

Sistema de Reputação e Ranqueamento para Redes Sociais Aplicado ao Participatório

Daniel Gomes¹, Diego Pasqualin¹, Fabiano Silva¹, Luis C. E. de Bona¹, Mateus Strey¹

¹Departamento de Informática - Universidade Federal do Paraná (UFPR)
Caixa Postal: 19081 - CEP 81531-980 – Curitiba – PR – Brasil

{dgp09, dpasqualin, fabiano, bona, mars11}@c3sl.ufpr.br

Abstract. *This paper presents a system to compute the user reputation and posterior application in the issue of ordering content on social networks. The study case is the Participatório, a virtual environment proposed by the Brazilian Federal Government to encourage discussions on topics related to politics and youth.*

Resumo. *O presente artigo apresenta uma proposta para um sistema de cálculo de reputação de usuários e posterior aplicação na ordenação de conteúdo de redes sociais. O caso de uso apresentado é o Participatório - Observatório Participativo da Juventude, ambiente virtual promovido pelo Governo Federal do Brasil para incentivo a debates sobre temas relacionados a política e juventude.*

1. Introdução

O Participatório [SNJ/C3SL 2014] – Observatório Participativo da Juventude, é um ambiente virtual interativo inspirado nas redes sociais e que oferece um espaço para debates e discussões sobre temas relacionados às políticas de juventude.

Em tal ambiente de alta interatividade e constante produção de conteúdo livre é desejável que a informação apresentada seja ordenada de forma contínua e autônoma [Lampe and Resnick 2004], exibindo em destaque os debates mais relevantes sem eliminar a possibilidade de ascensão de conteúdo recém criado. Com esse objetivo foi desenvolvido e implementado um sistema de reputação para os usuários, utilizado para ordenação dos debates e discussões exibidos com destaque nas páginas do Participatório.

Diferente do conceito de *karma* [Ganley and Lampe 2009], usado em algumas redes sociais e portais de notícias [Slashdot 2014] como uma medida do comportamento geral do usuário no sistema, a reputação dos usuários depende apenas de suas interações. A proposta é que a reputação seja um atributo gerado a partir da opinião (“gostei”/“não gostei”) de terceiros sobre os comentários e debates criados por um usuário.

A seção 2 define as entidades envolvidas no sistema e descreve as equações para cálculo da reputação dos usuários, em seguida a seção 3 retrata como a reputação afeta a ordenação dos debates no Participatório, por fim a implementação dos métodos de reputação e ordenação é descrita na seção 4.

2. Método de Cálculo de Reputação

Definimos primeiramente as entidades do sistema:

- Usuário: entidade que representa um usuário qualquer do sistema. Serão considerados três classes de usuários: o Administrador ou Moderador, denotado por A ; o usuário Participatório, um tipo especial de administrador, denotado por P ; e os demais usuários do sistema, denotados por U .
- Comentário: é um texto incluído no sistema por um determinado usuário.
- Debate: é um conjunto de comentários.
- Postagem: é um tipo especial de comentário que inicia um debate.
- Reputação: representa a opinião dos usuários do sistema sobre um determinado usuário em um dado momento.

A reputação dos usuários será dada por um valor numérico dentro do intervalo $[0, 1]$, sendo 1 o valor máximo de reputação 0 o valor mínimo. Para um determinado usuário $u \in U$, sua reputação será denotada por $R(u)$.

Os valores iniciais para a reputação, no caso do Participatório, estão dispostos a seguir.

O usuário P recebe valor de reputação inicial $R(P) = 1$, ao usuário A é atribuído $R(A) = 0.8$, enquanto que os demais usuários do sistema recebem a atribuição inicial $R(U) = 0.1$. Os valores para a classe Participatório (P) e Administrador (A) são permanentes.

O valor da reputação de um usuário $u \in U$ pode variar dentro do intervalo $[0.1, 0.7]$. O limite superior é utilizado para garantir que postagens dos administradores, no geral comunicados de relevância geral, possuam sempre prioridade sobre os demais debates em andamento.

