

Detecção Automática de Bolhas Sociais no Twitter em uma Rede de Usuários de Tecnologia

Bruno Evangelista, Gabriela Batista, Jaqueline Faria de Oliveira

¹Centro Universitário de Belo Horizonte (UNIBH)
Belo Horizonte – MG – Brasil

{brunocarvalho107, gabi.vianall, jaquefari}@gmail.com

Abstract. *The phenomenon of Social Bubbles is known in several areas and research is being done to analyze its impact on society. It is characterized by the limitation of individuals to access information that has an affinity and lack of access to information that differs or differs from those of their interest. Technology professionals are not safe from social bubbles, and to measure the phenomenon for these professionals, this work seeks to detect social bubbles in Twitter using complex network metrics in a network of users related to Programming Languages most used in 2016. We collected 1,226,744 tweets, 896,556 profiles, and their followers. The Homophily by language presented that the profiles connected to the JavaScript programming language present greater Homophily than the other languages analyzed, followed by Python and Java. In this way, it is observed that these communities are more inserted in a social bubble.*

Resumo. *O fenômeno de Bolhas Sociais é conhecido em diversas áreas e pesquisas estão sendo feitas para analisar seu impacto na sociedade. Se caracteriza pela limitação dos indivíduos ao acesso a informações que tem afinidade e a falta de acesso a informações divergentes ou diferentes das de seu interesse. Os profissionais de tecnologia não estão a salvo das bolhas sociais, e com o objetivo de medir o fenômeno para esses profissionais, esse trabalho busca detectar bolhas sociais no Twitter utilizando-se de métricas de redes complexas em uma rede de usuários relacionados às 10 Linguagens de Programação mais utilizadas em 2016. Foram coletados 1.226.744 tweets, 896.556 perfis e seus seguidores. A Homofilia por linguagem apresentou que os perfis ligados à linguagem de programação JavaScript apresentam maior homofilia que as demais linguagens analisadas, seguida por Python e Java. Desta forma observa-se que essas comunidades estão mais inseridas em uma bolha social.*

1. Introdução

Nos últimos anos as redes sociais tem tido ritmo constante de crescimento [Chaffey 2017], sendo que o Facebook possui a maior fatia de mercado em todo o mundo, e no Brasil domina cerca de 76% do mercado [Newman et al. 2017]. As redes sociais apresentam uma larga escala de adesão dos usuários para comunicação e acesso à informação. No Brasil 57% da população recebe informações pelo Facebook, seguido de 46% pelo WhatsApp e 36% pelo Youtube. Segundo Guedes [Guedes 2014], por possuírem papel informativo, essas redes se tornaram uma forma importante de exercer o direito à informação, possibilitando o acompanhamento das ações do governo, ou o acesso a informações de interesse público em geral.

Para lidar com a grande quantidade de informações que as pessoas recebem no seu dia a dia, é comum o emprego de filtros em seu comportamento on-line, de forma consciente ou não [Nikolov et al. 2015]. A aplicação desses filtros tem por objetivo prover informações aos usuários conforme suas preferências, porém, a aplicação de filtros sociais e algorítmicos, pode criar e fortalecer a polarização do acesso a novas informações, estreitando a quantidade de pontos de vista a que as pessoas são expostas. Dessa maneira, as informações fornecidas são selecionadas de acordo com o círculo social no qual o indivíduo está inserido, minimizando a possibilidade do mesmo encontrar um ponto de vista contrário ao seu. A tecnologia que ajuda o usuário a encontrar conteúdo relevante na Internet está criando uma “bolha” em torno das pessoas, pois o usuário sempre recebe informações que reforçam seu ponto de vista [CAPELAS 2017].

As comunidades nas redes sociais vem sendo fonte de diversos estudos, inclusive para identificação de bolhas sociais. O Twitter é hoje uma grande fonte de informação para o estudo das comunidades e seu comportamento. Dentre os estudos, destacam-se os estudos para identificação de comunidades [Soares et al. 2014], Homofilia [Colleoni et al. 2014] política, estudos de Bolhas Sociais políticas [Caetano et al. 2016] e acesso a notícias [Nikolov et al. 2015].

Os profissionais de tecnologia também formam uma grande comunidade nas redes sociais e apresentam muitas vezes preferências semelhantes, por Sistemas Operacionais, Plataformas de Desenvolvimento, Áreas de Atuação e mesmo Linguagens de Programação. Haverá, também nessas comunidades, a presença do fenômeno de Bolha Social? Com o objetivo de responder a esta pergunta, desenvolveu-se nesse trabalho a análise de uma rede de usuários do Twitter que possuem comentários sobre linguagens de programação, a relação desses usuários e seus seguidores. Mediu-se a exposição destes a conteúdos diversos sobre linguagens de programação, com o objetivo final de medir a propensão de existência de Bolhas Sociais. Para isso foram investigados usuários que realizaram postagens sobre as 10 linguagens de programação mais populares em 2016 [Diakopoulos and Cass 2016] apresentadas na Figura 1.

