Seleção de Atributos Utilizando Algoritmos Genéticos para Detecção do Vandalismo na Wikipedia

Maria I. M. Sumbana, Allan J. C. Silva,
Marcos A. Gonçalves, Jussara Almeida, Gisele Pappa

Universidade Federal de Minas Gerais
{inesumbana,allan,mgoncalv,jussara,glpappa}@dcc.ufmg.br

Abstract. This paper presents a genetic algorithm based approach for reducing the set of attributes for detecting vandalism in Wikipedia. Our experimental results show that the proposed approach is able to reduce the set of attribute in 83% with no significant impact in detection effectiveness. Moreover, our approach is able to select a number of attributes that leads to better results if compared to a state-of-the-art technique, the Information Gain.

Resumo. Este artigo apresenta uma abordagem baseada em algoritmos genéticos para reduzir o número de atributos utilizados para a detecção automática de vandalismo na Wikipedia. Nossos resultados experimentais mostram que a abordagem proposta é capaz de reduzir o número de atributos em até 83% sem alteração significativa na efetividade da detecção. Além disso, mostramos que a técnica proposta é capaz de selecionar um número reduzido de atributos que produzem uma efetividade na classificação superior ao conjunto selecionado por uma técnica de seleção de atributos considerada estado-da-arte, o Information Gain.

Categories and Subject Descriptors: H.Information Systems [**H.3.4.Systems and Software**]: Information Networks

General Terms: Algoritmos, Experimentação

Keywords: Detecção de vandalismo, classificação, algoritmos genéticos, seleção de atributos, Information Gain

1. INTRODUÇÃO

A proliferação de conteúdo gerado pelo usuário, em especial os artigos escritos colaborativamente em coleções de documentos, tornou-se um fenômeno crescente na Web 2.0. O mais popular desses serviços, Wikia[1], cresceu de cem para vários milhares de coleções em poucos anos, contendo atualmente mais de quatro milhões de páginas de conteúdo rico [Wang and McKeown 2010]. Outro exemplo de como as comunidades podem produzir conteúdo colaborativo em larga escala é a Wikipedia[2]. Na Wikipedia, qualquer um pode editar, modificar ou revisar artigos, dado que os direitos de cópia e de modificação sejam preservados [Belani 2010]. Esta enciclopédia on-line levou apenas dois anos para chegar a mais de dezessete milhões de artigos [Giles 2005], escritos em dezenas de línguas diferentes.

No entanto, nesses tipos de serviços colaborativos, atividades maliciosas como o vandalismo são um dos maiores problemas. Em outras palavras, algumas pessoas tentam explorar os serviços para seu próprio benefício (por exemplo, incluindo publicidade) ou com a intenção de degradar a integridade e a confiabilidade do sistema (por exemplo, incluindo pornografia ou informações incorretas). Na Wikipedia, em particular, o vandalismo é definido como qualquer adição, remoção ou mudança de conteúdo em uma tentativa deliberada de comprometer a integridade do sistema [Wikipedia 2012b]. Exemplos típicos de atos de vandalismo são a adição de obscenidades, informações claramente irrelevantes e "humor negro" em um artigo (e.g., *hey look at me, I just vandalized a page LOL*) [Wikipedia

[1]www.wikia.com
[2]www.wikipedia.com

2012a], remoção ilegítima de páginas, e inserção de conteúdo sem sentido em uma página (por exemplo, *gggfebdgs&%#*88*) [Chin *et al.* 2010]. Porém a detecção de vandalismo é freqüentemente feita de forma manual por voluntários [Wang and McKeown 2010], o que requer muito esforço por parte destes devido ao tamanho atual e à taxa de crescimento da Wikipedia. Além disso, os atos de vandalismo podem ser sutis, o que torna a detecção manual muito difícil.

Assim sendo, existe uma forte demanda por abordagens mais eficazes para detecção do vandalismo, principalmente de maneira automática. Métodos de classificação supervisionados têm sido utilizados para este fim demonstrando muito sucesso [Potthast et al. 2010]. Porém, nesses trabalhos a quantidade e relevância dos atributos utilizados no processo de classificação são raramente estudados com profundidade. Em certos casos, alguns atributos podem ser mesmos detrimentais ao processo de aprendizagem, atrapalhando a eficácia da classificação, ou ainda podem produzir um alto custo de aprendizado. Neste contexto, vê-se a importância de se reduzir o espaço de atributos.

