Aplicando Programação Genética na Geração de Classificadores de Sentimento

Airton Bordin-Junior[1], Celso Camilo-Junior[1], Nadia Felix[1], and Thierson Rosa[1]

[1]Universidade Federal de Goiás, Goiânia GO 74690-900
I.airtonbjunior@gmail.com, II.celsocamilo@gmail.com

Abstract. A WEB é comumente utilizada como plataforma para debates, opiniões, avaliações e etc. Esses dados permitiram que algumas áreas, como a Análise de Sentimento (AS), se desenvolvessem para extrair informação e conhecimento que possam ser utilizados em diferentes aplicações. Entre os desafios da AS podemos destacar a criação de classificadores com boa eficácia. Normalmente, os modelos gerados pelas técnicas não supervisionadas são heurísticas específicas, manualmente definidas e pouco adaptáveis a diferentes contextos. Assim, o presente trabalho propõe a utilização da Programação Genética (PG) para a geração automatizada de modelos de classificação de sentimento baseados em léxicos. Com isso, espera-se reduzir o custo de geração dos classificadores e aumentar a eficácia para cada domínio analisado. Para validar a proposta foi utilizado o *benchmark* SemEval 2014. Os resultados mostram que a abordagem de geração automatizada com a PG é promissora, pois os modelos gerados superam o *baseline* e são competitivos com outros trabalhos. Por fim, destaca-se a capacidade da proposta de customização dos modelos léxicos de acordo com o contexto abordado e a possibilidade de transferência de conhecimento dos usuários por meio das funções utilizadas pela PG.

Keywords: Análise de Sentimentos, Mineração de Opiniões, Programação Genética, Classificadores

1 Introdução

A Análise de Sentimentos (AS) é uma linha de pesquisa que tem por objetivo a classificação das emoções de um determinado texto, geralmente como positivo, negativo ou neutro [17, 22]. A área vem ganhando destaque nos últimos anos, principalmente por conta da popularização do acesso à Internet e do consequente aumento na quantidade de conteúdo produzido na rede. O uso das redes sociais, como *Twitter* [1] e Facebook[2] , e a forma como as pessoas compartilham suas opiniões e sentimentos sobre os mais diversos assuntos, tem motivado a pesquisa de classificadores para esses conteúdos.

[1] https://twitter.com/
[2] https://www.facebook.com/

É possível dividir as abordagens de classificação de sentimentos em duas classes principais [8, 26, 21]: técnicas supervisionadas e não supervisionadas. A primeira delas utiliza abordagens de aprendizado de máquina para a classificação das opiniões, realizando o treinamento com mensagens previamente classificadas. As abordagens não supervisionadas são heurísticas criadas manualmente, baseadas em aspectos estruturais do texto e, frequentemente, fazem uso de léxicos.

Para que um classificador tenha resultados generalizáveis, deve levar em consideração aspectos inerentes ao contexto das opiniões que serão avaliadas. Um modelo de classificação de *Tweets*, por exemplo, geralmente é diferente de um processo de classificação de avaliações de produtos ou comentários políticos. Em *Tweets*, por conta da limitação do tamanho das mensagens em 140 caracteres, o uso de abreviações é muito comum. Além disso, por se tratar de uma plataforma de rede social, os textos são frequentemente escritos de maneira informal, comumente fazendo uso de gírias [6].

Algumas abordagens [2, 13, 8, 12] geram o classificador de forma manual pois, assim, capturam o conhecimento prévio e a experiência do projetista sobre o domínio a ser analisado. No entanto, isso aumenta o custo de geração para cada domínio, além de prejudicar a generalização.

Com o intuito de automatizar essa tarefa, podemos formular o desafio de geração de um classificador de sentimento como um problema de busca e otimização, pois tem como objetivo encontrar um modelo, dentro do espaço de modelos possíveis, que maximize a acurácia da classificação.

Entre os métodos existentes utilizados para geração automatizada de modelos, a Programação Genética (PG), explanada em detalhes no seção 2.2, é uma das mais adequadas. A PG é uma abordagem da área da computação evolucionária que objetiva a criação automatizada de modelos baseados na função objetivo, que representa o problema.

O presente trabalho, portanto, propõe o uso da PG para a geração automatizada de modelos baseados em léxico para a classificação de sentimento.

