UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

CELSO ROMÃO CARDOSO DE ALMEIDA JÚNIOR

PROPOSTA DE UM SISTEMA AUTOMÁTICO DE AVALIAÇÃO DE REDAÇÕES DO ENEM, FOCO NA COMPETÊNCIA 1: DEMONSTRAR DOMÍNO DA MODALIDADE ESCRITA FORMAL DA LÍNGUA PORTUGUESA

VITÓRIA
2017

CELSO ROMÃO CARDOSO DE ALMEIDA JÚNIOR

PROPOSTA DE UM SISTEMA AUTOMÁTICO DE AVALIAÇÃO DE REDAÇÕES DO ENEM, FOCO NA COMPETÊNCIA 1: DEMONSTRAR DOMÍNO DA MODALIDADE ESCRITA FORMAL DA LÍNGUA PORTUGUESA

Dissertação apresentada ao Programa de Pós-graduação em Informática do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Informática.
Orientador: Prof. Dr. Elias de Oliveira.

VITÓRIA
2017

[reservado para a ficha catalográfica]

1.

Celso Romão Cardoso de Almeida Jr.

PROPOSTA DE UM SISTEMA AUTOMÁTICO DE AVALIAÇÃO DE REDAÇÕES DO ENEM, FOCO NA COMPETÊNCIA 1: DEMONSTRAR DOMÍNO DA MODALIDADE ESCRITA FORMAL DA LÍNGUA PORTUGUESA

Dissertação submetida ao programa de Pós-Graduação em Informática do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Informática.

Aprovada em 03 de agosto de 2017.

COMISSÃO EXAMINADORA

_____

Prof. Dr. Elias, de Oliveira

Universidade Federal do Espírito Santo

Orientador

_____

Profa. Dra. Priscila Machado Vieira Lima

Universidade Federal do Rio de Janeiro

_____

Prof. Dr. Davidson Cury

Universidade Federal do Espírito Santo

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus, por seu fiel apoio em todos os momentos, permitindo que eu chegasse até o fim desta jornada.

A minha esposa, Núbia, que sempre me apoiou, mesmo tendo eu estado muitas vezes distante nesse período.

A meu orientador, pela paciência e dedicação ao me guiar nesta empreitada, pelas suas correções e incentivos.

Também sou grato aos demais colegas de mestrado, aos professores do Programa de Pós-graduação em Informática e aos profissionais da Secretaria. Obrigado pelo companheirismo, pela contribuição nas diversas disciplinas oferecidas e pelo auxílio em tantos momentos durante o período do mestrado.

A meus pais, Wilma Maia de Almeida e Celso Romão Cardoso de Almeida, e a todos os que, direta ou indiretamente, fizeram parte da minha formação, o meu muito obrigado.

A Deus, minha esposa, minha família, amigos, colegas de trabalho e orientador, pelo apoio, força, incentivo, companheirismo e amizade. Sem eles, nada disso seria possível.

Só sei que nada sei e o fato de saber isso me coloca em vantagem sobre aqueles que acham que sabem alguma coisa.

Sócrates

PUBLICAÇÕES

Como parte deste estudo, foram desenvolvidos e publicados em anais de congressos os trabalhos relacionados a seguir, os quais apresentam, em maior ou menor grau, relação com o tema proposto:

- ALMEIDA, C. R. C.; OLIVEIRA, E. Proposta de um sistema de avaliação automática de redações do Enem. WORKSHOP DE PESQUISA E DESENVOLVIMENTO EM INTELIGÊNCIA ARTIFICIAL, INTELIGÊNCIA COLETIVA E CIÊNCIA DOS DADOS.2. 2016, Niterói.

- ALMEIDA, C. R. C.; SPALENZA, M.; OLIVEIRA, E. Proposta de um sistema de avaliação automática de redações do Enem utilizando técnicas de aprendizagem de máquina e processamento de linguagem natural. COMPUTER ON THE BEACH, 8. 2017, Florianópolis. (Premiação: Menção honrosa).

RESUMO

As avaliações automáticas de redações são amplamente praticadas na Língua Inglesa. Porém, na Língua Portuguesa, não podemos dizer o mesmo. A redação é uma das competências exigidas pelo Exame Nacional do Ensino Médio (Enem), porta de entrada para a maioria das universidades do Brasil. O alto custo e o grande número de profissionais que trabalham no processo de correção das redações do Enem são alguns dos fatores que motivam pesquisas na área de avaliação automática de redação. Este trabalho apresenta uma estratégia para melhorar a produtividade do avaliador, reduzindo o esforço em 20% do tempo ao avaliar a primeira das cinco competências avaliadas nas redações do referido exame. Para isso, propusemos a construção de um sistema de avaliação automática de redações, na Competência 1 do Enem (demonstrar domínio da modalidade escrita formal da Língua Portuguesa). Na construção do sistema, utilizamos técnicas e ferramentas de processamento de linguagem natural na etapa de pré-processamento das redações, além de técnicas de aprendizagem de máquina nas etapas de seleção de características e predição das notas. Os resultados dos experimentos realizados com redações do site UOL mostram que o sistema é capaz de apoiar o avaliador de redações do Enem, com um erro médio absoluto de 0,2354 em 2,0 em relação às notas atribuídas pelos especialistas do site.

Palavras-chave: processamento de linguagem natural, aprendizagem de máquina, redação, ENEM.

ABSTRACT

Automatic essay assessments are widely practiced in English, but in Portuguese, we cannot say the same. Writing is one of the competencies required by the National High School Examination (ENEM), gateway to most universities in Brazil. The high cost and the large number of professionals working in the correction process of the ENEM essays are some of the factors that motivate research in the area of automatic essay evaluation. This work presents a strategy to improve the evaluator's productivity, reducing the effort in 20% of the time, evaluating Competence 1, one of the five competences evaluated in the essays of ENEM. For this, we propose the construction of a system of automatic evaluation of essays, in Competence 1 of the ENEM; demonstrate mastery of the formal written form of the Portuguese language. In the construction of the system, we use Natural Language Processing techniques and tools, in the preprocessing stage of the essays as well as Machine Learning techniques in the stages of selection of characteristics and prediction of the grades. The results of the experiments carried out with the UOL website show that the system is able to support the ENEM essay evaluator with an absolute mean error of 0.2354 in 2.0 compared to the scores attributed by the site experts.

Keywords: natural language processing, machine learning, artificial intelligence, essay, ENEM.

LISTA DE SIGLAS

Enem – Exame Nacional do Ensino Médio

MAE – Erro Médio Absoluto

PLN – Processamento de Linguagem Natural

PSO – *Particle Swarm Optmizatiom*

SPLN – Sistemas de Processamento de Linguagem Natural

SVM – *Support Vector Machine*

GB – Gradient Boosting

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

O propósito deste capítulo é proporcionar ao leitor um melhor entendimento do contexto e problema que buscamos solucionar, a saber, minimizar o esforço humano e diminuir os custos no processo de avaliação de redações, especificamente, tomando como base a Competência 1 do Exame Nacional do Ensino Médio (Enem) – “demonstrar domínio da modalidade escrita formal da Língua Portuguesa”. Descrevemos, ainda, a motivação, os objetivos e os procedimentos metodológicos utilizados na realização do estudo aqui relatado. No fim do capítulo, apresentamos a estrutura da dissertação, detalhando o conteúdo de cada capítulo.

1.1 CONTEXTO

A redação vale 20% da pontuação na prova do Enem, sendo a única prova do exame que não é submetida ao método Teoria de Resposta ao Item (TRI) (PASQUALI, 2003), no qual o valor de cada questão varia de acordo com o percentual de acertos e erros do estudante. Portanto, escrever bem, seguindo as normas cultas da Língua Portuguesa, é essencial para quem deseja ingressar em uma instituição de ensino superior no Brasil.

O custo total estimado do Contrato n° 12/2016, firmado junto ao Consórcio Cesgranrio-Cebraspe, para a correção das redações do Enem é de R$ 117.419.455,93[1]. O Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP) repassou ao consórcio o valor de R$ 4,47 para a primeira e segunda correção e de R$ 5,85 para a terceira correção. O processo de mensuração da produtividade média diária apurada em relação aos avaliadores registrou que cerca de 47% deles avaliaram entre 50 e 100 redações por dia[2]. Entretanto o valor pago aos avaliadores é menor já que existem descontos que são repassados. Avaliadores relatam que o processo é massacrante devido à grande quantidade de trabalho, por isso passam uma grande quantidade de erros de correção. Os dados mostram a grandiosidade do que é a avaliação de redação do Enem. Esse cenário desafiador

[1] Informação enviada pelo Inep, após solicitação feita via portal do órgão.
[2] Idem.

configura-se uma excelente oportunidade para a aplicação de técnicas de avaliações automáticas realizadas por computadores.

Em consequência da importância da redação no Enem, escolas têm dedicado uma atenção especial à preparação dos alunos para a prova de redação, criando técnicas pedagógicas para tornar a escrita mais prazerosa e eficiente. Além disso, é importante destacar o surgimento de plataformas privadas para correção e avaliação de redações no modelo Enem, tal como a mostrada na Figura 1.1.Nelas, contratando um pacote mensal, os estudantes podem submeter suas redações para que os especialistas da plataforma as corrijam e comentem.

Figura 1.1 – Plataforma de correção de redação *Imaginie*

Fonte: Plataforma *Imaginie* (acesso em: 6 nov. 2016).

Um exemplo de plataforma gratuita é o site UOL, mostrado na Figura 1.2.O site permite que seus usuários submetam, mensalmente, suas redações, elaboradas com base em um tema proposto para avaliação. Foi justamente em cima desse conjunto de dados[3] que elaboramos estratégias para a construção de um sistema de avaliação automática de redações, na Competência 1 do Enem.

[3]http://educacao.uol.com.br/bancoderedacoes/

Figura 1.2 – Plataforma de correção de redação UOL

Redações corrigidas

Título	Nota
Busca constante por beleza	5,0
O desinteresse da beleza moderna	7,5
Estética e seu verdadeiro significado	7,0
A beleza e o capital	9,5
(Sem título 079)	6,0
Aceitação ou não, eis a questão	4,5
Corpolatria	9,5
Saúde e bem estar [bem-estar]	6,5
O zelo sem exagero	6,0
Violência Travestida	6,0
Relação beleza e saúde	5,0

Fonte: Brasil Escola (acesso em: 6nov. 2016).

1.1.1 O processo de avaliação de redação do Enem

Conforme os critérios estabelecidos pelo anexo IV do Edital Enem 2016 (INEP, 2016), para cada redação, cada um dos dois avaliadores atribui uma nota de 0 a 200 em cada uma das cinco competências, que são as seguintes:

I. demonstrar domínio da modalidade escrita formal da língua portuguesa;
II. compreender a proposta de redação e aplicar conceitos das várias áreas de conhecimento para desenvolver o tema, dentro dos limites estruturais do texto dissertativo-argumentativo em prosa;
III. selecionar, relacionar, organizar e interpretar informações, fatos, opiniões e argumentos em defesa de um ponto de vista;
IV. demonstrar conhecimento dos mecanismos linguísticos necessários para a construção da argumentação;
V. elaborar proposta de intervenção para o problema abordado, respeitando os direitos humanos.

A nota total atribuída por cada avaliador é a soma das notas atribuídas em cada competência. Caso não haja discrepância caracterizada pela diferença de mais de 100 pontos na nota total ou superior a 80 pontos em uma das competências, a nota final do estudante será a média aritmética das notas totais atribuídas pelos dois avaliadores. Existindo tal discrepância, a redação será corrigida por um terceiro avaliador e a nota final será a média aritmética das notas totais que mais se aproximarem. Se a nota do terceiro avaliador apresentar discrepância com as notas dos dois anteriores, forma-se uma banca composta pelo terceiro avaliador e outros dois novos, descartando-se as notas atribuídas pelos dois avaliadores que iniciaram a correção do texto.

Na avaliação da Competência 1, foco deste trabalho, são observados os desvios gramaticais, como sintaxe de concordância, regência e colocação, pontuação, flexão, entre outros. Em relação às convenções da escrita, espera-se do participante o domínio e o respeito às particularidades dessa modalidade de expressão. São avaliados também ortografia, acentuação e uso adequado de letras maiúsculas e minúsculas. Características comuns ao “internetês” e uso de gírias são consideradas desvio da norma culta. Importante ressaltar que existem outras normas além da norma padrão, como a norma culta urbana, em cada uma delas o que é ou não considerado erro é diferente e variável. No entanto, neste trabalho consideramos a norma padrão, adotada no ENEM.

A Tabela 1.1 mostra os níveis de conhecimento avaliados na Competência 1, estabelecidos pelo anexo IV do Edital Enem 2016.

Tabela 1.1 – Níveis de conceitos na Competência 1

Nível	Proficiência	Pontuação
0	Muito baixa ou ausente	0%
1	Baixa	20%
2	Mediana	40%
3	Boa	60%
4	Muito boa	80%
5	Excelente	100%

Fonte: INEP (2016).

1.2 MOTIVAÇÃO

O processo de avaliação de redações do Enem envolve centenas de pessoas avaliando dezenas de redações por dia. Esses especialistas relatam algumas dificuldades inerentes ao processo. O alto número de redações avaliadas por dia foi citado como fator determinante na queda da qualidade da correção. Outra reclamação dos avaliadores é a dificuldade de compreensão e diferenciação dos níveis de conhecimentos associados a cada competência (LUNA, 2009).

Assim, nossa motivação com este trabalho é minimizar o esforço e o tempo do docente na avaliação de redação no modelo do Enem, tomando como base a primeira das cinco competências avaliadas em uma redação do referido exame – Competência 1: demonstrar domínio da modalidade escrita formal da Língua Portuguesa.

Por fim, pretendemos com este trabalho disseminar uma nova estratégia para avaliação automática de redações. Nessa estratégia uma das 5 competências é avaliada separadamente das outras.

1.3 OBJETIVOS

Nosso objetivo é construir uma ferramenta de avaliação automática de redações no modelo do Enem, especificamente, tomando como base a Competência 1. Foram pesquisadas áreas da inteligência artificial e da linguística, destacando-se o processamento de linguagem natural e a aprendizagem de máquina.

Além disso, pretendemos:

I. disponibilizar as bases de dados utilizadas neste trabalho;
II. integrar o sistema com o ambiente de aprendizagem *Moodle*, para que sejam usados por docentes de Língua Portuguesa que se dedicam ao ensino e avaliação de produção textual.

1.4 PROCEDIMENTOS METODOLÓGICOS

Inicialmente, realizamos uma revisão bibliográfica, por meio da qual pesquisamos por trabalhos relacionados com avaliação automática de redações e similares. O propósito dessa etapa foi avaliar as principais estratégias, técnicas e ferramentas utilizadas.

Nesse processo, pesquisamos os principais revisores gramaticais para o português do Brasil, o *ReGra*[4] e *CoGrOO*[5]. Foram pesquisados também técnicas e *frameworks* de processamento de linguagem natural, como *Apache OpenNLP*[6], a serem utilizados em tarefas de pré-processamento de textos, que constituem parte do trabalho aqui relatado.

Com a revisão, analisamos algumas técnicas de classificação, em especial as que utilizam a estratégia *multclass*, pois em nosso problema uma redação pode pertencer a uma das cinco classes, que neste caso, se referem às notas que o avaliador atribuiu à redação em dada competência (0,0, 0,5, 1,0, 1,5 e 2,0). Outra estratégia de classificação pesquisada foi a combinação de vários classificadores, que forma estruturas conhecidas como *ensembles*. Para melhorar a eficiência do classificador, foram estudadas técnicas de redução de dimensionalidade e seleção de características, incluindo o uso do algoritmo evolucionário *Particle Swarm Optmizatiom* (PSO).

Na composição de nossa base de dados, utilizamos textos de bancos de redações do site UOL. As redações haviam sido corrigidas e comentadas pelo humano especialista do site em cada uma das cinco competências do Enem.

1.5 ESTRUTURA DO TRABALHO

Além desta Introdução, este trabalho está organizado do seguinte modo:

- Capítulo 2 (**Trabalhos relacionados**): apresenta trabalhos relacionados ao objeto da pesquisa aqui relatada.

---

[4] http://www.nilc.icmc.usp.br/nilc/projects/regra.htm
[5] http://cogroo.sourceforge.net/
[6] https://opennlp.apache.org/

- Capítulo 3 (**Conceitos básicos**): discorre sobre alguns conceitos de processamento de linguagem natural e aprendizagem de máquina, os quais foram utilizados neste trabalho.

- Capítulo 4 (**Avaliação automática de redações**): apresenta a estratégia utilizada na construção do sistema.

