



SICIT
Semana de Iniciação
Científica e Tecnológica

25 a 29 de setembro de 2017
Engenharias e Computação

 Universidade de Itaúna

Recuperação de Padrões na Valoração Textual de Redações

Eugênio Cunha¹

Marco Túlio Alves Nolasco Rodrigues¹

Resumo: Nos dias atuais, há uma quantidade intensa de redações sendo produzida e avaliada em vestibulares, concursos e exames. Diferentemente dos métodos existentes, que processam e avaliam as redações de maneira manual, este trabalho aborda uma forma automática, por meio de aprendizagem de máquina, capaz de generalizar, aprender e extrair padrões das classes de redações com base no conteúdo rotulado. O método precisa de pouca intervenção humana e permite a valoração de grandes quantidades de textos. Este trabalho fundamenta-se no problema de avaliação manual das competências exigidas em um texto de redação do tipo dissertativo-argumentativo com temas diversificados de ordem social, científica, cultural ou política. Dado um “corpus” de redações o objetivo principal é induzir um modelo a classificar automaticamente uma competência exigida em um texto de redação. Embasado nas principais métricas de análise dos classificadores citados na literatura de aprendizado de máquina, a solução proposta neste trabalho demonstrou ser útil e propícia a ser utilizada em problemas que envolva a valoração automática de texto.

Palavras-chave: Aprendizado de Máquina. classificação de textos. ENEM.

1. Ciência da Computação, Bacharelado, UIT, genio.py@gmail.com

2. Ciência da Computação, Doutorado, UIT, tulio.rodrigues@gmail.com

1. Introdução

O desenvolvimento de uma redação é uma atividade prática presente na cultura civilizada desde a invenção da escrita. Lara (1994) explica em seu trabalho que na década de 70 iniciou-se processo de redemocratização que conseqüentemente restituiu a palavra ao estudante. O decreto 79.298, de 24 de fevereiro de 1977 definiu a volta da redação à escola pela “inclusão obrigatória da prova ou questão de redação em língua portuguesa” nos concursos e vestibulares (Art. 1º, alínea d). A redação é aplicada no ENEM desde a primeira edição 1998, hoje o maior exame do Brasil, que no ano de 2016 contou com 8.627.195 escritos confirmados, e a participação direta de 11.360 profissionais externos na correção de 5.825.134 redações. Com o advento do ENEM ser um requisito para o processo seletivo de acesso às inúmeras universidades públicas (SISU, 2017) e a importantes programas de governo (CSF, 2017), este número tem aumentado incessantemente. Segundo o edital INEP (2016), cada redação foi avaliada por, pelo menos, dois avaliadores, de forma independente, uma estimativa mínima de 11.650.268 avaliações manuais das competências exigidas num texto pelo ENEM. Devido à grande quantidade de redações produzidas, torna-se humanamente difícil e caro organizar e avaliar as competências de uma redação manualmente.



2. TRABALHOS RELACIONADOS

Com o processamento computacional mais barato e poderoso, a crescente variedade e volume de dados disponíveis, e o armazenamento de forma acessível, o Aprendizado de Máquina está no centro de muitos avanços tecnológicos, alcançando as áreas antes exclusivas de seres humanos. Os carros autônomos do Google são o exemplo de uma atividade antes exclusivamente humana e hoje exercida e aperfeiçoada por algoritmos de Aprendizado de Máquina (WAYMO, 2017). O Aprendizado de Máquina está presente na nossa vida cotidiana como, resultados de pesquisa *web*, análise de sentimento baseado em texto e na detecção de fraudes em operações com cartões de crédito entre outras aplicações (BATISTA et al., 1999).

A avaliação de redações automática pode ser realizada utilizando sistemas especialistas ou algoritmos de Aprendizado de Máquina. A primeira hipótese dependente essencialmente da presença de especialistas que detêm o conhecimento sobre o domínio do problema para desenvolver um conjunto de regras. O sistema especialista deve ser capaz de tomar suas decisões, ou seja, as regras são disparadas para atingir determinada decisão (NEGNEVITSKY, 2005). Entretanto, regras desenvolvidas manualmente tem um processo de manutenção e atualização complexo, o que torna mais difícil a sua utilização em diferentes domínios do problema proposto. O uso de algoritmos de Aprendizado de Máquina para valoração de redações é uma alternativa ao sistema especialista, exige menor esforço humano com a abstração simples de extrair padrões ou características, aprender e generalizar. Dados os benefícios, a hipótese deste artigo é que um algoritmo de Aprendizado de Máquina pode ser útil e propício a ser utilizada em problemas que envolva a valoração de texto manual por profissionais capacitados.