Com isso, as principais contribuições são:

- Uma abordagem de geração automatizada de modelos léxicos de classificação de sentimento baseado em PG;
- Análise de desempenho da proposta automatizada em relação ao *baseline* e outros trabalhos da literatura.

Este trabalho está organizado da seguinte maneira: Inicialmente, conceitos essenciais para o entendimento do problema de pesquisa são apresentados na Seção 2. Na sequência, trabalhos relacionados são discutidos na Seção 3. A proposta deste trabalho é apresentada na Seção 4. Os experimentos são apresentados na Seção 5 e os resultados discutidos na Seção 6. Por fim, a Seção 7 discorre sobre as conclusões e trabalhos futuros.

2 Conceitos

Nesta seção serão apresentados, de forma sucinta, conceitos fundamentais para o entendimento do trabalho. Pontos principais dos temas serão discutidos, com foco nos conteúdos relevantes para a solução do problema de pesquisa.

2.1 Análise de Sentimentos

A Análise de Sentimentos, também chamada de Análise de Opiniões ou Mineração de Opiniões, é uma linha de pesquisa abrangente e que vem sendo tema de diversos trabalhos nos últimos anos. Como observado em [16], esse crescente interesse sobre o assunto ocorre principalmente devido ao aumento no número de usuários da Internet e o consequente crescimento da produção de conteúdo independente na rede, como opiniões, avaliações, entre outros.

Essa área de estudo tem como principal desafio a Análise de Opiniões, descritas em linguagem natural, para a identificação da polaridade implícita ou explícita no texto. Essa polaridade é, na maior parte das vezes, identificada como uma escala de pontuação de sua característica positiva, negativa ou neutra.

Quanto aos classificadores de sentimentos, podemos dividi-los em duas abordagens principais: [8, 26, 21]: supervisionada e não supervisionada. Na primeira, técnicas de aprendizado de máquina são aplicados à mensagens previamente rotuladas de forma a identificar características que auxiliem na distinção e detecção de sentimentos nas sentenças desconhecidas. Dentre as principais técnicas de aprendizado de máquina para a classificação de sentimentos, podemos citar o *Support Vector Machines* (SVM) [9], *Naïve Bayes* [11], *Adaboost* [7], Redes Neurais Artificiais, entre outros [23].

Técnicas não supervisionadas atuam principalmente em características sintáticas e semânticas do texto e, geralmente, baseiam-se em dicionários léxicos - conjunto de palavras e suas polaridades (grau de positividade e negatividade de uma mensagem). À partir desse dicionário, é feito o processamento das mensagens pelo classificador e retornada a polaridade das mesmas [1].

2.2 Programação Genética

Programação Genética (PG) é um campo da computação evolucionária que busca resolver problemas, de forma automatizada, sem demandar conhecimentos detalhados sobre a solução [14]. De forma geral, podemos definir a PG como um método sistemático, não dependente de um domínio específico, usado para permitir que computadores criem programas para solução de problemas de forma automática, inciando com um conhecimento de alto nível sobre as regras gerais dos possíveis modelos.

Nesse contexto, programa significa um modelo capaz de, à partir de uma ou mais entradas, produzir uma saída para as mesmas. Embora possam ser representadas por diversos tipos de estruturas, a forma mais comum é a representação

por meio de árvores, onde os nós internos representam funções e os nós folha representam terminais do problema. Um exemplo de programa é o código *if(X > Y) then { X * 6 + 1.9 } else { X / cos(X) }*

Na PG, assim como em outros algoritmos baseados na evolução humana, são criadas populações onde cada indivíduo representa uma possível solução para o problema. A inicialização aleatória é a forma mais comum de criação da população, evoluindo as mesmas no decorrer dos ciclos, chamados de gerações. A cada geração, indivíduos possivelmente melhores são criados, evoluindo os programas (modelos) gerados. Assim como a natureza, a PG é um processo estocástico, e não garante o resultado ótimo. Porém, essa aleatoriedade faz com que, frequentemente, as soluções fujam de problemas frequentemente enfrentados por métodos determinísticos gulosos, como máximos e mínimos locais [18].

3 Trabalhos relacionados

A quantidade de trabalhos na área de Análise de Sentimentos vem crescendo a cada ano, motivado, principalmente, pela importância da área no contexto atual de análise de grande quantidade de dados e informações.