- Capítulo 5 (**Experimentos e resultados**): descreve e discute os resultados obtidos nos experimentos com uso das várias versões da metodologia proposta, apresentando, por fim, alguns exemplos da avaliação de redação efetuada pelo sistema.

- Capítulo 6 (**Conclusões e trabalhos futuros**): apresenta as conclusões do estudo, suas contribuições e propostas futuras de aprimoramento da tarefa de avaliação de redação usando aplicações computacionais.

2 TRABALHOS RELACIONADOS

Neste capítulo, será realizada uma análise de trabalhos que de algum modo se relacionam com o objeto deste estudo e que lhe serviram de base. Assim, serão apresentadas pesquisas na área de avaliação de questões discursivas apoiadas por computador, revisores gramaticais para o português do Brasil e avaliação automática de redações na Língua Portuguesa.

2.1 BREVE HISTÓRICO DE AVALIAÇÃO AUTOMÁTICA

Pesquisas na área de avaliação automática de atividades escritas vêm crescendo de forma significativa com os avanços da computação. Page (1966) apresentou um trabalho que descreveu o *Project Essay Grade* (PEG), primeiro corretor completo de redações. Esse trabalho observou informações relacionadas ao padrão de escrita do aluno, como: quantidade e tamanho de palavras, número de preposições, conectivos, erros gramaticais, entre outros, num total de 30 variáveis. O PEG apresentou uma correlação múltipla de 71% entre as variáveis coletadas e as notas atribuídas. Entretanto, ao observar a qualidade da escrita, desconsiderava seu conteúdo.

Nos anos 1990, com o aparecimento de novas tecnologias na área de recuperação da informação (RI) e processamento de linguagem natural, surgiram novos trabalhos na área, como *Criterion Online Essay Evaluation* (BURSTEIN; CHODOROW; LEACOCK, 2003), um sistema que combina avaliação automática e diagnóstico de *feedback*. Tal sistema é composto por duas aplicações baseadas em método de processamento de linguagem natural. A primeira, *e-rater* (BURSTEIN et al., 1998), extrai as características linguísticas de uma redação e usa modelo estatístico para atribuir uma nota. A segunda aplicação, *Critique*, avalia erros gramaticais, mecânicos, estruturais e fornece um *feedback*.

2.2 AVALIAÇÃO AUTOMÁTICA DE REDAÇÃO NO BRASIL

Também utilizando textos do banco de redações do site UOL, tal como nosso trabalho, Bazelato e Amorim (2013) construíram um sistema de avaliação automática de redações, usando o teorema de Bayes, definido pela Equação 2.1. O algoritmo dos autores adaptou o teorema para seu problema por meio da Equação 2.2, representando cada redação por um conjunto de termos (palavras). A probabilidade condicional, definida como a probabilidade de uma redação ser atribuída a uma classe (nota), é o produto entre as probabilidades dos termos de uma redação ser atribuída a uma dada classe, conforme mostra a Equação 2.3.

$$P(A|B) = P(B|A) * P(A)/P(B) \tag{2.1}$$

$$P(grade|essay) = P(essay|grade) * P(grade)/P(essay) \tag{2.2}$$

$$P(essay|grade) = \prod P(term|grade) \tag{2.3}$$

Seu sistema considerou a nota total de cada redação variando de 0 a 10, com passos de 0,5. O acerto (*accuracy)* obtido pelo classificador foi de 52%, considerando as notas adjacentes classificadas no máximo 1,5 pontos de distância a partir da nota do avaliador humano.

Pissinati (2014) propôs uma ferramenta de predição semiautomática de notas para questões discursivas na Língua Portuguesa e uma de visualização de informação capaz de apresentar o desempenho dos alunos. A ferramenta de predição semiautomática agrupa as respostas mais similares usando a similaridade cosseno, seleciona uma resposta e envia para correção de um professor. A nota atribuída a essa resposta será replicada para as outras do grupo.

José, Paiva e Bittencourt (2015) propuseram um analisador ortográfico-gramatical para avaliação automática de atividades escritas, utilizando algoritmos genéticos (GA) e processamento de linguagem natural para automatizar a correção léxico-ortográfica. O sistema possui dois módulos, sendo um de correção ortográfica e o

outro, gramatical. O objetivo dos autores foi avaliar a estrutura ortográfica e gramatical de redações do Enem, oferecendo sugestões de correção.

O módulo ortográfico foi criado para realizar a correção léxica das palavras e sugerir outras para substituição. Outra função desse módulo é impedir que palavras escritas incorretamente prejudiquem a correção do módulo gramatical. Além do dicionário *JSpell*, o módulo ortográfico utiliza um dicionário auxiliar, com palavras ausentes no *JSpell*. Para aperfeiçoar as sugestões de correções feitas pelo algoritmo genético, o módulo utiliza a distância de *Levenshtein*, que é dada pelo número mínimo de operações necessárias (inserção, deleção ou substituição de caractere) para transformar uma cadeia de caracteres em outra.

O módulo gramatical reconhece erros de concordância verbal, concordância nominal, o uso de crase e colocação pronominal, entre outros. Utiliza ferramentas de processamento de linguagem natural, como *Apache OpenNLP*, para marcar as classes das palavras. O sistema de código aberto *CoGrOO* é usado para identificar os erros gramaticais.

O experimento foi realizado com 20 redações, dez das quais disponíveis na *Web*, todas escritas no modelo do Enem e corrigidas por professores/tutores. As redações foram escolhidas com base em erros ortográficos e sintáticos, pois o objetivo era avaliar a Competência 1 do Enem, tal como no estudo aqui relatado. A avaliação das redações foi feita em módulos isolados e em conjunto. As métricas utilizadas na avaliação *Precision* e *Recall* são obtidas pelas fórmulas:

$$Prec = \frac{PC}{PC + PI} \tag{2.4}$$

$$Recall = \frac{PC}{PC + AI} \tag{2.5}$$

Sendo:

- PC = Palavras que eram erros e foram corrigidas;
- PI = Palavras que eram erros e não foram corrigidas;
- AI = Palavras que não eram erros, mas foram identificadas como erro.

O módulo ortográfico obteve uma *Precision* igual a 1 e um *Recall* de 0,96. Em um total de134 erros ortográficos, identificou 139 erros, sendo que cinco palavras foram assim classificadas pelo fato de não existirem nos dicionários. Já o módulo gramatical obteve uma taxa de 60% de acertos com base nos erros esperados.

O sistema identifico que os erros ortográficos influenciaram na análise gramatical. Quando os dois módulos atuaram em conjunto, 80% das redações foram corrigidas com sucesso. No entanto, tal sistema, ao contrário do que foi desenvolvido neste trabalho, não atribuiu notas às redações, além de não considerar outras características importantes para a Competência 1 do Enem.

3 CONCEITOS BÁSICOS

Neste capítulo, faremos uma breve descrição de processamento de linguagem natural, aprendizagem de máquina e de outros conceitos técnicos utilizados neste trabalho.

3.1 PROCESSAMENTO DE LINGUAGEM NATURAL

O processamento de linguagem natural (PLN) começou na década de 1950, como uma interseção entre a inteligência artificial e a linguística. Inicialmente, o PLN foi separado da recuperação de informações de texto, que emprega técnicas altamente escalonáveis baseadas em estatísticas, para indexar e pesquisar grandes volumes de texto de forma eficiente (NADKARNI; OHNO-MACHADO; CHAPMAN, 2011).

Silva (2013) afirma que, assim como em outras áreas da inteligência artificial, o processamento de linguagem natural procura imitar o comportamento humano, mais especificamente no processamento e geração de linguagem natural. Para melhor lidar com a complexidade de compreender a linguagem natural, existem vários sistemas, cada um com um objetivo específico. Os principais são:

I. conversão de dados de um computador para a linguagem humana;
II. conversão da linguagem humana para dados de um computador;
III. tradução automática entre linguagens diferentes;
IV. reconhecimento de padrões em um texto como, datas ou telefones;
V. análise de sentimentos, determinando se um comentário é positivo ou negativo;
VI. verificação gramatical automática.

A seguir, descrevemos e exemplificamos algumas tarefas do PLN, como detecção de limites de oração, tokenização e etiquetagem, as quais são utilizadas no processamento das redações realizado no nosso trabalho. Os exemplos são gerados pelo *framework Apache OpenNLP*, mostrado na Seção 3.1.4.

3.1.1 Detecção de limites de oração

Segundo Silva (2013), a detecção de limites de oração tem como função identificar os limites de uma frase, tendo como base sinais de pontuação tais como ponto, ponto-e-vírgula, exclamação e interrogação. Além disso, colchetes, traços e citação somente são considerados se a frase contiver um verbo no infinitivo. Nadkarni, Ohno-Machado e Chapman (2011) alertam para alguns complicadores dessa tarefa, dentre eles, abreviações e separadores de casas decimais.

A seguir, destacamos um exemplo do que seriam a entrada e a saída do processo de detecção dos limites de oração.

- **Entrada:** *Economia doméstica, nada mais é que o núcleo econômico familiar, com todos os gastos pertinentes a uma família. Durante tempos de crise, como a que está ocorrendo no Brasil, deve-se equilibrar as finanças familiares, fazer o orçamento render e ainda prevenir-se para possíveis imprevistos*.

- **Saída:** [Economia doméstica, nada mais é que o núcleo econômico familiar, com todos os gastos pertinentes a uma família.] [Durante tempos de crise, como a que está ocorrendo no Brasil, deve-se equilibrar as finanças familiares, fazer o orçamento render e ainda prevenir-se para possíveis imprevistos.]

3.1.2 Tokenização

Tokens são grupos de textos separados por espaços em uma frase, podendo ser palavras, números, pontos, marcações, entre outros. Na identificação de *tokens*, podemos encontrar alguns problemas. Por exemplo, a falta de espaço entre uma palavra e um sinal de pontuação pode ocasionar uma etiquetagem incorreta do *token*.

A seguir, destacamos um exemplo do que seriam a entrada e a saída do processo de tokenização.

- **Entrada:** *Economia doméstica, nada mais é que o núcleo econômico familiar, com todos os gastos pertinentes a uma família.*

- **Saída:** [Economia] [doméstica] [,] [nada] [mais] [é] [que] [o] [núcleo] [econômico] [familiar] [,] [com] [todos] [os] [gastos] [pertinentes] [a] [uma] [família] [.]

3.1.3 Etiquetagem

Gonzalez e Lima (2003) explicam que a etiquetagem é responsável pela inclusão de uma etiqueta (*tag*) após a palavra, a qual pode conter informações sobre categorias morfológicas, como substantivo e adjetivo, ou funções sintáticas das palavras, como sujeito e objeto direto. Vieira (2000) também destaca a etiquetagem semântica, que anexa informação relacionada ao significado, podendo indicar os papéis dos itens lexicais na sentença, como agente, processo e estado.

Vale salientar que, segundo a Linguística a Norma Gramatical Brasileira possui uma série de incoerências (INFO ESCOLA). Porém não é escopo deste trabalho realizar um estudo sobre essas incoerências.

A seguir, destacamos um exemplo do que seriam a entrada e a saída do processo de etiquetagem.

- **Entrada:** *Economia doméstica, nada mais é que o núcleo econômico familiar, com todos os gastos pertinentes a uma família.*

- **Saída:** Economia_n doméstica,_adj nada_pron-indp mais_adv é_v-fin que_conj-s o_art núcleo_n conômico_adj familiar,_adj com_prp todos_pron-det os_art gastos_n pertinentes_adj a_prp uma_art família._n

3.1.4 *Apache OpenNLP*

No paradigma orientado a objetos, *framework* é um conjunto de classes que colaboram para realizar uma responsabilidade para um domínio de um subsistema da aplicação. Um dos principais *frameworks* de processamento de linguagem

natural, a biblioteca *Apache OpenNLP*[7] é um conjunto de ferramentas baseado na aprendizagem de máquinas para o processamento de texto em linguagem natural. Oferece suporte às tarefas mais comuns de PLN, tais como *tokenização*, segmentação de sentenças, etiquetagem, extração de entidade nomeada e fragmentação. O objetivo do projeto *OpenNLP* é criar um ferramental maduro para essas tarefas e, adicionalmente, fornecer um grande número de modelos pé-construídos para uma variedade de idiomas, bem como os recursos de texto anotado, dos quais esses modelos são derivados.

3.1.5 Recursos linguísticos

3.1.5.1 Dicionários

Guthrie et al. (1996) afirmam que a função dos dicionários (ou léxicos) é fornecer um grande número de informações sobre as palavras, como sentidos, etimologia, pronúncia, morfologia, sintaxe, entre outras. Quanto ao conteúdo, podemos classificar os dicionários em cinco categorias:

I. convencionais, com verbetes em ordem alfabética;
II. analógicos, que organizam os itens lexicais de acordo com seu significado;
III. etimológicos, que se ocupam exclusivamente da origem das palavras;
IV. morfológicos, que apresentam as formas flexionais dos lexemas;
V. de sinônimos e antônimos, com listagens de palavras semelhantes ou opostas em significado.

Segundo Wilks et al. (1996), os dicionários podem ser classificados quanto aos seus objetivos, podendo ser dos seguintes tipos:

I. padrão, que explicam os significados das palavras;
II. *thesauri*, que apontam relacionamentos entre os itens lexicais;
III. bilíngues, que buscam relacionar dois idiomas em nível de equivalência de sentidos das palavras;

[7] https://opennlp.apache.org/

IV. de estilo, que dão orientações sobre o bom uso das regras gramaticais;

V. de concordância, que são essencialmente ferramentas escolares.

Outra categoria de dicionários existe é o Dicionário de usos do Português do Brasil (DUP), organizado por BORBA (2002), que traz um registro lexicográfico da língua escrita no Brasil, na segunda metade do século XX.

Segundo WELKER (2006), o "DUP é o primeiro dicionário geral brasileiro a dar informações sintático-semânticas, imprescindíveis para o uso correto das palavras". O DUP fornece informação sobre a preposição exigida por determinados verbos, substantivos e adjetivos do português (ex.: habituar, confiança e crente) ao contrário do Aurélio, do Michaelis e do Houaiss.

Porém, neste trabalho, o dicionário é uma base de dados composta por palavras da Língua Portuguesa, onde o corretor ortográfico do sistema consulta a existência de cada palavra de uma redação. Os dicionários do sistema são detalhados na Seção 4.2.1.

3.1.5.2 Revisores gramaticais

O grande propósito do uso do processamento de linguagem natural está nos Sistemas de PLN (SPLNs), programas que buscam aproximar o computador do universo linguístico humano, na medida em que adotam a linguagem natural. Os SPLNs são projetados para executar a complexa tarefa de interpretar e gerar informações veiculadas por mensagens linguisticamente construídas. Revisores automáticos são tipos de SPLNs que têm como objetivo principal oferecer ao usuário um recurso de revisão da escrita. De modo geral, esses sistemas são específicos de uma língua (DIAS-DA-SILVA, 1996).

A seguir, faremos uma breve descrição de dois revisores gramaticais para o português brasileiro utilizados neste trabalho: o *ReGra*, propriedade da *Microsoft*,e o *CoGrOO*[8], revisor gramatical de código aberto.

[8] http://cogroo.sourceforge.net/

a) *ReGra*

O *ReGra* é um revisor gramatical para o idioma português do Brasil, desenvolvido pela parceria *Itautec-Philco* e o Núcleo Interinstitucional de Linguística Computacional da Universidade de São Paulo, em 1993. Sua arquitetura compõe-se de três módulos funcionais. O estatístico realiza uma série de cálculos de quantidade de parágrafos, sentenças, palavras. O módulo mecânico localiza erros como palavras ou símbolo repetidos, balanceamento de parênteses, aspas, entre outros. O módulo gramatical é o responsável por identificar 33 tipos de erros gramaticais, tais como concordâncias verbal e nominal, regências nominal e verbal, uso de crase (PINHEIRO, 2007).

b) *CoGrOO*

Construído em cima do *Apache OpenNLP*, o *CoGrOO* também é um corretor gramatical para o idioma português do Brasil. Utiliza um modelo híbrido, técnicas de estatística de processamento de linguagem natural para mapear o texto e um sistema que identifica os erros gramaticais por meio de regras (SILVA, 2013).

3.2 APRENDIZAGEM DE MÁQUINA

Baeza-Yates et al. (2011) definem aprendizagem de máquina como uma ampla área da inteligência artificial. Seu interesse é projetar e desenvolver algoritmos que aprendem padrões presentes nos dados de entrada. Esses padrões aprendidos são usados para a predição de dados desconhecidos.