Além disso, para avaliar e validar a hipótese, o método de construção do conhecimento deste trabalho terá como fundamento o problema de recuperação de padrões na valoração textual. Dado um *corpus* de redações o objetivo principal é induzir um modelo a classificar as competências exigidas compondo uma nota avaliativa sobre a redação. O presente estudo com base na proposta do problema descrito contribuirá na área do Aprendizado de Máquina e diretamente no processo de valoração de um texto em prosa do tipo dissertativo-argumentativo.

2. Trabalhos Relacionados

Segundo Silva e Carvalho (2017), a prova de redação do ENEM é avaliada considerando uma matriz de referência do INEP (2016). Essa matriz, foi desenvolvida com a colaboração de especialistas, com o objetivo de operacionalizar o exame. A matriz apresenta cinco competências, para cada competência expressa para redação existem níveis de conhecimento associados de 0 a 5. Braga (2015) explica no seu trabalho, que num texto de redação, o candidato defenderá uma opinião a respeito do tema proposto, de forma coerente e coesa, apoiado em argumentos consistentes. O texto será redigido a respeitar a escrita formal da Língua Portuguesa. Ao fim, o candidato elabora uma proposta de intervenção social para o problema apresentado no desenvolvimento do texto que respeite os direitos humanos.

No seu estudo Monard e Baranauskas (2003) cita: “A indução é a forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos”. Na indução, um conceito é aprendido efetuando-se inferência indutiva sobre as amostras apresentadas. O aprendizado indutivo pode ser dividido em supervisionado e não supervisionado como



2. TRABALHOS RELACIONADOS

ilustrada a Figura 1. No aprendizado não supervisionado, o algoritmo analisa os exemplos fornecidos e tenta determinar se alguns deles podem ser agrupados de alguma maneira, formando *clusters* ou agrupamentos. Já no aprendizado supervisionado é fornecido ao algoritmo de um conjunto de exemplos de treinamento para os quais o rótulo da classe associada é conhecido.

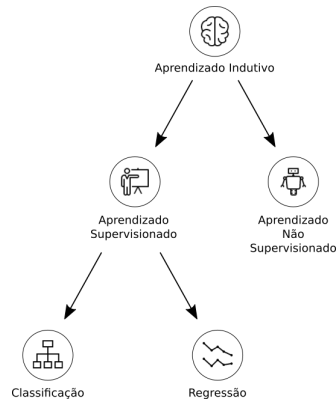


Figura 1. Árvore hierárquica do aprendizado indutivo, a qual é dividida em algoritmos supervisionado e não-supervisionado.

Classificadores são utilizados para a predição de classes de objetos e pode ser dita como o processo de generalização dos dados a partir de diferentes instâncias. Existe uma tendência de se referir a problemas com respostas quantitativas como “problemas de regressão” e aqueles com uma saída qualitativa como “problemas de classificação”. Dado um conjunto de exemplos como ilustrado na Figura 2, os classificadores devem encontrar uma função geral capaz de prever adequadamente as saídas para novas amostras. Após o treinamento, o classificador é avaliado e se necessário o processo de classificação pode ser ajustado usando o conhecimento sobre o domínio do problema, de acordo com o trabalho de [Motta \(2016\)](#).

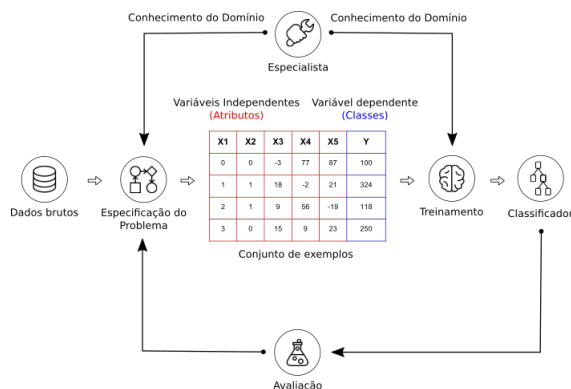


Figura 2. Fluxo do processo de classificação, o modelo encontra uma função geral capaz de prever as saídas, a especificação do problema pode ser reajustada com o conhecimento do domínio para obter um melhor resultado.

Diversas ferramentas disponíveis para exploração de dados dispõem de soluções para o



2. TRABALHOS RELACIONADOS

processamento e a análise das informações de forma ágil e simples. Numa análise comparativa [Boscarioli, Viterbo e Teixeira \(2014\)](#) demonstram que não existe uma única ferramenta com características melhores para todas as aplicações em mineração de dados. Em um estudo que comparou quatro ferramentas (KMINE, *Orange*, Tanagra, Weka), todas de código aberto, gratuitas e muito utilizadas na pesquisa e na academia, [Wahbeh et al. \(2011\)](#) concluíram: Weka apresentou o melhor desempenho, seguido pelo *Orange*, e, depois, pelo KMINE e Tanagra. De acordo com [\(DEMŠAR et al., 2013\)](#), a ferramenta *Orange* na atual versão 3.5 desenvolvida pelo laboratório de Inteligência Artificial da Faculdade de Computação e Ciência da Informação da Universidade de *Ljubljana* na *Eslovênia*, possui uma *interface* gráfica denominada *Orange Canvas*. Atravé de sua *interface*, é possível conectar e interligar os objetos montando um fluxo de trabalho para o desenvolvimento de modelos de classificação, incluindo *Adaboost*, *Naive Bayes*, Árvores de Decisão, etc.

