendo, coordenado pela Profa. Dra. Sabine Pompéia da UNIFESP, desde setembro de 2019, em várias etapas:

1. na remodelagem da tarefa de anotação linguística de disfluências da fala;
2. na remodelagem da geração da narrativa final, sem disfluências, para ser usada na extração de métricas de complexidade linguística;

¹ <https://adole-sendo.info/>

3. na extração automática de métricas linguísticas, usando o NILC METRIX²; e
4. na criação de classificadores de complexidade, tanto das 7 séries do projeto Adole-sendo, quanto do classificador de interesse deste trabalho de doutorado, de etapas de ensino da plataforma MEC-RED. Os detalhes do projeto Adole-sendo são apresentados na Seção 6.3.

A remodelagem (1) citada acima foi motivada pela necessidade de realizar uma anotação de disfluências seguindo uma metodologia adequada de anotação linguística, visando à criação de datasets para treinamento de preditores via aprendizado de máquina (PUSTEJOVSKY; STUBBS, 2012). A remodelagem (2) incluiu a automatização da geração de uma narrativa sem marcas de disfluências, diferente da proposta inicial de fazê-la de forma manual, o que dispndia mais tempo e também era passível de erros humanos.

Na análise de dados mais recente da Plataforma MEC-RED, realizada em setembro de 2020, 86% dos recursos da plataforma não possuíam classificação por etapa de ensino, dificultando a sua localização, o seu uso e sua expansão. Além disso, na avaliação dos recursos da MEC-RED realizada em 2019, 100% dos aplicativos móveis e 99% dos jogos não apresentavam informação sobre a etapa de ensino, inviabilizando a compilação de um grande *córpus* balanceado de REA para o treinamento de um preditor de complexidade textual. Essa foi a razão da busca de outros *córpus* com similaridade linguística de alguns tipos de textos da MEC-RED e com etapas de ensino previamente anotadas para compor o dataset de treinamento dos classificadores descritos neste capítulo.

As seções a seguir são organizadas da seguinte maneira. Na seção 6.2 trouxemos as motivações que levaram à avaliação da abordagem MTL com textos transcritos. Na seção 6.3 são apresentados os detalhes dos três *corpus* utilizados nesta pesquisa. Na seção 6.4 são apresentadas as *features* utilizadas com suas descrições. Na seção 6.5, apresentamos os experimentos realizados, as arquiteturas neurais desenvolvidas, mostrando os detalhes das abordagens de aprendizagem avaliadas e os resultados da avaliação intrínseca e extrínseca de todos os modelos treinados.

6.2 Avaliação da complexidade textual usando MTL com textos transcritos

O estudo final deste projeto de doutorado foi a avaliação da complexidade textual por meio de uma abordagem de MTL com o compartilhamento de parâmetros das camadas ocultas (CARUANA, 1997), chamada em inglês de *hard parameter sharing*. Foram treinados três modelos MLP que compartilham todos os parâmetros em suas camadas ocultas, inspirados

² <https://simpligo.sidle.al/nilcmatrixdoc>

pelos trabalhos de [Gonzalez-Garduno e Sogaard \(2017\)](#) e [Leal et al. \(2020\)](#). A tarefa principal é um classificador de complexidade textual de baixa granularidade de textos das quatro etapas da educação no Brasil (Ensino Fundamental I e II, Ensino Médio e Ensino Superior). As tarefas auxiliares (secundária e de domínio) são um classificador de complexidade textual de granularidade mais fina (ou alta) (séries escolares) baseado no corpus Adole-sendo e em outro classificador de complexidade textual de baixa granularidade de etapas escolares, baseado em um pequeno conjunto de recursos de REA da plataforma MEC-RED, respectivamente.

A razão da escolha da abordagem MTL foi evitar que o modelo treinado corresse o risco de *overfitting*. Nas abordagens MTL quanto mais tarefas são treinadas simultaneamente, menor a chance de *overfitting* da tarefa principal ([RUDER, 2017b](#)). Como a plataforma MEC-RED possui textos do gênero didático (apresentações, livros digitais, website externo de um curso, planos de aula) que têm o objetivo de ensinar novos conhecimentos, mas também textos do gênero instrucional (animação/simulação, aplicativo móvel, jogos, experimento prático e software educacional), que informam como fazer uma tarefa, acreditamos que a abordagem MTL é mais adequada ao nosso cenário de pesquisa, pois pode agregar datasets previamente anotados de vários gêneros para a criação de um modelo de complexidade textual da plataforma MEC-RED.

Há na literatura alguns trabalhos que tratam da avaliação da complexidade textual de transcrições, por exemplo, (i) [Chen e Zechner \(2011\)](#) que analisou transcrições faladas em termos de complexidade sintática, (ii) [Crossley e McNamara \(2013\)](#) que as analisou para o desenvolvimento de técnicas de pontuação automática usando índices automatizados retirados das ferramentas Coh-Metrix³, CPIDR (*Computerized Propositional Idea Density Rater*) e LIWC⁴, e (iii) [Vajjala e Meurers \(2014\)](#) que propôs uma abordagem de classificação para identificar programas de TV para diferentes grupos de idade, com base em features linguísticas derivadas de pesquisas sobre complexidade textual, proficiência em uma segunda língua e pesquisa psicolinguística. Mas neste trabalho de doutorado, exploramos a relação entre três tarefas de complexidade textual, sendo que uma delas usou transcrições, aprendendo conjuntamente a prever a complexidade textual.

6.3 Datasets

No estudo descrito neste capítulo, foram usados três datasets: (i) o conjunto de dados com 2.067 extratos de textos didáticos usados nas quatro etapas do Sistema Educacional Brasileiro que também foi utilizado nos estudos descritos no Capítulo 4; (ii) o dataset do domínio da MEC-RED, composto por 60 REA de experimentos práticos, animações/simulações e software educacional mais 40 REA de textos didáticos, para treinamento e mais

³ <http://cohmetrix.com/>

⁴ <https://liwc.wpengine.com/>

Tabela 27 – *Córpus de Textos Didáticos da Língua Portuguesa*

	Elementary School I	Elementary School II	High Education	Higher Education	Total
Sources	TextBooks + PSFL	TextBooks + SAEB + Ebooks	Wikilivros + ENEM 2015, 2016, 2017	Wikilivros	
# Docs	296	325	627	819	2.067
# Sents	5.258	5.598	9.316	10.416	30.588
ASL	20.58	24.31	29.81	39.15	31.35
Types	63.081	75.698	134.788	177.054	450.621
Tokens	101.911	127.705	241.267	342.534	813.417
TTR	0.618	0.592	0.558	0.516	0.553

Fonte: Elaborada pelo autor.

Foram utilizados dois conjuntos de textos de recursos da MEC-RED para treinamento: um dataset chamado de Dataset 1 (40 REA) composto por textos de recursos do tipo animação/-simulação, experimento prático e software educacional e outro dataset chamado de Dataset 2 (60 REA) de recursos da MEC-RED do tipo apresentações, livros digitais, website externo, planos de aula e textos que são considerados do gênero de textos didáticos. A Tabela 28, apresenta as características destes dados de treinamento e também do Dataset 3, utilizado para teste, composto de textos de recursos da MEC-RED que foram previamente classificados por uma editora quanto à etapa do Ensino, por apresentar mais garantias quanto a sua classificação.

6.3.1 *Córpus de Textos Didáticos dos Estágios Escolares do Sistema Educacional Brasileiro*

Compilamos um grande *córpus* de textos utilizados em quatro diferentes etapas de ensino do Sistema Educacional Brasileiro: Ensino Fundamental I (1^o ao 5^o ano), Ensino Fundamental II (6^o ao 9^o ano), Ensino Médio e Ensino Superior (Tabela 27). Essas quatro etapas de ensino são as mesmas utilizadas na MEC-RED para classificar os REA nos Estágios Escolares, embora a plataforma também permite pesquisar os REA pela etapa Educação Infantil, mas não estamos usando esse estágio em nosso trabalho. Este *córpus* já foi descrito em detalhes na Seção 4.

Esse *córpus* compreende 2.067 extratos (min = 300 palavras, max = 596 palavras, média = 448) dos textos amostrados. Como pode ser visto na Tabela 27, nosso *córpus* não é balanceado, pois o número de textos do Ensino Médio possui aproximadamente o dobro da quantidade do Ensino Fundamental I e do Ensino Fundamental II, por exemplo. Desta forma, métodos de *undersampling* e *oversampling* foram utilizados nesta pesquisa com a abordagem MTL.

6.3.2 *Córpus de REA*

Foram utilizados dois conjuntos de textos de recursos da MEC-RED para treinamento: um dataset chamado de Dataset 1 (40 REA) de recursos da MEC-RED do tipo apresentações,

Tabela 28 – Descrição do Córpus de Textos dos Recursos da MEC-RED

	Dataset 1 (D1)	Dataset 2 (D2)	Dataset 3 (D3)
# Docs	40	60	40
# Sent	675	1382	775
ASL	27.43	18.77	18.73
Type	8511	11572	6650
Token	15527	22110	12159
TTR	0.548	0.523	0.546

Fonte: Elaborada pelo autor.

Tabela 29 – Corpus Adole-sendo

Etapas Escolares	Ano	# Texts	# Sent	# Types	# Tokens	TTR	ASL
Ensino Fundamental I	4º	8	176	1045	2711	0.385	15.403
	5º	26	502	3043	7763	0.392	15.464
	6º	62	1067	7405	19566	0.359	18.286
Ensino Fundamental II	7º	39	908	5383	14899	0.361	16.409
	8º	14	310	1915	5512	0.347	17.781
	9º	55	650	5784	14039	0.412	21.598
Ensino Médio	1º	33	512	4088	10736	0.381	20.969
	TOTAL	237					

Fonte: Elaborada pelo autor.

livros digitais, website externo, planos de aula e textos que são considerados do gênero de textos didáticos, e outro chamado de Dataset 2 (60 REA) composto por textos de recursos do tipo animação/simulação, experimento prático e software educacional. A Tabela 28, apresenta as características destes dados de treinamento e também do Dataset 3, utilizado para teste, composto de textos de recursos da MEC-RED que foram previamente classificados por uma editora quanto à etapa do Ensino, por apresentar mais garantias quanto a sua classificação.

6.3.3 Córpus do Projeto Adole-sendo

O projeto Adole-sendo⁵ visa avaliar fatores biopsicossociais que afetam o desenvolvimento da autorregulação⁶ do comportamento de adolescentes de acordo com medidas de maturação biológica, ao contrário da maior parte da literatura, que considera apenas a idade cronológica como fator nesse desenvolvimento. O estudo prevê 330 participantes de 9-15 anos de uma ampla gama de níveis socioeconômicos, sendo que atualmente já foram coletados dados de 237 participantes, de acordo com a distribuição mostrada na Tabela 29.

Os dados deste projeto são compostos de narrativas transcritas, obtidas da tarefa de contação de histórias infantis (João e o Pé de Feijão ou da Chapeuzinho Vermelho) da própria

⁵ <https://adole-sendo.info/>

⁶ A autorregulação comportamental é a habilidade de poder monitorar e modular sentimentos, a cognição e o comportamento para adaptar-se às demandas cognitivas e sociais presentes na vida dos adolescentes.

memória (chamada aqui de reconto), por cada adolescente.

O processo de criação do corpus do projeto Adole-sendo se deu em três etapas:

1. transcrição dos áudios do reconto e anotação de 6 fenômenos linguísticos no nível das palavras;
2. anotação linguística de 5 tipos de disfluências em trechos de palavras, que foi realizada por pares de anotadores para avaliação da concordância da anotação;
3. geração automática da narrativa sem as disfluências e sem os fenômenos anotados na transcrição.

Estas etapas são descritas em detalhes nas próximas seções.

6.3.3.1 Transcrição de Áudios de Recontos de Histórias Infantis

As transcrições foram obtidas a partir de áudios dos recontos, gravados durante as baterias de medidas e entrevistas realizadas com as crianças e adolescentes das escolas da cidade de São Paulo (obtidas por amostra de conveniência).