A seleção de atributos é um processo estudado há muitos anos nas áreas de estatística e reconhecimento de padrões [Jain et al. 2000] e posteriormente na área de aprendizado de máquina [Pappa et al. 2002]. A seleção de atributos tem como objetivo descobrir um subconjunto de atributos mais relevantes para uma tarefa alvo, considerando os atributos iniciais, sendo importante por tornar o processo de aprendizagem mais eficiente, entre outras coisas. Segundo [Jain et al. 2000], a redução da dimensionalidade do espaço de atributos pode diminuir o custo de processamento, aumentar a acurácia da classificação, além de diminuir a chance de super-especialização do modelo gerado, que não generaliza para novos conjuntos de dados.

Nesse artigo, buscamos maximizar a eficácia e a eficiência do processo de detecção de vandalismo na Wikipedia. Para ta, baseamo-nos no método de detecção proposto em [Sumbana et al. 2012],, focando aqui em reduzir o numero de atributos usados, mantendo resultados em termos de MacroF1 comparáveis aos obtidos quando todos os atributos originalmente definidos na coleção de da dados adotada (discutida na Seção 4.1) são utilizados. Para reduzir o conjunto de atributos utilizamos algoritmo genéticos (AG), uma técnica que tem se mostrado eficaz na seleção de atributos [Pappa et al. 2002], para selecionar o subconjunto que produz melhores resultados. Em nossos experimentos, conseguimos reduções de até 83% no número de atributos enquanto mantendo a efetividade da detecção em níveis similares aos obtidos com o uso de todos os atributos (ou mesmo superando-os). Além disso, mostramos que a técnica proposta é capaz de selecionar um número reduzido de atributos que produzem uma efetividade na classificação superior ao conjunto selecionado por uma técnica do estado-da-arte de seleção de atributos, o Information Gain [Moore 2012], com o mesmo número de atributos.

O restante desse artigo está organizado da seguinte forma. A Seção 2 apresenta trabalhos relacionados à detecção de vandalismo enquanto a Seção 3 define formalmente o problema de detecção de vandalismo além de descrever o classificador aqui adotado assim como a técnica de algoritmos genéticos. A modelagem do problema utilizando algoritmos genéticos e a configuração experimental são apresentadas nas Seções 4 e 5. A Seção 6 descreve os resultados dos experimentos. Por fim, a Seção 7 lista nossas conclusões e dá diretivas de trabalhos futuros.

2. TRABALHOS RELACIONADOS

Em um dos primeiros estudos em que se abordou o problema de detecção automática de vandalismo, os autores definiram o problema como uma tarefa de classificação binária [Potthast et al. 2008]. Analisando o conteúdo e as categorias das revisões, eles identificaram alguns tipos de vandalismo (conforme reconhecido por seres humanos) e definiram os atributos necessários para identificá-los. Utilizando regressão logística eles foram capazes de classificar novos exemplos de vandalismo com certa acurácia. Mais recentemente, os autores de [Chin *et al.* 2010] definiram e identificaram sete tipos de vandalismo com base em uma taxonomia das revisões da Wikipedia construída a partir de

ações primárias, tais como inserir, remover e modificar. Eles aplicaram *Statistical Language Models* sobre a diferença entre duas revisões consecutivas para construir entradas para o classificador, que por sua vez foi utilizado para detectar os casos de vandalismo. Devido à complexidade do método, os autores trabalharam com revisões de apenas dois artigos da Wikipedia, Microsoft e Lincoln.