A reputação dos usuários do grupo U é alterada seguindo as regras abaixo:

1. A reputação de um usuário diminui com a passagem do tempo. A nova reputação de um usuário u , $R'(u)$, depois de um determinado intervalo de tempo Δ , é dada por: $R'(u) = a_t(\Delta) \times R(u)$, sendo $a_t(\Delta)$ o fator de atenuação calculado de acordo com o tempo decorrido Δ . O valor da função $a_t(\Delta)$ deve pertencer ao intervalo $[0, 1]$ e ser inversamente proporcional ao valor de Δ , ou seja, $a_t(\Delta)$ diminui quando Δ cresce.
2. A reputação de um usuário pode aumentar ou diminuir de acordo com a aceitação dos seus comentários por parte dos outros usuários. Os botões de ação “gostei” e “não-gostei” foram usados para implementar este tipo de ajuste na reputação. Considere dois usuários u e v pertencentes a U . O usuário v pressiona o botão “gostei” em algum comentário feito pelo usuário u . O novo valor da reputação do usuário u , $R'(u)$, é dado por: $R'(u) = R(u) + a_i \times R(v)$, onde a_i é o fator de atenuação da influência da reputação do usuário v sobre a reputação do usuário u . O valor de a_i deve pertencer ao intervalo $[-1, 0) \cup (0, 1]$. Caso v pressione o botão “não-gostei” o fator a_i deve ser negativo. Vale notar que um usuário pode executar a ação “gostei” ou “não-gostei” em um mesmo comentário uma única vez.

Os fatores de atenuação a_t e a_i determinam de forma indireta a velocidade com que as reputações variam no sistema com a passagem do tempo e a partir das avaliações “gostei” e “não-gostei” sobre os comentários dos usuários.

Fator de atenuação da influência:

Seja C_u o número de comentários do usuário u , C_k o número de comentários feitos pelo usuário $k \in U$ e s o sinal de a_i (1 para “gostei” e -1 para “não-gostei”), o fator de atenuação a_i é dado por:

$$a_i = s \times \frac{C_u}{\sum_{k \in U} C_k} \quad (1)$$

Exemplo:

Dado que $R'(u) = R(u) + a_i \times R(v)$, supondo que os usuários u e $v \in U$ possuem respectivamente $R(u) = 0.1$ e $R(v) = 0.2$ e que o número de comentários de ambos seja igual a $C_u = 15$ e $C_v = 23$. Para fins demonstrativos assumiremos que u e v são os únicos usuários do sistema e portanto $|U| = 2$.

O valor resultante de uma ação “gostei” do usuário v em um comentário do usuário u , como dito anteriormente, é igual a $R'(u) = R(u) + a_i \times R(v)$, e temos então que:

$$\begin{aligned} a_i &= [C_u / (C_u + C_v)] = 15 / (15 + 23) = 0.3947 \\ R'(u) &= 0.1 + 0.3947 \times 0.2 = 0.1 + 0.0789 = 0.1789 \end{aligned} \quad (2)$$

O valor resultante após uma ação de “não-gostei” de v sobre u seria:

$$R'(u) = 0.1 + -0.1789 \times 0.2 = 0.1 - 0.0789 = 0.0210 \quad (3)$$

Lembrando que $R(u)$ deve estar no intervalo $[0.1, 0.7]$ e em nosso último exemplo $R'(u) < 0.1$, então o valor final é forçado para $R(u) = 0.1$.

Fator de atenuação por tempo:

A atenuação temporal das reputações tem por objetivo valorizar os usuários com participação mais recente e controlar possíveis processos inflacionários nas reputações. Esta atenuação é aplicada diariamente (Δ referente a 24 horas) sobre todos os usuários de $u \in U$, atualizando suas reputações por:

$$R'(u) = a_t(\Delta) \times R(u) \quad (4)$$

O valor de a_t pode ser ajustado no intervalo de $[0, 1]$. É evidente que $a_t = 0$ implicaria em um reinício diário do sistema de reputação enquanto que $a_t = 1$ anularia o fator de atenuação temporal. Atualmente o fator de atenuação diária está definido em $a_t = 0,95$, ou seja, 5% da reputação do usuário é decrementada diariamente.