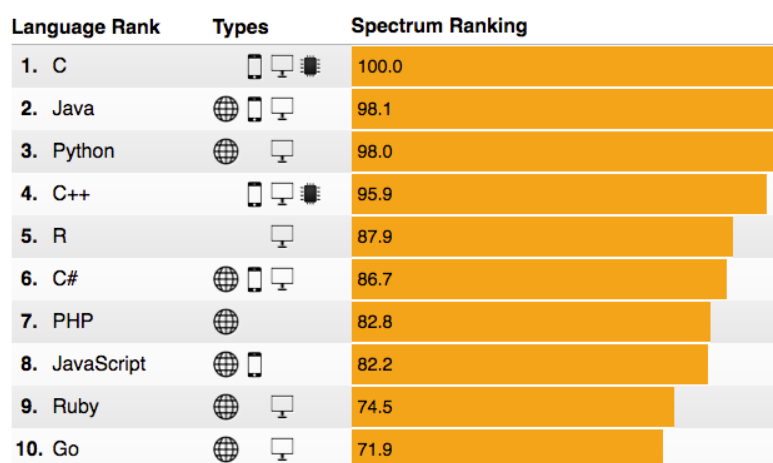


Figura 1. The Top Programming Languages 2016 [Diakopoulos and Cass 2016].

Foram coletados 1.226.744 tweets, 896.556 perfis e seus seguidores. A Homofilia por linguagem apresentou que os perfis ligados à linguagem de programação JavaScript

apresentam maior Homofilia que as demais linguagens analisadas, seguida por Python e Java.

As próximas sessões estão estruturadas da seguinte maneira. A seção 2 apresenta os trabalhos relacionados. A seção 3 traz informações sobre as características do Twitter e da área de *Text Mining* para dar suporte ao entendimento do estudo. A seção 4 apresenta a metodologia desenvolvida para a extração dos dados e obtenção dos resultados. A seção 5 apresenta a análise e avaliação dos resultados dos dados coletados. Por fim, a seção 6 apresenta a Conclusão e as propostas para trabalhos futuros.

2. Trabalhos Relacionados

Nesta seção são apresentados os trabalhos relacionados com o tema desse artigo. São apresentados trabalhos sobre identificação de Comunidades e Bolhas Sociais nas Redes Sociais On-Line.

No trabalho “Measuring On-line Filter Bubbles” de 2015 [Nikolov et al. 2015], Nikolov e seus colaboradores analisaram um conjunto de dados da Universidade de Indiana contendo postagens e registros de pesquisa na rede social Twitter e conteúdos pesquisados no mecanismo de pesquisas da América Online (AOL) e concluíram que, de forma geral, as pessoas acessam informações de um conjunto de fontes significativamente mais “rasas” por meio das mídias sociais, comparado a uma pesquisa mais tradicional. Nesse artigo, o autor discute sobre a existência de bolhas sociais coletivas, que são grupos que se relacionam com outros e sobre a existência de bolhas sociais individuais, aquelas que não se relacionam com outros grupos.

Um estudo sobre encontrar comunidades em redes complexas realizado por Soares, Oliveira e Brito [Soares et al. 2014], descreve a detecção de comunidades em redes complexas em geral, através de um algoritmo denominado FastGreedy. Após detectar as comunidades, foi aplicado um algoritmo de agrupamento (K-means) em cada grafo e realizada a análise desses componentes. Após a realização dos estudos, foi possível aplicar o algoritmo de clustering, encontrando agrupamentos próximos aos detectados pelas outras métricas.

O trabalho publicado por Caetano (2016), aplica o cálculo da Homofilia a uma rede coletada no Twitter contendo perfis de pessoas politicamente liberais ou conservadoras. No trabalho, os autores comparam seus resultados com o esperado para redes “off-line”, descobrindo que os valores de segregação ideológica encontrados são compatíveis com os presentes em relações interpessoais [Caetano et al. 2016].

3. Métricas de Redes complexas

Nesta seção são apresentados os conceitos teóricos necessários ao desenvolvimento desse projeto.

3.1. Redes Complexas

Uma rede pode ser definida genericamente como um conjunto de elementos que mantém conexões entre si [Brandão et al. 2007]. Na matemática, essas redes são reconhecidas como grafos, seus elementos são os vértices e suas conexões as arestas. Já na Ciência da Computação, segundo o autor, os elementos são conhecidos como nós e suas conexões como ligações.

