Partitioning and Social Scaling of Political Debates using Signed Bipartite Graphs

Sedat Gokalp
Computer Science
Arizona State University
Sedat.Gokalp@asu.edu

M'hamed Temkit
Mathematical Statistics
Arizona State University
Mhamed.Temkit@asu.edu

Hasan Davulcu
Computer Science
Arizona State University
Hasan.Davulcu@asu.edu

I. Hakki Toroslu
Computer Engineering
Middle East Tech. Uni.
Toroslu@ceng.metu.edu.tr

ABSTRACT

Blogosphere plays an increasingly important role as a forum for public debate. In this paper, given a mixed set of blogs debating a set of political issues from opposing camps, we use signed bipartite graphs for modeling debates, and we propose an algorithm for partitioning both the blogs, and the issues (i.e. topics, leaders, etc.) comprising the debate into binary opposing camps. Simultaneously, our algorithm scales both the blogs and the underlying issues on a univariate scale. Using this scale, a researcher can identify moderate and extreme blogs within each camp, and polarizing vs. unifying issues. Through performance evaluations we show that our proposed algorithm provides an effective solution to the problem, and performs much better than existing baseline algorithms adapted to solve this new problem. In our experiments, we used both real data from political blogosphere and US Congress records, as well as synthetic data which were obtained by varying polarization and degree distribution of the vertices of the graph to show the robustness of our algorithm.

I INTRODUCTION

Blogosphere plays an increasingly important role [1] as a forum of public debate, with knock-on consequences for the media, politics, and policy. Hotly debated issues span all spheres of human activity; from liberal vs. conservative politics, to extremist vs. counter-extremist religious debate, to climate change debate in scientific community, to globalization debate in economics, and to nuclear disarmament debate in security. There are many applications [2–6] for recognizing politically-oriented sentiment in texts. Previous work [7] studied linking patterns and discussion topics of political bloggers by measuring the degree of interaction between liberal and conservative blogs, and to uncover their differences. In this paper, given a mixed set of blogs debating a set of related issues from two opposing camps, we propose an algorithm to determine (i) which blog lies in which camp, (ii) what are the contested issues, and, (iii) who are mentioned as the key individuals within each camp.

Bipartite graphs [8–10] have been widely used to represent relationships between two sets of entities. We use bipartite graphs to model the relationships between blogs and issues (i.e. topics, individuals, etc.) mentioned within blogs. We use signed weighted edges to represent opinion strengths, where positive edges denote support, and negative edges denote opposition between a blog and an issue.

We develop algorithms to solve the following problems on signed bipartite graphs modeling blog debates:

1. **Partitioning** of both the blogs, and the underlying issues mentioned in blogs, into two opposing camps;
2. **Scaling** of both the blogs and the underlying issues on a univariate scale such that the position of a vertex is closer to the positions of the vertices it is connected with positive edges, and further away from the positions of the vertices it is connected with negative edges.

Using this scale, a researcher can identify both the moderate and extreme blogs within each camp, and the polarizing vs. unifying issues. Partitioning and scaling help a researcher to better understand the structure of a social, political or economic debate, or even the details of an emerging geopolitical conflict in the world. While extremist ends of a scale, may represent blogs with irreconcilable viewpoints, in some cases, moderate blogs may represent viewpoints that are more amenable to engage in a constructive dialog through a set of unifying issues. Moderates may sympathize with some of the claims and grievances of the other side. Longitudinal analysis using our proposed algorithms could reveal interesting dynamics, such as, moderates from opposing camps could be in the process of forming a coalition by making the necessary compromises to reach a consensus. All the while, moderates may be alienating extremists in their own camps who may choose to focus on polarizing issues only, and lash out violent or demonizing rhetoric on everyone else who do not share their exclusivist viewpoints.

To the best of our knowledge, simultaneous scaling and partitioning on signed weighted bipartite graphs has not been studied in the literature, and this paper is the first attempt to introduce the problem and provide an effective solution and evaluation strategies.

Major contributions of this paper are: (1) an iterative algorithm, named Alternatingly Normalized CO-HITS (ANCO-HITS), to propagate the scores on a signed bipartite graph to solve the partitioning and scaling problems described above; (2) a convergence proof for the proposed ANCO-HITS algorithm; (3) definition of a new coefficient to measure *structural equilibrium* for signed bipartite graphs using the multiplicative transitivity property presented in [11] exemplified by the phrase *the enemy of my enemy is my friend*; and (4) performance evaluations of blog/issue partitioning and scaling algorithms using synthetic data sets which were obtained by varying polarization and degree distribution of signed bipartite graphs, and analysis with two real data sets: (i) partitioning and scaling of Republicans/Democrats and their *roll call votes*[1] based on the 111^{th} US Congress voting records, and (iii) partitioning and scaling of the top 22 liberal and conservative blogs, and the most influential individuals mentioned in these blogs. In our experiments, variance in polarization relates to the distributions of the ratio of vertices corresponding to extremes vs. moderates.