Algoritmos de aprendizagem de máquina são totalmente dependentes da etapa de treinamento, na qual os dados de entrada são usados para encontrar um modelo ou uma função matemática representando o padrão neles presente. Dependendo do processo usado no treinamento, o algoritmo pode ser classificado como de aprendizagem supervisionada ou não supervisionada.

Na aprendizagem supervisionada, no caso de classificação de textos, durante a etapa de treinamento (aprendizagem), é fornecido o par documento-classe, sendo que a classe apropriada para um determinado documento é especificada por um humano especialista. No entanto, na aprendizagem não supervisionada, os dados

de entrada são fornecidos sem as suas classes. Nesse caso, a tarefa do classificador é separar os documentos em grupos ou classes, procedimento comumente conhecido como agrupamento (*clustering*).

Mitchell (1997) afirma que a aprendizagem de máquina possui grande valor prático para uma variedade de domínios de aplicações, sendo especialmente útil nos seguintes casos:

I. em problemas de mineração de dados, nos quais grandes bancos de dados são analisados automaticamente, existindo uma busca de regularidades implícitas que possam ser úteis;

II. em domínios ainda pouco entendidos, nos quais os humanos não possuem o conhecimento necessário para desenvolver algoritmos efetivos;

III. em domínios nos quais o programa necessita adaptar-se dinamicamente a mudanças;

IV. em domínios em que o custo da aquisição ou codificação manual do conhecimento é elevado.

Baeza-Yates (2011) afirmam que os algoritmos de aprendizagem de máquina podem der aplicados, por exemplo, em processamento de linguagem natural, diagnósticos médicos, detecção de fraude em cartão de crédito, recuperação de informação e análise de mercado de ações.

Em nosso trabalho, para proceder à classificação automática de redações na Competência 1 do Enem, utilizamos o método de aprendizagem supervisionado, de tal forma que cada redação nas bases de dados é pré-classificada com base na nota referente a essa competência, atribuída pelo especialista do *site*.

3.2.1 Treinamento, testes e validação

Na aprendizagem supervisionada, o processo de classificação automática de documentos é dividido em três fases: aprendizagem, validação e classificação. A

fase de aprendizagem é realizada com um conjunto de dados contendo os documentos já classificados. A validação ocorre quando ajustamos parâmetros do algoritmo de acordo com os resultados das métricas de desempenho em cima dos dados de treino. Por fim, a base de testes, composta por dados não contidos na base de treino, é submetida ao classificador para medir sua efetividade (SAÚDE, 2014). A seguir, apresentamos alguns tipos de validação.

3.2.1.1 Validação cruzada (cross-validation)

Uma tarefa típica de aprendizado de máquina é aprender um modelo a partir de dados disponíveis. O problema de avaliar um modelo é que ele pode demonstrar uma predição adequada sobre os dados de treinamento, mas, por outro lado, pode falhar para predizer dados futuros, chamado de *overfitting*.

Validação cruzada (*cross-validation*) é um método estatístico de avaliação e comparação de algoritmos de aprendizagem de máquina. Basicamente, consiste em dividir os dados em duas partes: uma usada para aprender ou treinar o modelo e outra, usada para validá-lo. Tipicamente, os dados de validação e treinamento devem ser cruzados em testes sucessivos, tal que cada parte tenha a chance de ser usado como treinamento e como teste. Na validação cruzada estratificada, as classes são apresentadas na mesma proporção no treino e no teste.

O desempenho do algoritmo sobre cada teste pode ser medido por meio de alguma métrica de desempenho. Os principais métodos de validação cruzada são: validação por ressubstituição, *hold-out validation*, *k-fold cross-validation*, *leave-one-out cross-validation*.

3.2.1.2 *K-foldcross-validation*

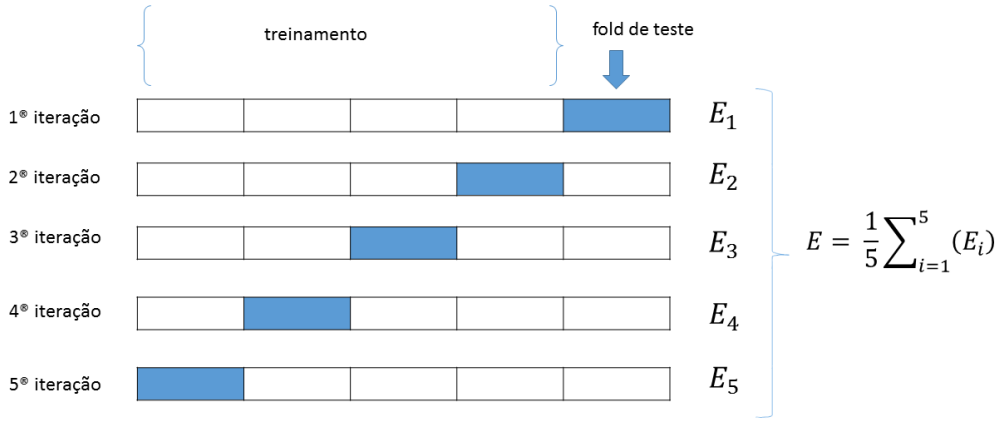
Neste trabalho, utilizamos o método *k-foldcross-validation*, cujos dados disponíveis são primeiramente divididos em *k* partes (*folds*) de tamanhos iguais (ou aproximadamente iguais), com amostras mutuamente exclusivas. Subsequentemente, *k* iterações de treinamento e teste são realizadas, tal que em cada iteração uma parte diferente é usada para teste e as outras *k-1* partes são usadas para treinamento, conforme mostra a Figura 3.1.Os conjuntos são

normalmente estratificados para garantir que cada parte seja uma boa representação do conjunto original de dados.

3.2.2 ALGORITMOS DE CLASSIFICAÇÃO

O estudo de Lorena e Carvalho (2008) destaca que diversos problemas envolvem a classificação de dados em categorias, também denominadas classes.Os algoritmos de classificação possuem diferentes características. Nesta seção, descrevemos os classificadores e suas estratégias de classificação utilizadas para a comparação dos resultados dos experimentos deste trabalho: o *Suport Vector Machine* (*SVM*), com estratégias de classificação multiclasses, e o *ensemble* de classificador *Gradient Boosting (GB)*.

Figura 3.1 – *5 - folds cross-validation*



Fonte:Saúde (2008)

3.2.2.1 *Suport Vector Machine*

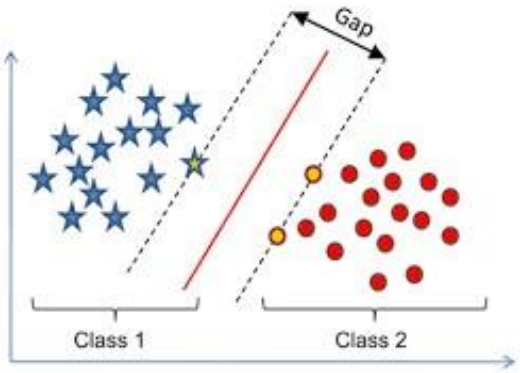
O classificador SVM é uma técnica de aprendizado de máquina introduzida por Cortes e Vapnik (1995), sendo utilizada pela primeira vez em problemas de categorização de documentos por Joachims (1998).

Oliveira et al. (2014) descrevem que o princípio do SVM é encontrar uma superfície de decisão (hiperplano), podendo ser usado para melhor separar os elementos de

duas classes C_1e C_2. Esse hiperplano, aprendido com os dados de treino, divide o espaço em duas regiões, nas quais os documentos que pertencem à classe C_1estão em uma região e os que pertencem à classe C_2estão em outra. Em um espaço com duas dimensões, o hiperplano é uma linha. Em um espaço tridimensional, o hiperplano é um plano. Um novo documento $d_j$será classificado de acordo com sua posição no hiperplano.

A Figura 3.2 mostra um hiperplano que separa as classes. C_1e C_2são duas classes linearmente separáveis e *x*, um documento da base de treino. Cada documento receberá um rótulo: $y = +1$ se $x \in C_1$, e $y_i = -1$ se $x \in C_2$.

Figura 3.2 – Hiperplano ótimo, que separa as classes C_1e C_2



Fonte: Adaptação de Saúde (2014).

A Equação 3.1 mostra a fórmula geral da função de decisão linear, na qual *w* é um vetor *m*–dimensional (pesos), *b* é o termo independente e*m* representa a dimensionalidade dos dados.

$$D(_x) = \sum_{i=1}^{m} w_i x + b \qquad (3.1)$$

A Equação 3.2, a seguir, é equivalente à Equação 3.1, porém, como produto interno entre dois vetores.

$$D(_x) = w^T x + b \qquad (3.2)$$

Sejam *m* o número de documentos da base de treino, e *w* e *x,* os vetores representados pó rw_i e x_i para i = 1,..., m. Para garantir que os elementos da base de treino sejam linearmente separáveis, seus elementos devem satisfazer às restrições mostradas nas equações 3.3 e3.4.

$$w^T x + b > 0, x \in C_1 (y = +1) \tag{3.3}$$

$$w^T x + b < 0, x \in C_2 (y = -1) \tag{3.4}$$

Combinando as desigualdades acima, obtêm-se a condição exibida na Equação 3.5, para i =1, 2, ..., m.

$$y_i (w^T x_i + b) \geq 1 \tag{3.5}$$

Portanto, o hiperplano que forma a superfície de separação entre as duas classes é obtido por meio da Equação 3.6.

$$D(_x) = w^T x + b = c, para - 1 \leq c \leq 1 \tag{3.6}$$

Para c = 0, a equação define um hiperplano situado à meia distância entre os dois hiperplanos nos extremos c = +1 e c = −1. A distância entre os dois hiperplanos extremos é denominada margem. A região entre os dois hiperplanos extremos é chamada região de generalização. O hiperplano D(x) = 0, ao maximizar o valor da margem, maximiza a região de generalização, sendo, portanto, considerado um hiperplano ótimo.

Uma das formas de utilização do SVM é combiná-lo com estratégias de classificação multiclasses. Segundo Lorena e Carvalho (2008), a generalização de técnicas de classificação binária para problemas multiclasses pode ser realizada basicamente por meio de duas estratégias. A primeira consiste na combinação de preditores

gerados em subproblemas binários, enquanto na segunda realizam-se adaptações nos algoritmos originais das técnicas consideradas.

Para as SVMs, em particular, Hsu e Lin (2002) observaram que a reformulação dessa técnica em versões multiclasses leva a algoritmos computacionalmente custosos. Assim, à alternativa de decompor o problema multiclasses em subproblemas binários, é comum recorrer-se a uma estratégia denominada decomposicional.

Carvalho (2008) afirma que as estratégias decomposicionais mais comuns encontradas na literatura são a um contra todos (*one-against-all*– OAA) e a todos contra todos (*all-against-all*, também denominada *one-against-one*– OAO), que são descritas a seguir.

Na estratégia um contra todos, dado um problema com k classes, k classificadores binários $f_i(x)$ são gerados. Cada um desses preditores é treinado de forma a distinguir uma classe i das demais. Já na decomposição todos contra todos, dadas k classes, $\frac{K\ (K-1)}{2}$ classificadores binários são gerados. Cada um deles é responsável por diferenciar um par de classes (i, j), em que $i = j$.

3.2.2.2 *Gradient Boosting Forests*

Bases de dados desbalanceadas comprometem o desempenho dos classificadores, pois eles assumem que elas possuem uma distribuição balanceada e o custo dos erros de classificação é igual para todas as classes. Uma das estratégias usadas para solucionar esse problema é selecionar uma porção balanceada da base de treinamentos. No entanto, essa estratégia pode não ser tão eficaz, uma vez que descarta instâncias que podem ser relevantes na discriminação entre as classes (FERNANDES et al., 2014).

Outra forma de lidar com tal situação é o uso de *ensemble* de classificadores, no qual vários classificadores são treinados para solucionar o mesmo problema. Nesse paradigma, um conjunto de hipóteses é induzido separadamente, sendo combinado por meio de algum método/operador de consenso (ZHOU, 2015).Os estudos de Tumer e Ghosh (1996) mostraram que a habilidade de generalização de um

ensemble é, em geral, maior que a dos classificadores isolados que o compõem, usualmente chamados de classificadores base.

Gradient Boosting é uma técnica de aprendizagem de máquina para regressão e problemas de classificação que produz um modelo de predição sob a forma de um conjunto (*ensembles*) de modelos de previsão fracos, normalmente árvores de decisão. Segundo Friedman (2001), G*radient Boosting* produz procedimentos competitivos, altamente robustos e interpretáveis para problemas de regressão e classificação, sendo apropriado para utilização em base de dados com grande número de ruídos – *outliers*. Trata-se do modelo melhor sucedido nas competições de aprendizagem de máquina do site *Kagle*[9], sendo usado em grande parte das soluções vencedoras.

3.2.3 Seleção de características

Segundo Baeza-Yateset al. (2011), um grande número de características pode tornar os classificadores de documentos impraticáveis, porque a classificação de novo documento levaria muito tempo. “Maldição da dimensionalidade” é a expressão que se refere a vários fenômenos que surgem na análise de dados em espaços com muitas dimensões (características ou atributos). De modo geral, o desempenho de um classificador tende a se degradar a partir de determinado número de características.

A solução para o problema é reduzir o número de características, selecionando um subconjunto formado por aquelas que melhor representem o documento. A seleção de características tem como objetivo descartar os atributos irrelevantes e redundantes. Normalmente, utiliza uma estratégia de busca que decide a maneira como as combinações de atributos são testadas, de acordo com um critério de qualidade. Como exemplos de algoritmos de seleção de características, podemos citar o *Sequence Forward Selection*, que inicia com um subconjunto vazio, isto é, sem nenhuma característica. A inclusão de uma característica é feita, se ela melhora o resultado.

[9]https://www.kaggle.com/competitions

Neste trabalho não utilizamos a seleção de características, pois pode ser que algumas não tenham uma frequência expressiva na etapa de treinamento, sendo, porém, essenciais nas redações separadas para teste. No entanto, utilizamos o algoritmo *Particle Swarm Optimization,* detalhado na próxima seção, para ponderar, isto é, atribuir pesos a cada característica.

3.2.4 *Particle Swarm Optimization*

Segundo Kennedy (2011), a otimização do enxame de partículas (*Particle Swarm Optimization* – PSO) possui muitas similaridades com as técnicas evolucionárias de computação, além de laços com algoritmos genéticos e programação evolutiva. O PSO foi criado por meio de interpretações do movimento de organismos em um bando de pássaros ou em um cardume de peixes. De fácil implementação, possui poucos parâmetros a serem ajustados, ao contrário do algoritmo genético. Pode ser aplicado na solução de vários tipos de problemas, entre eles a otimização de funções, treinamento de redes neurais artificiais, controle de sistemas *fuzzy* e outras áreas nas quais o algoritmo genético é aplicado. O pseudocódigo do algoritmo PSO é o seguinte:

Algoritmo PSO

```
for cada partícula do
    Inicializa partícula
end for
while máximo de iterações ou mínimo de erros não é obtido do
    for cada partícula do
        Calcula fitness
        if fitness maior que o maior fitness(pBest) then
            set valor fitness corrente para novo pBest
        end if
        if melhor fitness de todas as partículas then
            se melhor partícula de todas (gBest)
        end if
end for
for cada partícula do
    Calcula a velocidade da partícula
    Atualiza a posição da partícula
```

end for

end while

Primeiro, o PSO é inicializado com um grupo de partículas aleatórias que correspondem a soluções do problema. A cada iteração, procura-se o melhor valor *fitness*, calculado por uma função de qualidade, de cada partícula (*pBest*), e a melhor partícula de todas (*gBest*), aquela que possui o melhor valor *fitness*. Depois de encontrar os dois melhores valores, a partícula atualiza sua velocidade e posição com a Equação 3.7 e a Equação 3.8, a seguir.

$$v[\ ] = v[\ ] + c1 * rand() * (pbest[\ ] - present[\ ]) + c2 * rand() * (gbest[\ ] - present[\ ]) \quad (3.7)$$

$$present[\ ] = present[\ ] + v[\ ] \quad (3.8)$$

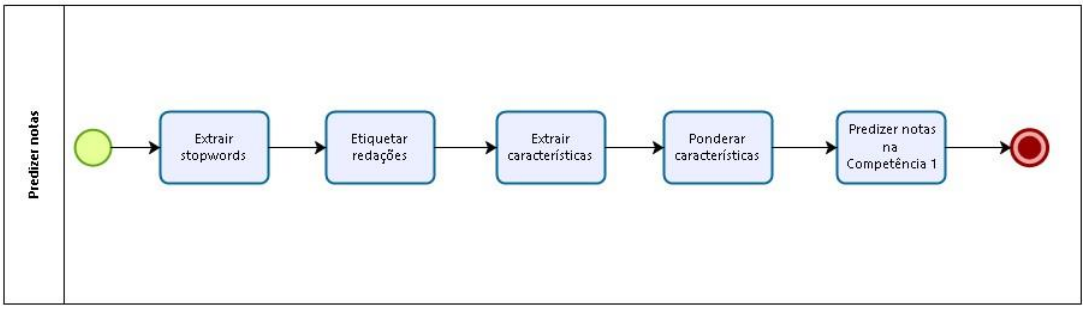
Sendo:

- *v[]*= velocidade da partícula
- *present []*= partícula corrente (solução)
- *randon()*= números randômicos entre (0,1)
- *c1* e *c2*= fatores de aprendizagem

4 *PESSAY* – SISTEMA DE AVALIAÇÃO AUTOMÁTICA DE REDAÇÕES

Nesse capítulo, apresentamos a arquitetura e funcionamento do sistema *Pessay*, desenvolvido para a correção automática de redações tendo como base a Competência 1 do Enem. Serão relatados os problemas encontrados e as justificativas para a escolha das soluções. A Figura 4.1, gerada pelo *software Bizagi*[10], mostra um resumo das etapas do processo, as quais passamos a detalhar.