Uma rede complexa é definida como uma estrutura que não possui um padrão regular, sendo compostas por estruturas complexas que possuem certo grau de imprecisão. Para solucionar este problema, ocorre a analogia entre a sua estrutura e a disposição de um grafo, com a finalidade de delimitar seu escopo [Barabási 2009].

Diversas métricas são utilizadas para caracterizar uma rede. Métricas de identificação de agrupamentos dentro de uma rede possibilitam a medida da aglomeração de conexões na rede [Blondel et al. 2008]). As medidas de centralidade indicam o grau de conectividade direta entre os nós de uma rede [Brandão and Parreiras 2010], dentre as métricas conhecidas na literatura, destacam-se a Centralidade de Grau e Betweenness [Vicente 2017].

Homofilia

O artigo de Mcpherson (2001), descreve o princípio da homofilia, que indica a tendência de indivíduos criarem laços com pessoas que possuam características e comportamentos sociais semelhantes ao utilizarem redes sociais [Mcpherson et al. 2001].

De acordo com Monge (2003), esse fenômeno já é conhecido e estudado também nos trabalhos "Similarity-Attraction hypothesis" de Byrne (1971) e "Theory of self-categorization" de Turner (1987), sendo esses fundamentais para dar suporte à teoria da homofilia [Monge and Contractor 2003].

Segundo [Colleoni et al. 2014], a homofilia de indivíduos de uma comunidade pode ser calculada através da Equação 1.

$$H_i = \frac{s_i}{s_i + d_i} \quad (1)$$

Na qual H_i é o valor da homofilia encontrado nesse grupo, s_i é o número de conexões que conectam os indivíduos dessa comunidade, ou seja, conexões homogêneas e d_i é o número de conexões que ligam os indivíduos desse conjunto, (grupo i), com indivíduos de outras comunidades (conexões heterogêneas).

Segundo trabalho publicado por Currarini et al. (2009), a fórmula mencionada não traduz a ideia de como um grupo tendencioso é comparado ao quão faccioso ele poderia ser potencialmente, apesar de fornecer informações interessantes [Currarini et al. 2009].

Para resolver esse problema, o autor sugere o uso da fórmula de Coleman (1958), "Inbreeding Homophily" (Equação 2), que realiza a normalização da homofilia pelo potencial de um grupo a ser tendencioso.

$$IH_i = \frac{H_i - w_i}{1 - w_i} \quad (2)$$

Nessa equação, IH_i é o indicador de Inbreeding Homophily, H_i é o valor calculado para a homofilia utilizando na Equação 1, e w_i é a probabilidade da ocorrência de indivíduos do grupo i, que consiste no total de indivíduos desse grupo dividido pelo total de indivíduos de uma rede T. Caso o valor de H_i seja maior que w_i , então existe homofilia. Quando ocorre o inverso, ou seja, o valor de w_i for maior, ocorre a heterofilia.

Ainda segundo o autor, nesse caso, a condição de homofilia e de heterofilia pode ser representada da seguinte forma:

$$IH_i > 0 \rightarrow \text{homofilia}$$

$$IH_i < 0 \rightarrow \text{heterofilia}$$

As medidas de Homofilia apresentadas nas Equações 1 e 2 serão utilizadas para medir a tendência de formação de Bolhas Sociais nas comunidades de Linguagens de Programação, que é a proposta deste trabalho.

4. Metodologia

Para realizar este trabalho foi feita a análise da rede formada pelos relacionamentos entre usuários a partir da coleta de tweets sobre 10 linguagens de programação distintas.

Foi adotado um processo metodológico que se dividiu em 6 passos. Esse processo inclui coleta de dados, tratamento dos dados, caracterização e análise de bolhas sociais. Os passos estão representados na Figura 2.

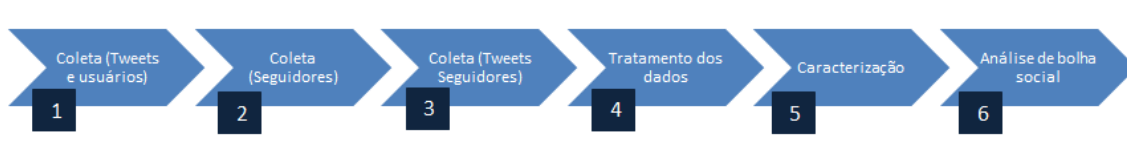


Figura 2. Processo adotado na metodologia.