A CLEF[3] (*Conference and Labs of the Evaluation Forum*) é uma conferencia anual que tem por foco, entre outros aspectos, a detecção do plagiarismo e a detecção do vandalismo na Wikipedia. Nesta conferência é disponibilizado um conjunto de dados constituído por revisões de artigos da Wikipedia, para uma competição de métodos de detecção de vandalismo nesse sistema. A maioria dos métodos vencedores da competição CLEF 2010[4] usaram variações da árvore de decisão nos seus detectores tais como *random forests, alternating decision trees, naive Bayes decision trees, and C4.5 decision trees* [Potthast et al. 2010]. Por exemplo, o detector vencedor usou *random forests* de 1000 árvores com 5 atributos aleatórios cada. Todos eles usaram o conjunto de dados PAN-WVC-10[5] - *Uncovering Plagiarism, Authorship, and Social Software Misuse - Task 2: Wikipedia Vandalism Detection* - dividido praticamente ao meio, sendo uma metade para treinamento e outra para teste. Os promotores da competição CLEF-2010 [Potthast et al. 2010] também combinaram as previsões dos oito melhores resultados usando *random forests* como um meta-classificador, o que levou a ganhos consideráveis de eficiência de detecção.

Em [Javanmardi *et al.* 2011], os autores descreveram um modelo para a detecção de vandalismo em UGC (*user generated content*). Utilizando como base a coleção PAN-WVC-10, eles extraíram vários novos atributos e os organizaram em quatro grupos, a saber: atributos do usuário, atributos textuais, atributos de metadados e atributos de modelo de linguagem. Em seguida, eles aplicaram a técnica de regularização *Lasso* para reduzir o número de atributos dentro dos grupos, mantendo apenas os mais discriminativos, e utilizaram *random forests* para aprender o modelo de classificação.

O processo de aprendizagem não depende apenas do conjunto de dados selecionado, como também nos atributos que representam este conjunto de dados. Porém, segundo [Kwasnicka and Orski 2004] encontrar um subconjunto ótimo de atributos é uma tarefa bastante complexa, mas que pode ser bem aproximada por algoritmos genéticos. Em [Kwasnicka and Orski 2004], por exemplo, os autores descreveram a utilização de algoritmos genéticos como uma ferramenta de seleção de atributos, utilizando o C4.5 para avaliar os atributos selecionados e redes neurais como classificador. Para os seus experimentos, eles utilizaram 4 coleções de dados. Já em [Pappa et al. 2002], os autores focam foca no uso de algoritmos genéticos para seleção de atributos em tarefas com múltiplos objetivos.

Em [Sumbana et al. 2012], nós desenvolvemos um método de deteccao de vandalismo para a Wikipedia utilizado o LAC – Lazy Associative Classifier [Veloso *et al.* 2006], algoritmo de classificação descrito na Seção 3.1. Naquele trabalho, foram utilizados 67 atributos, obtidos da coleção PAN-WVC-10, utilizada naquele trabalho.

Diferentemente de trabalhos anteriores, inclusive o nosso [Sumbana et al. 2012], o nosso foco no presente trabalho é na redução usando algoritmos genéticos, na quantidade de atributos utilizados, contribuindo assim para a melhor eficiência da detecção sem afetar significativamente a sua eficácia.

3. DETECÇÃO DE VANDALISMO

Assim como em trabalho anteriores [Potthast et al. 2008], nós também abordamos a tarefa de detecção de vandalismo na Wikipedia como um problema de classificação binária. Dada uma revisão e, introduzida por um usuário em um dado artigo, nosso objetivo é detectar automaticamente, se a revisão e é um ato de vandalismo ou não, no último caso referido como regular. Mais formalmente,

[3]http://clef2010.org/
[4]A CLEF 2011 focou na tarefa de detecção de vandalismo em coleções multilinguais.
[5]http://pan.webis.de

podemos definir a tarefa de detecção de vandalismo na Wikipedia da seguinte maneira. Sejam $\mathcal{D}$ a coleção de treinamento e $\mathcal{T}$ a coleção de teste. $\mathcal{D}$ consiste num conjunto de registros com formato $< e, l >$, onde e é uma revisão e l o rótulo que identifica a sua classe, isto é, regular (l=0) ou vandalismo (l=1). Cada revisão e é representada como uma lista de m valores dos atributos $\{f_1, f_2, \ldots, f_m\}$. Os atributos considerados neste trabalho são descritos na Seção 5. A coleção de treinamento $\mathcal{D}$ é usada para aprender um modelo de detecção de vandalismo $\mathcal{M}$, que relaciona os atributos de uma revisão às classes correspondentes. A coleção de teste $\mathcal{T}$ consiste em registros com revisões não rotuladas $< e, ? >$. O modelo de detecção de vandalismo $\mathcal{M}$ é usado para prever a classe de cada revisão no conjunto $\mathcal{T}$.