No processo de mineração de dados, segundo [Matsubara, Martins e Monard \(2003\)](#), na etapa de pré-processamento de textos, um dos métodos geralmente adotado é a representação usando a abordagem “*bag-of-words*”, uma das representações estruturadas mais simples. Utiliza técnicas de redução do termo ao seu radical e remoção de termos irrelevantes. Cada documento é representado como um vetor de palavras que ocorrem no texto, especificamente uma tabela atributo-valor.

O algoritmo Naive Bayes destaca-se entre os classificadores devido ao seu comportamento simplista, traz bons resultados em muitos casos. Baseado no Teorema de Bayes, criado por Thomas Bayes no século XVIII, é possível encontrar a probabilidade de um certo evento ocorrer, dada a probabilidade de outro evento que já ocorreu. [BRITO \(2017\)](#), cita na sua pesquisa o classificador Naive Bayes como um progenitor probabilístico, dado a fórmula:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

Onde: $P(c|x)$: probabilidade da classe c dado o vetor x ; $P(x|c)$: probabilidade do vetor x dada a classe c ; $P(c)$: probabilidade a “priori” da classe c ; $P(x)$: probabilidade a “priori” do vetor de treinamento x .

Naive Bayes desconsidera a correlação entre as variáveis (“features”), ou seja, se determinada fruta é considerada uma “Maçã” se ela for “Vermelha”, “Redonda” e possui “cerca de 10 cm de diâmetro”, o algoritmo não vai considerar a correlação entre esses fatores, tratando cada um de forma independente. [Chakrabarti \(2002\)](#) descreve este classificador como o mais eficiente na precisão e rotulagem de novas amostras.

AdaBoost ou *Adaptive Boosting* (“impulso ou estímulo adaptativo”), é um classificador do tipo *Boosting*, um dos algoritmos mais populares no Aprendizado de Máquina. Utiliza uma técnica que seleciona diversos algoritmos denominados classificadores fracos, com a finalidade de constituir um classificador forte. Segundo [Reis \(2015\)](#), o seu sucesso deve-se ao mérito de conseguir adaptar-se aos classificadores de base. Neste algoritmo, os classificadores são gerados de forma a ajudar os exemplos incorretamente classificados pelos classificadores antecedentes, ele aumenta os pesos dos exemplos em que os classificadores anteriores cometeram erros, para indicar importância do exemplo no conjunto. Dado uma base de dados de entrada, o AdaBoost tem a função de encontrar um conjunto de características que construirão o classificador forte,

3. METODOLOGIA

favorecendo uma melhor classificação do conjunto de entrada.

3. Metodologia

Para concluir com êxito o desenvolvimento deste trabalho e conseqüentemente os objetivos propostos, o método utilizado para solução do problema é composto das seguintes etapas sequenciais:

3.1. Coleta de textos

Para as avaliações experimentais e análises realizadas neste estudo foram coletadas redações de dois diferentes projetos que estimulam o estudante a treinar a produção de textos do gênero dissertativo-argumentativo, sugerindo um tema, avaliando e publicando (ESCOLA, 2017) e (UOL, 2017).

Nos dias atuais consegue-se facilmente coletar textos de páginas *web*, para esta tarefa, foi necessário criar um *crawler*. Existem diversas formas de implementar um *crawler*, dentre elas, uma muito utilizada é o *Scrapy*, utilizado neste trabalho (SCRAPY, 2017). O uso de um *crawler*, permite explorar a estrutura de grafo da *web*, navegar de uma página para outra identificando as *tags* HTML que contém os dados necessários para compilar um *dataset*. A figura 3 ilustra a etapa em que o *crawler* navega entre as páginas HTML, filtra as *tags*, coleta e armazena os dados em um *dataset*.

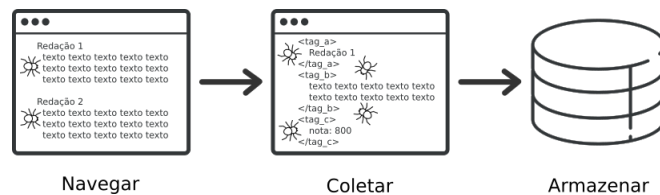


Figura 3. O *crawler*, navega entre as páginas HTML do banco de redações de forma metódica e automatizada indexando textos que posteriormente serão filtrados, coletados e armazenados.