A etapa 1 consiste na coleta de dados do Twitter. A Coleta foi realizada utilizando a biblioteca Tweepy¹, que implementa os métodos da API do Twitter na linguagem Python. Para busca dos tweets foi criada uma *bag of words* contendo termos relacionados às linguagens a serem pesquisadas. Esses termos são apresentados na Tabela 1. A coleta dos tweets ocorreu de 21 de maio a 30 de junho de 2017. Essa coleta foi realizada através de *streaming*, ou seja, os dados foram obtidos em tempo real através da API utilizada.

Na etapa 2 e 3 foram coletados, respectivamente, os seguidores dos usuários que realizaram as postagens e seus tweets. Essa coleta foi feita nos meses de julho a novembro de 2017. A diferença de período se justifica pela limitação da API do twitter, o que torna lenta a coleta de dados. Foram coletados os tweets dos usuários da base que postaram algum twitter referente à uma das linguagens de programação analisadas.

Na etapa 4, os dados foram tratados de forma a possibilitar a efetiva execução das etapas 5 e 6, respectivamente. Foram considerados somente os dados dos "seguidores dos usuário" que continham a menção a alguma das linguagens de programação analisadas, os demais dados foram desconsiderados nas análises.

Na etapa 5 foi feita uma caracterização da rede gerada a partir dos usuários coletados. Os vértices representam os usuários coletados e as arestas as interações "seguindo / seguidor", gerando um grafo direcionado, no qual as arestas saem do "seguidor" e incidem no "seguido".

¹<http://www.tweepy.org/>

Tabela 1. Termos utilizados para a busca dos tweets

Linguagens	Termos
C	clang; clanguage; cprogramming; ansic; learningansic; codingc; c programming; c language; coding c; learning c; ansi c
Java	javalang; javalanguage; java; learning java; coding java; java8; java7; java language; java lang
Python	csharp; csharp language; csharp programming; learning csharp; c#; c# programming; learning c#; c# language
C++	c++ language; c++ programming; cplusplusprogramming; learning cplusplus; learningc++; c plus plus; cpp; cpp programming; learning c++; learning cpp
R	rlang; rlanguage; rprogramming; learningr; codingr; rdevelopment; r lang; r language; r programming; learning r; coding r; r development
C#	csharp; csharp language; csharp programming; learning csharp; c#; c# programming; learning c#; c# language
PHP	php; php language; phplanguage; php programming; coding php; learning php; learningphp; phplang; php7; php5
JavaScript	javascript; javascript language; javascriptlang; jscode; jslanguage; codingjs; js programming; js; nodejs; angularjs; reactjs; vuejs; vue.js
Ruby	rubylang; ruby language; ruby programming; rubyprogramming; learnruby; codingruby; rubydevelopment; ruby; ruby lang
GO	goprogramming; go programming language; golang; golanguage; learnolang; godevelopment; learning go; learning golang

A partir da base inicial, definiu-se um conjunto de "Comunidades de Linguagens de Programação" às quais cada usuário pertence. Considera-se, no contexto desse trabalho, que um usuário pertence a uma Comunidade de Linguagem de Programação caso tenha mencionado em seus tweets alguma das linguagens analisadas. Esse fato possibilita que um usuário possa participar de mais de uma Comunidade. Métricas de Centralidade foram utilizadas para caracterização dos usuários da rede e Modularidade para a medida de agrupamentos de nós da rede.

Por fim, na etapa 6, foi medida a homofilia dos usuários das Comunidades de Linguagens de Programação com o intuito de identificar a similaridade entre os usuários. Na rede de usuários gerada, foram aplicadas as métricas de Homofilia (Equação 1 e Equação2) para verificar a tendência dos usuários das comunidades de conectarem-se com pessoas de interesses semelhantes.

5. Resultados

Esta seção fará a caracterização detalhada dos dados coletados, apresentando as informações relativas à quantidade de registros e como os mesmos se organizam. Serão apresentados ainda a caracterização da rede analisada e os resultados da homofilia da rede de usuários por Comunidades de Linguagens de Programação.

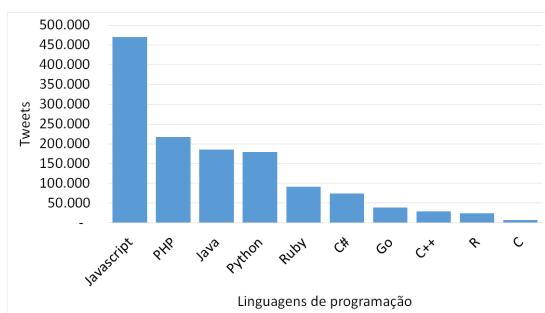
5.1. Conjunto de Dados

O processo de coleta consistiu nas etapas 1, 2 e 3 da metodologia. Foram obtidos 1.226.744 tweets e 896.556 usuários distintos que postaram sobre as linguagens de programação analisadas.