3.1 Classificador Associativo

Nós adotamos o LAC (Lazy Associative Classifier) [Veloso *et al.* 2006] para classificar cada edição do conjunto de teste $\mathcal{T}$ como vandalismo ou regular, devido aos ótimos resultados produzidos por ele na tarefa de detecção de vandalismo [Sumbana et al. 2012]. O LAC explora o fato de que geralmente há fortes associações entre os valores dos atributos e as classes. Tais associações são utilizadas para prever a classe das edições não rotulados (por exemplo adições em $\mathcal{T}$) e são expressas através de regras da forma $\mathcal{X} \rightarrow k$, indicando a associação entre o conjunto dos valores de atributos $\mathcal{X}$ e a classe k. LAC aprende um modelo $\mathcal{M}$ composto por regras de associação extraídas do conjunto de treinamento $\mathcal{D}$. O processo de aprendizado ocorre em duas fases: a extração de regras sob-demanda e a previsão da classe. A fim de assegurar a eficácia ao extrair regras de $\mathcal{D}$, LAC realiza uma extração sob demanda, isto é, o processo de extração de regras é realizado apenas no momento da classificação. O LAC projeta o espaço de busca das regras de acordo com as informações contidas nas revisões em $\mathcal{T}$ para permitir que a extração de regras seja eficiente. Ele projeta/filtra o conjunto de treinamento de acordo com os valores dos atributos da revisão $e \in \mathcal{T}$, e extrai regras para o conjunto de treinamento projetado, denominado $\mathcal{D}^e$. Isso garante que somente regras que carregam informações sobre a revisão e são extraídas do conjunto de treinamento, limitando drasticamente o número possível de regras.

A confiança de uma regra, denotado por $\theta(\mathcal{X} \rightarrow k)$ mede a força da associação entre $\mathcal{X}$ e k e é estimada pela probabilidade condicional de k ser a classe da revisão e dado que $\mathcal{X} \subseteq e$. O LAC prevê a classe de uma edição $e \in \mathcal{T}$ combinando as confidências de todas as regras úteis $\mathcal{X} \rightarrow k$. Mais especificamente, seja $\mathcal{R}_k^e$ o conjunto de regras que predizem a classe da revisão e como k, extraídas de $\mathcal{D}$. $\mathcal{R}_k^e$ é interpretado como uma enquete, em que cada regra $\mathcal{X} \rightarrow k \in \mathcal{R}_k^e$ é um voto dado pelos atributos em $\mathcal{X}$ para a classe k. O peso do voto $\mathcal{X} \rightarrow k$ depende da força da associação entre $\mathcal{X}$ e k, dada pela confiança $\theta(\mathcal{X} \rightarrow k)$. O processo de estimativa da probabilidade de k ser a classe de e começa pela soma dos votos ponderados para k. Em seguida, calcula-se a confiança média das regras em $\mathcal{R}_k^e$, $s(k, e)$, obtida pela razão entre a soma total dos votos e o número total de votos. A probabilidade estimada de k ser a classe de uma revisão e, denotada por $p(k|e)$, é estimada através da normalização da função $s(k, e)$ pela soma das pontuações obtidas por e para todas classes $p(k|e) = \frac{s(k,e)}{\sum_{j=1}^{n} s(j,e)}$. A classe de e é definida como aquela que tiver maior valor de $p(k|e)$.

3.2 Algoritmos Genéticos

Algoritmos genéticos foram inspirados na teoria da evolução de Darwin, baseada em seleção natural. Segundo essa teoria, em uma população de cromossomos que evoluem a cada geração, apenas os indivíduos mais aptos ao ambiente sobrevivem. Algoritmos genéticos são especialmente atrativos por não exigirem que se saiba como encontrar uma solução ótima para um problema, mas sim como reconhecê-la como ótima [Pappa et al. 2002].

