

Geração automática de avaliações de objetos de aprendizagem por meio de mineração de textos

Henrique L. dos Santos¹, Cristian Cechinel¹

¹Universidade Federal de Pelotas
Caixa Postal: 354 – 96010-900 - Pelotas, RS, Brasil

hldsantos@inf.ufpel.edu.br, contato@crisiancechinel.pro.br

Abstract. *The recommendation of educational resources is an important field of learning analytics. Using collaborative filtering, content-based recommendation or even hybrid approaches, this field has been frequently explored over the last years. However, these techniques, specially collaborative filtering, suffer from the lack of learning objects' evaluation data, required in order to perform some recommendation algorithms. The present work aims to investigate whether it is possible to automatically generate some kind of useful evaluation to recommender systems, through performing text mining algorithms over user comments about learning objects. Results shown here point out that it is possible to generate classification models with meaningfully results.*

Resumo. *A recomendação de recursos educacionais é uma importante sub-área da analítica da aprendizagem. O uso de filtragem colaborativa, recomendação baseada em conteúdo ou até mesmo abordagens híbridas vem sendo investigado frequentemente nos últimos anos. Contudo, tais técnicas, especialmente a filtragem colaborativa, carecem de dados avaliativos sobre os objetos de aprendizagem (OAs), necessários para o processo de recomendação. O presente trabalho visa investigar se é possível, através de mineração textual sobre comentários textuais de usuários, gerar automaticamente algum tipo de avaliação útil para sistemas de recomendação. Os resultados coletados indicam que é possível gerar modelos de classificação significativamente corretos para tal fim.*

1. Introdução

O estudo do processo de recomendação de conteúdo educacional é uma das áreas possíveis da analítica da aprendizagem (do inglês *Learning Analytics*). Como exposto por [Siemens 2012], sistemas de recomendação podem prover um caminho personalizado e adaptável ao ensino de um indivíduo através da comparação do perfil alvo com outros perfis de usuários do sistema. Nesse sentido, um sistema de recomendação de recursos educacionais deve coletar dados referentes a estes recursos (avaliações, revisões, metadados, etc), processá-los de acordo com a técnica de recomendação utilizada e, por fim, como resultado final, fornecer aos usuários recomendações de materiais que sejam úteis e de boa qualidade a fim de enriquecer a sua experiência de ensino do ambiente virtual em questão. Entretanto, no que diz respeito a repositórios de objetos de aprendizagem (ROAs), existe uma ausência significativa de revisões e/ou avaliações de grande parte dos OAs disponibilizados virtualmente [Cechinel et al. 2011]. Tal fato tende a enfraquecer o poder de recomendação, caso a mesma se baseie em filtragem colaborativa, por exemplo.

O trabalho aqui conduzido, investiga a possibilidade de gerar automaticamente avaliações (em formato binário) a partir de comentários (em formato de texto livre) feitos por usuários do repositório Merlot¹. Ressalta-se que tal repositório permite que usuários façam comentários acerca de um OA juntamente com uma pontuação, mas que em muitos casos os usuários não realizam a avaliação explícita (pontuação) e fornecem apenas os comentários. Nesse trabalho, técnicas de mineração de texto e algoritmos de classificação foram aplicados a um conjunto de comentários com pontuações para testar a acurácia de modelos de classificação automática da qualidade por meio de comentários. Nas seções seguintes, mostrar-se-á que os modelos obtiveram uma acurácia suficientemente significativa, mostrando ser possível a geração de avaliações para auxiliar em um processo de recomendação.

O trabalho está organizado da seguinte maneira. A seção 2 sumariza e descreve os trabalhos relacionados. Na seção 3 são apresentados os objetivos, a metodologia utilizada e as características da coleção de dados utilizada. Os resultados são discutidos na seção 4. Por fim, a seção 5 conclui o artigo, apresentando os benefícios obtidos pelo experimento e que deverão ser explorados no futuro.

2. Trabalhos correlatos

O trabalho aqui apresentado pode ser visto como uma alternativa à proposta apresentada por [Koukourikos et al. 2012], onde uma análise de sentimento foi conduzida sobre os comentários e revisões do site Merlot. Nessa proposta, os autores utilizaram diversos conjuntos de palavras com polaridade sentimental determinada (positiva, para palavras que expressam um sentimento positivo no comentário, e negativa, para o contrário) para classificar um comentário em neutro (nota 3), positivo (nota maior ou igual a 4) ou negativo (nota menor do que 3). Com esse trabalho se tem a intenção futura de popular o repositório Merlot com avaliações que se encontram implícitas nos comentários feitos por usuários do site. A ideia é que a inserção de novas avaliações possa auxiliar na melhoria de métodos de recomendação, por filtragem colaborativa, por exemplo, de OAs aos usuários.