O processo de transcrição foi baseado nos trabalhos de [Marcuschi \(1986\)](#) e [Gago \(2002\)](#) e consistiu na transcrição integral dos áudios em si, seguindo a ortografia padrão do português e segmentação em unidades sentenciais de acordo com as regras da língua, com base em prosódia e regras de sintaxe de textos escritos. Nas transcrições foram incluídos, também, erros sintáticos, marcadores discursivos, repetições de palavras, autocorreções e palavras interrompidas, sem modificação. Variações de pronúncia não foram transcritas. Nesta etapa, também foram anotados os 6 fenômenos linguísticos no nível das palavras, descritos abaixo:

1. Repetições de palavras intencionais, para dar ênfase, transcritas entre os símbolos ();
2. Palavras interrompidas, transcritas com o símbolo (.) unido à palavra interrompida;
3. Pausas
 - curtas entre palavras diferentes, transcritas com o símbolo (.) entre essas palavras;
 - curtas no meio de palavras, transcritas com o símbolo : dentro da palavra que recebeu pausa;
 - longas entre palavras diferentes, transcritas com o símbolo (...) entre essas palavras;
4. Pausas preenchidas e som prolongado, transcritas com o símbolo _ unido à palavra com som prolongado;

6.3.3.2 Anotação Linguística de Disfluências

Após o término da etapa da transcrição, os textos obtidos foram importados para a plataforma web Inception (KLIE *et al.*, 2018), para anotação de 5 fenômenos de disfluências, descritos em um manual de anotação:

1. Marcadores Discursivos, anotados com a etiqueta MD. São palavras que funcionam primariamente como uma unidade estrutural da língua falada. Ex: “af”, “daí”, “tá”, “né”, “assim”, “bom”, “meio que”, “beleza”, “tipo”. Também foram anotados os marcadores “E” e “Então” iniciando uma oração;
2. Comentários não relacionados com a história, anotados com a etiqueta COM;
3. Repetições de palavras não-intencionais, anotados com as etiquetas ORI_REP (para o trecho original) e REP (para a repetição do trecho original);
4. Autocorreções, anotados com as etiquetas ORI_AC (trecho original) e AC (correção do trecho original);
5. Pausas preenchidas, como eh, ah, hm, hum, anotadas com a etiqueta PP.

Após a anotação piloto, seis anotadores divididos em 3 duplas receberam, cada dupla, um terço das transcrições e anotaram de forma independente para o cálculo do Kappa de Cohen e da Medida Alpha de Krippendorff. A anotação é composta de 2 tarefas: (i) encontrar e segmentar corretamente o fenômeno e (ii) rotular corretamente o fenômeno delimitado.

Usamos 2 medidas de concordância, pois a medida Kappa de Cohen só analisa os segmentos (*chunks*) que o par segmentou de forma igual, descartando as anotações incompletas ou faltantes de uma anotação. A anotação obteve um Kappa alto (0,94 para todos os pares), indicando que os anotadores não tem dúvidas em distinguir os fenômenos, mas podem, ainda, ter problemas em encontrar os fenômenos e em os segmentar corretamente. Por esta razão, escolhemos a medida Alpha de Krippendorff que analisa a anotação caractere a caractere da narrativa, penalizando se um anotador ultrapassou um caractere (ou mais caracteres) ou se esqueceu de anotar um dado fenômeno. O valor da medida Alpha varia de 0-1 e pode assumir valores negativos. Na anotação realizada, ele foi negativo para todas as duplas (- 1.05, -0,92, -0.65), apontando que os anotadores tiveram dificuldades em seguir o manual de anotação ou que adotaram regras diferentes, consistentemente. A plataforma Inception, que calcula as medidas de concordância, informa ainda quantas anotações são distintas, incompletas (faltando a segmentação e anotação) e concordantes na segmentação. A medida Alpha de Krippendorff mostrou que o manual deveria ser revisto e acordado entre os pares de anotadores para se prosseguir com a etapa de adjudicação (*curation*), e se criar a anotação *gold standard* do corpus. O corpus foi adjudicado pelo anotador mais experiente do par, com discussão dos casos mais difíceis.

6.3.3.3 Geração Automática da narrativa sem disfluências

As anotações linguísticas foram computadas para serem usadas nas análises das narrativas. Uma narrativa modificada, sem as 6 anotações realizadas durante a transcrição e sem as 5 anotações de disfluências, foi gerada automaticamente. As narrativas foram então processadas pela plataforma NILC Metrix, que extrai automaticamente 182 métricas textuais para avaliação da complexidade textual, sendo detalhada na Seção 6.4.

6.4 Features

Todas as *features* utilizadas neste estudo foram extraídas do NILC Metrix⁷, disponível na plataforma Simpligo que é composta por um conjunto de métodos e ferramentas de suporte à simplificação de textos em português do Brasil. O NILC Metrix extrai automaticamente 182 métricas textuais, abrangendo vários níveis de análise linguística (lexical, sintática, semântica e textual). É baseado nas ferramentas Coh-Metrix-Port (ALUÍSIO *et al.*, 2010) e Coh-Metrix-Dementia (ALUÍSIO; CUNHA; SCARTON, 2016), e em 24 métricas psicolinguísticas criadas a partir de um repositório de 26.874 palavras em português brasileiro anotadas com valores de imageabilidade, concretude, familiaridade e idade de aquisição (SANTOS *et al.*, 2017b). Além disso, 39 métricas sintáticas também foram desenvolvidas para extrair voz passiva e outras informações de sentenças e orações, utilizando o parser PALAVRAS (BICK, 2000). Finalmente, cinco fórmulas clássicas de complexidade textual (Índice de Flesch, Índice de Inteligibilidade Dale-Chall, Gunning Fog Index, Índice de Brunet, Estatística de Honoré) também estão disponíveis.

6.4.1 Features usadas no *córpus de textos didáticos e nos conjuntos de dados da MEC-RED*

Para o *córpus de textos didáticos* e para os três conjuntos de recursos da MEC-RED (D1, D2 e D3), apresentados na Seção 6.3.2, o conjunto de métricas que utilizamos consiste em 79 métricas lexicais, morfossintáticas, sintáticas, textuais (anáforas e correferência) e psicolinguísticas disponíveis no NILC Metrix. Elas estão agrupadas naquelas relacionadas a palavras, sentenças e conexões entre sentenças, seguindo o trabalho de Graesser, McNamara e Kulikowich (2011), e detalhados abaixo.

6.4.1.1 Palavras

O conhecimento de um grande vocabulário tem impacto no tempo de leitura e também na compreensão textual. Assim, é importante capturar as várias características das palavras que são relevantes para a melhoria da leitura e para ajudar na compreensão para poder diferenciar

⁷ <https://simpligo.sidle.al/>

crianças que estão começando os primeiros anos escolares dos leitores mais hábeis das séries mais avançadas (GRAESSER; MCNAMARA; KULIKOWICH, 2011). Esse grupo consiste em 49 métricas (Tabela 30), dentre as quais estão as métricas de tamanho de palavras (em sílabas), Part of Speech (PoS), frequências de palavras, escores psicolinguísticos⁸ como concretude, imageabilidade, familiaridade e idade de aquisição (AoA). A imageabilidade da palavra é a facilidade e a velocidade com que uma palavra evoca uma imagem mental; concretude é o grau com que as palavras se referem a objetos, pessoas, lugares ou coisas que podem ser experimentadas pelos sentidos; familiaridade é a estimativa do número de vezes que uma palavra é encontrada por indivíduos em sua forma escrita ou falada, e AoA é a estimativa da idade em que uma palavra foi aprendida. Também compõem esse grupo medidas que avaliam a polissemia, que se refere ao número de sentidos de uma palavra (neste grupo há o cômputo da ambiguidade das palavras de conteúdo – substantivos, verbos, adjetivos e advérbios). Teoricamente, a ambiguidade é um fator que aumenta a complexidade das palavras, embora palavras raras (que são mais complexas) possuem menos sentidos e as palavras mais frequentes sejam as com mais sentidos. A lista das 49 métricas é mostrada abaixo e suas descrições podem ser encontradas no manual do NILC Metrix⁹.

6.4.1.2 Sentenças

Esse grupo de 13 métricas está relacionado com a sintaxe que avalia os grupos de palavras e sua organização na forma de uma estrutura de árvore. É fácil entender que sentenças curtas, com poucas palavras por sintagmas nominais, poucas palavras antes do verbo principal e poucos conectivos lógicos (e, ou, não) são mais fáceis de se entender. A proporção de sentenças na voz passiva em um texto também causa dificuldade para a compreensão e é computada, juntamente com a Complexidade de Yngve que mede o quanto uma árvore sintática, que naturalmente se ramifica para a direita (Português e Inglês tem essa característica), se desvia do padrão de ramificação. A Complexidade de Frasier é outra medida que avalia uma árvore sintática para julgar a complexidade de uma oração. A última métrica de árvores sintáticas é a distância em árvores de dependência que indica que quanto maiores as distâncias de dependência, maior a complexidade do texto. A lista das 13 métricas é mostrada abaixo e suas descrições podem ser encontradas no manual do NILC Metrix¹⁰.

1. words_per_sentence
2. sentence_length_max
3. sentence_length_min

⁸ Os escores são baseados em uma escala likert de 7 itens e as métricas avaliam quatro faixas de escores na escala.

⁹ <https://simpligo.sidle.al/nilcmetrixdoc>

¹⁰ <https://simpligo.sidle.al/nilcmetrixdoc>

Tabela 30 – *Features* de Palavras

Features de Palavras			
1	syllables_per_content_word	25	verbs_ambiguity
2	adjective_ratio	26	concretude_mean
3	adverbs_ratio	27	concretude_std
4	noun_ratio	28	concretude_1_25_ratio
5	verbs_ratio	29	concretude_25_4_ratio
6	negation_ratio	30	concretude_4_55_ratio
7	ratio_function_to_content_words	31	concretude_55_7_ratio
8	pronoun_ratio	32	familiaridade_mean
9	first_person_pronouns	33	familiaridade_std
10	third_person_pronouns	34	familiaridade_1_25_ratio
11	cw_freq	35	familiaridade_25_4_ratio
12	min_cw_freq	36	familiaridade_4_55_ratio
13	conn_ratio	37	familiaridade_55_7_ratio
14	add_neg_conn_ratio	38	idade_aquisicao_mean
15	add_pos_conn_ratio	39	idade_aquisicao_std
16	cau_neg_conn_ratio	40	idade_aquisicao_1_25_ratio
17	cau_pos_conn_ratio	41	idade_aquisicao_4_55_ratio
18	log_neg_conn_ratio	42	idade_aquisicao_55_7_ratio
19	log_pos_conn_ratio	43	idade_aquisicao_25_4_ratio
20	tmp_neg_conn_ratio	44	imageabilidade_mean
21	tmp_pos_conn_ratio	45	imageabilidade_std
22	adjectives_ambiguity	46	imageabilidade_1_25_ratio
23	adverbs_ambiguity	47	imageabilidade_25_4_ratio
24	nouns_ambiguity	48	imageabilidade_4_55_ratio
		49	Imageabilidade_55_7_ratio

Fonte: Elaborada pelo autor.

4. sentence_length_standard_deviation
5. yngve
6. frazier
7. dep_distance
8. words_before_main_verb
9. mean_noun_phrase
10. min_noun_phrase
11. max_noun_phrase
12. std_noun_phrase
13. passive_ratio

6.4.1.3 Conexões entre sentenças

Esse grupo de métricas traz 17 delas que ajudam a explicitar o significado das palavras e sentenças. A coesão referencial ocorre quando um nome, pronome ou sintagma nominal se refere a outro termo em um texto. Quando há falhas nesta conexão, o tempo de leitura aumenta e pode haver quebra de compreensão (GRAESSER; MCNAMARA; KULIKOWICH, 2011). Há 4 métricas para capturar a coesão referencial. A diversidade lexical é relacionada com a coesão porque quanto maior o número de palavras diferentes em um texto significa que novas palavras precisam ser integradas no contexto discursivo. Type-token ratio (TTR) é a medida de diversidade lexical mais conhecida. Há 8 métricas que medem a sobreposição conceitual entre as sentenças de um texto pelo modelo estatístico chamada Análise de Semântica Latente (LSA, em inglês) (LANDAUER; DUMAIS, 1997). LSA é um método importante para computar a similaridade porque além de considerar palavras ele também pode medir conhecimento implícito. E faz isto porque o significado de uma palavra é capturado pela companhia das outras palavras que a rodeiam em um dado texto (GRAESSER; MCNAMARA; KULIKOWICH, 2011). Os valores para essas métricas variam de 0 (indica baixa coesão) a 1 (indica alta coesão). A lista das 17 métricas é mostrada abaixo e suas descrições podem ser encontradas no manual do NILC Metrix¹¹.