Ao debatermos os trabalhos relacionados à área de AS, é importante iniciarmos citando [16]. Os autores conceitualizam o problema e propõem uma forma estruturada de organização dos dados não estruturados, característica intrínseca dos textos em linguagem natural, objeto de entrada da pesquisa.

A definição de opinião como uma quíntupla (entidade, aspecto da entidade, sentimento, autor e tempo) é utilizada em grande parte dos trabalhos na área, caracterizando-se, portanto, como elemento fundamental nas pesquisas sobre o assunto [16]. Visão geral sobre o tema e principais desafios e técnicas são vistos também em [19, 5, 8, 24, 3, 12].

Algumas abordagens para a AS são supervisionadas, com uma fase de treinamento geralmente utilizando textos de um contexto específico. Somente alguns desses trabalhos levam em consideração o léxico para gerar o modelo. A inclusão de aspectos sintáticos e semânticos (dicionários, valor semântico) em heurísticas manualmente geradas é comum na literatura. No entanto, a geração automatizada de modelos com o uso do léxico é menos frequente [8, 21, 1].

Entre as abordagens supervisionadas, as técnicas comumente utilizada são SVM [20], *Naïve Bayes* [11], *Adaboost* [7], Redes Neurais Artificiais, entre outros [23].

Considerando a importância da qualidade do léxico para a classificação, alguns trabalhos [25] apresentam uma abordagem de expansão léxica fazendo uso de *Pointwise Mutual Infomation* (PMI), com o objetivo de calcular a coocorrência para definir a polaridade de novas entradas. Nesse trabalho, amplamente referenciado por outras pesquisas, o autor compara o conjunto de palavras de polaridade desconhecida com as palavras *"excellent"* e *"poor"*, representando sentimentos positivo e negativo, respectivamente. Essas palavras previamente conhecidas utilizadas para a expansão do dicionário são chamadas de palavras semente (*seed words*). O trabalho obteve uma acurácia de 66% na análise de avaliações de

filmes. Apesar de usar o léxico ampliado, o modelo de classificação é simples e manualmente gerado.

Destaca-se, como apresentado em [1], que somente a melhoria do dicionário léxico não é suficiente para uma eficaz classificação dos sentimentos, já que alguns trabalhos aplicam apenas a soma das polaridades das palavras para definir o sentimento do texto. Outros trabalhos possuem heurísticas mais elaboradas, geralmente manualmente desenvolvidas, que trabalham em conjunto com os dicionários.

Essas estratégias heurísticas levam em consideração aspectos gramaticais e sintáticos que possuem uma importância na expressão do sentimento, como pontuação, negação, intensificação, capitalização, entre outros. Em contextos específicos, como a classificação de *Tweets*, pode-se levar em consideração a quantidade de *hashtags*, gírias, etc.

Especificamente sobre a geração de modelos de classificação, tema central deste trabalho, o trabalho [20] demonstra a utilização de SVM para a construção de um modelo de classificação de *Tweets* utilizando o *benchmark* SemEval 2014 (*International Workshop on Semantic Evaluation*). Para o treinamento, os autores levaram em consideração uma série de características das mensagens, como pontuação, presença de *emoticons*, presença de *hashtags*, palavras alongadas (como "simmmm"), entre outras. Além disso, em todos os *Tweets*, as URLs e menções a outros usuários passaram por um processo de normalização. O trabalho fez uso de 5 dicionários léxicos e obteve um F1 *score* de 0.6902 na classificação das mensagens.

Também fazendo uso de SVM, o trabalho [15] utiliza 3 APIs públicas de análise de sentimentos para apoiar o processo de classificação - Sentiment140[3], SentimentAnalyzer[4] e SentiStrength[5], além de 6 dicionários léxicos. Para o treinamento do modelo, fez uso do *benchmark* SemEval2014. Quanto às *features* das mensagens, o autor leva em consideração o tamanho das palavras, utilização de asteriscos e hífens, *emoticons*, *hashtags*, entre outros. Além disso, o trabalho utiliza as bibliotecas *CMU ARK Twitter NLP Tool*[6] e *Stanford CoreNLP*[7] para processar o PoS (*Part of Speech*) das mensagens. O trabalho obteve um F1 *score* de 0.7446 em uma das bases do *benchmark*, sendo seu melhor resultado.