Figura 4.1 – Etapas do processo de avaliação automática de redações



Fonte: elaborado pelo autor no *software Bizagi*.

4.1 Requisitos para utilização do sistema

4.1.1 **Requisitos de *hardware***

Nossos experimentos foram executados em um computador com as seguintes especificações:

- Processador Intel core i7 2.40 GHz;
- 8,0 GB de memória RAM;
- Sistema operacional Ubuntu 16.04.

[10]https://www.bizagi.com/pt

4.1.2 **Requisitos de *software***

Quanto aos requisitos de *software*, abaixo listamos os principais utilizados nos experimentos:

- Python versão3.5;
- Java 8;
- Apache OpenNLP versão 1.5.3;
- CoGrOO versão 3.0.5;
- Dicionário hunspell versão2.2;

4.2 PROCESSAMENTO DAS REDAÇÕES

Nas primeiras 3 etapas do diagrama da Figura 4.1 o sistema realiza o processamento das redações, retirando as correções dos avaliadores do *site*, removendo datas, números e outros termos considerados irrelevantes para o contexto. Em seguida, utilizando o *Apache OpenNLP*, o sistema inclui etiquetas com informações sobre categorias morfológicas, como substantivo, preposições, adjetivo, pronome, entre outras.

Na sequência, o sistema extrai as características de cada redação, que variam de acordo com o revisor gramatical utilizado. Quando o *ReGra* é usado, uma redação é representada por 64 características, apresentadas no Apêndice A.1.No *CoGrOO*, cada redação é representada por 160 características, mostradas no Apêndice A.3.

Para a extração dos erros ortográficos e gramaticais, o sistema possui os módulos: ortográfico e gramatical, que são explicados nas seções a seguir.

Importante relatar que, em muitos casos o avaliador humano pode relativizar o peso de um erro em função da ousadia ou originalidade da construção.

4.2.1 Módulo ortográfico

Nesse módulo, o sistema utiliza o dicionário *Hunspel* para identificar palavras escritas de forma incorreta. Porém, algumas não se encontram na base de dados desse dicionário. Para minimizar o problema, o sistema utiliza um dicionário auxiliar,

formado por palavras extraídas dos textos-temas, publicados no enunciado de cada tema no *site* UOL, e de textos do jornal *A Tribuna*, de circulação no Estado do Espírito Santo[11].

Além disso, outro problema que o sistema *Pessay* encontrou na identificação de erros ortográficos relaciona-se a palavras sintaticamente corretas, mas fora do contexto da frase. Como podemos ver na frase destacada a seguir, a palavra "transito", na primeira frase da redação, não foi considerada erro pelo sistema (como as que estão grafadas), pois sintaticamente está correta (primeira pessoa do verbo transitar).

"É possivel sim reduzir o nível de violencia no transito brasileiro. basta que as pessoas tenham mais respeito, educaçao e tolerância com o proximo."

Para solucionar esse problema, o sistema utiliza um algoritmo que se baseia na probabilidade bayesiana para analisar o contexto da palavra, isto é, sua colocação na frase. Esse algoritmo utiliza dicionários de *bigrams* e *trigrams*, ou seja, contabilizando na frase a frequência das palavras adjacentes com dois ou três termos.

4.2.2 Módulo gramatical

No módulo gramatical do sistema, testamos os revisores *ReGra*e *CoGrOO*, apresentados na Seção 3.1.5.

Para identificação dos erros gramaticais nas redações, o sistema divide cada redação em sentenças, essas são enviadas para o Módulo Detector de Erros Gramaticais, que no caso do CoGrOO, consulta um arquivo de regras, retornando os erros e sugestões, caso existam.

No caso do revisor ReGra, cada redação é enviada para a API_REGRA, esta gera um relatório com os erros gramaticais, após consultar uma base de erros.

---

[11] O Laboratório de Computação de Alto Desempenho da Universidade Federal do Espírito Santo possui uma base de dados com reportagens de mais uma década desse jornal. Em função disso, o dicionário que utilizamos foi complementado com as palavras dessa base.

Ao final da etapa de extração de características, o sistema gera o *dataset*, uma matriz *n* x *m*, sendo *n* o número de redações do conjunto de dados e *m*, o número de características de cada redação.

4.3 PONDERAÇÃO DAS CARACTERÍSTICAS

O sistema desenvolvido em nosso estudo adaptou o algoritmo PSO, apresentado na Seção 3.3.4, para ponderar, ou seja, atribuir pesos às características de uma redação. Como função de qualidade, usamos o Erro Médio Absoluto (*Mean Absolute Error* – MAE), descrito na Seção 4.4.2 e definido pela Equação 4.3.

4.4 CLASSIFICAÇÃO

Para a classificação das notas obtidas pelas redações na Competência 1, o sistema utiliza dois classificadores. Um deles é o SVM combinado à estratégia *multclass* todos contra todos, apresentados na Seção 3.3.1.O outro classificador utilizado é o *ensemble* de classificadores *Gradient Boosting*, mostrado na Seção 3.3.2.

4.5 MÉTRICAS

Para avaliar os resultados da classificação, foram adotadas métricas como precisão (*Precision*) (Equação 4.1), recuperação (*Recall*) (Equação 4.2) e o Erro Médio Absoluto (Equação 4.3), detalhados a seguir.

4.5.1 *Precision* e*Recall*

Precision (precisão) é a proporção do número de documentos relevantes recuperados ao número total de documentos para uma determinada consulta do usuário (Equação 4.1). *Recall* (recuperação) de um sistema de recuperação de texto, por sua vez, pode ser definido como a proporção do número de documentos relevantes retornados ao número total de documentos relevantes para a consulta do usuário na coleção (JUNKER; HOCH; DENGEL,1999) (Equação 4.2).

$$Precision(C_p) = \frac{TP(C_p)}{TP(C_p) + FP(C_p)} \quad (4.1)$$

$$Recall(C_p) = \frac{TP(C_p)}{TP(C_p) + FN(C_p)} \quad (4.2)$$

Para o nosso problema, *p* corresponde a uma das classes = {0,0, 0,5, 1,0, 1,5, 2,0} que são notas atribuídas por especialistas do site na Competência 1 do ENEM. *True Positive* (TP (C_p)) é a quantidade de redações atribuídas corretamente à classe C_p pelo classificador. *False Positive* (FP (C_p)) é a quantidade de redações atribuídas incorretamente à classe C_p pelo classificador. *False Negative* (FN (C_p)) é a quantidade de redações que pertencem à classe C_p, mas que foram classificadas incorretamente em outra classe.

4.5.2 MAE

Em estatística, o Erro Médio Absoluto (*Mean Absolute Error*– MAE), dado pela Equação4.3, é a média das diferenças entre os valores reais e preditos, sendo que *n* é o número de amostras, y_i é a classe real e y^t é classe predita pelo classificador.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - y_i^t| \quad (4.3)$$

4.5.3. Ajustando o *threshold*

Page (1994) destacou que notas atribuídas pelos avaliadores humanos são geralmente diferentes. Em seus experimentos, constatou que a correlação de Pearson entre dois humanos avaliadores foi em torno de 0,564. Bazelato e Amorim (2013) afirmam que é razoável avaliar não apenas o grau exato atribuído automaticamente, mas também notas adjacentes.

Conforme estabelecido pelo anexo IV do Edital Enem 2016, o Exame Nacional do Ensino Médio considera discrepância entre as notas dos avaliadores uma diferença

superior a 80 pontos em cada uma das competências (INEP, 2016). Dada essa afirmação, decidimos avaliar também as notas adjacentes, classificadas como 0,5,1,0 e 1,5 pontos longe da nota atribuída pelo avaliador do site UOL, considerada aqui como nota verdadeira.

Em nossos experimentos, configuramos o sistema para considerar notas adjacentes, relaxando o limite (*threshold*) do que queremos que o sistema desenvolvido considere como nota correta. Esse limite foi movido tanto para cima quanto para baixo. Em cada posição do limiar, valores distintos foram obtidos para *precision*, *recall* e MAE. Em nossos resultados, *k* indica o quanto variamos esse limite para longe da nota tomada como a correta.

5 EXPERIMENTOS E RESULTADOS

Nos capítulos anteriores, apresentamos as técnicas e ferramentas utilizadas no processamento das redações. Explicamos também o funcionamento do sistema de avaliação de redação na Competência 1 do Enem, apresentando as técnicas utilizadas para validação e teste. Neste capítulo, serão apresentadas as bases de dados utilizadas no estudo relatado nesta dissertação. Em seguida, analisaremos os resultados dos experimentos obtidos na aplicação do sistema. Por fim, traremos alguns exemplos de avaliação de redações efetuadas pelo sistema.

5.1 BASES DE DADOS

As bases de dados utilizadas neste trabalho são compostas por redações extraídas de dois bancos de redações do *site* UOL: UOL Educação e Brasil Escola. As redações foram corrigidas e comentadas por especialistas do *site*, que atribuíram uma nota em uma escala de 0,0 a 2,0, com passos de 0,5 em cada uma das cinco competências do Enem. As Figuras 5.1 e 5.2 mostram, respectivamente, uma redação corrigida, com as notas em cada competência do avaliador do *site*.

Figura 5.1 – Redação corrigida

Bandido bom é bandido recuperado

NOTA
3,0

Nosso país passa por uma severa crise financeira e social. Falta de emprego, alta de preços, desvalorização da moeda. Uma pessima [péssima] educação escolar, principalmente a pública, levando a uma desigualdade social grave.

Não seria desculpa a [para] uma pessoa que esteja desempregada e, necessita [e necessita] sustentar sua familia [família], entrar na vida do crime, onde "dinheiro fácil", e [fácil" é] bem vindo. Cada vez mais, jovens sem perspectiva de futuro, integra [futuro integram] a criminalidade. Seja pela venda de drogas, furtos, assaltos a mão armada, a uma crescente em nosso meios.

Policiais, deveriam proteger e frear essa violência, mas, péssimos salários, despreparo, levam a desvirtuar [salários e despreparo desvirtuam] seu caminho. Não sabemos distinguir os policias bons dos policiais corruptos [os policiais bons dos corruptos]. A imagem e [é] que todos são corruptiveis [corruptíveis].

O que disser as [dizer das] prisões brasileiras, verdadeiras faculdades do crime, em que adianta, prender, para que, [não adianta prender pois] quando sair de lá, voltara [o criminoso voltará] a fazer os mesmos atos, mas agora mais consciente. Porque não, criar [mais escolado no crime? Por que não criar] meios de reeducar os criminosos nas prisões, dando futuro de melhoras quando sair [dando-lhes um futuro melhor depois do cumprimento da pena?]. O mesmo para os policiais, melhorias em seu ambiente de trabalho, melhor preparo e salários.

Talvez, a unica solução, seja [a única solução sejam mais] investimentos em educação, geração de empregos e diminuição da corrupção, que deva ser o maior e [mais] complexo problema do Brasil. Nosso país arrecada o suficiente, mas os desvios de verbas, pagamentos [verbas e os pagamentos] de propina sugam os recursos, gerando todos os nossos grandes problemas sociais.

Fonte: UOL Educação (acesso em: 12 nov. 2016).

Figura 5.2 – Notas atribuídas pelo avaliador a cada competência

Competências avaliadas

Itens	Nota
Demonstrar domínio da norma culta da língua escrita.	1,0
Compreender a proposta da redação e aplicar conceito das várias áreas de conhecimento para desenvolver o tema, dentro dos limites estruturais do texto dissertativo-argumentativo.	0,5
Selecionar, relacionar, organizar e interpretar informações, fatos, opiniões e argumentos em defesa de um ponto de vista.	0,5
Demonstrar conhecimento dos mecanismos linguísticos necessários para a construção da argumentação.	0,5
Elaborar a proposta de solução para o problema abordado, mostrando respeito aos valores humanos e considerando a diversidade sociocultural.	0,5
Nota final	**3,0**

Fonte: UOL Educação (acesso em: 12 nov. 2016).

A Tabela 5.1 mostra um resumo das bases de dados. As linhas mostram a quantidade de redações, por nota na Competência 1, das duas bases de dados. A primeira coluna apresenta a distribuição da base UOL Educação, com 38 redações de notas zero, 222 com notas 0,5, 422 com notas 1,5 e 72 com notas 2,0, num total de 953 redações. A segunda coluna apresenta a distribuição das 4902 redações da base Brasil Escola.

Tabela 5.1 – Resumo das bases de dados com relação à nota atribuída na Competência 1 do Enem

	Bases de dados	
Notas Competência 1	UOL Educação	Brasil Escola
0,0	38	55
0,5	222	291
1,0	422	2456
1,5	199	1895
2,0	72	205
Total	953	4902

Fonte: elaborado pelo autor a partir dos dados obtidos nas bases de dados UOL Educação e Brasil Escola.

Ou seja, a avaliação do domínio da escrita não acontece in absentia, independentemente do que acordado sobre o “peso” que será dado na correção humana, Necessidade de uma regulagem no sistema.

Analisando a Tabela 5.1, notamos que cerca de 4% das redações da base UOL e 1,0% das redações da base Brasil Escola obtiveram a nota mínima na competência 1. Observa-se a baixa quantidade de redações nas quais os avaliadores do *site* atribuíram nota zero na Competência 1, isso deve-se ao interesse do avaliador em cativar o usuário do *site*.

5.2 METODOLOGIA EXPERIMENTAL

Realizamos experimentos para cada base de dados mostradas na Tabela 5.1. Em cada experimento, executamos as etapas do diagrama da Figura 4.1. Para a extração dos erros gramaticais, experimentamos os revisores gramaticais *ReGra* e *CoGrOO*, apresentados na Seção 3.1.5.

Cada redação é representada em nossas bases de dados por um vetor de características, as quais variam conforme o revisor gramatical utilizado.Quando utilizamos o *ReGra*, uma redação é representada por 64 características, mostradas na Seção B.1, já no uso do *CoGrOO*, cada redação é representada por 160 características, mostradas na Seção B.2.

Para a predição das notas na Competência 1, utilizamos os classificadores mostrados na Seção 3.3. O SVM, com a estratégia de classificação *multiclass* OAO, e o *ensemble* de classificadores *Gradient Boosting*.

Dado que algumas características podem não ser expressivas na etapa de treinamento, mas extremamente importantes em novas amostras utilizadas para testes, não utilizamos a etapa de seleção de características para reduzir a dimensionalidade. Utilizamos o PSO para ponderar, ou seja, atribuir pesos a cada característica.

Nas etapas de treinamento e testes, optamos por utilizar a técnica *K-foldcross-validation* com *k* igual a cinco. Por isso, dividimos a base em cinco partes iguais. Em seguida, realizamos cinco iterações. Em cada iteração, quatro partes são usadas para treinar o classificador e uma, para teste. Computamos a média dos resultados das cinco partes antes e depois da etapa de ponderação de características.

Conforme mencionamos na Seção 4.4.3, é razoável avaliar não exatamente a nota atribuída, mas também notas adjacentes. Por isso, em nossos resultados, mostramos também as métricas para as notas obtidas com distância de 0,5, 1,0 e 1,5 da nota atribuída pelo especialista do *site*.

5.2.1 Resumo dos experimentos

Na Tabela 5.2 apresentamos um resumo dos experimentos realizados. A tabela é dividida em duas partes: as primeiras quatros linhas apresentam as configurações dos experimentos com a Base UOL Educação, nas últimas quatro linhas apresentamos as configurações dos experimentos com a Base Brasil Escola. Na primeira coluna exibimos as tabelas com os resultados dos experimentos. Na segunda e terceira coluna exibimos o revisor gramatical e o classificador utilizado respectivamente. Já nas duas últimas colunas informamos a utilização ou não do

PSO para ponderar as características e a retirada ou não das redações zeradas por influência de outras competências.