Alongside our proposed ANCO-HITS, we also evaluated two baseline algorithms, namely CO-HITS [8] and spectral clustering [12]. Although Co-HITS was designed for scaling unsigned bipartite graphs, it can be directly applied for scaling signed bipartite graphs, and partitioning by considering the signs of vertex values. Spectral clustering algorithm was designed for partitioning of graphs, and it can also produce a scale by using the component values of the eigenvector associated with the second smallest positive eigenvalue of the graph Laplacian [13, 14]. Our experiments showed that the ANCO-HITS algorithm is the only robust algorithm in the presence of variance in polarization and vertex degrees.

The rest of this paper is organized as follows. We review related work in Section 2. Section 3 presents the problem formulation. In Sections 4 and 5, we present two baseline algorithms. In Section 6, we present the ANCO-HITS algorithm with its convergence proof. Section 7 presents the definition for structural equilibrium. We report experimental evaluations in Section 8, and conclude in Section 9.

[1] `http://thomas.loc.gov/home/rollcallvotes.html`

II RELATED WORK

Scaling vertices of a graph based on the network structure rather than individual properties has been of great interest for more than a decade. Two most well-known algorithms are the PageRank [15] and the HITS [16] algorithms. They were designed to rank the vertices of graphs with positive weighted edges. Spectral analysis show that both PageRank and HITS algorithms converge. An important distinction between the two algorithms is that; the HITS algorithm provides two different types of rankings corresponding to *hubs* and *authorities*, whereas PageRank provides only a single ranking.

Many data types from data mining applications can be modeled as bipartite graphs, examples include terms and documents in a text corpus, customers and items purchased in market basket analysis and bloggers writing about current issues.

Based on variations of HITS and PageRank, many researchers have proposed algorithms. In [8], the authors propose a modification of the HITS algorithm to work on bipartite graphs called CO-HITS. The main difference between HITS and CO-HITS is that; HITS provides two scores for each vertex, whereas CO-HITS provides one score for each type of vertex. In this paper, we use CO-HITS as one of the baseline algorithms, and in order to overcome its deficiencies, we extend it with normalization steps.

The *clustering coefficient* was first introduced in [17] to measure how much multiplicative transitivity property the graph exhibits, which reflects the tendency of the vertices to form small groups. In [11], authors define a new coefficient using the multiplicative transitivity for signed graphs to measure structural equilibrium. In this paper, we define another coefficient through multiplicative transitivity for signed bipartite graphs.

Data mining methods such as clustering have been used quite extensively for exploratory data mining applications [18, 19]. Clustering analysis [20] provides a partitioning of the data into subsets, called clusters such that the objects in a cluster are more similar than those in distinct clusters. Spectral clustering [12,13,21] is a powerful clustering method that is able to outperform K-means clustering [22], which has a major drawback of not being able to separate clusters that are non-linearly separable in input space [21]. The method is based on computing the

eigenvalues of the normalized version of the graph Laplacian, and has theoretical connections with the normalized cut of the graph. In particular when clustering a bipartite graph into two balanced clusters, the second smallest positive eigenvalue [13] is the solution to the normalized cut of the graph. In recent years, several authors have used spectral clustering to analyze bipartite graphs [10]. Furthermore, some work has been done to take into account a signed adjacency matrix by using an augmented adjacency matrix [23]. In this paper, spectral clustering was also used as one of our baseline methods for partitioning and scaling signed bipartite graphs.

III PROBLEM FORMULATION

1 COSCALING FOR SIGNED BIPARTITE GRAPHS

Given

- $G = (U \cup V, A)$ is a bipartite graph consisting of two disjoint sets of vertices U and V, and a signed adjacency matrix A
- $U = \{u_1, u_2, \ldots, u_m\}$, a set of m vertices
- $V = \{v_1, v_2, \ldots, v_n\}$, a set of n vertices
- $A \in \mathbb{R}^{m \times n}$, where a_{ij} represents the signed edge between u_i and v_j

Find

- $X = (x_1, x_2, \ldots, x_m)$, where $x_i \in \mathbb{R}$ is the assigned value of the vertex u_i
- $Y = (y_1, y_2, \ldots, y_n)$, where $y_i \in \mathbb{R}$ is the assigned value of the vertex v_i

such that

- $sgn(x_i)$ and $sgn(y_i)$ shall determine the *polarity* of the vertices i.e. -1 and $+1$ as the opposing polarities
- x_i value for a vertex u_i should be closer to the y_j values of the vertices that it supports (connects positively), and further away from the y_k values of the vertices that it opposes (connects negatively). The magnitudes of x_i and y_j denote the *extremity* of the nodes u_i and v_j. i.e. magnitudes closer to 0 meaning *more moderate* and larger magnitudes meaning *more extreme*.