Em [27] os autores apresentam uma abordagem de um classificador linear treinado utilizando *Stochastic Gradient Descent* (SGD). Além do pré-processamento das entradas, faz uso de características das mensagens, como a soma acumulada de polaridades positivas e negativas, usando o dicionário SentiWordNet[8] e o stem da frase. Além disso, o classificador trabalha com 3 variantes de cada palavra: a palavra original, uma versão normalizada com todas as letras minúsculas e todos

[3] http://www.sentiment140.com/
[4] http://sentimentanalyzer.appspot.com/
[5] http://sentistrength.wlv.ac.uk/
[6] http://www.cs.cmu.edu/ ark/TweetNLP/
[7] https://stanfordnlp.github.io/CoreNLP/
[8] http://sentiwordnet.isti.cnr.it/

os números convertidos para 0 e uma versão com letras repetidas suprimidas. Obteve um F1 *score* de 0.6554 no *benchmark* SemEval 2013.

O principal diferencial do presente trabalho em relação aos artigos supracitados é a capacidade de gerar automaticamente um modelo léxico de AS baseado na Programação Genética. Vale destacar que essa abordagem permite uma customização ou generalização - dependendo da base de treinamento e as funções utilizadas - e também um entendimento de como o modelo atribui classe para as mensagens.

4 Abordagem

O desafio de gerar o classificador de sentimentos pode ser descrito como um problema de otimização, com o objetivo de encontrar um modelo que represente a solução desejada.

Como explanado na seção 2.2, a Programação Genética pode ser utilizada para a criação de um modelo de solução - um programa - para a resolução de um dado problema. De posse de um conjunto de dados previamente classificados, podemos evoluir nossa população de possíveis soluções (indivíduos), avaliando seu *fitness* de acordo com a semelhança com o resultado esperado para determinada entrada. Espera-se, ao final, que o indivíduo mais apto (de melhor *fitness*) retornado pelo algoritmo seja um modelo de classificação de sentimento eficaz para o contexto abordado.

O primeiro passo para projetar uma solução de PG é determinar o conjunto de terminais e funções do modelo. Os terminais serão compostos pelas mensagens sob avaliação, por exemplo *Tweets*, bem como por uma constante efêmera, um número real escolhido aleatoriamente entre -3 e 3.

Já as funções serão responsáveis por realizar a manipulação das mensagem sob avaliação. A lista das principais funções definidas para a solução pode ser vista na Tabela 1. Para a criação dessas funções (Tabela 1), foram levadas em consideração heurísticas apresentadas em trabalhos anteriores sobre o tema em determinados contextos, como os apresentados em [1, 23, 25]. Desta forma, a PG pode combiná-las de acordo com o contexto abordado. Destaca-se que o uso de funções pode potencializar a transferência de conhecimento dos usuários para o processo, pois podem definir funcionalidades específicas de um domínio para a PG considerar na busca do modelo.

Table 1: Principais funções utilizadas na Programação Genética

Função	Retorno
polaritySum(str): float	Soma das polaridades de cada palavra
hashtagPolaritySum(str): float	Soma das polaridades de cada hashtag
emoticonPolaritySum(str): float	Soma das polaridades de cada emoticon
positiveWords(str): float	Quantidade de palavras positivas
negativeWords(str): float	Quantidade de palavras negativas
hasHashtags(str): bool	Verifica se o *Mensagem* possui hashtag
hasEmoticons(str): bool	Verifica se o *Mensagem* possui emoticon
removeStopWords(str): str	Remove os *stopwords* do *Mensagem*

Além das funções citadas na Tabela 1, também foram incluídas funções matemáticas como adição (add), subtração (sub), divisão (div), multiplicação (mul), logaritmo (log), raiz quadrada (sqrt), exponenciação (exp), seno (sin) e cosseno (cos).

Outras métricas são utilizadas para a verificação da qualidade do classificador: acurácia (soma das classificações corretas pelo total de mensagens), precisão (número de instâncias de uma classe avaliadas corretamente dividido pelo total de mensagens avaliadas nessa classe) e *recall* (número de mensagens de uma classe avaliadas corretamente dividido pela quantidade de mensagens pertencentes àquela classe).