Para a resolução de um dado problema utilizando algoritmos genéticos, alguns passos devem ser seguidos. Em primeiro lugar deve-se gerar, na maioria dos casos de forma aleatória uma população inicial, onde cada indivíduo representa um cromossomo composto por uma seqüência de bits que repre-

sentam uma possível solução para o problema em mãos. Em seguida cada indivíduo é avaliado através da função de aptidão (*fitness*) que determina quão bom um indivíduo candidato é para resolução do problema. A função de *fitness* pode ser calculada de diversos modos, dependendo do problema a ser solucionado. Por exemplo, em uma população natural, a função de *fitness* é determinada pela capacidade do indivíduo de sobreviver a predadores e outros obstáculos naturais, e depois se reproduzir [Pappa et al. 2002]. Através da função de *fitness*, os melhores indivíduos são selecionados para a próxima geração, isto é quanto maior for o valor da função da *fitness*, maior é a chance de um indivíduo sobreviver e se reproduzir. É também necessário determinar quais os operadores genéticos (cruzamento, mutação) serão aplicados sobre as soluções candidatas. A operação de cruzamento (ou *crossover*) consiste em gerar um indivíduo a partir do cruzamento entre os dois melhores indivíduos da população anterior. Neste caso, é definido um ponto de partição de modo que o indivíduo gerado é formado pelos bits anteriores a este ponto do primeiro indivíduo e pelos bits posteriores a este ponto do segundo. A mutação, consiste em inverter de forma aleatória os bits de um indivíduo da população. Esta operação garante a diversidade da população e, além disso, assegura que o indivíduo sempre cobrirá uma parte suficientemente grande do espaço de busca [Pappa et al. 2002]. O processo continua até que uma condição de parada seja verificada, por exemplo um determinado número de gerações seja atingido.

4. SELEÇÃO DE ATRIBUTOS BASEADA EM ALGORITMOS GENÉTICOS

A nossa abordagem utiliza algoritmos genéticos para efetuar a seleção de atributos. Em nossa modelagem, o indivíduo foi definido como uma cadeia de k bits, onde k corresponde ao número de atributos do conjunto de dados ($k = 67$ no nosso caso). Um bit com valor 1 em uma determinada posição i do indivíduo indica a presença do i-ésimo atributo na instância de treino $e \in \mathcal{D}$, e o valor 0 indica a ausência desse atributo em e.

Foram aplicados os operadores de *crossover*, mutação em um ponto e reprodução nas sucessivas gerações. Para cada indivíduo criado é gerado um novo conjunto de treinamento com os respectivos atributos ligados (bit com valor 1), que é submetido ao classificador. Por um processo de validação cruzada de 5 folds, descrito na Seção 5, são calculados os valores de F1 as classes não-vandalismo ($F1_{c0}$) e vandalismo ($F1_{c1}$), assim como de macro F1 (MF1). A *fitness* de um indivíduo é avaliada como: $Fitness = F1_{c1} + \frac{MF1}{|c|}$, onde $|c|$ é o número dos atributos utilizados pelo classificador. A intuição dessa fórmula é que um bom classificador é aquele que consegue boa eficácia na detecção de vandalismo (instâncias da classe c_1) conforme explicitado na primeira parcela da fórmula ($F1_{c1}$) e, ao mesmo tempo, consegue ser eficaz em detectar instâncias de não-vandalismo (elementos da classe $c0$), entretanto com um número reduzido de atributos, conforme expresso na segunda parcela da soma ($\frac{MF1}{|c|}$).

A partir dos dados de treino e dos indivíduos gerados são derivadas novas bases de treinamento que são submetidas ao processo de treinamento e teste do classificador. As métricas obtidas são utilizadas para cálculo das *fitness* dos indivíduos, conforme descrito anteriormente.