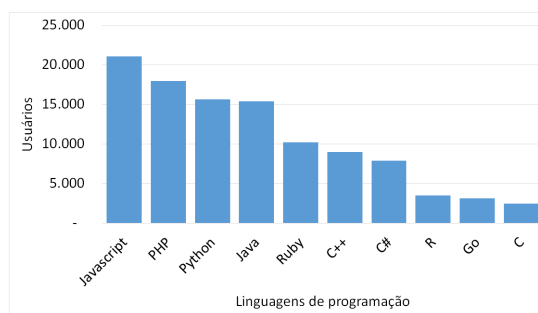
Com o objetivo de avaliar a popularidade das linguagens no Twitter, foram analisadas as distribuições de tweets por linguagem (Figura 3(a)) e usuários distintos por linguagem (Figura 3(b)).

Em comparação à popularidade das linguagens publicada pelo IEEE Spectrum em 2016 [Diakopoulos and Cass 2016], a quantidade de coleta dos tweets por linguagem demonstra um resultado diferente. A linguagem Javascript é a mais popular, enquanto o levantamento do IEEE mostra a linguagem C nesta posição. É importante notar que existe a possibilidade de um mesmo tweet falar a respeito de mais de uma linguagem, o que torna a soma das menções individuais, independente da quantidade total de tweets.

Ao se analisar a quantidade de usuários distintos que publicaram informações sobre cada linguagem, observa-se uma distribuição semelhante ao total de tweets por linguagem nas duas primeiras posições. Há variação no *ranking* entre Java e Python, as demais linguagens apresentam distribuições distintas.



(a) Tweets coletados por linguagem.



(b) Total de usuários que realizaram publicações por linguagem.

Figura 3. Na Figura 3(a) é apresentado o total de tweets coletados por linguagem. Na Figura 3(b) são apresentados os totais de usuários que realizaram publicações para cada uma da linguagem.

5.2. Rede de usuários

A rede de usuários, formada por seguidos e seguidores que fizeram menção às linguagens de programação, foi analisada. A Figura 4 apresenta a distribuição de frequência de conexões dos usuários com outros usuários que também falam sobre linguagens de programação. Pode-se verificar que a grande maioria dos usuários possui poucas conexões com demais usuários que falam sobre o tema. Cerca de 65% tem somente uma conexão, as demais conexões são formadas por 35% da base, mas representam um valor considerável com mais de 480 mil usuários.

Foi gerado um grafo para representar a rede de usuários, onde os vértices são os usuários e as arestas as interações “seguindo / seguidor”. O grafo é direcionado, no qual as arestas saem do “seguidor” e incidem no “seguido”. A rede possui 896.556 vértices

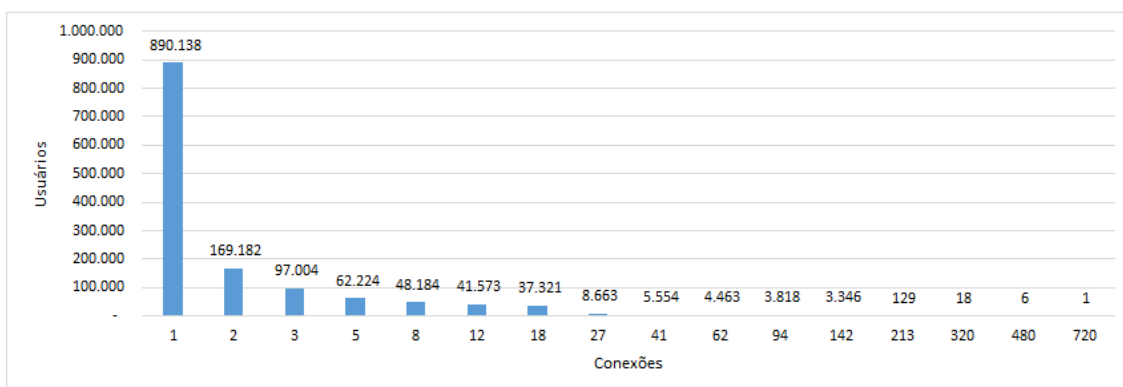
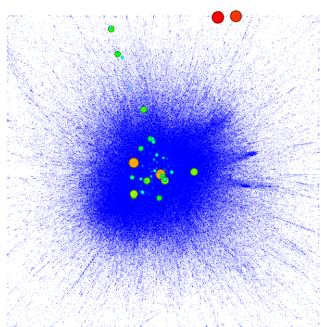


Figura 4. Frequência de conexões por usuário.