3.2. Balanceamento de dados

Em muitos domínios, os conjuntos de dados são naturalmente desbalanceados. Dados desbalanceados representam o domínio onde qualquer classe de um grupo de dados está representado por um amplo número de exemplos, enquanto as demais classes são representadas por poucos exemplos. Abordagens ao nível de dados equilibram a distribuição das classes no conjunto de dados, usar técnicas como *undersampling* e *oversampling* resolvem o problema do desbalanceamento de acordo com o estudo de Ferreira, Dosciatti e Paraiso (2014). A técnica *oversampling* replica de forma aleatória, exemplos da classe minoritária, enquanto a técnica *undersampling* utilizada neste estudo, elimina aleatoriamente exemplos da classe majoritária. Além disso, Machado (2009) em seu estudo indica o uso das técnicas de limpeza de dados de modo a, eliminar os exemplos ruidosos e *limítrofes*, respectivamente (*class-label noise*, *borderlines*). A figura 4 ilustra a etapa onde os dados naturalmente desbalanceados são submetidos a técnica *undersampling* e limpeza de dados, resultando um *dataset* menor e balanceado.

3. METODOLOGIA

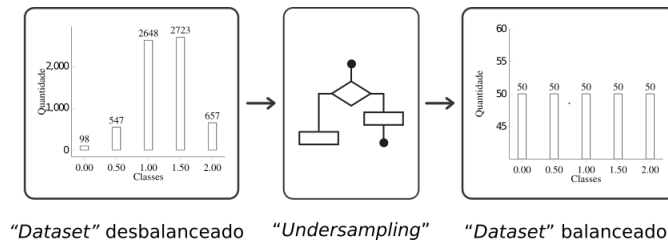


Figura 4. O *dataset* desbalanceado é submetido a técnica *undersampling* que gera um *dataset* menor e balanceado.

3.3. Pré-processamento, inferência indutiva e métricas de desempenho

A figura 5 ilustra as etapas necessárias para pré-processamento, indução e testes dos algoritmos classificadores. Devido à natureza textual não estruturada dos textos contidos no *dataset*, no primeiro passo os documentos armazenados necessitam de um pré-processamento. Cada sentença do texto é separada em *tokens* para transformar esses dados não estruturados em um formato estruturado, especificamente uma tabela atributo-valor, denominada *bag-of-words*. Nesta abordagem, palavras pouco significativas como artigos, preposições e conjunções que pouco caracterizam o texto pode ser ignorada com uma ou mais listas de *stopwords*. Segundo Matsubara, Martins e Monard (2003), este passo é importante, visto que a representação desses textos tem uma influência fundamental no resultado da indução dos algoritmos de Aprendizado de Máquina. No segundo passo é necessário definir os parâmetros da inferência indutiva de cada algoritmo e induzir os modelos classificadores *Adaboost* e *Naive Bayes*. O terceiro e último passo, o resultado da inferência dos classificadores são avaliados com as principais métricas de análise de classificadores citadas na literatura de Aprendizado de Máquina. Os passos dois e três são repetidos até que um dos classificadores apresente resultados relevantes ao estudo.

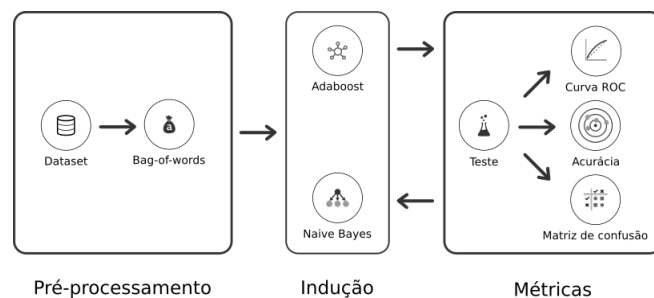


Figura 5. O *dataset* balanceado é submetido a técnica *bag-of-words* no pré-processamento, resultando em uma estrutura de atributo-valor utilizada na inferência indutiva do classificadores, por fim, os modelos induzidos são avaliados por métricas de desempenho.

3.4. Validação cruzada

Para avaliar e validar a hipótese proposta foi adotada a metodologia de validação cruzada, já prevista na própria ferramenta *Orange*. O estudo de Tavares, Lopes e Lima (2007) explica que esta abordagem consiste em fracionar o *dataset* em N partes (*folds*). Destas, $N-1$ partes são aplicadas na inferência indutiva e uma amostra é utilizada como base de testes. O método



4. RESULTADOS

é repetido N vezes, de forma que cada fração seja utilizada apenas uma vez como conjunto de testes. Por fim, é calculada pela média dos resultados atingidos em cada ciclo, obtendo-se assim uma estimativa da qualidade da inferência induzida, o que permite análises estatísticas. A Figura 6 ilustra o fracionamento do *dataset* em N partes, da qual, uma amostra é separada para testes e as demais para inferência indutiva, ao fim, é calculada a média dos resultados obtidos de cada métrica de desempenho.