Tabela 5.2 – Resumo dos Experimentos realizados

Resultados	Revisor Gramatical	Classificador	PSO	Redações zeradas
Experimentos Base UOL Educação				
Tabela 5.3	ReGra	SVM/GB	Não	Sim
Tabela 5.4	ReGra	SVM/GB	Sim	Sim
Tabela 5.5	ReGra	SVM/GB	Não	Não
Tabela 5.6	ReGra	SVM/GB	Sim	Não
Experimentos Base Brasil Escola				
Tabela 5.7	ReGra	SVM/GB	Não	Não
Tabela 5.8	ReGra	SVM/GB	Sim	Não
Tabela 5.9	CoGrOO	SVM/GB	Não	Não
Tabela 5.10	CoGrOO	SVM/GB	Sim	Não

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

5.3 EXPERIMENTOS 1 – BASE DE DADOS UOL EDUCAÇÃO

Em nosso primeiro experimento, utilizamos a base de dados UOL Educação, detalhada na primeira coluna da Tabela 5.1. Para extração dos erros gramaticais, utilizamos o revisor *ReGra*.

A Tabela 5.3, a seguir, mostra os primeiros resultados do experimento, variando a distância entre as notas do sistema e as notas atribuídas pelo especialista de 0,0 até 1,5. Cada linha da Tabela 5.3 mostra a média dos cinco *folds* para as métricas MAE, *Precision* e *Recall*. Nas primeiras três colunas das linhas, exibimos os resultados do classificador SVM e nas três últimas colunas são mostrados os resultados do classificador *Gradient Boosting*.

Tabela 5.3 – Resultados iniciais – Base de dados UOL Educação

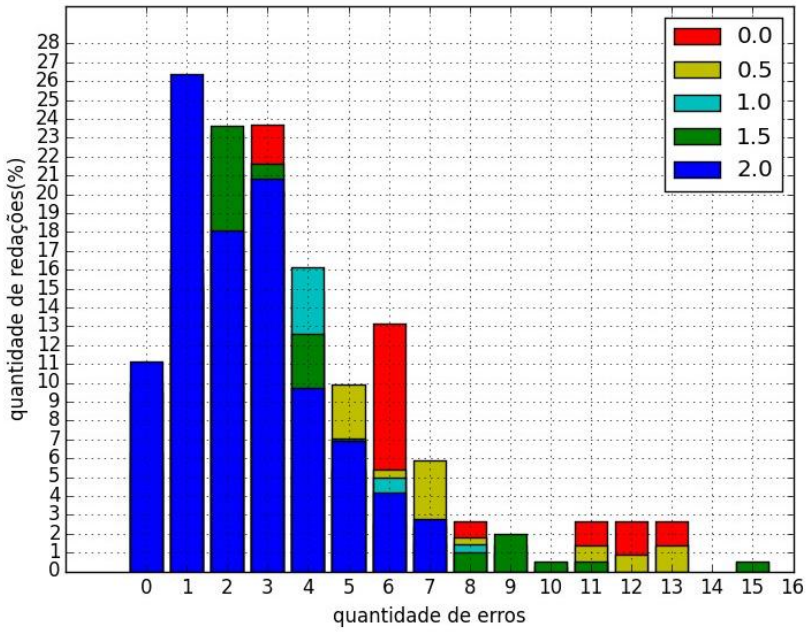
	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,3404	0,3837	0,4430	0,3273	0,4069	0,4534
0,5	0,1201	0,8900	0,8836	0,1069	0,8795	0,8941
1,0	0,0110	0,9933	0,9927	0,0031	0,9981	0,9979
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

As colunas destacadas da primeira linha da Tabela 5.3 mostram os MAE nos dois classificadores com relaxamento zero. Notamos um melhor desempenho do classificador *Gradient Boosting*. Na segunda linha, podemos verificar uma queda significativa dos MAE, 0,1201 e 0,1069, com um relaxamento de 0,5 ponto (um nível), isto é, a distância entre a nota atribuída pelo sistema e a do especialista foi de 0,5 ponto.

O gráfico da Figura 5.3, a seguir, mostra a distribuição dos erros ortográficos e gramaticais por nota na Competência 1. Analisando o gráfico, podemos verificar que existem redações com a mesma quantidade de erros, mas com notas diferentes. Também verificamos redações com a mesma nota, mas com quantidade de erros diferentes. A sexta coluna mostra redações com seis erros, ortográficos e gramaticais, com notas 0,0, 0,5, 1,0 e 2,0. As colunas de cor azul mostram redações com nota 2,0, com quantidade de erros entre zero e sete. Uma das causas dessas características são os erros cometidos pelos corretores, principalmente o gramatical. A outra causa são os pesos atribuídos pelos avaliadores a tipos de erros diferentes.

Figura 5.3 – Distribuição de erros ortográficos e gramaticais por nota na Competência 1 –Base de dados UOL Educação



Fonte: elaborado pelo autor no *software Python* a partir dos dados do sistema *Pessay*.

Como o intuito de melhorar o resultado da classificação, alguns estudos apresentaram métodos que aumentam a importância de algumas características, que melhor representam uma classe, atribuindo um peso maior a elas (SOUZA; CIARELLI; OLIVEIRA, 2014). Em nosso trabalho, utilizamos o PSO e a medida MAE, mostrada na Seção 4.3, como função de qualidade para selecionar os pesos para cada característica e, assim, minimizar essa medida. A Tabela 5.4 mostra os resultados após a ponderação das características. A primeira coluna da primeira linha mostra uma pequena melhora no MAE nos dois classificadores.

Tabela 5.4 – Resultados base de dados UOL Educação, após a ponderação das características

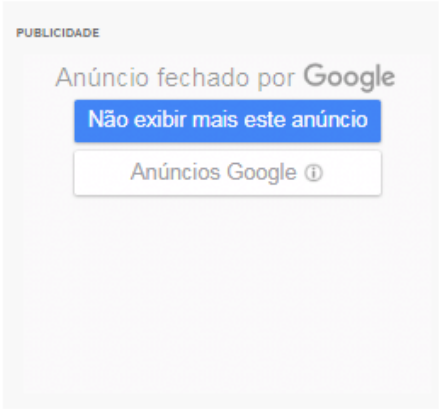
	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,3283	0,3929	0,4556	0,3268	0,4280	0,4545
0,5	0,1107	0,8622	0,8909	0,1069	0,8806	0,8941
1,0	0,0047	0,9970	0,9969	0,0031	0,9980	0,9979
1,5	0,0000	1,0000	0,0000	1,0000	1,0000	0,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Outra característica de nossa base de dados são as redações que foram zeradas por influência de outras competências. A Figura 5.4 mostra uma redação que

recebeu nota zero em todas as competências, por fugir ao tema proposto, (Competência 2 do Enem). Como podemos verificar na Figura 5.4, a redação não apresenta erros sintáticos e estruturais que justifiquem uma nota zero na Competência 1. Situações como a exposta podem prejudicar o desempenho do sistema *Pessay*.

Figura 5.4 – Redação nota zero



A mobilidade urbana tem sido cada vez mais questionada no Brasil. Devido ao aumento de automóveis nas metrópoles, o trânsito caótico tornou-se um desafio para sociedade, devido as horas que o cidadão gasta no trafego e pelo os altos niveis de monóxido de carbono prejudicando a saúde dos individuos.

Visto que o transporte rodoviário teve impulso a partir da década de 1920 do século xx,por iniciativos de governantes como Juscelino Kubistchek consagrado pela frase 'Governar é abrir estradas' com intuito de ampliar e melhorar as rodovias.

Atualmente, nota-se que há urgência em reduzir o fluxo de carros e ampliar outros meios de locomoção nas grandes cidades do Brasil, segundo o G1 pais tem um automóvel para cada quatro habitante, essa intensificação é devido a má qualidade do transporte público e pelo custo das tarifas que são altas.

Dessa forma, para reverter esse crescimento de veiculos,há serviços sendo oferecidos por empresas privadas ,como o aplicativo UBER que oferece transporte remunerado usando diminuir o uso de automóvel particular,temos que ter parceria com a prefeitura e empresas para disponibilizar o aluguel de bicicletas com intuito de diminuir o trânsito nas cidades.

Fonte: Brasil Escola (acesso em 19 nov. 2016).

A fim de solucionar esse problema, retiramos as 26 redações zeradas por influência de outras competências. As Tabelas 5.5 e 5.6 mostram, respectivamente, os resultados antes e depois da ponderação de características, após a retirada dessas redações.

Tabela 5.5 – Resultados iniciais após a retirada de 26 redações com nota total zero – Base de dados UOL Educação

	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,3251	0,4074	0,4565	0,3090	0,4351	0,4746
0,5	0,1046	0,8994	0,8976	0,0915	0,9012	0,9095
1,0	0,0065	0,9957	0,9957	0,0032	0,9979	0,9979
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Tabela 5.6 – Resultados iniciais após a retirada de 26 redações com nota total zero e ponderação das características – Base de dados UOL Educação

	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,3107	0,4276	0,4801	0,3009	0,4738	0,4865
0,5	0,0998	0,9012	0,9018	0,0878	0,9039	0,9127
1,0	0,0049	0,9968	0,9968	0,0016	0,9990	0,9989
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Assim, utilizando a base UOL Educação, nosso primeiro experimento mostrou algumas características do processo e da base de dados que podem dificultar a tarefa de predizer a nota na Competência 1. Uma dificuldade encontrada pelo sistema foi detectar padrões nos pesos atribuídos pelos avaliadores, atribuindo um peso menor aos erros cometidos em frases com um vocabulário mais rico.

Outro problema foram as redações zeradas por influência de outras competências, num total de 26 redações, que foram retiradas da base.

Após a retirada das 26 redações com nota total zero e ponderação das características, conseguimos, utilizando o classificador *Gradient Boosting*, diminuir o MAE para 0,3009, o *Precision* para 0,4738 e o Recall para 0,4865, mostrados nas últimas três colunas da primeira linha da Tabela 5.6.Já considerando uma distância de 0,5 ponto (um nível) da nota do especialista, melhoramos o MAE para 0,0878, o *Precision* para 0,9039 e o *Recall* para 0,9127.

Consideramos tais resultados razoáveis, pois, conforme mencionamos, o próprio Enem, por meio do edital, só considera discrepância uma diferença maior do que 80 pontos, dois níveis, entre as notas dos avaliadores em uma competência.

5.4 EXPERIMENTO 2 – BASE DE DADOS BRASIL ESCOLA

No segundo experimento, utilizamos o banco de redações Brasil Escola, também do *site* UOL. A base de dados é detalhada na segunda coluna da Tabela 5.1.Nesse experimento, testamos os revisores gramaticais *CoGrOO* e *ReGra*.

Retiramos da base Brasil Escola 49 redações zeradas por influência de outras competências. As Tabelas 5.7 e 5.8 mostramos resultados utilizando o revisor gramatical *ReGra*, respectivamente, antes e depois da ponderação das características.

Tabela 5.7 – Resultados base de dados Brasil Escola, usando o *ReGra* como revisor

	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,2365	0,5288	0,5516	0,2385	0,5343	0,5495
0,5	0,0245	0,9751	0,9755	0,0265	0,9732	0,9736
1,0	0,0000	1,0000	1,0000	0,0003	0,9996	0,9998
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Tabela 5.8 – Resultados base de dados Brasil Escola, usando o *ReGra* como revisor, após a ponderação das características

	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,2374	0,5277	0,5510	0,2397	0,5318	0,5485
0,5	0,0258	0,9740	0,9742	0,0277	0,9721	0,9724
1,0	0,0000	1,0000	1,0000	0,0003	0,9996	0,9998
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Em seguida, experimentamos também o revisor gramatical *CoGrOO*. As Tabelas 5.9 e 5.10, a seguir, apresentam, respectivamente, os resultados antes e depois da utilização do PSO, utilizado para ponderaras características.

Tabela 5.9 – Resultados base de dados Brasil Escola, usando o *CoGrOO* como revisor

	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,2354	0,5303	0,5510	0,2385	0,5343	0,5495
0,5	0,0217	0,9779	0,9784	0,0265	0,9732	0,9736
1,0	0,0003	0,9996	0,9998	0,0003	0,9996	0,9998
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Tabela 5.10 – Resultados base de dados Brasil Escola, usando o *CoGrOO* após a ponderação das características

	SVM			*Gradient Boosting*		
K	MAE	*Precision*	*Recall*	MAE	*Precision*	*Recall*
0,0	0,2337	0,5345	0,5551	0,2407	0,5219	0,5460
0,5	0,0224	0,9773	0,9777	0,0272	0,9726	0,9730
1,0	0,0000	1,0000	1,0000	0,0006	0,9996	0,9996
1,5	0,0000	1,0000	0,0000	0,0000	1,0000	1,0000

Fonte: elaborado pelo autor a partir do sistema *Pessay*.

Na Tabela 5.10, podemos verificar uma melhora nos resultados com a utilização da base Brasil Escola. A primeira linha da Tabela 5.9 mostra um MAE de 0,2337, *Precision* de 0,5303 e *Recall* de 0,5510 considerando notas até 0,0 de distância para as notas atribuídas pelo especialista.

5.5 ANÁLISE DOS RESULTADOS DOS EXPERIMENTOS

A seguir fazemos uma análise dos resultados obtidos pelos classificadores nas duas bases de dados.

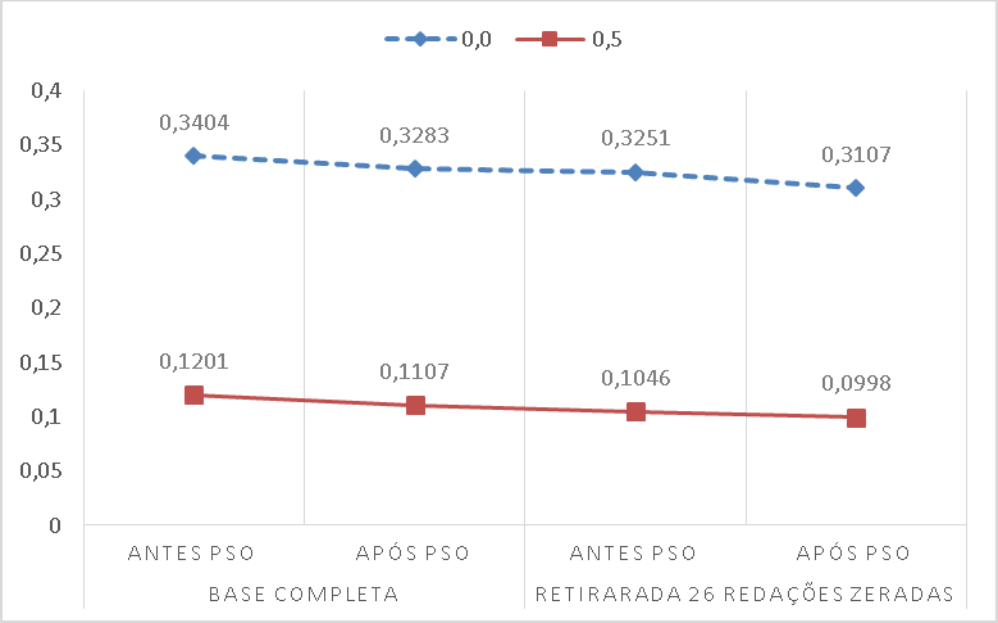
5.5.1Gráficos comparativos dos resultados obtidos pelos experimentos aplicados à base UOL educação

Os gráficos das Figuras 5.5 e 5.6 apresentam as medidas, erro médio absoluto, obtidos pelos classificadores SVM e *Gradient Boosting, respectivamente,* nos experimentos aplicados à base de dados UOL Educação. Estes gráficos mostram o desempenho dos classificadores nos experimentos com a base completa e após a retirada de 26 redações zeradas por influência de outras competências. Nestes gráficos também são mostrados o desempenho dos classificadores antes e após a

aplicação do PSO, utilizado para atribuir peso a cada característica. A linha tracejada mostra a evolução dos resultados com 0,0 de distância para a nota do especialista e a linha sólida mostra os resultados considerando notas de até 0,5 de distância para a nota do especialista.