5 Experimentos

Para validar a proposta, os experimentos seguem o fluxo mostrado na Figura 1.

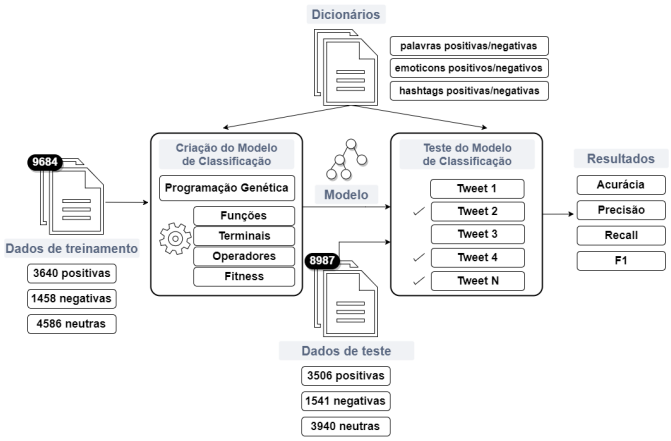


Fig. 1: Diagrama simplificado da experimentação

Para apoio no desenvolvimento da PG foi utilizada a biblioteca DEAP[9] (*Distributed Evolutionary Algorithms in Python*), escrita na linguagem *Python* e disponível para uso gratuito. Fornece abstrações para a implementação de várias classes de algoritmos evolucionários, como Algoritmos Genéticos, Programação Genética, entre outros [4].

Especificamente para o contexto de Programação Genética, DEAP fornece funcionalidades para controle de criação das estruturas de árvores, operadores genéticos, parametrização das operações, *logs*, entre outras. Para a stemização - processo de redução de palavras flexionadas para sua forma raiz - foi utilizada a biblioteca stemming 1.0[10] do *python*. Para a criação de uma lista de *stopwords*

[9] https://github.com/DEAP/deap
[10] https://pypi.python.org/pypi/stemming/1.0

- palavras que podem ser consideradas irrelevantes para a análise do texto - foi utilizada a biblioteca nltk[11].

5.1 Base de dados

Para o presente trabalho, foi utilizada a base SemEvaL 2014[12] (*International Workshop on Semantic Evaluation*), uma das principais competições na área de Análise de Sentimentos em mensagens web.

O evento é dividido por tarefas (*Tasks*), que possuem objetivos distintos dentro da área de pesquisa. Para este trabalho, utilizou-se a base de dados da *Task 9 - Sentiment Analysis in Twitter*. São disponibilizadas bases de treinamento e de testes para *download*[13] no site do evento. A base de treinamento aplicada no trabalho possui 9684 mensagens, com a seguinte divisão de polaridades: 3640 mensagens positivas, 1458 negativas e 4586 neutras.

O evento também disponibiliza uma base de teste com 8987 mensagens, que serve como critério de avaliação e comparação dos trabalhos submetidos para cada *Task*. A base fornecida é dividida em 5 sub-bases: *Tweets2013* (3813 mensagens), *Tweets2014* (1853 mensagens), *Tweets2014Sarcasm* (86 mensagens), *SMS2013* (2093 mensagens) e *LiveJournal2014* (1142 mensagens)

Com isso, o objetivo do experimento é gerar modelos de classificação generalistas, ou seja, que apresentem bons resultados nas diferentes bases.

5.2 Dicionários

Foram utilizados os dicionários de palavras positivas e negativas de [10][14]. Os léxicos fornecem um conjunto de 4783 palavras negativas e 2006 palavras positivas para apoiar no processo de Análise de Sentimentos.

Utilizou-se, também, o dicionário de *emoticons* SentiStrength[15], que fornece 46 *emoticons* positivos e 58 negativos. A escolha desses dicionários deu-se, principalmente, por terem sido utilizados como base para o SemEval 2014, *Task* 9.

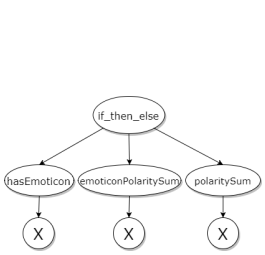
5.3 Modelos gerados

O desenho e parametrização são partes importantes da PG. Considerando o objetivo de criar modelos que melhorem a eficácia da classificação, a *fitness* da PG é o F1 médio das mensagens positivas e negativas (métrica utilizada no SemEval 2014). O cruzamento é feito utilizando o operador *one-point crossover*, e, para a mutação, o operador *uniform mutation*. Além disso, foi usada uma estratégia de sobrevivência elitista.