1. TTR
2. content_word_diversity
3. verb_diversity
4. adj_mean
5. adj_std
6. all_mean
7. all_std
8. givenness_mean
9. givenness_std
10. span_mean
11. span_std
12. content_density
13. adj_arg_ovl

¹¹ <https://simpligo.sidle.al/nilcmatrixdoc>

14. arg_ovl
15. adj_stem_ovl
16. stem_ovl
17. adj_cw_ovl

6.4.2 Features usadas no corpus Adole-sendo

Para o corpus Adole-sendo, por se tratar de um corpus de transcrições de áudios gravados, utilizamos um método de seleção de features para extrair as features mais importantes para a tarefa de classificação. Para entender quais features contribuem para a classificação de complexidade textual, aplicamos a seleção de features em todo o conjunto NILC Metrix, usando o algoritmo Recursive Feature Elimination and Cross-Validated (RFECV) ¹² com o classificador Random Forest para selecionar automaticamente o melhor número de features. O conjunto resultante foi as seguintes 55 features, agrupados por aqueles relacionados a palavras (31 features), sentenças (14 features) e conexões entre sentenças (10 features), a fim de serem facilmente comparados com aquelas features usadas nos outros conjuntos de dados.

6.5 Experimentos e Resultados

Nessa seção, apresentamos os resultados das avaliações da abordagem Single Task Learning (STL) e duas implementações Multi-Task Learning (MTL) – com 2 tarefas, chamada aqui de MTC-DTG Simplex, e com 3 tarefas, chamada de MTC-DTG, (Tabela 32). Neste trabalho, avaliamos o efeito do balanceamento de classes com abordagem *oversampling* e *undersampling*. Também, utilizamos técnicas de aumento de dados (Data Augmentation), que foram necessárias para as arquiteturas MTL para ter entrada de dados iguais. Além disso, avaliamos o modelo MTL com três tarefas de forma intrínseca e extrínseca usando o Dataset 3.

Setup

Em todas as avaliações intrínsecas, nós utilizamos cross-validation, k-fold=10, e utilizamos a média ponderada da F-measure. Nós usamos a linguagem Python e o framework Keras 2 e Tensor Flow, e as bibliotecas pandas e numpy do scikit-learn, para implementação dos modelos. Também, utilizamos a função de ativação ReLU nas camadas ocultas e a função de ativação softmax para estimar a probabilidade para cada classe, na camada de saída. Nossas categorias foram transformadas em one hot-encoding para as camadas de saída e usamos a função de perda¹³ categorical cross entropy e otimizador Adam para compilar o modelo.

¹² <https://scikit-learn.org/stable/auto_examples/feature_selection/plot_rfe_with_cross_validation.html>

¹³ Também chamada de função de erro ou função de custo

Tabela 31 – As 55 features usados no corpus Adole-sendo.

Metric name	Definition
Words	
Brunet's Statistics	Brunet's Statistics is a form of type/token ratio that is less sensitive to text size.
Dale-Chall formula adapted to BP	Combines the number of unfamiliar words with the average number of words per sentence.
Adjective ratio	Proportion of adjectives in relation to the number of words in the text.
Function words	Percentage of functional words in relation to the total words of the text.
Pronouns ratio	The ratio of pronouns to the number of words in the text.
Nouns ratio	The ratio of nouns to the number of words in the text.
Connectives Ratio	Proportion of connectives in relation to the number of words in the text.
Pronouns std	Standard deviation of the pronouns in relation to the number of words in the sentences.
Oblique pronouns ratio	Proportion of oblique pronouns in relation to all pronouns in the text.
Relative pronouns ratio	Proportion of Relative Pronouns in relation to the number of pronouns in the text.
Verbs max	Maximum proportion of verbs in relation to the number of words in the text.
Verbs std	Standard deviation of the verbs in relation to the number of words in the text.
Temporal positive conn ratio	Proportion of positive temporal connectors in relation to the number of words in the text.
Pronoun ratio	Proportion of relative pronouns in relation to the number of pronouns in the text.
Adjectives ambiguity	Proportion between the amount of meanings of the text adjectives in the T&P 2.0 and the amount of adjectives in the text.
Nouns ambiguity	Proportion of meanings of nouns in relation to the quantity of nouns.
Hypernyms verbs	Average number of hyperonyms per text verb.
Concreteness mean	Average of the concreteness values of the words of the sentence.
Concreteness std	Standard deviation of the Concreteness values of the sentence content words.
Concreteness 2.5-4 ratio	Proportion of content words with Concreteness values between 2.5 to 4, in relation to all content words.
Concreteness 4-5.5 ratio	Proportion of content words with Concreteness values between 4 to 5.5 4, in relation to all content words.
Familiarity std	Standard deviation of the Familiarity values of the sentence content words.
Familiarity 2.5-4 ratio	Proportion of content words with Familiarity values between 2.5 to 4, in relation to all content words.
Familiarity 4-5.5 ratio	Proportion of content words with Familiarity values between 4 to 5.5 4, in relation to all content words.
Familiarity 5.5-7 ratio	Proportion of content words with Familiarity values between 5.5 to 7, in relation to all content words.
AoA std	Standard deviation of the Age of Acquisition values of the sentence content words.
AoA 1-2.5 ratio	Proportion of content words with Age of Acquisition (AoA) values between 1 to 2.5, in relation to all content words.
AoA 4-5.5 ratio	Proportion of content words with Age of Acquisition values between 4 to 5.5, in relation to all content words.
Imageability 2.5-4 ratio	Proportion of content words with Imageability values between 2.5 to 4, in relation to all content words.
Min content words freq in brwac	Average of the absolute frequencies of the rarest content words in the text.
Content word freq	Average of the absolute frequencies of the content words in the text.
Content word std	Standard deviation of the proportions between the content words and the number of words in the sentences.
Sentences	
Words per sentence	Number of words divided by the number of sentences.
Sentence length min	Minimum number of words per sentence.
Std noun phrase	Standard deviation of the size of the noun phrases of the text.
Dependence Distance	The dependence distance using a dependency tree.
Words before main verb	Average number of words before the main verbs of the sentences.
Prepositions per clause	Average prepositions per clause.
Prepositions per sentence	Average prepositions per sentence.
Infinitive verbs	Proportion of verbs in the infinitive in relation to all verbs in the text.
Inflected verbs	Proportion of inflected verbs in relation to all verbs in the text.
Non-inflected verbs	Proportion of non-inflected verbs in relation to all verbs in the text.
Indicative imperfect ratio	Proportion of verbs in the imperfect past tense, in relation to the total number of verbs in the text.
Indicative present ratio	Proportion of verbs in the present indicative, in relation to the total number of inflected verbs in the text.
Indicative preterite perfect ratio	Proportion of verbs in the simple perfect past tense, in relation to the total number of verbs in the text.
Subordinate clauses	Proportion of subordinate clauses reduced in relation to the number of clauses in the text.
Connections between Sentence	
Type-Token Ratio	The proportion of words without repetition (types) in relation to the total of words (tokens).
Words	Number of words in the text.
Content words density	Average proportion of content words in relation to the number of functional words in the sentences.
Anaphoric refs	Average referent candidates by anaphoric pronoun, in up to 5 previous sentences.
Adj arg ovl	Average number of referents that are repeated in pairs of adjacent sentences in the text.
Function word diversity	Proportion of functional word types in relation to the number of functional word tokens in the text.
Content word diversity	Proportion of content word types in relation to the number of content word tokens in the text.
Noun diversity	Proportion of noun types in relation to the number of noun tokens in the text.
Coreference pronoun ratio (subjective personal pronouns).	Average referent candidates, in the previous sentence, by anaphoric pronoun
Verb diversity	Proportion of verb types in relation to the number of verb tokens in the text.

Fonte: Elaborada pelo autor.

Single Task Learning (STL)

A Tabela 32 mostra os resultados das arquiteturas STL para o dataset de transcrições do projeto Adole-sendo e para o dataset de Textos Didáticos dos Estágios Escolares do Sistema Educacional Brasileiro nas linhas 1 e 2, respectivamente. A média ponderada do F1 foi 0.90 para a primeira tarefa e 0.99 para a segunda tarefa.

Tabela 32 – Resultados dos Modelos na Avaliação Intrínseca

Forma de Avaliação	Modelos	F1 (weight)	
Avaliação Intrínseca	Oversampling	Single task Transcrições (1)	0.90
	OverSampling	Single task Book (2)	0.99
	Oversampling com	Multi-task Book (3)	0.93
	Data augmentation	Multi-task Transcrições (4)	0.915
	Undersampling com	Multi-task Book (5)	0.955
	Data augmentation	Multi-task Transcrições (6)	0.88
		Multi-task D1+D2 (7)	0.98

Fonte: Elaborada pelo autor.

No caso da arquitetura STL de textos didáticos, usamos uma única task com uma camada de entrada de 79 neurônios. Essa camada se conecta a uma camada densa de 120 neurônios, conectada a uma camada oculta de 28 neurônios, e uma outra de 14; todas essas com a camada de ativação ReLU e otimizador Adam. Por fim, uma camada com 4 saídas e a função de ativação softmax. Também utilizamos a função de perda categorical cross entropy. Concluindo com o treinamento de 13.454 parâmetros da rede. Para a arquitetura STL de transcrições, mantivemos a configuração acima, mudando apenas a camada de saída de 4 para 7 neurônios (devido ao número maior de classes) e a camada de entrada de 79 para 55 (menos features na entrada); sendo o treinamento de 10.619 parâmetros.

MTL com 2 tarefas (MTC-DTG Simplex)

Realizamos experimentos envolvendo o treinamento das duas tarefas (Textos Didáticos e Transcrições) de forma compartilhada, via MTL, e a predição individual para cada tarefa é mostrada nas linhas 3 e 4 da Tabela 32. Para a predição dos textos didáticos usando Multi-Task Learning, a F1 foi de 0.93 (linha 3), enquanto o Single Task Learning obteve a F1 de 0.99 (linha 2).

MTL com 3 tarefas (MTC-DTG)

Os experimentos usando a abordagem MTL com três tarefas de predição de complexidade foram realizados com a técnica de amostragem *undersampling* e *data augmentation*, exigida pela arquitetura neural. Os valores da F-measure dos modelos treinados são apresentados nas linhas 5, 6 e 7 da Tabela 32. Observa-se o melhor valor de F-measure (F1) para o preditor da tarefa principal, treinado no dataset de Textos Didáticos, que tem um desempenho de 0.955 de F1. Esse valor supera o baseline apresentado em Gazzola, Leal e Aluísio (2019) que obteve 0.80 de F1.

Tabela 33 – Resultados dos Modelos na Avaliação Extrínseca com datasets da MEC-RED

Dataset de Avaliação	Modelos	F1 (weight)
D1	Modelo STL Books, Oversampling	0.25
D2	Modelo STL Books, Oversampling	0.30
D1+D2	MTC-DTG SimpleX Undersampling, Data augmentation	0.32
D3	MTC-DTG Undersampling, Data augmentation	0.575

Fonte: Elaborada pelo autor.

6.5.1 Matriz de Confusão da Avaliação Extrínseca da Complexidade Textual

Os experimentos de avaliação extrínseca com datasets da MEC-RED (Tabela 33) foram realizados com os modelos STL, MTL de 2 tarefas (MTC-DTG Simplex) e MTL com 3 tarefas (MTC-DTG), porém não foram realizados nos mesmos datasets, pois os datasets D1 e D2 foram usados no treinamento do modelo MTL com 3 tarefas. Na avaliação extrínseca do modelo MTC-DTG Simplex com os datasets D1+D2, o resultado foi 0.32 de F1 enquanto usando MTC-DTG e avaliação com o dataset D3 o resultado foi de 0.54 de F1, mostrando um aumento na performance ao trazermos dados do domínio no treinamento do modelo de complexidade textual. As duas arquiteturas chamadas de Multi-task approach to Textual Complexity using Different Text Genres (MTC-DTG) Simplex com 47.053 parâmetros e MTC-DTG com 71.947 parâmetros treinados são mostradas na Figura 49.