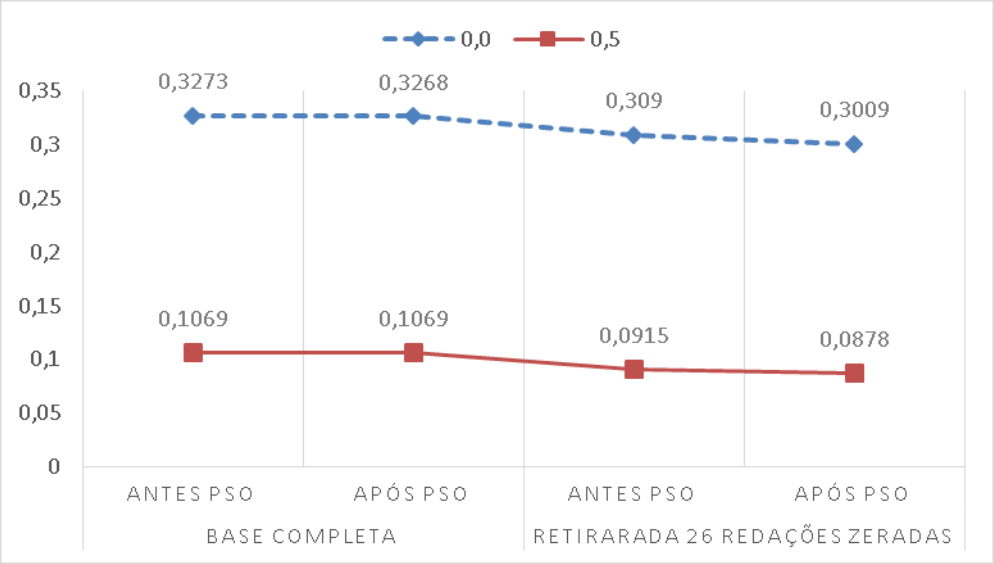
O gráfico da Figura 5.5 mostra as medidas de classificação alcançadas, erro médio absoluto, MAE, pelo classificador SVM. Analisando a linha tracejada do gráfico é possível observar uma queda no MAE, de 0,3404 para 0,3251, após a retirada das 26 redações zeradas da base de dados. Continuando a análise verificamos uma melhora nos resultados após a aplicação do PSO, chegando a MAE de 0,3107. A linha sólida mostra uma pequena diminuição do MAE após a retirada das redações zeradas e a aplicação do PSO, chegando a 0,0998 de MAE.

Figura 5.5 – Evolução dos resultados das medidas de classificação da base UOL Educação, com o classificador SVM



Fonte: elaborado pelo autor no *software Python* a partir dos resultados do sistema *Pessay*

Figura 5.6 – Evolução dos resultados das medidas de classificação da base UOL Educação, com o classificador *Gradient Boosting*



Fonte: elaborado pelo autor no *software Python* a partir dos resultados do sistema *Pessay*

O gráfico da Figura 5.6 mostra as medidas alcançadas, erro médio absoluto, MAE, pelo classificador *Gradient Boosting*. Podemos notar, através das linhas tracejadas e sólidas uma melhora nos resultados em relação aos obtidos pelo SVM, chegando a um MAE de 0,3009 com 0.0 de distância para a nota do especialista e 0,0878 considerando notas com até 0,5 de distância.

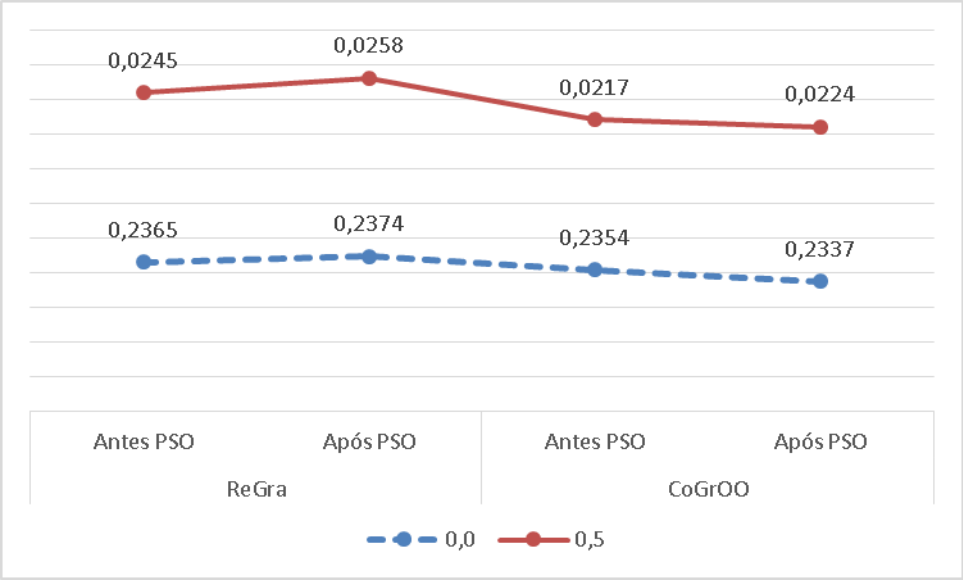
5.5.2 Gráficos comparativos dos resultados obtidos pelos experimentos aplicados à base Brasil Escola.

Os gráficos das Figuras 5.7 e 5.8 apresentam os resultados obtidos pelos classificadores SVM e *Gradient Boosting* nos experimentos aplicados à base de dados Brasil Escola. Estes gráficos mostram o desempenho dos classificadores SVM e *Gradient Boosting* nos experimentos realizados com revisor gramatical ReGra e CoGrOO. Nestes gráficos também são mostrados o desempenho do SVM e *Gradient Boosting* antes e após a aplicação do PSO para atribuir peso a cada característica. Importante ressaltar que nestes experimentos foram extraídas da base de dados 49 redações zeradas por influência de outras competências.

O gráfico da Figura 5.7 mostra a evolução dos resultados dos experimentos realizados utilizando o classificador SVM, os revisores gramaticais ReGra e CoGrOO, antes e depois da aplicação do PSO para ponderar as características.

Podemos observar, através da linha tracejada, uma queda significativa do MAE em relação aos resultados da base UOL, chegando a 0,2365 considerando notas com até 0,0 de distância para as notas atribuídas pelos especialistas do *site,* antes da utilização do PSO e utilizando o ReGra como revisor gramatical.

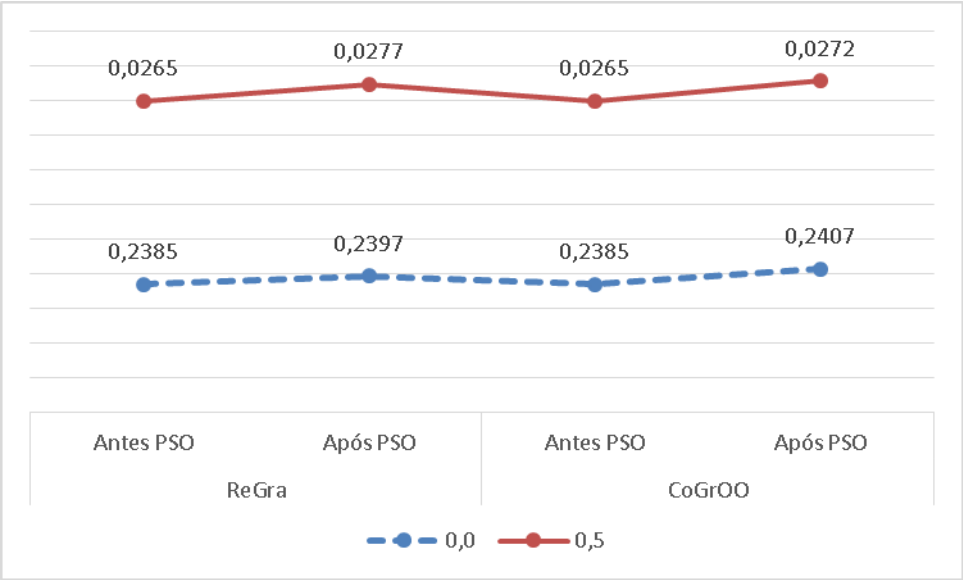
Figura 5.7 – Evolução dos resultados das medidas de classificação da base Brasil Escola, com o classificador SVM



Fonte: elaborado pelo autor no *software Python* a partir os resultados do sistema *Pessay*

O gráfico da Figura 5.8 mostra a evolução dos resultados obtidos com o uso do classificador *Gradient Boosting.* A linha tracejada mostra o comportamento do MAE, considerando notas com até 0,0 pontos de distância das notas atribuídas pelos especialistas e linha sólida mostra o comportamento do MAE considerando notas com até 0,5 pontos de distância das notas atribuídas pelos especialistas.

Figura 5.8 – Evolução dos resultados das medidas de classificação da base Brasil Escola, com o classificador *Gradient Boosting*



Fonte: elaborado pelo autor no *software Python* a partir dos resultados do sistema *Pessay*

Analisando os resultados dos gráficos das Figuras 5.7 e 5.8, observamos que o sistema obteve o melhor resultado, na base de dados Brasil Escola, com o classificador SVM, o revisor gramatical CoGrOO e após o uso do PSO. Chegamos a um MAE de 0,2354 com 0,0 de distância e 0,0224 com 0,5 de distância para a nota do especialista.

Nossa estratégia de ponderar, ou seja, atribuir peso a cada característica, obteve uma pequena melhoria nos resultados na base UOL Educação, ao contrário do que ocorreu na base Brasil Escola, na qual eles se mantiveram estáveis.

Um dos problemas encontrados nas duas bases de dados é o desbalanceamento, baixa quantidade de redações com notas 0 e 2 e uma grande quantidade de redações com notas 1 e 1,5. Os classificadores gerados a partir de bases de treinamento desbalanceadas apresentam altas taxas de erros para as classes raras, pois esses tendem a predizer para as classes majoritárias (MACHADO; LADEIRA, 2007). Conforme mencionamos na Seção 3.3.2, um dos motivos da utilização do *ensemble* de classificadores *Gradient Boosting Forests* é o seu bom desempenho em base de dados desbalanceada.

Outro problema encontrado na predição de notas foram os erros cometidos pelos corretores gramaticais *ReGra* e *CoGrOO*:o revisor não interveio ou o fez incorretamente. Pinheiro (2007) destacou que existem desvios da norma culta para os quais o *ReGra* não está preparado, em função da complexidade linguística imputada ao processamento automático das redações. O mesmo foi observado para o *CoGrOO*. O mau uso dos sinais de pontuação e a escolha inadequada de palavras estão entre os erros de difícil detecção pelo sistema. Por fim, salientamos que os revisores gramaticais *ReGra* e *CoGrOO* deixam a desejar quando existem erros de concordância entre palavras não adjacentes na frase. Apresentamos alguns desses problemas na próxima seção.

Na próxima seção, apresentamos alguns exemplos de avaliação automática de redação executadas pelo sistema construído no âmbito do estudo aqui relatado.

5.6 EXEMPLOS

Nesta seção, apresentamos exemplos de redação do nosso conjunto de teste, selecionados manualmente para ilustrar os pontos fortes e fracos do sistema, além dos problemas encontrados.

5.6.1 Exemplo 1

Nosso primeiro exemplo relata alguns problemas encontrados na identificação dos erros ortográficos e gramaticais. A Figura 5.9 mostra uma redação à qual o sistema Pessay atribuiu nota 1,0, enquanto o avaliador atribuiu nota máxima 2,0.

Figura 5.9 – Redação nota 2,0

Banco de Redações Título: Bipartição dos Três Poderes

Banco de Redações
Vestibular Brasil Escola

Redação enviada em 06/05/2014 00:22

Platão já advertia em "A República" sobre os perigos da concentração do poder estatal nas mãos de uma só secção, e ali concebia a teoria da separação dos poderes do Estado, que dois mil anos depois culminou na tripartição de poderes moderna apresentada por Montesquieu em "O Espírito das Leis". No Brasil, esse modelo, também conhecido como tripartite, foi adotado na constituição de 1988 e perpertua até hoje.

O benefício central desse modelo é a limitação e fiscalização de uma repartição sobre a outra, impedindo abusos e parcialidades. Mas dentro do jogo político firmado no Brasil, assegurar tal imparcialidade é artigo de luxo. Leis são vetadas e colocadas novamente em pauta favorecendo partidos e alianças políticas, onde andam legislativo e executivo a passos coordenados e cabendo ao judiciário ser o tripé que mantém a integridade e neutralidade do modelo.

Cabe, ainda, apontar que essa divisão de poderes não é equiparada, notando por exemplo a grande supremacia do Palácio do Planalto sobre o Congresso Nacional, uma vez que o segundo não tem meios para negociação com o primeiro, estando "à mercê" de seus desmandos asseguradas até por constituição. Em Alguns países da América Latina, certos poderes do legislativo são concedidos ao executivo, exemplificando a visão preeminente sobre esta partição do poder.*

Portanto, verifica-se uma falha de ~~equanimidade~~ equidade na distribuição de pesos dentro do modelo no Brasil, a exceção fica ao judiciário que tem total autonomia. Já os poderes legislativo e executivo usufruem de uma protocooperação onde ambas são beneficiadas desta convivência, com certa independência do segundo sobre o primeiro, em uma singela bipartição dos três poderes pensados por Montesquieu.

Fonte: Brasil Escola (acesso em: 6 nov. 2016).

A Tabela 5.10 mostra parte da frase com os erros e os tipos de erros detectados pelo revisor *ReGra*. Eles foram grifados em laranja na Figura 5.9, para melhor visualização. Os erros ortográficos identificados pelo sistema são apresentados na Tabela 5.11 e estão marcados em verde na Figura 5.7.

A Tabela 5.11 mostra algumas correções equivocadas do *ReGra*. Na primeira linha, podemos verificar que o *ReGra* detectou um erro de concordância verbal na palavra "culminou", pois considerou que ela estava se referindo a "esses anos". Porém, podemos observar na Figura 5.9 que a palavra "culminou" estava concordando com "a teoria da separação dos poderes do Estado".

Na segunda linha da Tabela 5.11, encontramos mais um equívoco do revisor *ReGra*, pois legislativo e executivo não flexionam quanto ao número nesse contexto. Outro erro do revisor *ReGra* foi na identificação da falta de crase, mostrado na terceira linha. Por fim, na última linha, o *ReGra* detectou, de forma correta, um erro na palavra "ambas", pois ela deveria concordar em gênero com a palavra "poderes", ao contrário do avaliador, que não identificou tal erro.

Tabela 5.11 – Erros gramaticais identificados pelo sistema *Pessay* na redação do Exemplo 1 utilizando o ReGra como revisor

Erro identificado	Regra	Sugestão
dois mil anos depois culminou na tripartição	Concordância Verbal	Culminaram
onde andam legislativo e executivo a passos	Concordância Verbal	legislativos e executivos
a exceção fica ao judiciário	Uso de crase	à exceção
onde ambas	Concordância Nominal	onde ambos

Fonte: elaborado pelo autor a partir de dados gerados pelo sistema *Pessay.*

Tabela 5.12 – Erros ortográficos identificados pelo Sistema *Pessay*

Palavra errada	Sugestão
Protocooperação	-----
Perpetua	Perpetua

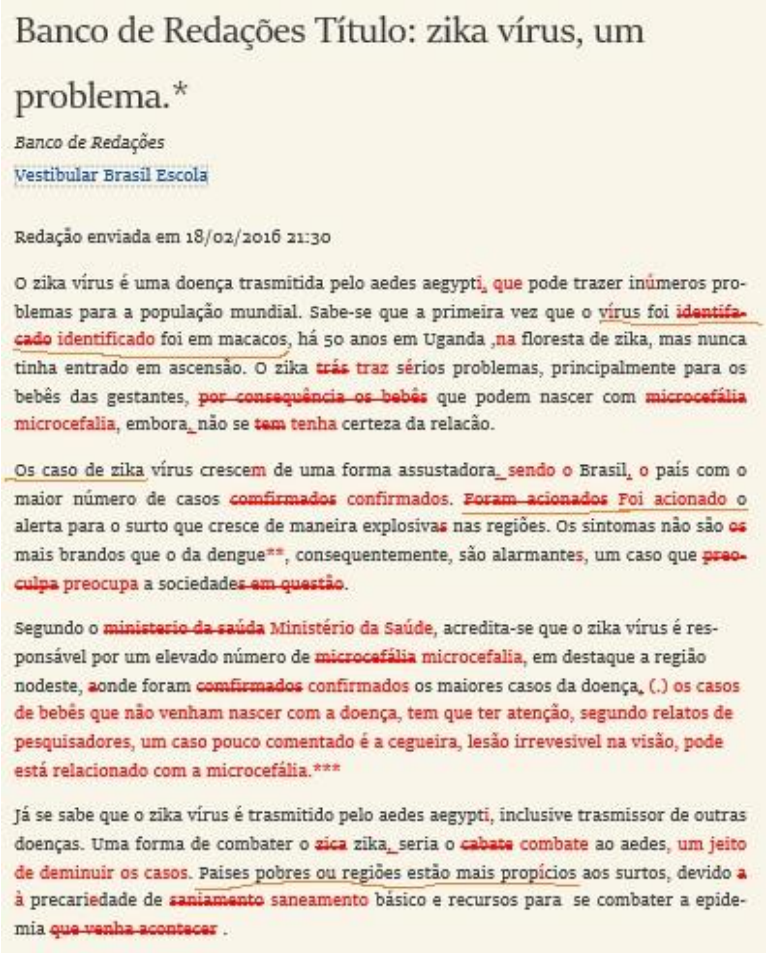
Fonte: elaborado pelo autor a partir de dados gerados pelo sistema Pessay.