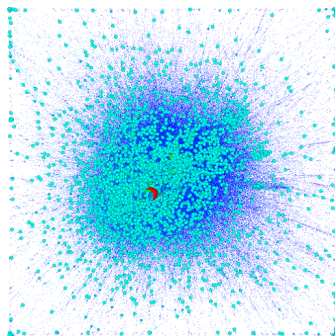
e 1.354.698 arestas e 538 componentes. Essa rede é analisada por meio de métricas de centralidade e de identificação de agrupamentos.

Betweenness

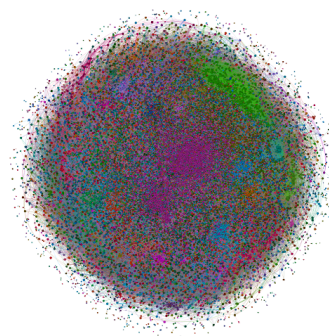
Foi calculada a métrica de Centralidade de Intermediação (Betweenness) ao grafo para verificar a importância dos nós na propagação de informação dentro da rede. Para destacar estes nós, foi configurada uma visualização em que o tamanho e a cor dos nós refletissem esta medida, como pode ser observado na Figura 5(a). Observa-se que a grande maioria dos vértices do grafo possuem um valor Betweenness muito baixo. Estes resultados indicam que uma quantidade muito pequena de usuários têm uma importância muito grande no fluxo de informações na rede coletada.



(a) Métrica Betweenness.



(b) Distribuição de grau.



(c) Comunidades da rede.

Figura 5. As Figuras apresentam os resultados de métricas de redes complexas aplicadas à rede em estudo. Na Figura 5(a) os vértices são coloridos e dimensionados de acordo com seu valor de Betweenness. Na Figura 5(b) também são apresentados os resultados através das cores e dimensão dos vértices, neste caso para representar a distribuição de grau. Na Figura 5(c) é apresentado o resultado da aplicação da métrica Modularidade para identificação de agrupamentos da rede (comunidades), sendo que as comunidades são representadas por diferentes cores.

Distribuição de Graus

A distribuição de graus foi calculada para verificar como as conexões do grafo se encontram distribuídas, para identificar quais os usuários mais conectados e quantas conexões eles possuem. Para obter essas informações foi calculado o grau médio do grafo, que representa a quantidade média de conexões por nós. Esse grau foi aferido em 3,039. Mais uma vez, a visualização do grafo foi configurada, desta vez para refletir o grau de cada vértice. A visualização é demonstrada na Figura 5(b) e mostra que muitos vértices possuem poucas conexões, o que justificaria o baixo valor do grau médio obtido para o grafo.

Modularidade

A métrica modularidade foi utilizada para detectar os agrupamentos (comunidades) existentes no grafo. O cálculo foi realizado para identificar esses grupos, sendo aplicado à toda rede, conforme representado na Figura 5(c).

A métrica de modularidade identificou 1.003 comunidades, obtendo um valor 0,735, que indica a existência de comunidades mais densamente conectadas internamente que externamente, ou seja, existem comunidades em que os indivíduos se conectam mais entre si do que com indivíduos externos. A maioria das comunidades possui poucos indivíduos, enquanto o número de comunidades com mais que 30 mil indivíduos é extremamente baixo.

5.3. Análise de Bolhas Sociais

Com o objetivo de analisar se há presença de Bolhas Sociais entre as Comunidades de usuários de Linguagens de Programação, foi aplicada a métrica de homofilia, procurando medir a similaridade das conexões entre os usuários.

Para cada Comunidade de Linguagens de Programação, formado por usuários que falaram das mesmas linguagens de programação, analisa-se se os usuários pertencentes às comunidades estão conectados a usuários que também estão em outras comunidades de linguagens. Para isso verifica-se as conexões homogêneas, conexões em que ambos os usuários falam sobre a linguagem, e conexões heterogêneas, conexões em que os usuários falam de linguagens distintas.

Foi feito o cálculo da homofilia dos usuários de cada Comunidade de Linguagem de Programação, representado pelo valor de H_i . O indicador *inbreeding homophily* (IH_i) também foi calculado para demonstrar um comparativo dos resultados. O valor de W_i é a probabilidade da ocorrência de indivíduos do tipo i e, quando H_i é maior que W_i , há homofilia na comunidade.

Pode-se verificar, conforme demonstrado na Figura 6, que os valores de H_i e IH_i estão muito próximos, embora os valores de IH_i sejam superiores aos valores de H_i para a maioria das Comunidades. Ambos os valores, H_i e IH_i , demonstram que todas as Comunidades de Linguagens de Programação apresentam homofilia e nenhuma apresenta heterofilia. Isso quer dizer que um grande número de usuários das comunidades se conectam a outros usuários da mesma comunidade.