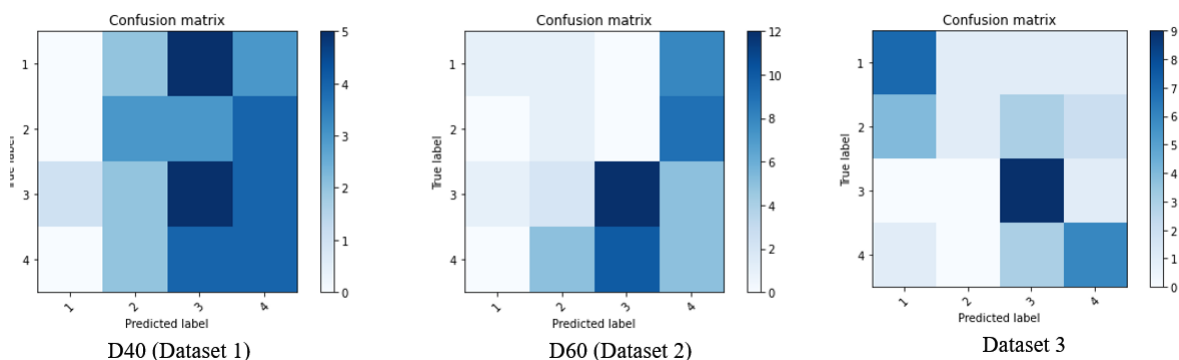
Foram analisadas as matrizes de confusão da avaliação extrínseca nos datasets D1, D2 e D3, mostrados na Tabela 33. Na Figura 48, são mostradas as matrizes de confusão.

No D1, que representa o dataset de textos didáticos, para as classes Ensino Médio e o Ensino Superior o modelo acertou 5 amostras de 10 e 4 amostras de 10, respectivamente. Enquanto que, para as classes Ensino Fundamental II o modelo acertou 3 amostras de 10 e o pior desempenho ficou com a classe Ensino Fundamental I para a qual o preditor errou todas as 10 amostras.

No D2, que representa o dataset de textos de animação/simulação, experimento prático e software educacional, para a classe Ensino Médio o modelo obteve o melhor desempenho acertando 12 amostras de 20 e para a classe Ensino Superior o modelo acertou 5 amostras de 20. Enquanto que, para as classes Ensino Fundamental I e Ensino Fundamental II o modelo obteve o pior desempenho, com 1 acerto de 10 amostras em uma das etapas escolares.

No D3, as classes Ensino Médio e Ensino Superior obtiveram um ótimo desempenho acertando 9 das 10 amostras e 6 acertos de 10 amostras, respectivamente. Em relação ao Ensino Fundamental I foram 7 acertos de 10 amostras. Por fim, o Ensino Fundamental II demonstra ser a etapa escolar mais difícil para nosso classificador de complexidade textual, acertando apenas 1 amostra de 10.

Figura 48 – Matrizes de confusão de 3 modelos da Tabela 33 (linhas 1, 2 e 4)



Fonte: Elaborada pelo autor.

Na Figura 49 são mostradas as duas arquiteturas finais, que chamamos de MTC-DTG Simplex e MTC-DTG. Na MTC-DTG Simplex temos 2 camadas paralelas de entrada uma para cada documento com 55 e 79 neurônios que representam a quantidade de *features*. Essas camadas são conectadas em 2 camadas ocultas densas paralelas com 120 neurônios cada, representando 16.320 mil parâmetros a serem treinados para aprendizagem; usamos a função de ativação ReLU. Em seguida, as 2 camadas ocultas densas são totalmente conectadas em uma camada de concatenação para serem usadas na camada compartilhada (neste ponto temos 40.420 parâmetros), nossa camada compartilhada também utiliza a função de ativação *relu*, como também as próximas camadas até as 2 últimas camadas de saídas que utilizamos a função de perda *categorical cross entropy* e as entradas como *one hot encoding* e otimizador Adam. Por fim, concluímos a tarefa de predição para 2 tasks com 47.053 parâmetros treinados para predição de transcrições e de livros didáticos. Para a arquitetura MTC-DTG, fizemos modificações para incluir uma nova tarefa de predição similar para recursos da MEC-RED, de forma semelhante ao MTC-DTG Simplex, porém adicionando uma nova task com 79 features de forma paralela às outras 2 tasks, alterando a camada de concatenação que passa ser 360 e a camada compartilhada com 150 neurônios. Concluindo a tarefa de predição para 3 tasks com 71.957 parâmetros a serem treinados.

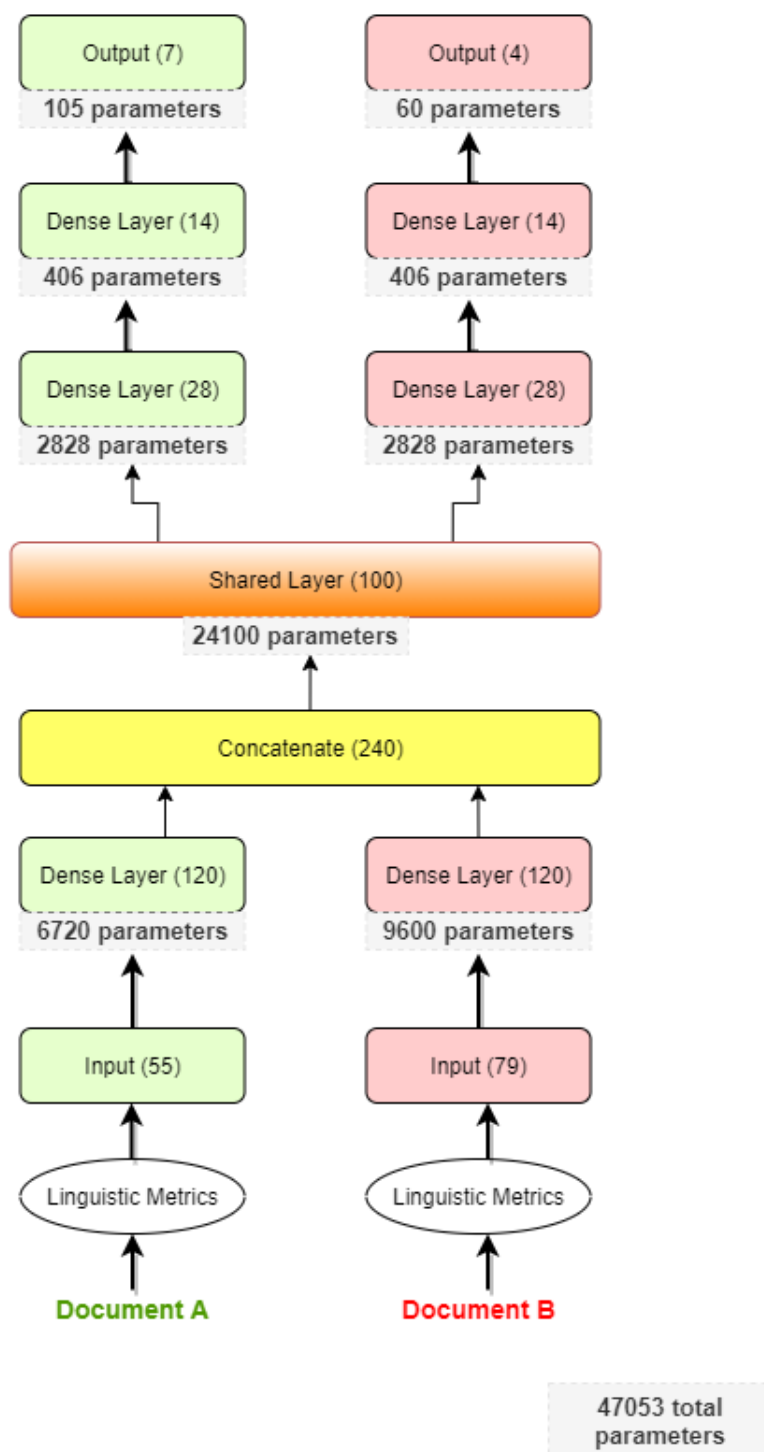
6.6 Conclusões e Discussões

O principal objetivo deste estudo foi explorar a relação entre três tarefas de complexidade textual com uma abordagem de MTL, usando conjuntos de dados com granularidade mais fina (ou alta) e de granularidade fina de textos escritos e textos transcritos manualmente, para superar a falta de textos classificados do MEC-RED. Neste estudo, tivemos duas questões relacionadas a classificadores de MTL:

- Um classificador desenvolvido com dois registros de língua (fala e escrita) via MTL tem

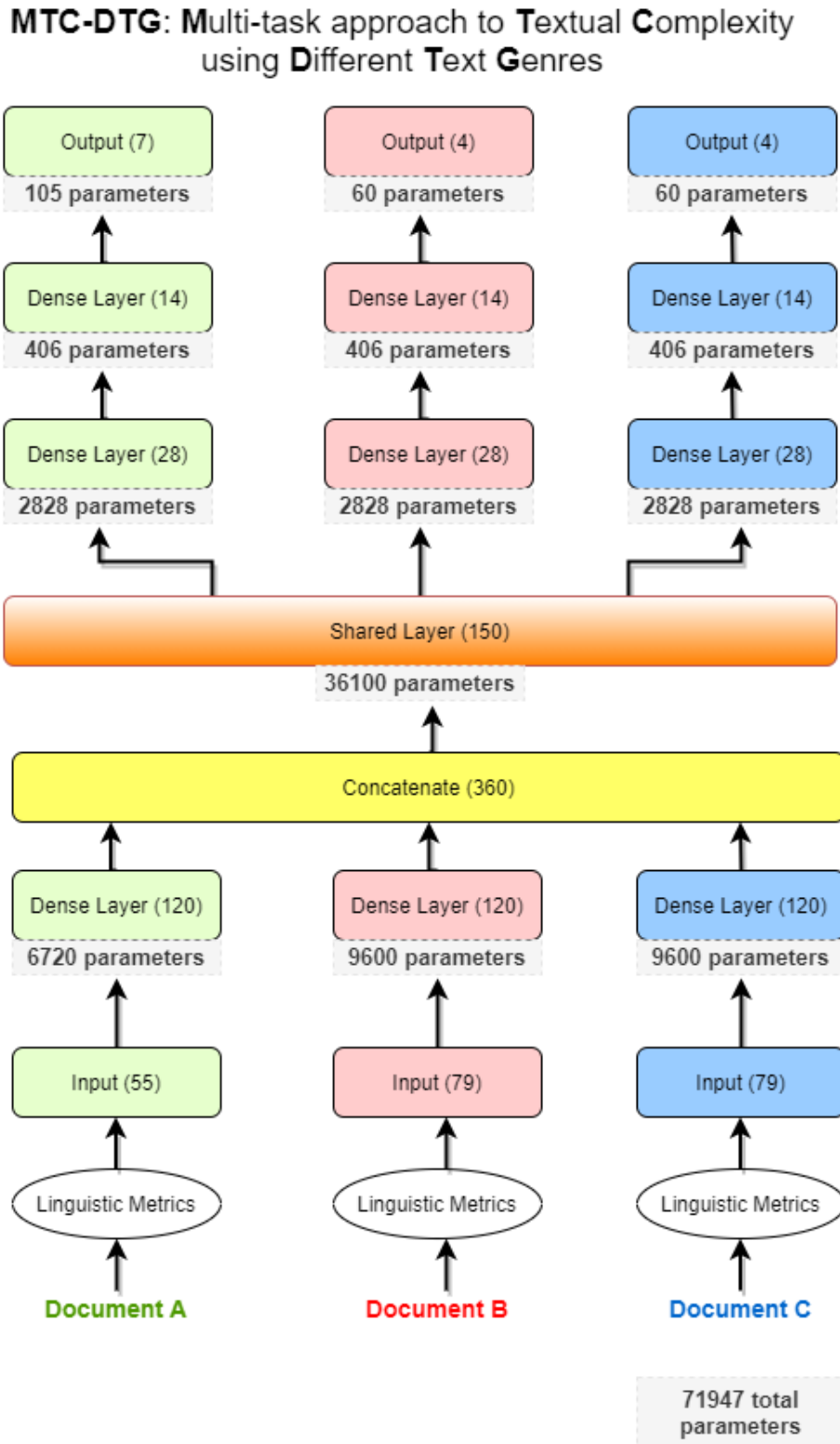
Figura 49 – Arquitetura do MTC-DTG Simplex

MTC-DTG Simplex: Multi-task approach to Textual Complexity using Different Text Genres



Fonte: Elaborada pelo autor.