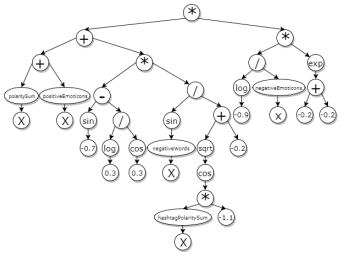
[11] http://www.nltk.org/
[12] http://alt.qcri.org/semeval2014/
[13] http://alt.qcri.org/semeval2014/task9/index.php?id=data-and-tools
[14] https://www.cs.uic.edu/ liub/FBS/sentiment-analysis.html#lexicon
[15] http://sentistrength.wlv.ac.uk/

Diferentes configurações de parâmetros da PG foram aplicadas para a geração de três modelos distintos. Optou-se por alterar os parâmetros que apresentaram maior sensibilidade, identificados em testes anteriores. Para o modelo A, foi configurada uma população de 50 indivíduos, taxa de *crossover* de 35% e de mutação de 15%, processando 500 gerações. No modelo B, foi mantido o valor de população anterior, e foi incrementada a taxa de *crossover* para 95% e de mutação para 35%, com 600 gerações de processamento. Por fim, no modelo C, a população foi configurada em 100 indivíduos e 650 gerações de processamento, com as taxas de *crossover* e mutação em 45% e 25%, respectivamente.

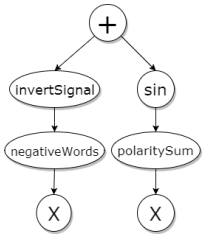
Esses modelos são apresentados nas Figuras 2a, 2b e 2c (a variável x representa a mensagem de entrada).



(a) Modelo A



(b) Modelo B



(c) Modelo C

Além dos modelos criados pela PG, foram realizados testes com um modelo de classificador padrão simples (*baseline*), que faz a soma das polaridades das palavras contidas nas mensagens, representado pela função *polaritySum(x)*. O objetivo é comparar um modelo simples com os modelos gerados pelo processo proposto neste trabalho, de forma a identificar os possíveis ganhos nos resultados dos novos classificadores.

6 Resultados

Nesta seção, são apresentados os resultados obtidos com os modelos gerados pela abordagem proposta.

A Tabela 2 apresenta o resultado de cada modelo gerado em cada base de teste, segundo as métricas.

Dentre os modelos testados, o B (figura 2b) obteve melhores resultados em 4 das 5 bases, em relação ao modelo A (figura 2a), apresentando resultado inferior somente na base *Tweets2014*. Além disso, considerando a avaliação de todas as mensagens, teve um desempenho superior de aproximadamente 4%.

Como podemos perceber, os resultados do melhor classificador (modelo C) para a base *TwitterSarcasm* (que possui sarcasmo em seu conteúdo) tiveram um resultado consideravelmente baixo: F1 de 0.2854. Isso acontece pela dificuldade de identificação dessa figura de linguagem pelo modelo. Normalmente, mensagens