Pelo fato do LAC ser um classificador sob demanda, que incorre num custo adicional de geração de um modelo específico para cada instância de teste, além de bastante sensível à questão do desbalanceamento dos dados [Sumbana et al. 2012], utilizamos o J48, uma implementação da técnica baseada em árvores de decisão C4.5, disponível no Weka [6], para selecionar o subconjunto dos atributos que produz melhor resultado nessa fase. O J48 apresentou ótimo desempenho nesta fase superando outras alternativas testadas tais como kNN e SVM.

[6] http:// www.cs.waikato.ac.nz/ml/weka/

5. CONFIGURAÇÃO EXPERIMENTAL

Para a tarefa de detecção do vandalismo na Wikipedia, utilizamos o conjunto de dados do PAN-WVC-10 que compreende 32.452 edições em Inglês de 28.468 artigos diferentes, das quais 2.391 edições são de vandalismo [Potthast 2010], conforme detecção manual. Cada edição representa a diferença entre duas revisões consecutivas de um artigo, e a respectiva classe indica se a edição é vandalismo ou não. Como pode-se notar, este conjunto de dados é muito desbalanceado, com 92,7% dos casos pertencentes à classe de não vandalismo e apenas 7,3% sendo casos de vandalismo (a classe positiva). Conforme procedimento descrito em [Javanmardi *et al.* 2011], foram derivados 67 atributos para cada instância, que podem ser agrupados em quatro categorias:

—**Atributos do usuário:** 12 atributos, obtidos através da mineração do histórico das revisões até uma data específica (18-11-2009).

—**Atributos de texto:** 30 atributos, calculados a partir do conteúdo do texto inserido ou apagado dos artigos.

—**Atributos de Metadados:** 22 atributos, extraídos dos comentários associados às revisões.

—**Atributos do Modelo de Linguagem:**: 3 atributos, derivados através do cálculo da distância de Kullback-Leibler (KLD) entre: a aplicação de dois modelos de linguagem unigrama entre a revisão anterior e a atual; o conteúdo inserido e a revisão anterior; e o conteúdo apagado e a revisão anterior.

A lista completa de atributos está em http://dl.dropbox.com/u/20663184/sbbd2012/listaatributos.pdf

Os experimentos realizados tiveram como objetivo analisar os resultados do classificador LAC na detecção do vandalismo na Wikipedia, utilizando os atributos selecionados pelo Algoritmo Genético (AG), comparando-os com os resultados do LAC utilizando os atributos selecionados pela técnica estado-da-arte Information Gain (IG). Todos os experimentos de classificação foram realizados utilizando validação cruzada com 5-*folds*. Ou seja, a amostra original foi particionada em 5 sub-amostras, das quais quatro foram utilizadas como dados de treinamento, e uma foi usada para testar o classificador. O processo foi então repetido 5 vezes, com cada uma das 5 sub-amostras sendo usada como teste, produzindo assim 5 resultados. Os resultados apresentados na Seção 6 são portanto médias dessas 5 execuções com respectivos intervalos de confiança de 95%.

6. RESULTADOS EXPERIMENTAIS

Nesta seção apresentamos os resultados mais relevantes da classificação com todos os atributos e com os atributos selecionados pelo algoritmo genético (AG), comparando estes resultados com os obtidos pelos atributos selecionados pelo IG. Os resultados são relatados em termos de precisão, revocação e Macro-F_1. Devido à sensibilidade do classificador escolhido ao desbalanceamento do dados [Sumbana et al. 2012], os resultados apresentados neste trabalho foram realizados aplicando uma técnica de balanceamento (*undersampling*) sobre os dados de treinamento, descrita a seguir.

6.1 Balanceamento dos Dados de Treinamento

A técnica de *undersampling* consiste em reduzir aleatoriamente O número de instâncias da classe maior no conjunto de treinamento, visando equilibrar, ainda que aproximadamente, as duas classes. Assim, no presente contexto, nós reduzimos o número de instâncias da classe 0, a classe negativa que representa edições regulares. A redução é feita definindo uma proporção alvo p do número de instâncias da classe 0 sobre o número de instâncias da classe 1. Sejam C_0 e C_1 os números de instâncias das classes 0 e 1 respectivamente que existem no conjunto de treinamento original. Dado uma proporção alvo p, eliminamos, de forma aleatória, instâncias negativas do conjunto de treinamento até restar apenas $p \times C_1$ instâncias dessa classe.