Figura 50 – Arquitetura do MTC-DTG



Fonte: Elaborada pelo autor.

melhor desempenho em REA em português do que um modelo de tarefa única treinado apenas com textos escritos de várias séries (granularidade mais fina)?

- Há impacto no desempenho se textos do domínio MEC-RED forem incluídos no treinamento de classificadores usando aprendizado *Multi-Task Learning*?

Nosso modelo de aprendizado multitarefa com duas tarefas auxiliares (MTC-DTG) apresenta uma medida F1 de 0,95%, uma melhoria de 0,15 pontos em relação ao melhor modelo de [Gazzola, Leal e Aluísio \(2019\)](#). Também leva a melhorias significativas em relação ao Single Task Learning (STL) e Multi-task Learning com uma tarefa auxiliar (MTC-DTG Simplex). Os resultados mostram que o principal modelo MTC-DTG supera ambos os modelos quando avaliado em um conjunto de dados REA não vistos de recursos do MEC-RED — um aumento de 28 pontos sobre o Single Task Learning e 22 pontos sobre o MTC-DTG Simplex.

O modelo baseline [Gazzola, Leal e Aluísio \(2019\)](#), quando avaliado com dados da MEC-RED, obteve F1 de 0.52 para o Dataset 1 e F1 de 0.39 para o Dataset 2. Enquanto que, nossa proposta de MTL de 2 tasks (MTC-DTG Simplex) demonstrou ter resultados inferiores para os 2 datasets (D1+D2) (0.32 de F1). Entretanto, com o modelo MTC-DTG obtivemos um resultado com F1 de 0.575 na avaliação extrínseca com dados da MEC-RED. Não é possível afirmar que o modelo final (MTC-DTG) seja superior ao melhor modelo de [Gazzola, Leal e Aluísio \(2019\)](#), pois os datasets de avaliação podem não ter a mesma distribuição dos dados, mas esse estudo aponta um caminho a ser seguido — incluir mais textos do domínio do MEC-RED, que tenham um lastro de alguma editora, no treinamento dos classificadores usando MTL.

O modelo baseline [Gazzola, Leal e Aluísio \(2019\)](#), quando avaliado com dados da MEC-RED, obteve F1 de 0.52 para o Dataset 1 e F1 de 0.39 para o Dataset 2. Enquanto que, nossa proposta de MTL de 2 tasks (MTC-DTG Simplex) demonstrou ter resultados inferiores para os 2 datasets (D1+D2) (0.32 de F1). Entretanto, com o modelo MTC-DTG obtivemos um resultado com F1 de 0.54 na avaliação extrínseca com dados da MEC-RED. Não é possível afirmar que o modelo final (MTC-DTG) seja superior ao melhor modelo de [Gazzola, Leal e Aluísio \(2019\)](#), pois os datasets de avaliação podem não ter a mesma distribuição dos dados, mas esse estudo aponta um caminho a ser seguido — incluir mais textos do domínio do MEC-RED, que tenham um lastro de alguma editora, no treinamento dos classificadores usando MTL.

6.7 Considerações Finais

Em relação aos dados, as contribuições deste capítulo são quatro. Disponibilizamos publicamente: (1) o cópulo de 2.067 extratos de textos didáticos, classificados nas 4 etapas escolares do sistema educacional brasileiro; (2) o cópulo Adole-sendo com 237 narrativas transcritas e com disfluências removidas; (3) os três conjuntos de textos da MEC-REC totalizando amostras de 140 recursos; e (4) uma recolha de recursos da MEC-RED datada de abril de 2018.

Todos os recursos estão disponibilizados no GitHub do projeto (<https://github.com/gazzola/MTC-DTG>).

O modelo MTC-DTG é uma proposta a ser aprimorada com a inclusão de mais recursos da MEC-RED para avaliar a complexidade textual com uma maior quantidade de amostras de textos. Além disso, considerando o F1 de 0.575, acreditamos que este resultado precisa ser melhorado antes de ser disponibilizado como uma ferramenta para a plataforma MEC-RED. Como trabalhos futuros, pretendemos comparar a complexidade textual de cada grupo de recursos da MEC-RED:

- apresentações, livros digitais, website externo, planos de aula e textos (que são recursos que se aproximam do gênero de textos didático, com o objetivo de ensinar novos conhecimentos), e
- animação/simulação, aplicativo móvel, jogos, experimento prático e software educacional (que são recursos do gênero instrucional, que informam como fazer uma tarefa).

para poder treinar um modelo final com melhor desempenho e que possa ser usado como um suporte ao autor dos recursos quando trazer um novo recurso para a MEC-RED.

CONCLUSÕES

O objetivo geral deste trabalho foi contribuir a área de pesquisa de Complexidade Textual no Português Brasileiro e a qualidade de REA para esta língua, avaliando tanto a qualidade dos metadados de REA quanto a complexidade textual de REA para predição da etapa escolar. Além disso, propusemos alguns objetivos específicos:

1. Avaliar as abordagens de seleção de *features* (CFS, Lasso e PCA) no uso da complexidade textual para predição da etapa escolar.
2. Propor arquiteturas de *Deep Learning* para estudar e avaliar a complexidade textual para predição da etapa escolar.
3. Propor arquiteturas *Single e Multi-task Learning* para estudar e avaliar a complexidade textual para predição da etapa escolar.
4. Usar e implementar métodos de aprendizado de máquina tradicional para estudar e avaliar a complexidade textual para predição da etapa escolar.
5. Implementar métodos de aprendizado de máquina tradicional e de *Deep Learning* para estudar e avaliar a predição automática da qualidade dos metadados de REA.
6. Comparar o desempenho dos métodos criados de aprendizado de máquina tradicional, *Single* e MTL para predição da etapa escolar.

Podemos descrever as contribuições dessa monografia de doutorado como sendo o primeiro trabalho em tratar qualidade de REA (metadados e conteúdo) com uso de técnicas de Processamento de Línguas Naturais para o Português do Brasil.

Além disso, esse trabalho propôs uma abordagem de *Deep Learning* para avaliação da qualidade de metadados de REA, que era novidade para a área da Educação em 2017, e propôs

novos modelos de avaliação da complexidade textual para o Português que podem ser usados como *baseline*, futuramente.

Este trabalho mostrou que avaliar a complexidade textual para predizer as etapas de ensino é um desafio dentro de repositórios abertos, como o MEC-RED, para os tipos de materiais como: experimentos práticos, software educacional, textos, apresentação, imagens, áudios e vídeos, que são diferentes linguisticamente dos textos didáticos usados geralmente em *datasets* para treinamento da tarefa.

7.1 Contribuições

As contribuições dessa monografia de doutorado podem ser divididas em 2 conjuntos, para cada tarefa:

Avaliar a qualidade de metadados de REA

- Criação e anotação de uma grande recolha de REA da MEC-RED, em 2018, formando um corpus com título e resumo para avaliar a qualidade de metadados de REA;
- Implementação de método de aprendizagem de máquina tradicional para avaliar a qualidade de metadados de REA.
- Criação de duas arquiteturas de Deep Learning (DeepMD e DeepMDX) baseadas na LSTM com objetivo predizer a qualidade dos metadados de REA.

Avaliar a complexidade textual para predição da etapa escolar

- A recolha de todos os recursos disponibilizados na MEC-RED em 2018, além de metadados trouxe material/link, que foram armazenados em um banco de dados relacional.
- Compilação e disponibilização pública (no github do projeto <<https://github.com/gazzola/MTC-DTG>>) de um grande cópulo com textos didáticos de diversas fontes, classificados em diferentes etapas de ensino (ou níveis escolares) do Sistema Educacional Brasileiro, organizado nas seguintes etapas: Ensino Fundamental I (1^o ao 5^o ano), Ensino Fundamental II (6^o ao 9^o ano), Ensino Médio e Ensino Superior. Essas quatro etapas de ensino são as mesmas utilizadas na MEC-RED para classificar os REA nos Estágios Escolares.
- Compilação de um conjunto composto por três *datasets*, para avaliação extrínseca da tarefa de Complexidade Textual da MEC-RED. Um *dataset* chamado de Dataset 1 (40 REA) de recursos da MEC-RED do tipo apresentações, livros digitais, website externo, planos de aula e textos que são considerados do gênero de textos didáticos, e outro chamado de Dataset 2 (60 REA) composto por textos de recursos do tipo animação/simulação,

experimento prático e software educacional, considerados do gênero de texto instrucional, que indica o que fazer, isto é, fornece mais frequentemente os passos para se executar uma tarefa/ação. Por fim, um dataset chamado de Dataset 3 (40 REA), composto por textos de recursos da MEC-RED de 4 etapas escolares vindos de editoras, para garantir um crivo melhor quanto à etapa escolar dos textos.

- Avaliação de 3 abordagens de seleção de *features* (CFS, Lasso e PCA), aplicadas aos modelos de aprendizado de máquina tradicional na tarefa de complexidade textual. Os resultados foram avaliados de forma comparativa com e sem uso de seleção de *features*. Também levantamos as *features* mais importantes para classificação automática da etapa escolar.
- Utilização e implementação de métodos de aprendizagem de máquina tradicional usando *features* linguísticas para predição da etapa escolar.
- Criação de seis arquiteturas de Deep Learning para avaliação da complexidade textual: (i) BiLSTM + Word2Vec + Self.Attention, (ii) BiLSTM + Wang2Vec + Self.Attention, (iii) BiLSTM + Wang2Vec, (iv) BiLSTM + Wang2Vec+ Self.Attention + Elmo, (v) BiLSTM + GloVe + Self.Attention e (vi) BiLSTM + FastText + Self.Attention.
- Avaliações da complexidade textual, tanto de forma intrínseca quanto extrínseca, com uso de aprendizado de máquina tradicional (2 avaliações) e com métodos de Deep Learning (2 avaliações).
- Anotação e criação de um cópús de transcrições do Projeto Adole-sendo com 237 narrativas transcritas. Esse cópús está disponível publicamente no github do projeto <<https://github.com/gazzola/MTC-DTG>>.
- Criação de classificadores desenvolvidos com dois registros de língua (fala e escrita) via Multi-task Learning para REA em português, sendo eles: Modelo Single Task Learning, MTC-DTG Simplex e MTC-DTG.

Nossa hipótese envolvendo o uso de métodos de *Deep Learning* e MTL se confirmaram em trazer um melhor desempenho na avaliação da complexidade textual para predição da etapa escolar em relação aos métodos tradicionais de aprendizado de máquina, quando há um grande volume de dados para o aprendizado. Também, nossa segunda hipótese, se confirmou como sendo uma alternativa na avaliação da qualidade do REA, em materiais que não possuem conteúdo textual, como vídeos e áudios.

7.2 Limitações

As limitações desse trabalho se relacionam com a compilação dos datasets de avaliação extrínseca da MEC-RED.

Usando a recolha realizada em 2018, conseguimos poucos dados para 2 etapas educacionais (Ensino Fundamental I e Ensino Fundamental II) dentro da plataforma MEC-RED, pois áudios e vídeos não foram considerados (havia um número maior de recursos deste tipo) dado que esta tese que focou em dados textuais.

Outra limitação foi a não consideração da transcrição automática de vídeos e áudios para avaliar a etapa de ensino. Sabemos que métodos automáticos de transcrição para o Português não são tão bons quanto para a língua inglesa, mas poderíamos ter feito a transcrição automática com correção manual.