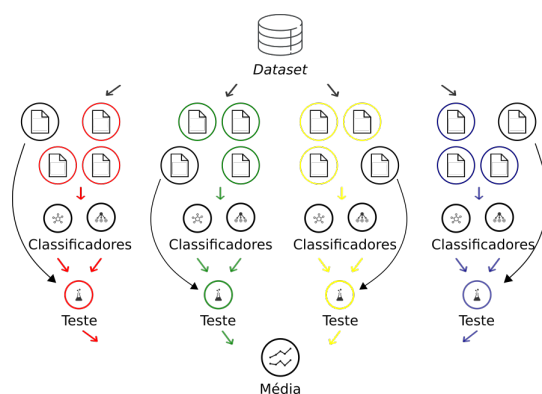


Figura 6. O *dataset* balanceado é fracionados em N partes, sendo uma parte separada para testes e as demais utilizadas na indução dos classificadores, por fim, é calculada a média dos resultados obtidos.

4. Resultados

Este tópico é dedicado a apresentar os resultados, adversidades e contribuições alcançadas durante o desenvolvimento do estudo referente ao problema proposto. Por fim, são apresentadas considerações sobre as limitações ocorridas no desenvolvimento deste trabalho. Nos resultados do problema proposto, este estudo utilizou as principais métricas da literatura para análise de desempenho dos classificadores.

4.1. Configuração do experimento

Alvo – Dada a matriz de referência do [INEP \(2016\)](#), a competência III foi selecionada aleatoriamente como o alvo da inferência indutiva dos classificadores.

Naive Bayes – Para o algoritmo Naive Bayes não foi preciso ajustar os parâmetros pois ele é não paramétrico.

AdaBoost – O Classificador base utilizado pelo AdaBoost foi a Árvore de decisão, com uma taxa de aprendizado configurado em 1,0 (um) e o número de iterações foi ajustado para 50 (cinquenta).

Validação cruzada – O *dataset* foi dividido em 10 conjuntos disjuntos com 69 textos. Os classificadores são treinados 10 vezes, cada vez com um conjunto diferente sendo deixado de fora para fazer a validação.



4. RESULTADOS

4.2. Disposição das classes no *dataset*

Dada as 6.663 redações coletadas originalmente, com temas diversificados que passaram em um processo de avaliação manual com diferentes avaliadores, a aplicação dos métodos de balanceamento e limpeza de dados, filtrou um segundo *dataset*, dispondo de 690 redações. O Gráfico 7 demonstra a disposição das classes distintas (0.00, 0.50, 1.00, 1.50, 2.00) sobre as cinco competências exigidas.

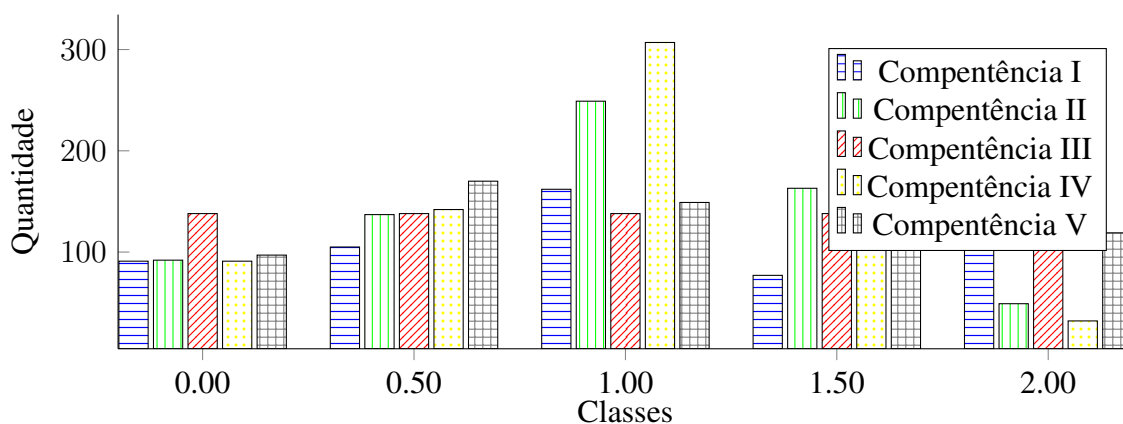


Figura 7. Distribuição das classes sobre a competência III de 690 redações no *dataset* balanceado, cada classe da competência III possui uma amostragem de 138 redações.