Table 2: Resultado de cada modelo por base de teste

	Base	Acurácia	Precisão	Recall	F1	F1 (SemEval)
Modelo A	Tweets2013	0.5969	0.5886	0.5552	0.5599	0.5137
	Tweets2014	0.5413	0.5309	0.514	0.4967	**0.458**
	TwitterSarcasm	0.2791	0.3623	0.4098	0.2597	0.2229
	SMS2013	0.645	0.5898	0.5363	0.5498	0.4472
	LiveJournal	0.613	0.6381	0.5973	0.601	0.5889
	Todas	0.5956	0.5901	0.5456	0.5531	0.4999
Modelo B	Tweets2013	0.6001	0.5815	0.5702	0.5671	**0.5193**
	Tweets2014	0.537	0.5141	0.5197	0.4929	0.4507
	TwitterSarcasm	0.2907	0.3426	0.4182	0.2772	0.2412
	SMS2013	0.6474	0.583	0.5451	0.5569	**0.4549**
	LiveJournal	0.627	0.6364	0.6169	0.6189	**0.6052**
	Todas	0.5985	0.5814	0.5586	0.5605	**0.5054**
Modelo C	Tweets2013	0.5846	0.5734	0.5623	0.5529	0.5
	Tweets2014	0.5084	0.5001	0.511	0.4704	0.4205
	TwitterSarcasm	0.314	0.3943	0.4348	0.3067	**0.2854**
	SMS2013	0.6421	0.5705	0.5325	0.5446	0.435
	LiveJournal	0.6165	0.6265	0.6087	0.6079	0.5861
	Todas	0.5837	0.5701	0.5485	0.5451	0.4833
polaritySum(x)	Tweets2013	0.5796	0.5753	0.5354	0.5377	0.4831
	Tweets2014	0.5175	0.5163	0.4955	0.4726	0.4262
	TwitterSarcasm	0.2791	0.3623	0.4098	0.2597	0.2229
	SMS2013	0.6436	0.5859	0.5283	0.5428	0.4358
	LiveJournal	0.6121	0.6419	0.5968	0.5999	0.5845
	Todas	0.583	0.5807	0.5314	0.5367	0.476

sarcásticas tem pelo menos uma sentença positiva e uma outra negativa. Com isso, acredita-se que a inclusão de novas funções na PG e a utilização de bases de treinamento com mais frases contendo sarcasmo trarão melhorias para os futuros modelos.

Os melhores resultados dos modelos gerados puderam ser observados na base *LiveJournal*, com um F1 médio de 0.6052. Isso demonstra que os modelos se adequaram às mensagens com palavras mais comuns e pouco uso de gírias e abreviações.

Somado a isso, percebe-se um baixo valor de *Recall* das classes positiva (0,5) e negativa (0,4) no melhor modelo B, demonstrando que uma parte das mensagens não receberam valor semântico e, por isso, foram classificadas como neutras. A ausência das palavras das mensagens nos dicionários utilizados provocam esse efeito.

Tanto o baixo valor de *Recall* quanto o bom desempenho na base *LiveJournal*, demonstram que a quantidade e qualidade dos dicionários utilizados tem grande impacto na eficácia da classificação. Assim, faz-se necessário a inclusão de mais dicionários para melhorar o desempenho dos modelos.

Em comparação com o modelo de soma simples de polaridades (*baseline*), utilizado em grande parte dos trabalhos (inclusive em alguns que participaram da competição SemEval2014) o melhor modelo gerado pela PG obteve um ganho de 6% na média F1 de todas as bases. Destaca-se a melhoria de 6,97% na base Tweets2013 do modelo B.

7 Conclusão

Considerando a importância dos modelos de classificação de sentimento e o custo para gerá-los manualmente, o presente trabalho propõe o uso de PG para automatizar a geração desses modelos.

Para validação da abordagem, foram construídos três modelos pela PG utilizando uma base de treinamento com 9684 mensagens. Posteriormente, os modelos foram avaliados utilizando uma base de teste com 8987 mensagens (*benchmark* SemEval 2014).

Percebe-se que alguns modelos apresentaram melhores resultados em determinadas sub-bases de teste que outras, mas em todas as bases o F1 médio dos modelos gerados pela PG foram superiores ao *baseline* e, por isso, podem ser considerados satisfatórios.

A exceção é o resultado para as mensagens com sacarmo, que apresentaram um valor de F1 médio baixo, apesar de superior ao *baseline*.

Por fim, destaca-se que a média F1 dos três modelos para todas as mensagens de teste é de 55%, o que é satisfatório dada a dificuldade do *benchmark* utilizado. Assim, conclui-se que a abordagem é promissora.

Para melhorar o processo pretende-se, em trabalhos futuros, incluir novas funções para uso da PG, fazer modificações no *design* do algoritmo (diferentes operadores), ampliar o conjunto de dicionários utilizados e, por fim, usar bases de treinamento maiores e, principalmente, mais diversas.

8 Agradecimentos

Agradecemos a FAPEG - Fundação de Amparo a Pesquisa do Estado de Goiás - pelo suporte na apresentação do trabalho e ao CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico - pela concessão da bolsa de pesquisa.