7.3 Trabalhos Futuros

Inclusão de novas métricas de complexidade, como as 17 métricas do trabalho de [Graesser e McNamara \(2011\)](#), citadas na Seção 5.3, que não foram incluídas neste estudo atual, pois não foram implementadas para o Português do Brasil.

Criação de corpúscos maiores envolvendo a complexidade textual e avaliação automática da etapa escolar, incluindo repositórios abertos como MEC-RED e outras como a plataforma de Educação em Saúde e Diversidade da USP, recentemente criada¹.

7.4 Lista de Publicações

Seguem abaixo todos os artigos produzidos (aceitos e submetidos) durante o doutorado. Foram separadas aquelas publicações/submissões que estão diretamente vinculadas a esta tese de doutorado e aquelas que foram produzidas em parcerias com pesquisadores durante o doutorado.

Trabalhos relacionados com a Tese de Doutorado

- GAZZOLA, Murilo. Um Método para Avaliação Automática da Qualidade de Recursos Educacionais Abertos usando Deep Learning. Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2017. p. 1477. <<http://dx.doi.org/10.5753/cbie.sbie.2017.1477>>.
- GAZZOLA, M. G., Leal, S. E., Aluísio, S. M. Predição da complexidade textual de recursos educacionais abertos em português. Symposium in Information and Human Language Technology (Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana - STIL). 2019. p. 61. <<http://comissoes.sbc.org.br/ce-pln/stil2019/proceedings-stil-2019-Final.pdf>>

¹ <https://prceu.usp.br/repositorio-uspdiversidade/>

Trabalhos produzidos em parcerias com pesquisadores durante o doutorado

- AMARO, Edson et al. Utilização de Inteligência Artificial em Saúde. 2020. Nic.br. <https://www.nic.br/media/docs/publicacoes/6/20200908170853/panorama_setorial_ano-xii_n_2_Ano%20XII%20-%20N.%202%20-%20inteligencia_artificial_e_saude.pdf>
- AMARO, Edson et al. Use of Artificial Intelligence in Health: Lessons learned while addressing the COVID-19 outbreak. 2020. Cetic.br. <https://cetic.br/media/docs/publicacoes/6/20201027153704/panorama_setorial_ano-xii_n_2_Year%20XII%20-%20N.%202%20-%20Use%20of%20Artificial%20Intelligence%20in%20Health_Lessons%20learned%20while%20addressing%20the%20COVID%2019%20outbreak.pdf>
- FUCHS, Felipe et al. Easynegex: Tool For Negation And Uncertainty. Detection In Portuguese Radiology Reports. 2020. Sociedade Paulista de Radiologia e Diagnóstico por Imagem. JPR. <https://jpr2020.org.br/wp-content/uploads/2020/05/CADERNO_JPR_2020_VIRTUAL-PORTUGUES-FINAL.pdf>
- FERREIRA, Lucas et al. Métodos de classificação aplicados à detecção automática de Estilos de Aprendizagem em um ambiente real de ensino. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2017. p. 1517. <<http://dx.doi.org/10.5753/cbie.sbie.2017.1517>>

REFERÊNCIAS

ABERTA, I. E. **Quais são as quatro liberdades dos REA?** 2020. <<https://aberta.org.br/faq/>>. Acessado em: 2020-11-01. Citado nas páginas 31 e 32.

AHMED, F.; FUGE, M. Capturing winning ideas in online design communities. In: **Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing**. New York, NY, USA: ACM, 2017. (CSCW '17), p. 1675–1687. ISBN 978-1-4503-4335-0. Disponível em: <<http://doi.acm.org/10.1145/2998181.2998249>>. Citado nas páginas 26, 64, 65, 66 e 83.

ALUÍSIO, S.; CUNHA, A.; SCARTON, C. Evaluating progression of alzheimer's disease by regression and classification methods in a narrative language test in portuguese. In: SILVA, J.; RIBEIRO, R.; QUARESMA, P.; ADAMI, A.; BRANCO, A. (Ed.). **Computational Processing of the Portuguese Language**. Cham: Springer International Publishing, 2016. p. 109–114. ISBN 978-3-319-41552-9. Citado na página 114.

ALUÍSIO, S.; SPECIA, L.; GASPERIN, C.; SCARTON, C. Readability assessment for text simplification. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications**. [S.l.], 2010. p. 1–9. Citado na página 114.

ANDRYCHOWICZ, M.; DENIL, M.; GOMEZ, S.; HOFFMAN, M. W.; PFAU, D.; SCHAUL, T.; FREITAS, N. de. Learning to learn by gradient descent by gradient descent. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2016. p. 3981–3989. Citado na página 45.

ATKINS, D. E.; BROWN, J. S.; HAMMOND, A. L. **A review of the open educational resources (OER) movement: Achievements, challenges, and new opportunities**. [S.l.]: Creative common, 2007. Citado nas páginas 23 e 31.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. **arXiv preprint arXiv:1409.0473**, 2014. Citado na página 44.

BARANIUK, R. G.; BURRUS, C. S. Viewpoint global warming toward open educational resources. **Communications of the ACM**, ACM, v. 51, n. 9, p. 30–32, 2008. Citado na página 23.

BENGIO, Y. *et al.* Learning deep architectures for ai. **Foundations and trends in Machine Learning**, Now Publishers, Inc., v. 2, n. 1, p. 1–127, 2009. Citado na página 43.

BETHARD, S.; WETZER, P.; BUTCHER, K.; MARTIN, J. H.; SUMNER, T. Automatically characterizing resource quality for educational digital libraries. In: ACM. **Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries**. [S.l.], 2009. p. 221–230. Citado nas páginas 26, 51, 52, 54, 59, 60 e 83.

- BICK, E. **The Parsing System “Palavras”. Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework**. Århus: University of Aarhus, 2000. Citado na página 114.
- BOJANOWSKI, P.; GRAVE, E.; JOULIN, A.; MIKOLOV, T. Enriching word vectors with subword information. **Transactions of the Association of Computational Linguistics**, v. 5, p. 135–146, 2017. Citado nas páginas 47 e 48.
- BOWMAN, S. R.; ANGELI, G.; POTTS, C.; MANNING, C. D. A large annotated corpus for learning natural language inference. **arXiv preprint arXiv:1508.05326**, 2015. Citado na página 44.
- CAEIRO-RODRÍGUEZ, M.; RODRÍGUEZ-BERMÚDEZ, M.; LLAMAS-NISTAL, M.; SANTOS-GAGO, J. M. Toward the implementation of relevance and reputation indicators in edu-area. **IEEE Revista Iberoamericana de Tecnologías del Aprendizaje**, IEEE, v. 11, n. 2, p. 107–114, 2016. Citado nas páginas 26, 63, 64 e 65.
- CAFOLLA, R. Project merlot: Bringing peer review to web-based educational resources. **Journal of Technology and Teacher Education**, Association for the Advancement of Computing in Education, v. 14, n. 2, p. 313, 2006. Citado na página 62.
- CARUANA, R. Multitask learning. **Machine learning**, Springer, v. 28, n. 1, p. 41–75, 1997. Citado nas páginas 48 e 108.
- CECHINEL, C.; SANCHEZ-ALONSO, S.; GARCIA-BARRIOCANAL, E. Statistical profiles of highly-rated learning objects. **Comput. Educ.**, Elsevier Science Ltd., Oxford, UK, UK, v. 57, n. 1, p. 1255–1269, ago. 2011. ISSN 0360-1315. Disponível em: <<http://dx.doi.org/10.1016/j.compedu.2011.01.012>>. Citado nas páginas 26, 55, 56 e 83.
- CHEN, M.; ZECHNER, K. Computing and evaluating syntactic complexity features for automated scoring of spontaneous non-native speech. In: **Proceedings of the 49th annual meeting of the Association for Computational Linguistics: Human Language Technologies**. [S.l.: s.n.], 2011. p. 722–731. Citado na página 109.
- CHO, K.; MERRIËNBOER, B. van; BAHDANAU, D.; BENGIO, Y. On the properties of neural machine translation: Encoder-decoder approaches. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation**. [S.l.], 2014. p. 103—111. Citado na página 44.
- COLLOBERT, R.; WESTON, J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In: **Proceedings of the 25th international conference on Machine learning**. [S.l.: s.n.], 2008. p. 160–167. Citado na página 48.
- COLLOBERT, R.; WESTON, J.; BOTTOU, L.; KARLEN, M.; KAVUKCUOGLU, K.; KUKSA, P. Natural language processing (almost) from scratch. **Journal of Machine Learning Research**, v. 12, n. Aug, p. 2493–2537, 2011. Citado na página 47.
- COMMONS, C. **Open Education**. 2020. <<https://creativecommons.org/about/program-areas/education-oer/>>. Acessado em: 2020-11-01. Citado na página 31.
- CROSSLEY, S.; MCNAMARA, D. Applications of text analysis tools for spoken response grading. **Language Learning & Technology**, University of Hawaii National Foreign Language Resource Center, v. 17, n. 2, p. 171–192, 2013. Citado na página 109.

CUNHA, A. L. V. da. **Coh-Matrix-Dementia: análise automática de distúrbios de linguagem nas demências utilizando Processamento de Línguas Naturais**. Dissertação (Mestrado) — Universidade de São Paulo, ICMC - USP São Carlos, 2015. Citado na página 92.

DALIP, D. H.; GONÇALVES, M. A.; CRISTO, M.; CALADO, P. Automatic assessment of document quality in web collaborative digital libraries. **Journal of Data and Information Quality (JDIQ)**, ACM, v. 2, n. 3, p. 14, 2011. Citado nas páginas 26, 53, 54 e 83.

DAVIS, H. C.; CARR, L. A.; HEY, J. M.; HOWARD, Y.; MILLARD, D.; MORRIS, D.; WHITE, S. Bootstrapping a culture of sharing to facilitate open educational resources. **IEEE transactions on learning technologies**, IEEE, v. 3, n. 2, p. 96–109, 2010. Citado na página 33.

DEUTSCH, T.; JASBI, M.; SHIEBER, S. Linguistic features for readability assessment. In: **Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications**. Seattle, WA, USA â† Online: Association for Computational Linguistics, 2020. p. 1–17. Disponível em: <<https://www.aclweb.org/anthology/2020.bea-1.1>>. Citado nas páginas 27 e 72.

DONAHUE, J.; HENDRICKS, L. A.; GUADARRAMA, S.; ROHRBACH, M.; VENUGOPALAN, S.; SAENKO, K.; DARRELL, T. Long-term recurrent convolutional networks for visual recognition and description. In: **The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2015. Citado nas páginas 45, 46 e 83.

DRAELOS, T. J.; MINER, N. E.; LAMB, C. C.; COX, J. A.; VINEYARD, C. M.; CARLSON, K. D.; SEVERA, W. M.; JAMES, C. D.; AIMONE, J. B. Neurogenesis deep learning: Extending deep networks to accommodate new classes. In: **IEEE. Neural Networks (IJCNN), 2017 International Joint Conference on**. [S.l.], 2017. p. 526–533. Citado na página 43.

ESTEVA, A.; KUPREL, B.; NOVOA, R. A.; KO, J.; SWETTER, S. M.; BLAU, H. M.; THRUN, S. Dermatologist-level classification of skin cancer with deep neural networks. **Nature**, v. 542, n. 7639, p. 115–118, 2017. Citado na página 44.

FILHO, J. A. W.; WILKENS, R.; ZILIO, L.; IDIART, M.; VILLAVICENCIO, A. Crawling by readability level. In: SPRINGER. **International Conference on Computational Processing of the Portuguese Language**. [S.l.], 2016. p. 306–318. Citado nas páginas 27 e 71.

GAGO, P. C. Questões de transcrição em análise da conversa. **Veredas-Revista de Estudos Linguístico**, v. 6, n. 2, p. 89–113, 2002. Citado na página 112.

GAZZOLA, M. G. **Uma arquitetura para mecanismos de buscas na web usando integração de esquemas e padrões de metadados heterogêneos de recursos educacionais abertos em repositórios dispersos**. Dissertação (Mestrado) — Universidade de São Paulo (USP), 2015. Acesso em: 2017-04-17. Citado nas páginas 31, 33 e 34.