O uso de mineração de texto para classificar opiniões, em forma de texto livre, de usuários vem sendo amplamente aplicado nos últimos anos, especialmente no contexto de *e-commerce* [Cao et al. 2011] [Gamon et al. 2005] [Hu and Liu 2004]. Entretanto, no contexto educacional, esforços nesse sentido foram aplicados apenas recentemente. Dentro desse escopo, pode-se destacar o trabalho de [Yu and Luna 2013], onde os *feedbacks* de alunos de um sistema de *e-learning* são analisados e categorizados via mineração textual. Ainda, destaca-se a proposta de [Hsu and Chang 2012] que visa aplicar mineração de texto nos diários de classe de professores que lecionam atividades educacionais que envolvem não somente o aprendizado em sala de aula, como também atividades aplicadas a alguma comunidade externa à Universidade de Harvard. Nesse sentido, os autores buscam identificar fatores que possam melhorar a qualidade desse método de ensino, chamado de *service-learning*.

Além desses, outros trabalhos utilizando mineração de texto em um contexto educacional podem ser destacados. Por exemplo, a proposta de [Suryani and Hasibuan 2013] tenciona personalizar o processo de disponibilização de conteúdo educacional a alunos

¹www.merlot.org

através de duas abordagens: mineração de texto combinada com regras de filtragem determinísticas e ontologias. Ainda, o trabalho de [da Costa Pinho et al. 2013] propõe a construção de um agente mediador automatizado para analisar e auxiliar em intervenções em um fórum online de discussão de alunos de turmas de aprendizado de língua estrangeira. Este agente tem por intuito principal auxiliar tutores humanos nas estratégias de desenvolvimento das discussões, além de identificar possíveis desvios de assunto nas conversações entre participantes da discussão. Já [Ou et al. 2005] utiliza mineração textual em um sistema de *e-learning* para identificar os papéis de cada aluno dentro do seu grupo, assim, os resultados de cada grupo e de seus membros (associados a algum dos 11 papéis identificados) são analisados e a influência desses papéis no desempenho dos grupos é demonstrada.

No contexto brasileiro, pode-se ressaltar o trabalho de [Severo et al. 2013] onde técnicas de mineração textual são aplicadas em conversações virtuais (dentro de um ambiente de *e-learning*) com o objetivo de identificar e categorizar tipos de mediação existentes entre os alunos e professores. Ainda, em [Goularte et al. 2014] é realizada uma tentativa de sumarizar atividades dissertativas (disponibilizadas aos alunos através de um ambiente virtual de aprendizagem) para automatizar o processo de correção por parte dos professores.

3. Objetivos e metodologia

3.1. Objetivos e descrição dos dados

O principal objetivo do trabalho aqui exposto é testar modelos para classificar comentários (em formato de texto livre) de usuários sobre OAs de forma automatizada. Assim, de posse dessa classificação (numérica ou nominal), a avaliação do usuário sobre determinado OA pode ser contabilizada e utilizada dentro de sistemas de recomendação, como por exemplo os baseados em filtragem colaborativa [Cechinel et al. 2013].

Atualmente, cerca de 15,4% dos comentários do repositório Merlot contém apenas texto livre, desprovidos de uma nota avaliativa. Nesse sentido, aplicamos algoritmos de classificação aos comentários restantes (que contém o texto e também a nota, contabilizando 9877 avaliações). Estas notas variam de 1 a 5 e são atribuídas ao OA diretamente pelo usuário.

Ainda, é importante ressaltar que tais comentários passaram por uma breve etapa de pré-processamento onde ocorreu a remoção de hiperlinks e a separação de palavras e pontuação de interrogação e exclamação, de modo que esses dois símbolos de pontuação pudessem ser tratados como expressões individuais pelos algoritmos de mineração de texto aplicados posteriormente. Os algoritmos utilizados estão disponíveis no software de mineração de dados Weka²

Entretanto, como a Figura 1 mostra a seguir, as avaliações se encontravam naturalmente desbalanceadas, com cerca de 85% das avaliações concentradas em notas 4 e 5. Dessa forma, optamos por não tentar classificar um comentário em uma nota específica (de 1 a 5), já que esse método exigiria uma demanda de reamostragem muito grande para gerar um classificador minimamente satisfatório. Portanto, utilizando a divisão por tercil,

²<http://www.cs.waikato.ac.nz/ml/weka/>

obtivemos apenas duas classes para a posterior classificação, sejam elas: *Excelente* (para nota exatamente igual a 5) e *Nao Excelente* (para qualquer nota abaixo de 5).