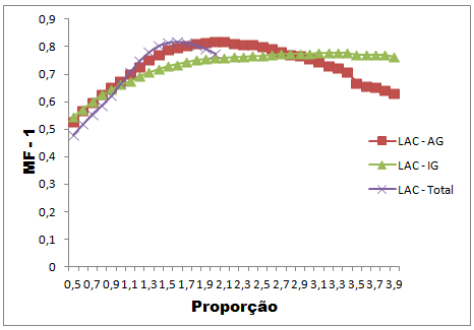

Fig. 1. Resultados da Macro-F_1 para o LAC com 67 atributos e com 11 atributos selecionados pelo AG e pelo IG em função da proporção p de exemplos negativos sobre os exemplos positivos no conjunto de treinamento

Table I. Resultados do LAC, AG E IG com proporções p de 1.5:1, 1.9:1 e 2.8:1 respectivamente

Algoritmo de Classificação	Número de Atributos	Regulares		Vandalismo		Macro $F-1$
		Precisão	Revocação	Precisão	Revocação	
LAC-Total	67(54)	97.9±0.0023	95.5±0.0064	57.3±0.0161	75.1±0.0265	80.9±0.0065
LAC-AG	11	97.82±0.0062	96.18±0.0025	60.04±0.0173	72.9±0.0602	81.42±0.0163
LAC-IG	11	97.39±0,0073	95.26±0,001	52.9±0,026	67.58±0.0887	77.42±0,015

Executamos vários experimentos, para determinar a melhor proporção de balanceamento tomando uma porção do conjunto de treinamento (20%), como conjunto de validação e variando os valores de p entre 0.5 e 4. A Figura 1 mostra os valores de Macro-F_1 obtidos em função de p, para o LAC utilizando todos os atributos (LAC-Total) e utilizando os atributos obtidos com cada método de seleção de atributos. Note que as 3 abordagens atingiram o valor máximo de Macro-F1 em proporções diferentes. Utilizando todos os atributos, o LAC produziu os melhores resultados para p entre 1.5 e 1.6. Utilizando os atributos selecionados pelo método proposto (baseado em Algoritmos Genéticos) o melhor resultado foi obtido para p entre 1.9 e 2.1. Já utilizando o Information Gain para seleção de atributos, o melhor valor de p está entre 2.8 e 3.3.

6.2 Resultados de Detecção

A Tabela I mostra os resultados obtidos com o LAC utilizando todos os atributos e os atributos selecionados pelos métodos AG e IG, aplicando a técnica de *undersampling* com valores de p escolhidos dentre os que levaram aos melhores resultados para cada método. Ou seja, utilizou p igual a 1.5, 1.9 e 3.3 para o LAC com todos os atributos, com os atributos selecionados pelo AG e por IG, respectivamente. A seleção de atributos usando AG foi feita a partir da geração de uma população inicial onde cada indivíduo possui exatamente 54 atributos, obtidos após o processo da discretização, necessário ao uso do LAC[7]. Nas gerações posteriores, conforme a aplicação dos operadores genéticos, o número de atributos associado a cada indivíduo pode variar. Foram executadas 200 iterações com elitismo[8], passando sempre o melhor indivíduo de uma geração para outra. Como podemos ver ao fim das 200 iterações a nossa abordagem foi capaz de reduzir o número de atributos para 11. No caso do IG foi selecionada a mesma quantidade de atributos, selecionados pelo AG em cada iteração, de acordo com o poder discriminativo.