GAZZOLA, M. G.; CIFERRI, C. D.; GIMENES, I. M. Seoer: Uma arquitetura para mecanismo de busca na web por recursos educacionais abertos. In: **Anais do Simpósio Brasileiro de Informática na Educação**. [S.l.: s.n.], 2014. p. 1013–1022. Citado na página 32.

GAZZOLA, M. G.; LEAL, S. E.; ALUÍSIO, S. M. Predição da complexidade textual de recursos educacionais abertos em português. In: **Symposium in Information and Human Language Technology - STIL**. [S.l.]: SBC, 2019. Citado nas páginas 71, 120 e 125.

- GIMENES, I. M.; BARROCA, L.; FELTRIM, V. D. Tendências na educação a distância e educação aberta na computação. **CSBC**, 2012. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/jai/2012/001.pdf>>. Citado nas páginas 23, 31 e 33.
- GONZALEZ-GARDUNO, A. V.; SØGAARD, A. Using gaze to predict text readability. In: **Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications**. [S.l.: s.n.], 2017. p. 438–443. Citado na página 109.
- GONZALEZ-GARDUÑO, A. V.; SØGAARD, A. Using gaze to predict text readability. In: **Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications**. [S.l.]: Association for Computational Linguistics, 2017. p. 438–443. Citado nas páginas 27 e 69.
- GRAESSER, A. C.; MCNAMARA, D. S. Computational analyses of multilevel discourse comprehension. **Topics in cognitive science**, Wiley Online Library, v. 3, n. 2, p. 371–398, 2011. Citado nas páginas 27, 69, 91, 92, 101 e 130.
- GRAESSER, A. C.; MCNAMARA, D. S.; KULIKOWICH, J. M. Coh-metrix: Providing multi-level analyses of text characteristics. **Educational researcher**, Sage Publications Sage CA: Los Angeles, CA, v. 40, n. 5, p. 223–234, 2011. Citado nas páginas 27, 69, 114, 115 e 117.
- GRAHAM, W. Facebook developer tools. In: **Beginning Facebook Game Apps Development**. [S.l.]: Apress, 2012. p. 201–229. ISBN 978-1-4302-4170-6. Citado na página 33.
- GRAVES, A. **Supervised sequence labelling with recurrent neural networks**. [S.l.]: Springer, 2012. v. 385. Citado na página 45.
- _____. Generating sequences with recurrent neural networks. **arXiv preprint arXiv:1308.0850**, 2013. Citado na página 44.
- GRAVES, A.; MOHAMED, A.-r.; HINTON, G. Speech recognition with deep recurrent neural networks. In: IEEE. **Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on**. [S.l.], 2013. p. 6645–6649. Citado na página 44.
- GRAVES, A.; SCHMIDHUBER, J. Offline handwriting recognition with multidimensional recurrent neural networks. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2009. p. 545–552. Citado na página 44.
- HALL, M.; FRANK, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; WITTEN, I. H. The WEKA data mining software: an update. **SIGKDD Explorations**, v. 11, n. 1, p. 10–18, 2009. Citado na página 56.
- HARTMANN, N.; CUCATTO, L.; BRANTS, D.; ALUÍSIO, S. Automatic classification of the complexity of nonfiction texts in portuguese for early school years. In: SPRINGER. **International Conference on Computational Processing of the Portuguese Language**. [S.l.], 2016. p. 12–24. Citado nas páginas 27 e 71.
- Hartmann, N.; Fonseca, E.; Shulby, C.; Treviso, M.; Rodrigues, J.; Aluísio, S. Portuguese Word Embeddings: Evaluating on Word Analogies and Natural Language Tasks. **ArXiv e-prints**, ago. 2017. Citado na página 47.
- HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. **Neural computation**, MIT Press, v. 9, n. 8, p. 1735–1780, 1997. Citado nas páginas 45 e 83.

JAIN, S.; KOTSAMPASAKOU, E.; ECKER, G. F. Comparing the performance of meta-classifiers—a case study on selected imbalanced data sets relevant for prediction of liver toxicity. **Journal of computer-aided molecular design**, Springer, p. 1–8, 2018. Citado na página 100.

JOZEFOWICZ, R.; ZAREMBA, W.; SUTSKEVER, I. An empirical exploration of recurrent network architectures. In: **Proceedings of the 32nd International Conference on Machine Learning (ICML-15)**. [S.l.: s.n.], 2015. p. 2342–2350. Citado na página 45.

KANWAR, A.; UVALIĆ-TRUMBIĆ, S.; BUTCHER, N. **A basic guide to open educational resources (OER)**. [S.l.]: Vancouver: Commonwealth of Learning; Paris: UNESCO, 2011. Citado nas páginas 23 e 31.

KLIE, J.-C.; BUGERT, M.; BOULLOSA, B.; CASTILHO, R. E. de; GUREVYCH, I. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In: **Proceedings of the 27th international conference on computational linguistics: system demonstrations**. [S.l.: s.n.], 2018. p. 5–9. Citado na página 113.

KOCHMAR, E. **Introduction to Natural Language Processing**. 2017. <<https://www.youtube.com/watch?v=8S3qHHUKqYk>>. Acessado em 2017-09-04. Citado na página 43.

KOUTSOMITROPOULOS, D. A.; SOLOMOU, G. D.; PAPTAEODOROU, T. S.; ALEXOPOULOS, A. D. The use of metadata for educational resources in digital repositories: Practices and perspectives. **D-Lib Magazine**, Corporation for National Research Initiatives, v. 16, n. 1, p. 3, 2010. Citado na página 33.

LANDAUER, T. K.; DUMAIS, S. T. A solution to plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. **Psychological Review**, v. 104, n. 2, p. 211–240, 1997. Citado na página 117.

LEAL, S.; RODRIGUES, E. dos S.; ALUÍSIO, S. Using eye-tracking data to predict the readability of brazilian portuguese sentences in single-task, multi-task and sequential transfer learning approaches. In: **Proceedings of the 28th International Conference on Computational Linguistics (COLING)**. [S.l.]: Association for Computational Linguistics, 2020. p. 1–9. Citado nas páginas 27, 69 e 72.

LEAL, S. E.; VIEIRA, J. M. M.; RODRIGUES, E. dos S.; TEIXEIRA, E. N.; ALUÍSIO, S. M. Using eye-tracking data to predict the readability of brazilian portuguese sentences in single-task, multi-task and sequential transfer learning approaches. In: **The 28th International Conference on Computational Linguistics (COLING)**. Barcelona, Spain: [s.n.], 2020. p. ?–? Citado na página 109.

LEARY, H.; RECKER, M.; WALKER, A.; WETZLER, P.; SUMNER, T.; MARTIN, J. Automating open educational resources assessments: a machine learning generalization study. In: ACM. **Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries**. [S.l.], 2011. p. 283–286. Citado nas páginas 26, 54 e 55.

LEARY, H.; WALKER, A.; RECKER, M.; GIERSCH, S. **Developing and Using a Guide to Assess Learning Resource Quality in Educational Digital Libraries**. [S.l.]: INTECH Open Access Publisher, 2011. Citado nas páginas 36, 37, 38 e 39.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, Nature Research, v. 521, n. 7553, p. 436–444, 2015. Citado nas páginas 43, 44 e 45.

- LI, J.; LIU, X.; YIN, W.; YANG, M.; MA, L.; JIN, Y. Empirical evaluation of multi-task learning in deep neural networks for natural language processing. **Neural Computing and Applications**, Springer, p. 1–12, 2020. Citado na página 48.
- LING, W.; DYER, C.; BLACK, A. W.; TRANCOSO, I. Two/too simple adaptations of word2vec for syntax problems. In: **Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**. [S.l.: s.n.], 2015. p. 1299–1304. Citado na página 47.
- LITTLE, S.; MIKROYANNIDIS, A.; OKADA, A.; SCOTT, P. Formal metadata and shared experiences for discovering tools to adapt open educational resources. In: **Signal-Image Technology and Internet-Based Systems (SITIS), 2011 Seventh International Conference on**. [S.l.: s.n.], 2011. p. 147–153. Citado na página 32.
- LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2015. p. 3431–3440. Citado na página 42.
- LU, X. Automatic analysis of syntactic complexity in second language writing. **International Journal of Corpus Linguistics**, John Benjamins, v. 15, n. 4, p. 474–496, 2010. ISSN 1384-6655. Disponível em: <<https://www.jbe-platform.com/content/journals/10.1075/ijcl.15.4.02lu>>. Citado na página 72.
- MANNING, C. **CS224n: Natural Language Processing with Deep Learning**. [S.l.]: Stanford University, 2019. Citado nas páginas 49 e 50.
- MARCUSCHI, L. A. **Análise da conversação**. [S.l.]: Ática, 1986. (Série Princípios). Citado na página 112.
- MARTÍN, A. *et al.* Minds on fire: Open education, the long tail and learning 2.0. *Càtedra UNESCO en e-Learning*, 2008. Citado na página 23.
- MATKIN, G. W. Open educational resources in the post mooc era. **eLearn**, ACM, New York, NY, USA, v. 2013, n. 4, abr. 2013. ISSN 1535-394X. Disponível em: <<http://dl.acm.org/citation.cfm?id=2460459.2460460>>. Citado na página 32.
- MATSUNE, H. **Using headings to improve visual readability**. 2007. <<https://goo.gl/EXw1r4>>. Acessado em: 2017-09-04. Citado na página 86.
- MCCLELLAND, M. Metadata standards for educational resources. **Computer**, v. 36, n. 11, p. 107–109, 2003. ISSN 0018-9162. Citado na página 32.
- MEC. **Termos de Serviços - Plataforma MEC-RED**. 2019. <<https://plataformaintegrada.mec.gov.br/termos-de-uso>>. Acessado em: 2019-01-24. Citado nas páginas 24, 25 e 39.
- MEDELYAN, O.; MILNE, D.; LEGG, C.; WITTEN, I. H. Mining meaning from wikipedia. **International Journal of Human-Computer Studies**, Elsevier, v. 67, n. 9, p. 716–754, 2009. Citado na página 86.
- MIAO, F.; MISHRA, S.; MCGREAL, R. **Open educational resources: policy, costs, transformation**. [S.l.]: UNESCO Publishing, 2016. Citado nas páginas 24 e 39.
- MIKOLOV, T. **Statistical Language Models Based on Neural Networks**. Tese (Doutorado) — Brno University of Technology, 2012. Citado na página 44.

MIKOLOV, T. Statistical language models based on neural networks. **Presentation at Google, Mountain View, 2nd April**, 2012. Citado na página 47.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient estimation of word representations in vector space. In: **Proceedings of International Conference on Learning Representations Workshop (ICLR-2013)**. [S.l.: s.n.], 2013. Citado na página 47.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed representations of words and phrases and their compositionality. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2013. p. 3111–3119. Citado nas páginas 42 e 43.

MIKOLOV, T.; YIH, W.-t.; ZWEIG, G. Linguistic regularities in continuous space word representations. In: **hlt-Naacl**. [S.l.: s.n.], 2013. v. 13, p. 746–751. Citado na página 47.

MOISE, G.; VLADOIU, M.; CONSTANTINESCU, Z. Maseco—a multi-agent system for evaluation and classification of oers and ocw based on quality criteria. 2011. Citado nas páginas 26, 56, 57 e 58.

NADEEM, F.; OSTENDORF, M. Estimating linguistic complexity for science texts. In: **Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications**. [S.l.: s.n.], 2018. p. 45–55. Citado nas páginas 27 e 71.

OKADA, A. Knowledge media technologies for open learning in online communities. **The International Journal of Technology, knowledge & Society**, v. 3, 2007. Citado na página 32.

OPENCONTENT. **Open Content**. 2020. <<http://opencontent.org/definition/>>. Acessado em: 2020-11-01. Citado na página 31.

_____. **Recursos Educacionais Abertos**. 2020. <<https://www.aprendizagemaberta.com.br/page/recursos-educacionais-abertos>>. Acessado em: 2020-11-01. Citado na página 31.