As Tabelas 1 e 2 apresentam uma breve comparação entre a média de comprimento (em termos de quantidade de palavras) dos comentários coletados, em termos de classes numéricas e nominais (criadas a partir da regra do tercil), respectivamente. Como pode ser observado na Tabela 1, só há uma diferença relevante no comprimento dos comentários entre valores 1 e os demais. Na Tabela 2, que mostra os resultados para uma divisão binária já aplicada, se nota que já não é mais possível atribuir um comprimento de comentário a uma classe específica.

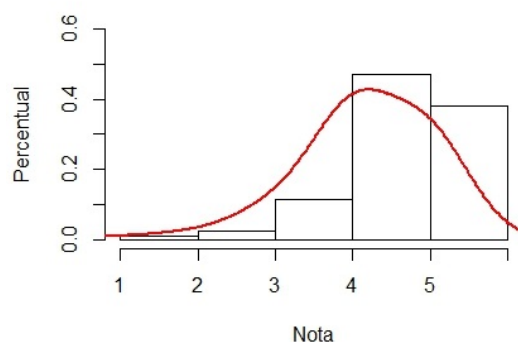


Figura 1. Histograma com linha de densidade da distribuição das notas dos comentários de usuários

Tabela 1. Média de palavras utilizadas nos comentários (para as 5 classes possíveis)

Valor da nota	Média de palavras (com stop-words) utilizadas	Média de palavras (sem stop-words) utilizadas
1	43	18
2	66	28
3	72	30
4	76	33
5	72	32

Tabela 2. Média de palavras utilizadas nos comentários (para 2 classes)

Classe	Média de palavras (com stop-words) utilizadas	Média de palavras (sem stop-words) utilizadas
Excelente	72	32
Não excelente	75	32

3.2. Mineração textual e classificação

3.2.1. Transformação de texto para vetor de palavras

O primeiro passo do processo consistiu em converter o arquivo *.csv* contendo o comentário e em seguida a nota do usuário (já considerada dentro de duas classes: excelente e não excelente) para um arquivo contendo valores referentes à palavras, mantendo a informação da nota como classe no final de cada linha. Para tanto, o filtro *StringToWordVector* foi utilizado. Foram testadas diversas configurações na aplicação desse filtro, entre elas: valor designado a palavra (contagem da palavra no registro *word count* ou valor $TF - IDF$ da palavra), uso de *stop-words* e uso de *n*-gramas.

O conjunto de *stop-words*, quando utilizado, foi retirado do projeto *Stop words*³. Foram utilizadas *stop-words* do idioma inglês, espanhol e português brasileiro. No que se refere ao uso de *n*-gramas, o valor de *n* foi definido dentro do intervalo fechado de 1 a 3. O número total de palavras mantidas após o processo foi definido em 1000.

3.2.2. Seleção de atributos e reamostragem

Na sequência, optamos por utilizar um filtro de seleção de atributos, no intuito de diminuir a quantidade de palavras a serem consideradas pelos classificadores ao final do processo. Com classe-alvo sendo a avaliação final do usuário, cada palavra teve o seu ganho de informação (algoritmo *InfoGainAttributeEval*), em relação a classe-alvo, calculado da seguinte forma:

$$GanhoInfo(Classe, Atributo) = H(Classe) - H(Classe|Atributo)$$

onde a função $H(Classe)$ é a entropia anterior, calculada sem considerar o corte com o atributo em questão, e $H(Classe | Atributo)$ é a entropia posterior, calculada considerando a poda com valores do atributo corrente.

Após esse cálculo, os atributos foram ranqueados de acordo com o seu ganho de informação. As primeiras 200 palavras foram mantidas para o passo seguinte.