3. Método de Cálculo de Ordenação de Debates

Ordenação considerando o tempo da postagem e voto dos usuários:

Optamos por uma abordagem que permita que os debates novos sejam privilegiados em detrimento dos antigos, utilizando também a quantidade de ações “gostei” e “não gostei” atribuídas aos debates. Utilizamos para tal o algoritmo que calcula o *hot*

score [Salihefendic 2014] presente no Reddit [Reddit 2014], popular rede social de compartilhamento de conteúdos, criados ou referenciados pelos seus usuários. O sistema está em uso desde 2005 e atende a demanda de ordenação semelhante ao desejado no Participatório. O algoritmo funciona da seguinte maneira:

Dado um debate A postado em uma data A_t , posterior a data B_t de início de implementação do algoritmo, temos t como a diferença em segundos entre a data de criação dos debates (equação 5) e x como a diferença em valor absoluto da quantidade de “gostei” (G) e “não gostei” (N) atribuídos a A (equação 6):

$$t = A_t - B_t \quad (5)$$

$$x = G - N \quad (6)$$

Temos $g(x)$ definida da seguinte forma:

$$g(x) = \begin{cases} 1, & \text{se } x > 0 \\ 0, & \text{se } x = 0 \\ -1, & \text{se } x < 0 \end{cases} \quad (7)$$

E $h(x)$ como o máximo entre o valor absoluto de $|x|$ e 1:

$$h(x) = \begin{cases} |x|, & \text{se } |x| \geq 1 \\ 1, & \text{se } |x| < 1 \end{cases} \quad (8)$$

Sendo $t_{frame} = 45000$ segundos (12,5 horas) definindo o intervalo de tempo em que as postagens ganham 1 ponto a mais que as anteriores, temos o ranqueamento como a função $f(t, t_{frame}, g(x), h(x))$ definida a seguir:

$$f(t, t_{frame}, g(x), h(x)) = \log_{10} h(x) \times g(x) + \frac{t}{t_{frame}} \quad (9)$$

Exemplo 1: Variação com o x (equação 6), dados $A_t = 1395014400$ (17/03/2014 00:00:00 UTC), $B_t = 1356998400$ (01/01/2013 00:00:00 UTC), temos a Figura 1 e Figura 2.

Exemplo 2: Variação com o t , dados $B_t = 1356998400$ (01/01/2013 00:00:00 UTC), $G = 1000$ e $N = 400$, temos a Figura 3.

Visão geral: Dado um tempo $t_{postagem}$ em segundos, $t_{frame} = 45000$ segundos temos a seguinte ordenação dos debates [Billard 2014]:

1. Postagens com $x \geq 0$ no tempo $t_{postagem}$
2. Postagens com $x < 0$ no tempo $t_{postagem}$
3. Postagens com $x \geq 0$ no tempo $t_{postagem} - t_{frame}$
4. Postagens com $x < 0$ no tempo $t_{postagem} - t_{frame}$
5. Postagens com $x \geq 0$ no tempo $t_{postagem} - 2t_{frame}$