4.3. Resultado da inferência indutiva

A inferência indutiva dos classificadores *Adaboost* e *Naive Bayes*, utilizando o *dataset* originou o Gráfico 8, onde está delineado os resultados da *acurácia* de cada classe distinta sobre domínio do problema. Com isso, nota-se que em relação ao algoritmo *Adaboost*, a indução do *Naive Bayes* proveu uma melhor *acurácia* na maior parte das classes.

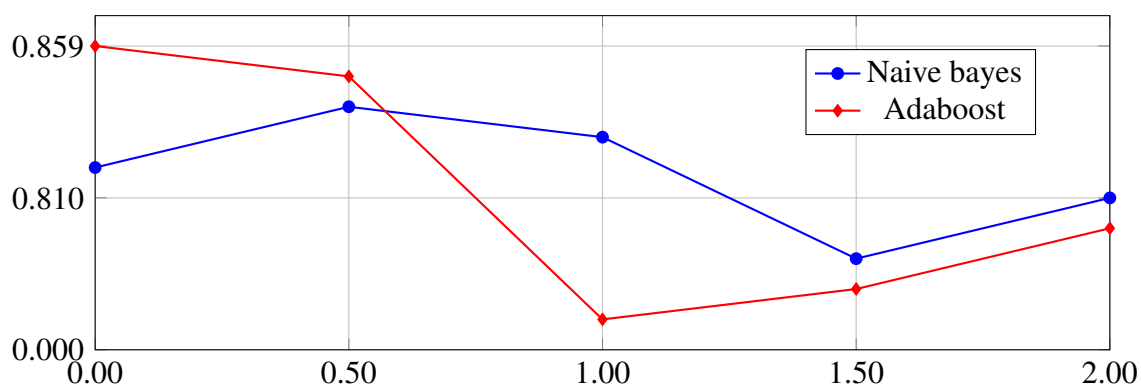


Figura 8. Sobreposição dos resultados de *acurácia* na inferência indutiva dos algoritmos *Adaboost* e *Naive Bayes*.

No Gráfico 9 é apresentado os resultados referentes ao ponto de corte da curva ROC



4. RESULTADOS

correspondente a cada classe distinta. Através deste ponto avalia-se que o poder de discriminação das classes do algoritmo *Naive Bayes* foi superior em relação ao *Adaboost*.

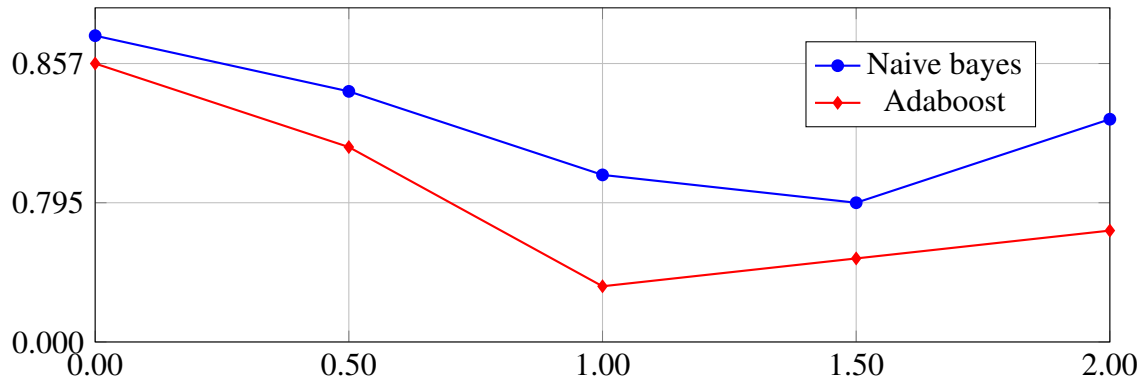


Figura 9. Sobreposição dos resultados da curva ROC na inferência indutiva dos algoritmos *Adaboost* e *Naive Bayes*.

A matriz de confusão ou tabela de contingência é uma ferramenta importante para avaliar os resultados da predição, facilita visualmente o entendimento e reage aos efeitos de predições falsas.

		Naive Bayes					Σ
		0.00	0.50	1.00	1.50	2.00	
Atual	0.00	92	23	9	6	8	138
	0.50	20	83	28	4	3	138
	1.00	24	18	68	19	9	138
	1.50	19	5	12	75	27	138
	2.00	19	6	7	52	54	138
	Σ	172	135	124	156	101	690

Tabela 1. Matriz de confusão resultante da indução do classificador *Naive Bayes*.