Apesar da divisão em duas classes feita anteriormente, o problema de desbalanceamento persistiu (em um nível menor), já que aproximadamente 65% dos comentários estavam atrelados a classe *Nao Excelente*. Por isso, decidimos aplicar, após a seleção de atributos, o algoritmo *SMOTE* (*Synthetic Minority Over-sampling Technique*[Chawla et al. 2002] a fim de reamostrar as instâncias pertencentes a classe minoritária. O algoritmo *SMOTE* sintetiza instâncias da classe minoritária a partir da seguinte construção: dada uma instância existente da classe minoritária, os *k* vizinhos mais próximos dessa instância são selecionados. Um desses vizinhos é selecionado aleatoriamente e a geração da nova instância se dá pelo cômputo:

$$NovaInstancia = InstanciaPivo + rand(0 - 1) * (InstanciaVizinho - InstanciaPivo)$$

onde $rand(0 - 1)$ é uma função que gera um número randômico entre 0 e 1. Naturalmente, todas as operações da fórmula acima são vetoriais.

³<https://code.google.com/p/stop-words/>

No presente trabalho, a classe minoritária (*Excelente*) foi reamostrada em 50%, de forma que a nova divisão de amostras se tornou: 52% de comentários pertencentes a *Nao Excelente* e 48% pertencentes a *Excelente*. Ao final dessa etapa, os dados já se encontram prontos para a classificação (ver Figura 2).

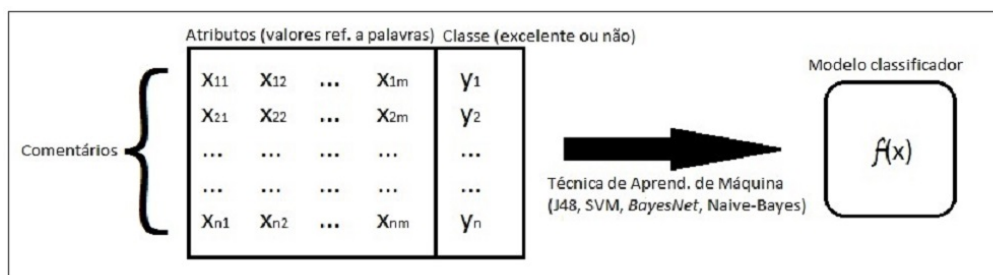


Figura 2. Ilustração da última etapa do processo (classificação) após os dados terem sido devidamente filtrados e balanceados. Nesse caso, $n = 9877$ e $m = 200$

3.2.3. Classificação

Para as tarefas de classificação, foram utilizados os seguintes algoritmos: J48 (árvore de decisão), máquina de vetores de suporte (*Support Vector Machine*, SVM), Rede Bayesiana (*BayesNet*) e Classificador *Naive-Bayes*. É importante ressaltar que o algoritmo SVM apresenta historicamente um bom desempenho em tarefas de mineração textual, como pode ser visto em [Li and Wu 2010], [Yu 2009], e [Joachims 2002]. Os outros algoritmos aqui avaliados também já foram aplicados em experimentos voltados a mineração de texto, tais como [Jain et al. 2004] e [AL-Wehaibi and Khan 2014](J48 e *Naive-Bayes*), [Li and Tsai 2013] (J48, Redes Bayesianas e SVM) e [Ur-Rahman and Harding 2012] (*Naive-Bayes* e SVM).

Para efeito de avaliação, foi utilizado o método de validação cruzada com 5 pastas, onde o conjunto total de dados é dividido em 5 pastas e 4 delas são utilizadas para treino, sendo a restante utilizada para teste. Esse processo se repete 5 vezes, a cada turno a pasta de teste é alternada. As taxas de erro médias são, então, calculadas.

4. Resultados e discussões

Ao todo, 5 algoritmos foram avaliados e para cada uma delas diversas configurações do filtro *StringToWordVector* foram testadas, tais como uso de *stop-words*, uso de *TF-IDF* ou contagem da palavra e uso de *n*-gramas. Os dois melhores resultados, em função de acurácia geral média e coeficiente *kappa*, podem ser analisados na Tabela 3.

Como pode ser visto, os coeficientes *kappa* mostram que os modelos construídos apresentam uma concordância regular ou moderada, de acordo com [Landis and Koch 1977]. Ainda assim, apesar de apresentar uma acurácia acima de 65%, outro fator importante deve ser analisado, qual seja: acurácia interna de cada classe. Tal métrica define se o modelo construído é realmente útil, uma vez que um modelo que classifica corretamente, por exemplo, 90% das instâncias de uma classe mas apenas 30% das instâncias de outra, é um modelo claramente falho. Além disso, outras medidas importantes para a avaliação de um classificador são a precisão e o *recall*.