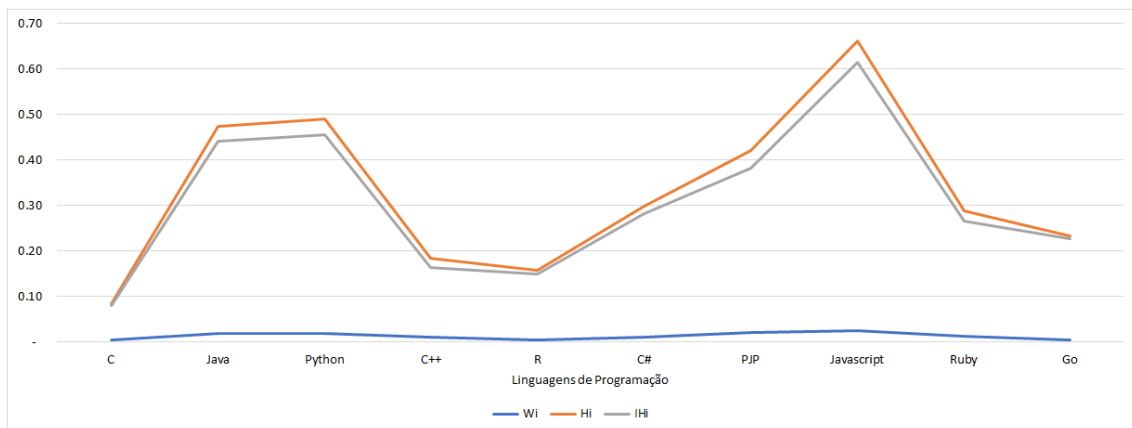
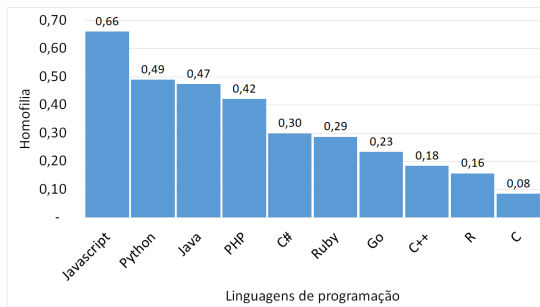
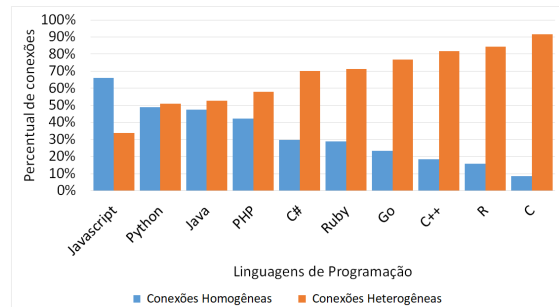


Figura 6. Demonstrativo dos resultados do cálculo da homofilia (H_i e IH_i) e a probabilidade de ocorrência de indivíduos da comunidade (W_i) por cada Comunidade de Linguagem de Programação.

Nota-se que o valor da homofilia para a comunidade de linguagem Javascript é o maior apresentado entre as comunidades, como demonstrado na Figura 7(a), seguida pelas comunidades Python, Java e PHP com valores superiores a 0,4 de homofilia. As demais apresentam homofilia, mas em valores mais baixos.



(a) Homofilia



(b) Conexões homogêneas e heterogêneas

Figura 7. Na Figura 7(a) é apresentada a medida de homofilia por Comunidade de Linguagem de Programação. A Figura 7(b) mostra um comparativo de conexões homogêneas e heterogêneas dos usuários por Comunidade de Linguagem de Programação.

Pode-se perceber na Figura 7(b) que é proporcional o aumento das conexões heterogêneas e a diminuição da homofilia apresentada por comunidade (Figura 7(a)). E, apesar dos resultados apontarem que todas as comunidades contém conexões homogêneas, a linguagem Javascript é a única que possui mais conexões homogêneas que heterogêneas. As comunidades Python, Java e PHP apresentam um percentual de conexões heterogêneas e homogêneas balanceado. As demais linguagens apresentam valores discrepantes de conexões, sendo sua maioria conexões heterogêneas.

Pode-se presumir que a comunidade JavaScript está mais inserida em um cenário de Bolha Sociais que as demais comunidades. Das Comunidades de Linguagens de Programação analisadas, a Comunidade da Linguagem C foi a que menos demonstrou

conexões homogêneas, tendo conseqüentemente a menor probabilidade de estar envolvida numa Bolha Social.