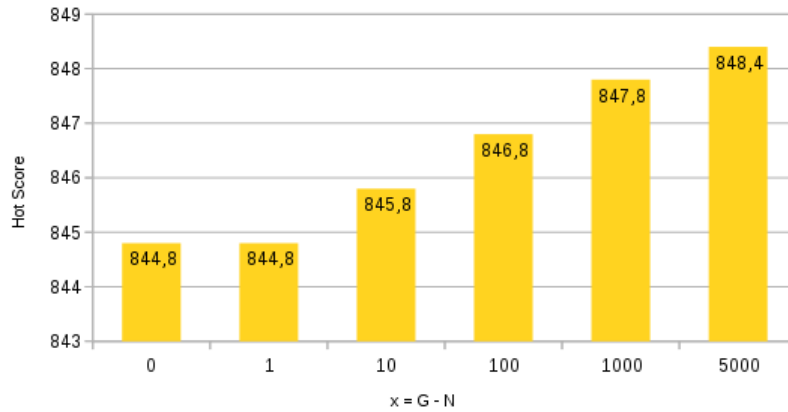


Figura 1. $x \geq 0$

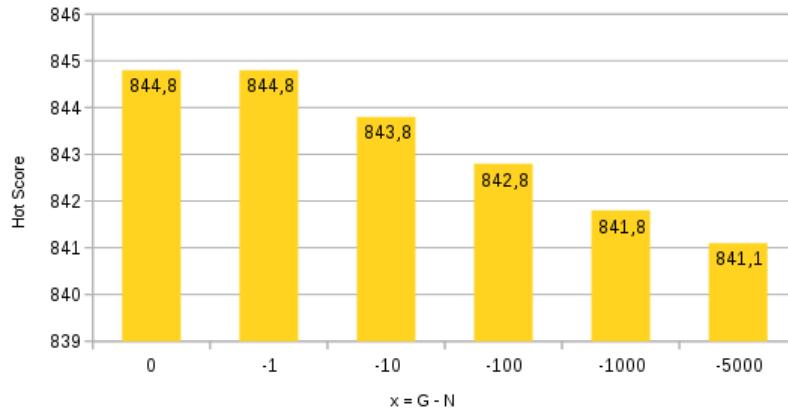


Figura 2. $x \leq 0$

6. Postagens com $x < 0$ no tempo $t_{postagem} - 2t_{frame}$
7. ...

Ordenação considerando também a reputação dos usuários:

Além dos fatores de tempo e voto, desejamos que o sistema de reputação tenha influência na ordenação dos debates. Criamos uma reputação para cada debate, formado através de um somatório R_D da reputação dos usuários únicos que contribuíram com ele.

Dado um conjunto de usuários D que participaram do debate e $R(u_i)$ a reputação de um usuário $u_i \in D$:

$$R_D = \sum_{i \in D} R(u_i) \quad (10)$$

Ordenação dos debates:

Dados o *hot score* (equação 9) e a reputação do debate (equação 10), temos a

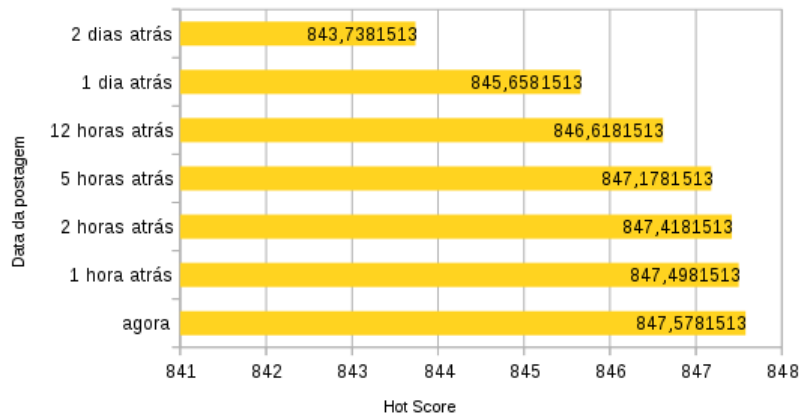


Figura 3. t variável, x constante

fórmula final de ordenação O como a multiplicação das duas:

$$O = \left[\log_{10} h(x) \times g(x) + \frac{t}{t_{frame}} \right] \times \left[\sum_{i \in D} R(u_i) \right] \quad (11)$$

Ressaltamos que a multiplicação simples entre as equações pode apresentar anomalias, por exemplo em debates com participação de muitos usuários, onde o valor do somatório das reputações pode ser muito elevado, tornando irrelevante o valor do *hot score*. Para resolver tais questões o uso de funções de normalização dos dados pode ser necessário [Hofmann 2003].

4. Implementação

Implementamos o sistema de reputação usando a estrutura de *plugins* da plataforma Elgg [Elgg 2014]¹, alterando *plugins* existentes e criando um novo chamado *reputation*.

O *plugin* da reputação abrange todas as funções necessárias para seu funcionamento, enquanto os *plugins* alterados recebem chamadas à essas funções e a alteração para a visualização dos debates ranqueados.

4.1. Resumo de funcionamento

Pela ação do usuário: O sistema de reputação usa a quantidade de comentários dos usuários na base do cálculo, assim foi implementado uma função cuja funcionalidade é incrementar ou decrementar tal quantidade dentro de *plugins* específicos (*groups* e *hy-peAlive/hypeFramework*) que são responsáveis por armazenar e remover comentários de acordo com as ações requisitadas pelo usuário, assim temos acesso rápido a soma de comentários que cada usuário possui.

O sistema também armazena o valor total de comentários no sistema em uma variável interna do banco de dados da plataforma Elgg. Com estas informações, juntamente com a reputação corrente do usuário que recebeu a ação “gostei” ou “não gostei” e do usuário que efetuou a ação, é calculada a nova reputação do usuário.