PARK, J.-R.; TOSAKA, Y. Metadata quality control in digital repositories and collections: Criteria, semantics, and mechanisms. **Cataloging & Classification Quarterly**, Taylor & Francis, v. 48, n. 8, p. 696–715, 2009. Citado na página 35.

PASCANU, R. On recurrent and deep neural networks. 2014. Citado na página 46.

PASCANU, R.; MIKOLOV, T.; BENGIO, Y. On the difficulty of training recurrent neural networks. In: **Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013**. [s.n.], 2013. p. 1310–1318. Disponível em: <<http://jmlr.org/proceedings/papers/v28/pascanu13.html>>. Citado na página 44.

PENNINGTON, J.; SOCHER, R.; MANNING, C. D. Glove: Global vectors for word representation. **Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP-2014)**, v. 12, p. 1532–1543, 2014. Citado na página 47.

PETERS, M. E.; NEUMANN, M.; IYYER, M.; GARDNER, M.; CLARK, C.; LEE, K.; ZETLEMOYER, L. Deep contextualized word representations. In: **Proc. of NAACL**. [S.l.: s.n.], 2018. Citado na página 101.

PLOTKIN, H. Free to learn: An open educational resources policy development guidebook for community college governance officials. **Creative Commons, San Francisco, CA**, 2010. Citado na página 23.

- PONS, D.; HILERA, J. R.; FERNANDEZ, L. *et al.* Managing the quality of e-learning resources in repositories. **Computer Applications in Engineering Education**, Wiley Online Library, v. 23, n. 4, p. 477–488, 2015. Citado nas páginas 26, 62 e 63.
- POWELL, A.; NILSSON, M.; A., N.; JOHNSTON, P.; BAKER, T. **DCMI Abstract Model**. 2007. <<http://dublincore.org/documents/2007/06/04/abstract-model>>. Acessado em: 2013-09-11. Citado na página 33.
- PUSTEJOVSKY, J.; STUBBS, A. **Natural Language Annotation for Machine Learning - a Guide to Corpus-Building for Applications**. O'Reilly, 2012. ISBN 978-1-449-30666-3. Disponível em: <<http://www.oreilly.de/catalog/9781449306663/index.html>>. Citado na página 108.
- RAMOS, J. *et al.* Using tf-idf to determine word relevance in document queries. In: **Proceedings of the first instructional conference on machine learning**. [S.l.: s.n.], 2003. v. 242, p. 133–142. Citado na página 82.
- RAZMERITA, L.; ANGEHRN, A.; MAEDCHE, A. Ontology-based user modeling for knowledge management systems. In: SPRINGER. **International Conference on User Modeling**. [S.l.], 2003. p. 213–217. Citado na página 61.
- ROCKTÄSCHEL, T.; GREFENSTETTE, E.; HERMANN, K. M.; KOČISKÝ, T.; BLUNSOM, P. Reasoning about entailment with neural attention. **arXiv preprint arXiv:1509.06664**, 2015. Citado na página 44.
- RODRIGUES, S. B.; SILVA, A. C. L. da; CORRÊA, J. M.; NETO, A. C. Combinação preditiva das técnicas arima e support vector regression (svr). **Trabalho técnico - UFPR**, 2014. Citado na página 53.
- RUDER, S. An overview of multi-task learning in deep neural networks. **arXiv preprint arXiv:1706.05098**, 2017. Citado nas páginas 48 e 49.
- _____. An overview of multi-task learning in deep neural networks. **CoRR**, abs/1706.05098, 2017. Disponível em: <<http://arxiv.org/abs/1706.05098>>. Citado na página 109.
- RUSSELL, S.; NORVIG, P. Ai a modern approach. **Learning**, v. 2, n. 3, p. 4, 2005. Citado na página 42.
- SANTOS, L. B. dos; DURAN, M. S.; HARTMANN, N. S.; JR., A. C.; PAETZOLD, G. H.; ALUÍSIO, S. M. A lightweight regression method to infer psycholinguistic properties for brazilian portuguese. **CoRR**, abs/1705.07008, 2017. Citado na página 92.
- _____. A lightweight regression method to infer psycholinguistic properties for brazilian portuguese. In: EKSTEIN, K.; MATOUSEK, V. (Ed.). **Text, Speech, and Dialogue - 20th International Conference, TSD 2017, Prague, Czech Republic, August 27-31, 2017, Proceedings**. Springer, 2017. (Lecture Notes in Computer Science, v. 10415), p. 281–289. Disponível em: <https://doi.org/10.1007/978-3-319-64206-2_32>. Citado na página 114.
- SAWADOGO, D.; CHAMPAGNAT, R.; ESTRAILLIER, P. User Profile Modelling for Digital Resource Management Systems. In: **The 22nd Conference on User Modeling, Adaptation and Personalization**. Aalborg, Denmark: [s.n.], 2014. Disponível em: <<https://hal.archives-ouvertes.fr/hal-01086269>>. Citado nas páginas 26, 60 e 61.

SCARTON, C.; GASPERIN, C.; ALUÍSIO, S. Revisiting the readability assessment of texts in portuguese. **Advances in Artificial Intelligence – IBERAMIA - Volume 6433 of Lecture Notes in Computer Science**, Springer Berlin Heidelberg, p. 306–315, 2010. Citado na página 92.

SCARTON, C. E.; ALUÍSIO, S. M. Análise da inteligibilidade de textos via ferramentas de processamento de língua natural: adaptando as métricas do coh-metrix para o português. **Linguamática**, v. 2, n. 1, p. 45–61, 2010. Citado nas páginas 27 e 70.

SILVER, D.; HUANG, A.; MADDISON, C. J.; GUEZ, A.; SIFRE, L.; DRIESSCHE, G. V. D.; SCHRITTWIESER, J.; ANTONOGLU, I.; PANNEERSHELVAM, V.; LANCTOT, M. *et al.* Mastering the game of go with deep neural networks and tree search. **Nature**, Nature Publishing Group, v. 529, n. 7587, p. 484–489, 2016. Citado na página 44.

SMITH, M. S.; CASSERLY, C. M. The promise of open educational resources. **Change: The Magazine of Higher Learning**, Taylor & Francis, v. 38, n. 5, p. 8–17, 2006. Citado nas páginas 23 e 24.

SUTSKEVER, I.; MARTENS, J.; HINTON, G. E. Generating text with recurrent neural networks. In: **Proceedings of the 28th International Conference on Machine Learning (ICML-11)**. [S.l.: s.n.], 2011. p. 1017–1024. Citado na página 44.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to sequence learning with neural networks. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2014. p. 3104–3112. Citado na página 44.

UNESCO. **Forum on the impact of open courseware for higher education in developing countries:: final report**. [S.l.]: UNESCO Paris, 2002. Citado na página 23.

VAJJALA, S.; MEURERS, D. On improving the accuracy of readability classification using insights from second language acquisition. In: **Proceedings of the Seventh Workshop on Building Educational Applications Using NLP**. Montréal, Canada: Association for Computational Linguistics, 2012. p. 163–173. Disponível em: <<https://www.aclweb.org/anthology/W12-2019>>. Citado nas páginas 71 e 72.

_____. Assessing the relative reading level of sentence pairs for text simplification. In: **Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics**. [S.l.: s.n.], 2014. p. 288–297. Citado na página 109.

VASAN, K. K.; SURENDIRAN, B. Dimensionality reduction using principal component analysis for network intrusion detection. **Perspectives in Science**, v. 8, p. 510 – 512, 2016. ISSN 2213-0209. Recent Trends in Engineering and Material Sciences. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2213020916301446>>. Citado na página 96.

WETZLER, P.; BETHARD, S.; LEARY, H.; BUTCHER, K.; BAHREINI, S. D.; ZHAO, J.; MARTIN, J. H.; SUMNER, T. Characterizing and predicting the multifaceted nature of quality in educational web resources. **ACM Transactions on Interactive Intelligent Systems (TiiS)**, ACM, v. 3, n. 3, p. 15, 2013. Citado nas páginas 26, 59 e 60.

WILEY, D.; BLISS, T.; MCEWEN, M. Open educational resources: A review of the literature. Springer New York, p. 781–789, 2014. Citado nas páginas 23, 24 e 39.

WILEY, D. A. **The Learning Objects Literature**. [s.n.], 2002. Disponível em: <www.opencontent.org/docs/wiley-lo-review-final.pdf?> Citado na página 23.

WU, C.-Y.; AHMED, A.; BEUTEL, A.; SMOLA, A. J.; JING, H. Recurrent recommender networks. In: **Proceedings of the Tenth ACM International Conference on Web Search and Data Mining**. New York, NY, USA: ACM, 2017. (WSDM '17), p. 495–503. ISBN 978-1-4503-4675-7. Citado na página 44.

PCA USANDO VARIMAX

Tabela 34 – PCA aplicado usando a Linguagem R, normalização Z-Score e Varimax para 75 features selecionadas - Parte I

Principal Componentes	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
	Words							
Syllables per content word				(-)				
Noun ratio		(-)						
Verbs		(+)						
Adjective ratio								
Adverbs		(+)						
Pronoun ratio		(+)						(+)
Personal pronouns		(+)						
Third person pronouns								
Ratio function to content words		(+)		(-)				
Conn ratio		(+)						(-)
Cau pos conn ratio		(+)					(-)	
Cau neg conn ratio								
Tmp neg conn ratio								
Tmp pos conn ratio								
Log neg conn ratio								
Log pos conn ratio	(+)	(+)					(-)	
Add neg conn ratio								
Add pos conn ratio								
Cw freq								
Min cw freq								
Idade aquisicao mean	(+)	(-)	(+)					
Idade aquisicao std						(-)		
Idade aquisicao 1 of 2.5 ratio	(-)		(-)					
Idade aquisicao 4 of 5.5 ratio								
Idade aquisicao 5.5 of 7 ratio	(+)		(+)					
Idade aquisicao 2.5 of 4 ratio		(+)	(-)					
Concretude mean	(-)							
Concretude std	(-)							
Concretude 1 of 2.5 ratio								
Concretude 2.5 of 4 ratio	(+)	(+)						
Concretude 4 of 5.5 ratio	(-)	(-)						
Concretude 5.5 of 7 ratio	(-)		(-)				(-)	
Imageabilidade mean	(-)		(-)					
Imageabilidade std	(-)							
Imageabilidade 1 of 2.5 ratio								
Imageabilidade 2.5 of 4 ratio	(+)							
Imageabilidade 4 of 5.5 ratio	(-)							
Imageabilidade 5.5 of 7 ratio	(-)		(-)					
Familiaridade mean		(+)			(-)			
Familiaridade std					(+)	(-)		
Familiaridade 1 of 2.5 ratio								
Familiaridade 4 of 5.5 ratio					(-)	(+)		
Familiaridade 2.5 of 4 ratio					(+)			
Familiaridade 5.5 of 7 ratio		(+)	(-)					
Negation ratio		(+)						
Adjectives ambiguity								
Adverbs ambiguity								
Nouns ambiguity	(+)							
Verbs ambiguity								

Fonte: Elaborada pelo autor.

Tabela 35 – PCA aplicado usando a Linguagem R, normalização Z-Score e Varimax para 75 features selecionadas - Parte II

Principal Componentes	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
Sentence structure								
words_per_sentence	(+)		(-)	(-)				
sentence_length_max	(+)		(-)					
sentence_length_min	(+)		(-)	(-)				
sentence_length_standard_deviation	(+)							
np_ratio								
words_before_main_verb								
Connections between sentences								
adj_cw_ovl	(+)		(-)					
adj_arg_ovl	(+)		(-)					
arg_ovl	(+)		(-)					
adj_stem_ovl	(+)		(-)					
stem_ovl	(+)		(-)					
ttr	(-)						(+)	
content_word_diversity	(-)			(-)				
verb_diversity	(-)							
givenness_mean	(+)			(+)				
adj_mean	(+)							
all_mean	(+)							
yngve	(+)		(-)					
frazier	(+)							
span_mean	(+)			(+)				
dep_distance	(+)		(-)	(-)				
all_std			(+)	(+)				
content_density		(-)						
adj_std			(+)	(+)				
givenness_std			(+)	(+)				
span_std	(-)		(+)					

Fonte: Elaborada pelo autor.