		Adaboost					Σ
		0.00	0.50	1.00	1.50	2.00	
Atual	0.00	83	10	27	11	7	138
	0.50	17	74	38	8	1	138
	1.00	10	19	77	19	13	138
	1.50	3	2	21	74	38	138
	2.00	12	3	12	50	61	138
	Σ	125	108	175	162	120	690

Tabela 2. Matriz de confusão resultante da indução do classificador *Adaboost*.

A análise da matriz nas Tabelas 1 e 2 respectivamente dos algoritmos *Naive Bayes*



5. DISCUSSÃO

REFERÊNCIAS

e *Adaboost* foi fundamental para a avaliação dos classificadores. Em ambos classificadores o resultado poderia ser melhor, se caso o padrão encontrado dentro do texto pudessem ser mensurado com maior representatividade obtendo uma melhor separação entre as valorações de cada competência, entretanto, este resultado corrobora com a hipótese proposta para este estudo. De acordo ainda com a análise da matriz de confusão apresentada nas Tabelas 1 e 2, o número de predições corretas do classificador *Naive Bayes* apresentou um resultado melhor em relação ao algoritmo *Adaboost*.

5. Discussão

Como explicado antes, a definição de um melhor algoritmo entre os analisados é inviável, e não faz parte da proposta deste trabalho. Entretanto o classificador *Naive Bayes* apresentou um resultado significativamente maior, no entanto, isto não significa que tal algoritmo seja de fato seja melhor que o *Adaboost*, todavia, atestou a hipótese proposta e demonstrou que ambos os algoritmos, que possuem lógica de predição distintas, quando induzidos, recuperam padrões implícitos no texto da redação. Contudo, as métricas aqui calculadas poderão ser utilizadas para guiar uma escolha de algoritmos para elaboração de trabalhos futuros.

6. Conclusão

Este trabalho teve por objetivo o estudo da recuperação de padrões na valoração textual de redações, através da classificação de textos. Destaca-se que foram realizadas extensas avaliações empíricas sobre os classificadores *Naive Bayes* e *Adaboost*, no decorrer das atividades desenvolvidas para atingir os objetivos propostos, no entanto, por ser um campo de estudo relativamente recente e em contínuo desenvolvimento, acredito que ainda exista um grande espaço para novas descobertas.

Como contribuição, este trabalho demonstra que é possível se beneficiar com os padrões recuperados em textos. A recuperação de padrões implícitos em textos abre precedente a explorar novas soluções na valoração automática dos textos de redação.

7. Trabalhos Futuros

Os conhecimentos obtidos através do desenvolvimento deste artigo, podem ser consideravelmente ampliados através de trabalhos futuros. Como dito anteriormente, a representação extruturada do texto, de uma forma equivalente ao original, foi uma difícil etapa durante a realização deste trabalho. Em trabalhos futuros, pretende-se utilizar “Deep Learning”, para extrair um vetor numérico de características do texto, com o objetivo de mensurar com maior representatividade os padrões encontrados.

Referências

BATISTA, G. et al. Aplicando seleção unilateral em conjuntos de exemplos desbalanceados: Resultados iniciais. In: XIX CONGRESSO NACIONAL DA SOCIEDADE BRASILEIRA DE