Podemos notar que o resultado da nossa abordagem supera os outros dois método, com uma Macro-F_1 de 81,4%, sendo capaz de detectar corretamente mais de 60% de instâncias de vandalismo, embora com uma pequena queda na revocação se comparado aos resultados quando todos os atributos são

[7]O processo de discretização utilizado pode remover atributos altamente correlacionados a outros atributos. Ele foi aplicado aos 3 métodos.
[8]Elitismo consiste em garantir a presença do melhor indivíduo na próxima geração.

usados. De fato, o resultado de Macro-F1 obtidos com os 11 atributos selecionados pela AG é até mesmo levemente superior ao uso de todos os atributos em termos de Macro-F1. Isto vem ainda com uma redução significativa do número de atributos necessários para a detectar atos de vandalismo na Wikipedia. Comparado com o IG, o resultado obtido com a mesma quantidade de atributos selecionados pelo AG superam em 4% em termos de Macro-F1. Apesar de não apresentarmos esses resultados por questões de espaço, testamos o AG contra o IG com todos os possíveis tamanhos de rankings de atributos gerados, e o AG foi sempre superior em todos os casos.

7. CONCLUSÃO E TRABALHOS FUTUROS

Neste artigo foi proposta uma abordagem baseada em algoritmos genéticos para a redução do número de atributos utilizados na detecção automática de vandalismo na Wikipedia e, conseqüentemente, redução do custo. Nossos resultados mostraram que a nossa abordagem, associada a uma técnica simples de balanceamento dos dados de treinamento, superou em termos de eficácia tanto a técnica tradicional de seleção de atributos *Information Gain* como a detecção utilizando todos os atributos disponíveis. Isso viabiliza a construção de classificadores com modelos compactos e eficazes para o problema de detecção de vandalismo na Wikipedia. Como trabalho futuro, planeja-se usar algoritmos genéticos para selecionar instâncias relevantes para o conjunto de treinamento.

REFERENCES

Belani, A. Vandalism detection in wikipedia: a bag-of-words classifier approach. *CoRR* vol. abs/1001.0700, 2010.

Chin *et al.*, S.-C. Detecting wikipedia vandalism with active learning and statistical language models. In *WICOW '10*. pp. 3–10, 2010.

Giles, J. Internet encyclopaedias go head to head. *Nature* 438 (7070): 901–902, 2005.

Jain, A. K., Duin, R. P. W., and Mao, J. Statistical pattern recognition: A review. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* 22 (1): 4–37, 2000.

Javanmardi *et al.*, S. Vandalism detection in wikipedia: a high-performing, feature-rich model and its reduction through lasso. In *WikiSym '11*. pp. 82–90, 2011.

Kwasnicka, H. and Orski, P. Genetic algorithm as an attributes selection tool for learning algorithms. In *Intelligent Information Systems'04*. pp. 449–453, 2004.

Moore, A. W. Information gain. `http://ftp.utcluj.ro/pub/users/nedevschi/AV/12_FeatureSelectionPerformanceEvaluation/`, 2012.

Pappa, G. L., Freitas, A. A., and Kaestner, C. A. A. Attribute selection with a multi-objective genetic algorithm. In *Proceedings of the 16th Brazilian Symposium on Artificial Intelligence: Advances in Artificial Intelligence*, 2002.

Potthast, M. Crowdsourcing a wikipedia vandalism corpus. In *SIGIR '10*. pp. 789–790, 2010.

Potthast, M., Stein, B., and Gerling, R. Automatic vandalism detection in wikipedia. In *Proceedings of the IR research, 30th European conference on Advances in information retrieval*. pp. 663–668, 2008.

Potthast, M., Stein, B., and Holfeld, T. Overview of the 1st International Competition on Wikipedia Vandalism Detection. In *CLEF (Notebook Papers/LABs/Workshops)*, M. Braschler, D. Harman, and E. Pianta (Eds.), 2010.

Sumbana, M., Marcos, G., Almeida, J., Silva, R., and Veloso, A. Automatic vandalism detection in wikipedia with active associative classification. In *TPDL'12*, 2012.

Veloso *et al.*, A. Lazy associative classification. In *ICDM '06*. pp. 645–654, 2006.

Wang, W. Y. and McKeown, K. "Got You!": Automatic Vandalism Detection in Wikipedia with Web-based Shallow Syntactic-Semantic Modeling. In *COLING'10*. Tsinghua University Press, 2010.

Wikipedia. The motivation of a vandal. `http://en.wikipedia.org/wiki/Wikipedia:The_motivation_of_a_vandal`, 2012a.

Wikipedia. Vandalism on wikipedia, 2012b.