Quanto aos erros ortográficos exibidos na Tabela 5.12, verificamos que o sistema considerou "protocooperação" como erro, pois a palavra não consta no dicionário usado pelo sistema *Pessay*. Já na segunda linha, o sistema considerou, de forma correta, palavra "perpertua" como erro, ao contrário do que fez o avaliador, que não o identificou.

5.6.2 Exemplo 2

Nesse exemplo, mostramos uma redação pontuada com 0,5 pelo sistema *Pessay* e pelo avaliador. A Figura 5.10 mostra a redação.

Figura 5.10 – Redação nota 0,5



Banco de Redações Título: zika vírus, um problema.*

Banco de Redações

Vestibular Brasil Escola

Redação enviada em 18/02/2016 21:30

O zika vírus é uma doença trasmitida pelo aedes aegypti, que pode trazer inúmeros problemas para a população mundial. Sabe-se que a primeira vez que o vírus foi ~~identifacado~~ identificado foi em macacos, há 50 anos em Uganda ,na floresta de zika, mas nunca tinha entrado em ascensão. O zika ~~trás~~ traz sérios problemas, principalmente para os bebês das gestantes, ~~por consequência os bebês~~ que podem nascer com ~~microcefália~~ microcefalia, embora não se ~~tem~~ tenha certeza da relacão.

Os caso de zika vírus crescem de uma forma assustadora, sendo o Brasil, o país com o maior número de casos ~~comfirmados~~ confirmados. ~~Foram acionados~~ Foi acionado o alerta para o surto que cresce de maneira explosiva~~s~~ nas regiões. Os sintomas não são ~~os~~ mais brandos que o da dengue**, consequentemente, são alarmantes, um caso que ~~preocupa~~ preocupa a sociedade~~s em questão~~.

Segundo o ~~ministerio da saúda~~ Ministério da Saúde, acredita-se que o zika vírus é responsável por um elevado número de ~~microcefália~~ microcefalia, em destaque a região nodeste, ~~a~~onde foram ~~comfirmados~~ confirmados os maiores casos da doença, (.) os casos de bebês que não venham nascer com a doença, tem que ter atenção, segundo relatos de pesquisadores, um caso pouco comentado é a cegueira, lesão irrevesivel na visão, pode está relacionado com a microcefália.***

Já se sabe que o zika vírus é trasmitido pelo aedes aegypti, inclusive trasmissor de outras doenças. Uma forma de combater o ~~zica~~ zika, seria o ~~cabate~~ combate ao aedes, um jeito de deminuir os casos. Paises pobres ou regiões estão mais propícios aos surtos, devido ~~a~~ à precariedade de ~~saniamento~~ saneamento básico e recursos para se combater a epidemia ~~que venha acontecer~~ .

Fonte: Brasil Escola (acesso em: 8 nov. 2016).

A Tabela 5.13 exibe os erros gramaticais identificados pelo *ReGra*, um dos revisores usados com o sistema *Pessay*.

Tabela 5.13 – Erros gramaticais identificados pelo sistema *Pessay* com o *ReGra*

Erro identificado	Regra	Sugestão
Os caso de zika vírus crescem.	Concordância Verbal	Os casos de zika crescem
Foram acionados o alerta para	Concordância Verbal	os alertas
Paises pobres ou regiões estão mais propícios	Concordância Nominal	propícias
que o vírus foi identifacado foi em macacos	Regência Verbal	foi a

Fonte: elaborado pelo autor a partir de dados gerados pelo sistema *Pessay.*

Na Tabela 5.13, podemos verificar que o *ReGra* identificou quatro erros gramaticais, sendo que somente o erro mostrado na primeira linha não foi identificado pelo avaliador.

O sistema *Pessay* identificou 14 erros ortográficos, entre eles: “transmitida”, “identifacado”, “microcefalia”, “relação”, “preocupa”, “ministério”, “saúda”, “nodeste”, “transmissor”, “cabate”, “deminuir”, “países” e “saneamento”.

5.6.3 Exemplo 3

Nesse exemplo, mostramos uma redação pontuada com 1.5 pelo sistema *Pessay* e com 0,5 pelo especialista do site. A Figura 5.11 mostra a redação.

Figura 5.11 – Redação nota 1,5

Desde o processo de **urbanização, e, com ele [urbanização e, com ele,]** o problema do "inchaço urbano" das grandes cidades, que **ouve-se [se ouve]** falar acerca da falta de moradia, saúde e educação de qualidade. O fato de não ter uma **infra-estrutura [infraestrutura]** planejada nos grandes centros **urbanos, faz [sem vírgula]** com que haja inúmeras pessoas vivendo em precárarias condições de vida, em lugares com estrutura sub humana.

Vemos como resultado de uma constante desigualdade e injustiças **sociais, a favelização [sem vírgula]**, que faz com que famílias inteiras se submetam a essas moradias, **devido a não ter [por não terem]** outro lugar para morar. Como consequência da favelização, **lembramos [lembramo-nos]** do deplorável episódio que ocorreu no começo do século XX, mais precisamente, no ano de 1902, na cidade do Rio de **Janeiro. Onde [Janeiro, quando]** a população das comunidades carentes foram acometidas por varíola e outras doenças, decorrentes da falta de saneamento básico e atendimento.

A falta de estrutura, a carência de planejamento, foi decorrendo, de modo que, hoje, existem milhões de pessoas , em todo o mundo, habitando lugares como esses, na maioria das vezes, sem segurança alguma. Diariamente, correndo riscos incalculáveis. São mães, pais, trabalhadores, que quando voltam do trabalho **[,]** não **tem [têm]** a certeza que chegarão vivos em casa, devido ao elevado nível de violência. Pessoas essas que, constantemente, adoecem por causa da precária condição de vida, moradia.

Diante disso, é preciso que haja intervenção para a não proliferação das favelas. **um [Um]** planejamento adequado, organizado, é viável, em meio a uma população que cresce dia após dia. Pois todo ser **humano, é [sem vírgula]** assegurado perante a lei a ter saúde, educação, segurança e moradia de qualidade, pois é um direito inalienável de todos, independente da etnia, classe, religião, todos, sem exceção. **Porantanto [Portanto,]** tem que haver o cumprimento das leis.

Fonte: Brasil Escola (acesso em: 10 nov. 2016).

A Tabela 5.14 exibe os erros gramaticais identificados pelo *CoGrOO*. Pelas correções do avaliador, podemos notar certa quantidade de erros de pontuação, principalmente de colocação de vírgula, erros que o *CoGrOO* e o módulo ortográfico não conseguem identificar. Outro erro não identificado pelo sistema mostrado no exemplo foi a escolha de expressões que não agradam ao avaliador. Por exemplo, no segundo parágrafo, o avaliador considera a expressão "por não terem" mais adequada do que "devido a não ter".

Tabela 5.14 – Erros gramaticais identificados pelo sistema *Pessay* na redação do Exemplo 1 utilizando o *CoGrOO* como revisor

Erro identificado	Regra	Sugestão
São mães, pais, trabalhadores, que quando voltam do trabalho	Concordância adjetivo com substantivo	

Fonte: elaborado pelo autor a partir de dados gerados pelo sistema *Pessay.*

O módulo ortográfico identificou quatro erros: “infra-estrutura”, “precárarias”, “vivos” e “porantanto”. Ressaltamos que a palavra “vivos” foi identificada como erro, provavelmente, em função de falha no algoritmo de probabilidade bayesiana, usado para analisar o contexto da palavra.

6 CONCLUSÕES E TRABALHOS FUTUROS

6.1 CONCLUSÃO

Segundo CHUIEIRE (2008), reduzir a avaliação à medida ou mais especificamente à prova implica aceitar a confiabilidade da prova como instrumento de medida e desconsiderar que a subjetividade do avaliador pode interferir nos resultados da avaliação. No entanto, este trabalho visa imitar o comportamento do humano avaliador de redação ENEM, atribuindo nota para uma redação na Competência 1. Contudo, há na literatura instrumentos para a avaliação e, se for desejável, minimizar a subjetividade dos avaliadores humanos em processos semelhantes a este, de avaliação de redações (JONHSON, NADAS, BELL, 2010)

Diferente de algumas propostas na área de avaliação automática de redações, que avaliam a redação como um todo, este trabalho apresentou um sistema de avaliação automática de redações do Enem com foco na Competência 1 – demonstrar domínio da modalidade escrita formal da Língua Portuguesa.

O sistema foi treinado e testado com redações extraídas de dois bancos de redações dos sites UOL: Educação UOL e Brasil Escola.

Experimentamos dois revisores gramaticais para o português do Brasil, *ReGra* e *CoGrOO*. Apesar de o *ReGra* obter resultados um pouco melhores, este é um *software* proprietário e difícil de embarcar, ou seja, de ser incluído no sistema. Já o *CoGrOO* é um *software open source*, desenvolvido na linguagem de programação *Java* e de fácil utilização. Conforme demonstramos na Seção 5.5.2, ambos possuem dificuldade para identificar alguns tipos de erros.

A predição de notas feitas pelo sistema foi comparada com as notas atribuídas pelos avaliadores das redações disponibilizadas no *site*, chegando a um erro médio absoluto de 0,2354 em 2,0 na exatidão. Com uma distância de 0,5 ponto (um nível) da nota atribuída pelo especialista do *site*, o sistema desenvolvido no âmbito deste

estudo obteve excelentes resultados, chegando a 0,0224, 0.9773 e 0.9777 para erro médio absoluto, *precision* e *recall*, respectivamente.

O método de avaliação de cada competência definida pelo Enem considera discrepância a diferença de mais de 80 pontos (dois níveis) entre duas notas. Não havendo tal discrepância, a nota da competência é dada pela média aritmética das duas notas atribuídas pelos dois avaliadores que corrigiram o texto. Assim, podemos afirmar que, pelos bons resultados obtidos, o uso do sistema de avaliação automática *Pessay* é capaz de apoiar os avaliadores nas correções, diminuindo seu trabalho na avaliação da Competência 1.

O sistema também pode ser adaptado para avaliar a norma culta em textos e questões discursivas. Além disso, pode ser integrado a plataformas de correções de redações no modelo do Enem, como a plataforma *Moodle*. Ainda, constitui um passo inicial para o desenvolvimento de uma ferramenta de avaliação automática de redações do Enem com foco nas outras competências do exame.

6.2 TRABALHOS FUTUROS

A partir da realização do estudo aqui relatado, sugerimos que trabalhos futuros sejam desenvolvidos com os seguintes enfoques:

I. implementar um algoritmo para identificar redações zeradas por influência de outras competências;

II. experimentar o sistema com uma base mais balanceada, com mais redações com notas zero, 0.5 e 2;

III. melhorar a identificação de erros ortográficos, incluindo dicionário com capacidade de aprender, além de aprimorar o algoritmo de identificação de erros de colocação de palavras, assim como os dicionários de *bigrams* e *trigrams*;

IV. acreditamos que o aperfeiçoamento da identificação de erros gramaticais é crucial para obter melhores resultados, de modo que é importante solucionar algumas deficiências encontradas no *ReGra* e *CoGrOO*, visando ao

reconhecimento de erros de concordância e regência entre palavras distantes na frase; outra melhoria seria estudar uma forma de atribuir pesos para cada tipo de erro;

V. testar novos classificadores, como *Deep Learning*, redes neurais de alta profundidade.

REFERÊNCIAS

BAEZA-YATES et al. **Modern information retrieval**. 2. ed. Boston, MA, USA: ACM Press New York, 2011.

BAZELATO, B. S.; AMORIM, E. C. de. A bayesian classifier to automatic correction of portuguese essays. In: CONGRESSO INTERNACIONAL DE INFORMÁTICA EDUCATIVA, 18., Porto Alegre, 2013. **Anais...** v. 9, p. 779-782, 2013. Disponível em: <http://www.tise.cl/volumen9/TISE2013/779-782.pdf>. Acesso em: 14 fev. 2017.

BRASIL ESCOLA. Disponível em: <http://vestibular.brasilescola.uol.com.br/banco-de-redacoes/>. Acesso em: 4 jul. 2017.

_____. Disponível em: <http://vestibular.brasilescola.uol.com.br/banco-de-redacoes/7939/>. Acesso em: 6 nov. 2016.

_____. Disponível em: <http://vestibular.brasilescola.uol.com.br/banco-de-redacoes/10826/>. Acesso em: 8 nov. 2016.

_____. Disponível em: <http://vestibular.brasilescola.uol.com.br/banco-de-redacoes/7037/>. Acesso em: 10 nov. 2016.

_____. Disponível em: <http://vestibular.brasilescola.uol.com.br/banco-de-redacoes/7037/>. Acesso em: 22nov. 2016.

BORBA, F. S. Dicionário de Usos do Português do Brasil. São Paulo: Ática, 2002.

BURSTEIN, J.; CHODOROW, M.; LEACOCK, C. Criterion^{SM} online essay evaluation: an application for automated evaluation of student essays. In: CONFERENCE ON INNOVATIVE APPLICATIONS OF ARTIFICIAL INTELLIGENCE, 15., Acapulco, México, 2003. **Proceedings…** p. 3-10, 2003.Disponível em: <https://www.ets.org/Media/Research/pdf/erater_iaai03_burstein.pdf>. Acesso em: 14 fev. 2017.

BURSTEIN, J. et al. Automated scoring using a hybrid feature identification technique. In: ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS AND 17TH INTERNATIONAL CONFERENCE ON COMPUTATIONAL LINGUISTICS, 36., 1998, Montreal, Canadá. **Proceedings…** v. 1., p. 206-210, 1998 Disponível em: <https://www.ets.org/Media/Research/pdf/erater_acl98.pdf>. Acesso em: 14 fev. 2017.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine learning**, v. 20, n. 3, p. 273-297, Springer, 1995.

CHUIEIRE, Mary Stela Ferreira. "**Concepções sobre a avaliação escolar**." *Estudos em Avaliação Educacional* 19.39 (2008): 49-64.

DIAS-DA-SILVA, B. C. **A face tecnológica dos estudos da linguagem: o processamento automático das línguas naturais**. Tese (Doutorado em Letras) – Universidade Estadual Paulista, Araraquara, 1996. Disponível em: <http://wiki.icmc.usp.br/images/a/ad/DiasDaSilva1996.pdf>. Acesso em: 14 fev. 2017.

FERNANDES, E. R. Q. et al. Ensembles de classificadores para bases de dados desbalanceadas: uma abordagem baseada em amostragem evolucionária. In: SYMPOSIUM ON KNOWLEDGE DISCOVERY, MINING AND LEARNING, 2., São Carlos, 2014. Disponível em: <http://www.producao.usp.br/bitstream/handle/BDPI/48647/2521782.pdf?sequence=1&isAllowed=y>. Acesso em: 14 fev. 2017.

FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. **Annalsofstatistics**, Jstor, p. 1189-1232, 2001. Disponível em: <https://statweb.stanford.edu/~jhf/ftp/trebst.pdf>. Acesso em: 14 fev. 2017.

GONZALEZ, M.; LIMA, V. L. S. Recuperação de informação e processamento da linguagem natural. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 23., 2003, Campinas.**Anais…** v. 3, p. 347-395, 2003. Disponível em: <http://www.inf.pucrs.br/gonzalez/docs/minicurso-jaia2003.pdf>. Acesso em: 14 fev. 2017.

GUTHRIE, L. et al. The role of lexicons in natural language processing. **Communications of the ACM**, ACM, v. 39, n. 1, p. 63-72, 1996.

HSU, C.-W.; LIN, C.-J. A comparison of methods for multiclass support vector machines.**IEEE transactions on Neural Networks**, IEEE, v. 13, n. 2, p. 415-425, mar. 2002.

INEP. **Edital nº 10, de 14 de abril de 2016 – Exame Nacional do Ensino Médio– Enem 2016**. Disponível em: <http://download.inep.gov.br/educacao_basica/enem/edital/2016/edital_enem_2016.pdf>. Acesso em: 4 jul. 2017.