REFERÊNCIAS

REFERÊNCIAS

- COMPUTAÇÃO “EDUCAÇÃO E APRENDIZAGEM NA SOCIEDADE DA INFORMAÇÃO. [S.l.: s.n.], 1999. v. 20, p. 327–340. Citado na página 2.
- BOSCARIOLI, C.; VITERBO, J.; TEIXEIRA, M. F. Avaliação de aspectos de usabilidade em ferramentas para mineração de dados. *Anais da I Escola Regional de Sistemas de Informação do Rio de Janeiro*, v. 1, n. 1, p. 107–114, 2014. Citado na página 4.
- BRAGA, B. M. d. A. *Teoria da resposta ao item: o uso do modelo de Samejima como proposta de correção para itens discursivos*. Dissertação (Mestrado) — Universidade de Brasília Instituto de Ciências Exatas Departamento de matemática, 7 2015. Online; acessado 06 Junho 2017. Citado na página 2.
- BRITO, E. M. N. D. Mineração de textos: detecção automática de sentimentos em comentários nas mídias sociais. *Projetos e Dissertações em Sistemas de Informação e Gestão do Conhecimento*, v. 6, n. 1, 2017. Citado na página 4.
- CHAKRABARTI, S. *Mining the Web: Discovering knowledge from hypertext data*. [S.l.]: Elsevier, 2002. Citado na página 4.
- CSF, C. s. F. *Estudante de Graduação*. 2017. Online; acessado 07 Abril 2017. Disponível em: <http://www.cienciasemfronteiras.gov.br/web/csf/estudante>. Citado na página 1.
- DEMŠAR, J. et al. Orange: Data mining toolbox in python. *Journal of Machine Learning Research*, v. 14, p. 2349–2353, 2013. Disponível em: <http://jmlr.org/papers/v14/demsar13a.html>. Citado na página 4.
- ESCOLA, B. *Banco de redações*. 2017. Online; acessado 04 Setembro 2017. Disponível em: <http://vestibular.brasile scola.uol.com.br/banco-de-redacoes/>. Citado na página 5.
- FERREIRA, L. P. C.; DOSCIATTI, M. M.; PARAISO, E. C. Estudo do impacto de um corpus desbalanceado na identificação de emoções em textos. p. 3, 2014. Disponível em: <https://www.researchgate.net/publication/277813458>. Citado na página 5.
- INEP. Edital anual do exame nacional do ensino médio, *EDITAL No 10, DE 14 DE ABRIL DE 2016*. 2016. Online; acessado 05 Junho 2017. Disponível em: http://download.inep.gov.br/educacao_basica/enem/edital/2016/edital_enem_2016.pdf. Citado 3 vezes nas páginas 1, 2 e 7.
- LARA, G. M. P. A redação como tema de pesquisa. In: *Leitura: Teoria e Prática*. [S.l.]: 1994, 1994. v. 13, n. 24, p. 62–82. Citado na página 1.
- MACHADO, E. L. Um estudo de limpeza em base de dados desbalanceada e com sobreposição de classes. p. 19, 2009. Online; acessado 09 Setembro 2017. Disponível em: <http://repositorio.unb.br/handle/10482/1397>. Citado na página 5.
- MATSUBARA, E. T.; MARTINS, C. A.; MONARD, M. C. Pretext: Uma ferramenta para pré-processamento de textos utilizando a abordagem bag-of-words. *Technical Report*, v. 209, 2003. Citado 2 vezes nas páginas 4 e 6.
- MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre aprendizado de máquina. *Sistemas Inteligentes-Fundamentos e Aplicações*, v. 1, n. 1, 2003. Citado na página 2.



REFERÊNCIAS

REFERÊNCIAS

- MOTTA, P. R. d. A. *Estudo Exploratório do Uso de Classificadores para a Predição de Desempenho e Abandono em Universidades*. Dissertação (Mestrado) — Universidade Federal de Goiás Instituto de Informática, 11 2016. Online; acessado 06 Junho 2017. Citado na página 3.
- NEGNEVITSKY, M. *Artificial intelligence: a guide to intelligent systems*. [S.l.]: Pearson Education, 2005. Citado na página 2.
- ORANGE, D. M. *Orange Visual Programming*. 2017. Online; acessado 07 Setembro 2017. Disponível em: <https://docs.orange.biolab.si/3/visual-programming/>. Nenhuma citação no texto.
- PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011. Nenhuma citação no texto.
- REIS, W. A. D. dos. Detecção de sinais de trânsito através do método de classificação adaboost. *UNOPAR Científica Ciências Exatas e Tecnológicas*, v. 12, n. 1, p. 28, 2015. Citado na página 4.
- SCRAPY. *Scrapy*. 2017. Online; acessado 04 Setembro 2017. Disponível em: <https://scrapy.org/>. Citado na página 5.
- SILVA, S. R. da; CARVALHO, T. L. Produção de texto escrito no ensino médio: Competências requeridas pela avaliação de redação do enem em (des)uso no livro didático de português. *Caminhos em linguística aplicada*, 1o sem 2017, v. 16, n. 1, p. 1–25, 2017. Disponível em: <http://periodicos.unitau.br/ojs-2.2/index.php/caminhoslinguistica>. Citado na página 2.
- SISU, S. d. s. u. *O que é o Sisu*. 2017. Online; acessado 07 Abril 2017. Disponível em: <http://sisu.mec.gov.br/>. Citado na página 1.
- TAVARES, L. G.; LOPES, H. S.; LIMA, C. R. E. Estudo comparativo de métodos de aprendizado de máquina na detecção de regiões promotoras de genes de escherichia coli. *Anais do I Simpósio Brasileiro de Inteligência Computacional*, p. 8–11, 2007. Citado na página 6.
- UOL. *Banco de redações*. 2017. Online; acessado 01 Junho 2017. Disponível em: <https://educacao.uol.com.br/bancoderedacoes/>. Citado na página 5.
- WAHBEH, A. H. et al. A comparison study between data mining tools over some classification methods. *International Journal of Advanced Computer Science and Applications*, v. 8, n. 2, p. 18–26, 2011. Citado na página 4.
- WAYMO. *We're building a safer driver for everyone*. 2017. Online; acessado 07 Abril 2017. Disponível em: <https://waymo.com/>. Citado na página 2.