¹Motor de código aberto para redes sociais utilizado no desenvolvimento do Participatório

Pela atenuação por tempo: Ao final de cada dia a reputação dos usuário é decrementada pelo fator de atenuação por tempo a_t com o uso do programa **cron** do sistema Unix, que executa a função afetando todos os usuários do sistema simultaneamente.

Pela ordenação dos debates: A cada 30 minutos a fórmula de ordenamento (equação 11) é recalculada para todos os debates da plataforma, utilizando o programa **cron**.

4.2. Estrutura do sistema

O *plugin* de reputação (*reputation*) foi desenvolvido e é composto pelas funções que executam os métodos definidos neste artigo.

O *plugin groups*, responsável pela criação dos debates da plataforma, executa chamadas ao atualizador do número de debates/comentários do usuário e a requisição dos debates ordenados ao banco de dados.

O *plugin hypeAlive*, responsável pelos comentários e as ações “gostar”/“não gostar”, executa a chamada para a função de calcular a reputação do usuário após as ações “gostar”/“não gostar”.

O *plugin hypeFramework*, responsável por parte do funcionamento do *plugin hypeAlive*, executa também algumas chamadas ao atualizador do número de debates/comentários do usuário.

4.3. Código Fonte

O código fonte para o *plugin reputation* assim como as modificações nos demais *plugins* estão disponíveis sob licença *GPL* no repositório *GIT* do projeto Participatório, que pode ser encontrado no endereço <https://gitlab.c3sl.ufpr.br/participatorio/participatorio>.

5. Conclusão

O método apresentado nesse artigo pode ser utilizado em redes sociais e outras aplicações que necessitem de ordenação de conteúdo, como sites de notícia e comércio eletrônico. Através da classificação dos usuários em diferentes grupos hierárquicos como apresentado na seção 2 também é possível implementar a promoção de usuários no sistema, onde uma vez atingido um valor de reputação limite o usuário ganharia novos privilégios. Ideia semelhante foi implementada no sistema online para ensino e prática do Xadrez [Direne et al. 2004].

Como pode ser observado ao longo do artigo existem algumas variáveis subjetivas a se considerar que dependem do escopo do sistema, como o valor de a_t , os intervalos de atualização para $R(u)$, além dos parâmetros t_{frame} e o próprio critério para consideração da reputação na ordenação dos debates. No Participatório, diversas combinações estão sendo analisadas e os valores definitivos ainda não foram completamente definidos. O objetivo é, através do sistema de reputação, dar destaque aos debates de maior participação, mas sem excluir novos debates em potencial. Além da ordenação dos debates a solução atual exibida no site consiste de duas seções distintas, uma com os debates mais importantes e outra com os debates recém iniciados, reforçando assim o objetivo e compensando por possíveis desvios de valores ideais para as variáveis supracitadas.

Referências

- Billard, M. (2014). Reddit's empire no longer founded on a flawed algorithm. <http://www.outofscope.com/reddits-empire-no-longer-founded-on-a-flawed-algorithm/>.
- Direne, A., Bona, L., Silva, F., Santos, G., Guedes, A., Castilho, M., Sunye, M., Hartmann, C., Neto, P. A., Mello, S., Neto, J. S., and Silva, W. (2004). Conceitos e ferramentas de apoio ao ensino de xadrez nas escolas brasileiras.
- Elgg (2014). Elgg. <http://elgg.org>.
- Ganley, D. and Lampe, C. (2009). The ties that bind: Social network principles in online communities. *Decision Support Systems*, 47(3):266 – 274. Online Communities and Social Network.
- Hofmann, T. (2003). Collaborative filtering via gaussian probabilistic latent semantic analysis. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, pages 259–266, New York, NY, USA. ACM.
- Lampe, C. and Resnick, P. (2004). Slash(dot) and burn: Distributed moderation in a large online conversation space. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 543–550, New York, NY, USA. ACM.
- Reddit (2014). Reddit. <http://www.reddit.com/>.
- Salihefendic, A. (2014). How reddit ranking algorithms work. <http://amix.dk/blog/post/19588>.
- Slashdot (2014). Slashdot. <http://slashdot.org/>.
- SNJ/C3SL (2014). Participatório. <http://participatorio.juventude.gov.br/>.