INFO ESCOLA. Disponível em: < https://www.infoescola.com/portugues/incoerencias-da-ngb-em-relacao-a-classificacao-dos-vocabulos//>. Acesso em: 10 set. 2017.

JOACHIMS, T. Making large-scale SVM learning practical. In: **Advances in kernel methods:** support vector learning. SCHÖLKOPF, B.; BURGES, C. J. C.; SMOLA, A. J. (Eds.). Cambridge, USA: MIT Press, 1998. p. 169-184. Disponível em: <https://www.cs.cornell.edu/people/tj/publications/joachims_99a.pdf>. Acesso em: 14 fev. 2017.

JONHOSON, M.; NADAS, R.; BELL, J.B. Marking essays on screen: An investigation into the reliability of marking extended subjective texts. British Journal of Educational Technology. 2010

JOSÉ, J.; PAIVA, R.; BITTENCOURT, I. I. Avaliação automática de atividades escritas baseada em algoritmo genético e processamento de linguagem natural: Avaliador ortográfico-gramatical. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 4., 2015, Maceió. **Anais dos workshops...** v. 4, n. 1, p. 95. Disponível em: <http://www.br-ie.org/pub/index.php/wcbie/article/view/5936/4164>. Acesso em: 14 fev. 2017.

JUNKER, M.; HOCH, R.; DENGEL, A. On the evaluation of document analysis components by recall, precision, and accuracy. In: INTERNATIONAL CONFERENCE ON DOCUMENT ANATYSIS AND RECOGNITION, 5., Bangalore, Índia.

Proceedings… p. 713-716, IEEE, 1999. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=791711>. Acesso em: 14 fev. 2017.

KENNEDY, J. Dualidade educacional. In: SAMMUT, C.; WEBB, G. I. (Eds.). ENCYCLOPEDIA OF MACHINE LEARNING. New York: Springer, 2011. p. 136-141.

LORENA, A. C.; CARVALHO, A. C. de. Estratégias para a combinação de classificadores binários em soluções multiclasses. **Revista de Informática Teórica e Aplicada**, v. 15, n. 2, p. 65-86, 2008. Disponível em: <www.seer.ufrgs.br/rita/article/download/rita_v15_n2_p65-86/4486>. Acesso em: 14 fev. 2017.

LUNA, E. Á. D. A. **Avaliação da produção escrita no Enem:** como se faz e o que pensam os avaliadores. Dissertação (Mestrado em Letras). Programa de Pós-Graduação em Letras, Universidade Federal de Pernambuco, Recife, 2009. Disponível em: <http://repositorio.ufpe.br/xmlui/bitstream/handle/123456789/7499/arquivo3889_1.pdf?sequence=1&isAllowed=y>. Acesso em: 14 fev. 2017.

MACHADO, E. L.; LADEIRA, M. Um estudo de limpeza em base de dados desbalanceada e com sobreposição de classes. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 17., Brasília, 2007. **Anais...** p. 330-340, 2007.Disponível em: <http://www.dcc.fc.up.pt/~ines/enia07_html/pdf/28076.pdf>. Acesso em: 14 fev. 2017.

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

NADKARNI, P. M.; OHNO-MACHADO, L.; CHAPMAN, W. W. Natural language processing: an introduction. **Journal of the American Medical Informatics Association**, The Oxford University Press, v. 18, n. 5, p. 544-551, 2011. Disponível em: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3168328/>. Acesso em: 14 fev. 2017.

APACHEOPENNLP. Disponível em: <https://opennlp.apache.org//>. Acesso em: 4 fev. 2016.

SAÚDE, M. R. **Uma estratégia para moderação de um grande conjunto de comentários de usuários**. Dissertação (Mestrado em Informática) – Programa de Pós-graduação em Informática, Centro Tecnológico, Universidade Federal do Espírito Santo, Vitória, 2014. Disponível em: <http://repositorio.ufes.br/bitstream/10/4272/1/tese_8208_dissertacao_marcos_r_saude.pdf>. Acesso em: 14 fev. 2017.

PAGE, E. B. The imminence of... grading essays by computer. **The Phi Delta KappanInternacional**, Jstor, v. 47, n. 5, p. 238-243, 1966. Disponível em: <https://www.jstor.org/stable/20371545?seq=1#page_scan_tab_contents>. Acesso em: 14 fev. 2017.

PAGE, E. B. Computer grading of student prose, using modern concepts and software. **The Journal of experimental education**, Taylor & Francis, v. 62, n. 2, p. 127-142, 1994.

PASQUALI, L. **Psicometria:** teoria dos testes na psicologia e na educação. Petrópolis: Vozes, 2003.

PINHEIRO, G. M. **Redações do Enem:** estudo dos desvios da norma padrão sob a perspectiva de corpos. Dissertação (Mestrado *em* Letras) – Programa de Pós-graduação em Estudos Linguísticos e Literários em Inglês, Universidade de São Paulo, São Paulo, 2007. Disponível em: <http://www.teses.usp.br/teses/disponiveis/8/8147/tde-30072008-104245/pt-br.php>. Acesso em: 14 fev. 2017.

PISSINATI, E. **Uma proposta de correção semiautomática de questões discursivas e de visualização de atividades para apoio à atuação do docente**. Dissertação (Mestrado em Informática) – Programa de Pós-graduação em Informática, Centro Tecnológico, Universidade Federal do Espírito Santo, Vitória, 2014.

SILVA, W. D. C. de M. **Aprimorando o corretor gramatical *CoGrOO***. Tese (Mestrado Ciência da Computação) – Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2013. Disponível em: <http://www.teses.usp.br/teses/disponiveis/45/45134/tde-02052013-135414/publico/WilliamColen_Dissertation.pdf>. Acesso em: 14 fev. 2017.

SOUZA, F. P. de; CIARELLI, P. M.; OLIVEIRA, E. de. Combinando fatores de ponderação para melhorar a classificação de textos. In: COMPUTER ON THE BEACH, Florianópolis, 2014. **Anais…** p. 32-41, 2014.Disponível em: <http://siaiap32.univali.br/seer/index.php/acotb/article/view/5293/2763>. Acesso em: 14 fev. 2017.

TUMER, K.; GHOSH, J. Analysis of decision boundaries in linearly combined neural classifiers. **PatternRecognition**, Elsevier, v. 29, n. 2, p. 341-348, 1996.Disponível em: <http://www.ideal.ece.utexas.edu/pubs/pdf/1996/tugh96.pdf>. Acesso em: 14 fev. 2017.

UOL EDUCAÇÃO. Disponível em: <https://educacao.uol.com.br/bancoderedacoes/redacoes/bandido-bom-e-bandido-recuperado.htm>. Acessoem: 12 nov. 2016.

VIEIRA, R. **Textual co-reference annotation:** a study on definite descriptions. 2000. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.9051&rep=rep1&type=pdf>. Acesso em: 14 fev. 2017.

WELKER, H. A. Breve histórico da metalexicografia no Brasil e dos dicionários gerais brasileiros. Matraga, Rio de Janeiro, ano 13, n. 19, 2006.

WILKS, Y. et al. **Dictionaries, computers and meanings**. Cambridge, Mass: The MIT Press, 1996.

ZHOU, Z.-H. Ensemble learning. In: LI, S. Z. (Ed.). ENCYCLOPEDIA OF BIOMETRICS. New York: Springer, 2015, p. 411-416.

APÊNDICE A - PROCESSAMENTO DE LINGUAGEM NATURAL

A.1 REGRAS GRAMATICAIS DO *REGRA*

1. Uso de Letras Maiúsculas
2. Uso de Arcaísmos
3. Uso de Parônimos
4. Inadequação Lexical
5. Concordância de Modos e Tempos Verbais
6. Máximo de Palavras Repetidas por Frase
7. Uso do Adjetivo
8. Uso de Estrangeirismos
9. Uso de Preposições
10. Uso de "mau"e "mal"
11. Uso de Neologismos
12. Concordância Nominal
13. Repetição de Palavras ou Símbolos
14. Uso de "onde"e "aonde"
15. Regência Verbal
16. Regência Nominal
17. Uso de Crase
18. Balanceamento de Delimitadores
19. Uso de Siglas e abreviaturas
20. Uso do Hífen
21. Uso de Artigos e Determinantes
22. Uso de Plebeísmos
23. Uso de Conjunções e Locuções Conjuncionais

24. Uso do Particípio Passado
25. Uso de Clichês
26. Uso dos Sinais de Pontuação
27. Uso de "por que"e variantes
28. Acentuação Gráfica
29. Uso de Pronomes
30. Colocação Pronominal
31. Uso de "há" e "a"
32. Máximo de Palavras por Frase
33. Pleonasmo Vicioso
34. Concordância Verbal

A.2 ETIQUETAS MORFOLÓGICAS

Tabela A.1 – Etiquetas morfológicas *Apache OpenNLP*

Símbolo	Categoria
n	substantivo
prop	nome próprio
adj	adjetivo
v	verbo
v-fin	verbo finito verbo
v-inf	infinitivo
v-pcp	verbo particípio
v-ger	gerúndio
art	artigo
det	artigo determinado
pron-pers	pronome pessoal
adv	advérbio
num	numeral
prp	preposição
int / in	interjeição
punc	pontuação

Fonte: APACHE OPENNLP (acesso em: 4fev. 2016).

A.3 REGRAS GRAMATICAIS DO *COGROO*

Tabela A.2 – Regras gramaticais *CoGrOO*

(continua)

Tipo	Grupo
Crase	à + substantivo masculino
Crase	à + substantivo masculino
Crase	à + substantivo feminino plural
Crase	à + verbo
Crase	a + indicador de horas
Crase	a + indicador de horas
Crase	a + lado de
Crase	a + lado de
Crase	em relação a + substantivo feminino
Crase	em relação a + substantivo feminino
Crase	com relação a
Crase	com relação a
Crase	devido a
Crase	devido a
Crase	à + pronomes de tratamento
Crase	à + pronomes de tratamento
Concordância adjetivo-substantivo	substantivo + adjetivo
Concordância adjetivo-substantivo	substantivo + adjetivo
Concordância adjetivo-substantivo	substantivo + adjetivo
Concordância adjetivo-substantivo	substantivo + adjetivo
Concordância adjetivo-substantivo	adjetivo + substantivo
Concordância adjetivo-substantivo	adjetivo + substantivo
Concordância adjetivo-substantivo	adjetivo + substantivo
Concordância adjetivo-substantivo	adjetivo + substantivo
Em anexo	em anexo
Em anexo	em anexo
Em anexo	em anexo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo

Em anexo	anexo como adjetivo
Em anexo	anexo como adjetivo
Uso de meio	meio no sentido de "um pouco"

Tabela A.2 – Regras gramaticais *CoGrOO*

(continuação)

Tipo	Grupo
Uso de meio	meio como adjetivo
Uso de meio	meio como adjetivo
Verbo fazer	fazer indicando tempo
Verbo fazer	fazer indicando tempo
Verbo fazer	verbo auxiliar + fazer indicando tempo
Verbo fazer	verbo auxiliar + fazer indicando tempo
Verbo haver	haver + denotação de tempo
Verbo haver	haver + denotação de tempo
Verbo haver	haver + denotação de tempo
Verbo haver	haver + denotação de tempo
Verbo haver	haver no sentido de existir
Verbo haver	haver no sentido de existir
Verbo haver	verbo auxiliar + haver no sentido de existir
Verbo haver	verbo auxiliar + haver no sentido de existir
Emprego do mim e ti	mim + verbo no infinitivo
Emprego do mim e ti	eu regido por preposição
Emprego do mim e ti	eu regido por preposição
Emprego do mim e ti	eu regido por preposição
Emprego do mim e ti	eu regido por preposição
Emprego de mau e mal	uso de mau
Emprego de mau e mal	uso de mau
Verbo preferir	preferir + redundância
Verbo preferir	regência do verbo preferir
Colocação pronominal	palavras de sentido negativo + verbo+ pron
Colocação pronominal	palavras de sentido negativo + verbo+ pron
Colocação pronominal	palavras de sentido negativo + subst.+ verbo
Colocação pronominal	palavras de sentido negativo + subst.+ verbo
Colocação pronominal	pronome relativo ou conjunção subordinativa
Colocação pronominal	pronome relativo ou conjunção subordinativa
Colocação pronominal	advérbio + verbo + pronome oblíquo
Colocação pronominal	advérbio + verbo + pronome oblíquo
Colocação pronominal	advérbio + verbo + pronome oblíquo
Colocação pronominal	advérbio + verbo + pronome oblíquo
Colocação pronominal	pronome indefinido + verbo + pronome oblíquo
Colocação pronominal	pronome indefinido + verbo + pronome oblíquo
Colocação pronominal	só, ou, ora ou quer + pronome oblíquo
Colocação pronominal	só, ou, ora ou quer + pronome oblíquo

Se eu ver	conjugação de um verbo irregular no futuro do subjuntivo
Se eu ver	conjugação de um verbo irregular no futuro do subjuntivo

Tabela A.2 – Regras gramaticais *CoGrOO*

(continuação)

Tipo	Grupo
Se eu ver	conjugação de um verbo irregular no futuro do subjuntivo
Crase	regência verbal
Crase	regência verbal
Crase	regência verbal
Crase	regência verbal
Crase	regência verbal
Concord. do sujeito com o adj. predicativo	pronome + verbo de ligação + adj predicativo
Crase	Crase - regência de alguns nomes
Crase	regência verbal - crase
regência verbal	regência do verbo obedecer/desobedecer
Crase	à + pronomes pessoais
Crase	à + pronomes pessoais
emprego de eu e mim	a + eu
regência verbal	regência do verbo namorar
Crase	crase - regência de alguns nomes - compl. Plural
Concordância adjetivo-substantivo	meio-dia e meia
Crase	regência verbal – crase
Crase	regência verbal – crase
Concordância artigo-substantivo	artigo plural + substantivo singular
regência verbal	regência do verbo evitar, usufruir
regência verbal	regência de demorar, torcer, votar
regência verbal	regência do verbo arrasar
regência verbal	regência verbo habituar-se
regência verbal	regência habituar com próclise
regência verbal	regência verbo habituar-se
regência verbal	regência verbo habituar-se
Concordância artigo-substantivo	artigo singular + substantivo plural
Concordância artigo-substantivo	artigo feminino + substantivo masculino
Concordância artigo-substantivo	artigo masculino + substantivo feminino
à medida em que/à medida que	vícios de expressão
regência verbal	verbo acarretar
Crase	segunda a sexta
regência verbal	assistir com o sentido de presenciar
regência nominal	valorização de
emprego de vírgulas	expressões entre vírgulas
emprego de vírgulas	expressões entre vírgulas

emprego de vírgulas	expressões entre vírgulas
Concordância determinante-substantivo	determinante singular + substantivo plural
Concordância numeral-substantivo	Concordância numeral-substantivo

Tabela A.2 – Regras gramaticais *CoGrOO*

(conclusão)

Tipo	Grupo
Concordância sujeito-verbo	sujeito plural + verbo singular
Concordância sujeito-verbo	sujeito plural + verbo singular
Gerundismo	Gerundismo
Uso do verbo haver	Redundância semântica
Expressões redundantes	Redundância semântica
Concordância do sujeito com o adjetivo predicativo	pronome + verbo de ligação + adjetivo predicativo

Fonte: Arquivo de regras do *CoGrOO, rules.xml*

APÊNDICE B - APRENDIZAGEM DE MÁQUINA

B.1 CARACTERÍSTICAS *REGRA*

Erros ortográficos, quantidade de parágrafos, frases, palavras, caracteres, *stopWords*, interrogações, exclamações, vírgulas, pontos, parágrafos grandes e pequenos, frases grandes e pequenas, parágrafos com vocabulário repetitivo, frases com palavras repetidas, erros gramaticais identificados pelo *ReGra* ,v-fin, pp, conj-c, art, adj, num, v-ger, punc, n, prp, pron-pers, conj-s, v-pcp, adv, prop, pron-indp, v-inf, ec, pron-det, mais os 34 erros gramaticais identificados pelo *ReGra* (Seção A.1).

B.2 CARACTERÍSTICAS *COGROO*

Erros ortográficos, quantidade de parágrafos, frases, palavras, caracteres, *stopWords*, interrogações, exclamações, vírgulas, pontos, parágrafos grandes e pequenos, frases grandes e pequenas, parágrafos com vocabulário repetitivo, frases com palavras repetidas, erros gramaticais identificados pelo *CoGrOO*, v-fin, pp, conj-c, art, adj, num, v-ger, punc, n, prp, pron-pers, conjs, v-pcp, adv, prop, pron-indp, v-inf, ec, pron-det, mais as 124 regras gramaticais do *CoGrOO* (Seção A.3).