References

1. Araújo, M., Gonçalves, P., Benevenuto, F.: Métodos para análise de sentimentos no twitter (2013)
2. Becker, L., Erhart, G., Skiba, D., Matula, V.: Avaya: Sentiment analysis on twitter with self-training and polarity lexicon expansion. In: Second Joint Conference on Lexical and Computational Semantics (* SEM). vol. 2, pp. 333–340 (2013)
3. D'Andrea, A., Ferri, F., Grifoni, P., Guzzo, T.: Article: Approaches, tools and applications for sentiment analysis implementation. International Journal of Computer Applications 125(3), 26–33 (September 2015)
4. Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M., Gagné, C.: DEAP: Evolutionary algorithms made easy. Journal of Machine Learning Research 13, 2171–2175 (jul 2012)
5. Ghaleb, O.A.M., Vijendran, A.S.: Survey and analysis of recent sentiment analysis schemes relating to social media. Indian Journal of Science and Technology 9(41) (2016)
6. Giachanou, A., Crestani, F.: Like it or not: A survey of twitter sentiment analysis methods. ACM Comput. Surv. 49(2), 28:1–28:41 (Jun 2016)
7. Graff, M., Tellez, E.S., Escalante, H.J., Miranda-Jiménez, S.: Semantic genetic programming for sentiment analysis. In: NEO 2015, pp. 43–65. Springer (2017)
8. Guimaraes, N., Torgo, L., Figueira, A.: Lexicon expansion system for domain and time oriented sentiment analysis. In: Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016). pp. 463–471 (2016)

9. Haddi, E., Liu, X., Shi, Y.: The role of text pre-processing in sentiment analysis. Procedia Computer Science 17, 26 – 32 (2013), first International Conference on Information Technology and Quantitative Management
10. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 168–177 (2004)
11. Iqbal, M., Karim, A., Kamiran, F.: Bias-aware lexicon-based sentiment analysis. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing. pp. 845–850. SAC '15 (2015)
12. Kaji, N., Kitsuregawa, M.: Building lexicon for sentiment analysis from massive collection of html documents. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). pp. 1075–1083. Association for Computational Linguistics, Prague, Czech Republic (June 2007)
13. Kanayama, H., Nasukawa, T.: Fully automatic lexicon expansion for domain-oriented sentiment analysis. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. pp. 355–363. EMNLP '06, Association for Computational Linguistics, Stroudsburg, PA, USA (2006)
14. Koza, J.R.: Genetic programming: on the programming of computers by means of natural selection, vol. 1. MIT press (1992)
15. Leal, J., Pinto, S., Bento, A., Oliveira, H.G., Gomes, P.: Cisuc-kis: Tackling message polarity classification with a large and diverse set of features. (2014)
16. Liu, B.: Sentiment analysis: A multifaceted problem. IEEE Intelligent Systems 25(3), 76–80 (8 2010)
17. Liu, B.: Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers (2012)
18. McPhee, N.F., Poli, R., Langdon, W.B.: Field guide to genetic programming (2008)
19. Mohammad, S.M.: Challenges in sentiment analysis. A Practical Guide to Sentiment Analysis, D. Das, E. Cambria, and S. Bandyopadhyay, Eds. Springer (2016)
20. Mohammad, S.M., Kiritchenko, S., Zhu, X.: Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. arXiv preprint arXiv:1308.6242 (2013)
21. Musto, C., Semeraro, G., Polignano, M.: A comparison of lexicon-based approaches for sentiment analysis of microblog posts. Information Filtering and Retrieval 59 (2014)
22. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(1-2), 1–135 (2008)
23. Rodrigues, R.G., das Dores, R.M., Camilo-Junior, C.G., Rosa, T.C.: Sentihealth-cancer: A sentiment analysis tool to help detecting mood of patients in online social networks. International Journal of Medical Informatics 85(1), 80 – 95 (2016)
24. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Comput. Linguist. 37(2), 267–307 (Jun 2011)
25. Turney, P.D.: Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. pp. 417–424. ACL '02 (2002)
26. Vohra, S., Teraiya, J.: A comparative study of sentiment analysis techniques. Journal JIKRCE 2(2), 313–317 (2013)
27. Wijksgatan, O., Furrer, L.: Gu-mlt-lt: Sentiment analysis of short messages using linguistic features and stochastic gradient descent. Atlanta, Georgia, USA 328 (2013)