Tabela 3. Configurações e resultados dos experimentos conduzidos

Referência (algoritmo + índice)	Uso de stop-words	Valor ref. a palavra	N-gramas	Acurácia Média (Excelente/Não Excelente) (%)	Precisão (%)	Coefficiente Kappa
Rede Bayesiana.1	Não	Contagem	1 a 3	72,2 (55,5/87,5)	74	0,4355
Rede Bayesiana.2	Não	TF-IDF	1 a 3	71,8 (58,3/84,3)	72,8	0,4301
Naive-Bayes.1	Não	TF-IDF	1 a 3	71,5 (61,1/80,9)	71,9	0,4237
SVM.1	Não	TF-IDF	1	69,6 (67,3/71,7)	69,6	0,3902
Naive-Bayes.2	Sim	TF-IDF	1	68,9 (42,5/93)	73,9	0,3632
SVM.2	Não	TF-IDF	1 a 3	68,6 (63/73,9)	68,7	0,3695
J48.1	Não	Contagem	1	68,1 (63/72,8)	68,1	0,359
J48.2	Não	TF-IDF	1 a 3	67,6 (62/72,7)	67,6	0,348

Nesse caso, o *recall* é a própria taxa de verdadeiro positivo (*TP Rate*) calculada sob a matriz de confusão, ou seja, o *recall* é a própria acurácia média do classificador. Já a precisão pode ser vista como a razão entre a quantidade de instâncias corretamente atribuídas a uma classe e a quantidade total de instâncias atribuídas a mesma classe [Gasparini et al. 2005].

Além disso, deve-se ressaltar que uma acurácia geral alta (tal como a do experimento Rede Bayesiana.1) não indica necessariamente um ótimo modelo, já que a acurácia do modelo para uma das classes é relativamente baixa. Dentre os experimentos, destacam-se aqueles que conseguiram manter ambas as acurácias perto dos 70%, tais como SVM.1 (especialmente), SVM.2, J48.1 e Naive-Bayes.1 (minimamente).

5. Considerações finais

O trabalho desenvolvido neste artigo compara diversas configurações de filtragem de texto e algoritmos de classificação a fim de investigar a possibilidade de classificação (binária) de avaliações, em formato de texto livre, de usuários acerca de objetos de aprendizagem dentro de um repositório. Nossa ideia é que tal classificação, se implementada com uma acurácia significativa para ambas as classes, possa tornar comentários, desprovidos de nota numérica explícita, úteis, por exemplo, para um processo de recomendação por filtragem colaborativa.

Os resultados obtidos mostram que é possível construir um modelo classificador, através de mineração de texto, que classifica os comentários com cerca de 70% de acurácia para ambas as classes. Apesar disso, vale ressaltar que foi necessário considerar uma divisão de classes diferente da proposta original do repositório, isto é, consideramos a divisão em apenas duas classes e, além disso, também foi necessário aplicar uma técnica de reamostragem a fim de balancear as instâncias das duas classes que escolhemos.

No futuro, repopularemos a base de dados, classificando os comentários desprovidos de notas com os modelos aqui apresentados, e faremos experimentos com técnicas de recomendação baseadas em filtragem colaborativa. Atualmente, o repositório Merlot possui cerca de 2000 comentários desprovidos de nota, e, portanto, inúteis para um processo de recomendação. Esperamos que o uso desses comentários melhore não somente as taxas de erro de recomendação como também aumente a cobertura, em termos de usuários, das recomendações geradas.

Agradecimentos

Este trabalho foi financiado pela FAPERGS (Fundação de Amparo à Pesquisa do Rio Grande do Sul) através do seu programa Pesquisador Gaúcho - PQG - Edital 02/2014, pelo CYTED (Programa Ibero-Americano para Ciência, Tecnologia e Desenvolvimento) como parte do seu projeto "RIURE - Red iberoamericana para la usabilidad de repositorios educativos" e pela Universidade Federal de Pelotas (UFPel-Brasil).