5.4. Discussões

Os valores obtidos sobre a homofilia e sobre as conexões homogêneas e heterogêneas dos usuários das comunidades apontam na mesma direção, especialmente para a linguagem Javascript, pois apresenta a maior diferença entre conexões homogêneas e heterogêneas se comparada às de outras linguagens (Figura 7(b)). Por este motivo, os usuários da comunidade da linguagem Javascript tendem a estar mais inseridos em bolhas sociais que os das demais comunidades.

Os resultados são similares aos encontrados por [Halberstam and Knight 2013], embora o mesmo tenha efetuado a medição da homofilia dentro de dois grupos de eleitores americanos, encontrando valores numericamente próximos ao encontrados no presente trabalho.

6. Conclusão

Com a aplicação dos métodos expostos neste trabalho, foi possível fazer a análise da rede estudada culminando na medição da homofilia dos usuários das comunidades de linguagens de programação. A rede foi obtida a partir de usuários do Twitter que mencionaram alguma das linguagens de programação analisadas e seus seguidores que também postaram sobre essas linguagens. A rede demonstra um baixo valor de centralidade de grau e poucos usuários da rede possuem alto valor de Betweenness.

Os valores obtidos a partir da métrica de homofilia apontam para a possível existência de bolhas sociais, notadamente no grupo da linguagem Javascript, que possui um índice muito elevado de conexões homogêneas em relação às demais. Sua alta homofilia e a disparidade em suas conexões, se comparadas com as de outras linguagens, são os maiores indicativos da presença de bolhas sociais dentro do grupo.

Deve-se levar em consideração que os resultados apresentados foram obtidos a partir de uma amostra de tweets fornecidos pela plataforma do Twitter, e que pode não representar fielmente a realidade.

Apesar de o foco do trabalho ter sido a detecção de bolhas sociais nas Comunidades de Linguagens de Programação, a metodologia apresentada poderia estender-se aos mais variados assuntos, não limitando-se ao que foi tratado neste documento. O presente trabalho buscou detectar bolhas sociais dentro dos grupos de usuários do Twitter que falam sobre Linguagens de Programação, porém, ainda há a possibilidade do desenvolvimento de pesquisas buscando caracterizar tais bolhas, seus usuários e fluxo de informação.

7. Referências

- Barabási, A.-L. (2009). Scale-free networks: a decade and beyond. *science*, 325(5939):412–413.
- Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

- Brandão, W. C. and Parreiras, F. S. (2010). Uma abordagem baseada em métricas de redes complexas para o estabelecimento do grau de influencia de termos em documentos. *ENCONTRO NACIONAL DE PESQUISA EM CIÊNCIA DA INFORMAÇÃO INOVAÇÃO E INCLUSÃO SOCIAL*, 11.
- Brandão, W. C., Parreiras, F. S., and Silva, A. B. d. O. (2007). Redes em ciência da informação: evidências comportamentais dos pesquisadores e tendências evolutivas das redes de coautoria. *Informação & Informação*.
- Caetano, A. J., Lima, H. S., and Santos, Mateus Freira, M.-N. H. T. (2016). **Utilizando análise de sentimentos para definição da homofilia política dos usuários do twitter durante a eleição presidencial americana de 2016.**
- CAPELAS, B; MANS, M. (2017). Saiba como os algoritmos das redes sociais podem mudar a política.
- Chaffey, D. (2017). Global social media research summary 2017.
- Colleoni, E., Rozza, A., and Arvidsson, A. (2014). Echo chamber or public sphere? predicting political orientation and measuring political homophily in twitter using big data. *Journal of Communication*, 64(2):317–332.
- Currarini, L., Jackson, M. O., and Pin, P. (2009). An economic model of friendship: Homophily, minorities, and segregation. *Journal of Communication*, 64(2):317–332.
- Diakopoulos, N. and Cass, S. (2016). Interactive: The top programming languages 2016.
- Guedes, T. M. (2014). As redes sociais-facebook e twitter-e suas influências nos movimentos sociais.
- Halberstam, Y. and Knight, B. (2013). Are social media more social than media? measuring ideological homophily and segregation on twitter. Technical report, working paper.
- Mcpherson, M., Smith-Lovin, L., and Cook, L. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444.
- Monge, P. R. and Contractor, N. (2003). *Theories of Communication Networks*.
- Newman, N., Fletcher, R., Kalogeropoulos, A., Levy, D. A., and Nielsen, R. K. (2017). Reuters institute digital news report 2017.
- Nikolov, D., Oliveira, D. F., Flammini, A., and Menczer, F. (2015). Measuring online social bubbles. *PeerJ Computer Science*, 1:e38.
- Soares, I., Oliveira, C. S., and de Moura Brito, J. A. (2014). Um estudo do problema de detecção de comunidades em redes. *Sistemas & Gestão*, 9(4):566–574.
- Vicente, R. (2017). Redes complexas.