Referências

- AL-Wehaibi, R. N. and Khan, M. B. (2014). Predicting arabic tweet popularity by use of data and text mining techniques. In *Proceedings of the 6th International Conference on Management of Emergent Digital EcoSystems*, MEDES '14, pages 31:183–31:189, New York, NY, USA. ACM.
- Cao, Q., Duan, W., and Gan, Q. (2011). Exploring determinants of voting for the "helpfulness" of online user reviews: A text mining approach. *Decision Support Systems*, 50(2):511–521.
- Cechinel, C., Sicilia, M.-Á., Sánchez-Alonso, S., and García-Barriocanal, E. (2013). Evaluating collaborative filtering recommendations inside large learning object repositories. *Information Processing & Management*, 49(1):34 – 50.
- Cechinel, C., Sánchez-Alonso, S., and García-Barriocanal, E. (2011). Statistical profiles of highly-rated learning objects. *Computers & Education*, 57(1):1255 – 1269.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321–357.
- da Costa Pinho, I., Epstein, D., Berni Reategui, E., Correa, Y., and Polonia, E. (2013). The use of text mining to build a pedagogical agent capable of mediating synchronous online discussions in the context of foreign language learning. In *Frontiers in Education Conference, 2013 IEEE*, pages 393–399.
- Gamon, M., Gamon, M., Aue, a., Aue, a., Corston-Oliver, S., Corston-Oliver, S., Ringger, E., and Ringger, E. (2005). Pulse: Mining customer opinions from free text. *Lecture Notes in Computer Science*, 3646:121–132.
- Gasparini, F., Corchs, S., and Schettini, R. (2005). A recall or precision oriented skin classifier using binary combining strategies. *Pattern Recognition*, 38(11):2204 – 2207.
- Goularte, F. B., Wilges, B., Nassar, S. M., and Cislighi, R. (2014). Métricas de sumarização automática de texto em tarefas de um Ambiente Virtual de Aprendizagem. In *Anais do XXV Simpósio Brasileiro de Informática na Educação*, pages 752–761.
- Hsu, C.-L. and Chang, Y.-F. (2012). Qualitative text mining in student's service learning diary. In *Innovations in Bio-Inspired Computing and Applications (IBICA), 2012 Third International Conference on*, pages 350–354.
- Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining KDD 04*, 04:168.
- Jain, G., Ginwala, A., and Aslandogan, Y. (2004). An approach to text classification using dimensionality reduction and combination of classifiers. In *Information Reuse and*

- Integration, 2004. IRI 2004. Proceedings of the 2004 IEEE International Conference on*, pages 564–569.
- Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA.
- Koukourikos, A., Stoitsis, G., and Karampiperis, P. (2012). Sentiment analysis: A tool for rating attribution to content in recommender systems. *CEUR Workshop Proceedings*, 896:61–70.
- Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):pp. 159–174.
- Li, N. and Wu, D. D. (2010). Using text mining and sentiment analysis for online forums hotspot detection and forecast. *Decision Support Systems*, 48(2):354 – 368.
- Li, S.-T. and Tsai, F.-C. (2013). A fuzzy conceptualization model for text mining with application in opinion polarity classification. *Knowledge-Based Systems*, 39(0):23 – 33.
- Ou, K.-L., Wang, C.-Y., and Chen, G.-D. (2005). Identify group roles by text mining on group discussion in a web-based learning system. In *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, volume 9, pages 5566–5572 Vol. 9.
- Severo, C. E. P., Passerino, L. M., and De Lima, J. V. (2013). Mapeamento de Índícios de Mediação em Ambiente Virtual de Ensino e Aprendizagem por Meio de Um Sistema de Mineração de Textos. In *Anais do XXIV Simpósio Brasileiro de Informática na Educação*, pages 567–576.
- Siemens, G. (2012). Learning analytics: Envisioning a research discipline and a domain of practice. In *Proceedings of the 2Nd International Conference on Learning Analytics and Knowledge, LAK '12*, pages 4–8, New York, NY, USA. ACM.
- Suryani, M. and Hasibuan, Z. (2013). The study of dynamic delivery adaptive learning content in e-learning personalization using text mining and ontology approach. In *Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on*, pages 21–26.
- Ur-Rahman, N. and Harding, J. (2012). Textual data mining for industrial knowledge management and text classification: A business oriented approach. *Expert Systems with Applications*, 39(5):4729 – 4739.
- Yu, C. (2009). Mining product features from free-text customer reviews: An svm-based approach. In *Information Science and Engineering (ICISE), 2009 1st International Conference on*, pages 900–903.
- Yu, W.-B. and Luna, R. (2013). Exploring user feedback of a e-learning system: A text mining approach. In Yamamoto, S., editor, *Human Interface and the Management of Information. Information and Interaction for Learning, Culture, Collaboration and Business*, volume 8018 of *Lecture Notes in Computer Science*, pages 182–191. Springer Berlin Heidelberg.