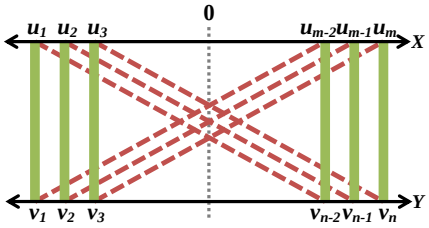


Fig 1. Perfectly polarized bipartite graph

Figure 1 depicts a perfectly polarized bipartite graph. The two axes X and Y represent the univariate scale for the nodes in U and V. The vertices to the right of zero have positive values, and the vertices to the left have negative values on the scale. A green solid line between the nodes u_i and v_j represents support, and a red dashed line represents opposition.

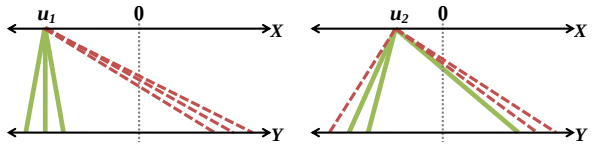


Fig 2. Extreme vs. Moderate vertices

Figure Figure 2 shows an example of two vertices; u_1 being extreme and u_2 being more moderate. u_1 supports the vertices of same polarity, and opposes the vertices of the opposite polarity. However, u_2 has mixed support and opposition.

Although partitioning algorithms can be utilized to detect the polarity of vertices, it is not possible to distinguish extremes from moderates. Scaling overcomes this problem and makes it possible to compare two vertices of same polarity. In this paper, we are not only able to compare pairs of vertices, but also provide the exact locations on the scale, therefore providing valuable information about the shape of the distribution as well.

To solve this co-scaling problem, we present two baseline methods. The first one is a common modification [10, 24, 25] of the well-known *Spectral Clustering* approach to work on graphs with signed edges. The second one is the CO-HITS [8] algorithm, that is a modification of the well-known HITS algorithm, designed for bipartite graphs.

Finally, we compare these baseline methods with a novel algorithm we developed for co-scaling problem, named Alternatingly Normalized CO-HITS (ANCO-HITS).

IV SPECTRAL CLUSTERING

Spectral clustering [13] uses spectral graph theory, which is the study of graphs using linear algebra methods. In this context, for a given graph, its edge set is represented by an adjacency matrix. The eigenvectors of the normalized Laplacian of the adjacency matrix are used to partition the graph into clusters, where objects in a cluster are more similar than those in distinct clusters. Spectral clustering incorporates the properties of a graph via the adjacency matrix and is able to outperform K-means clustering in many situations, especially in the presence of non-convex groups of data. This method has close connections with the normalized cut [12] of the graph. In particular when clustering a bipartite graph into two balanced clusters, the second smallest positive eigenvalue of the Laplacian matrix [21] is the solution to the problem of minimizing the normalized cut of the graph.

Spectral clustering embeds the data into the subspace of the eigenvectors of the Laplacian. In this paper, we will use spectral clustering to partition both types of vertices of a signed bipartite graph into two polarities and provide a scale for the vertices. We will do this by using the sign and the value of the components of the eigenvector associated with the smallest positive eigenvalue of the Laplacian.

Spectral clustering uses an adjacency matrix with all positive entries. However, our problem assumes a signed adjacency matrix. One of the common techniques to circumvent this problem is to augment the matrix into a bigger matrix [10, 24, 25], such that all entries are positive. The first half of the augmented matrix is reserved for the entries with positive values, and the second half is reserved for the entries with negative values.

Define $\tilde{A} \in \mathbb{R}^{m \times 2n}$ such that

$$\tilde{A} = \left[A^+, A^-\right]$$

where

$$a_{ij}^+ = \begin{cases} a_{ij}, & \text{if } a_{ij} > 0 \\ 0, & \text{otherwise} \end{cases}$$

and similarly

$$a_{ij}^- = \begin{cases} -a_{ij}, & \text{if } a_{ij} < 0 \\ 0, & \text{otherwise} \end{cases}$$

In order to partition and scale the nodes $u_i \in U$ and $v_i \in V$, we define the following matrix:

$$B = \begin{pmatrix} 0_{m \times m} & \tilde{A} \\ \tilde{A}^T & 0_{2n \times 2n} \end{pmatrix}$$

We define the Laplacian of B as $L = D - B$ where D is the diagonal degree matrix and $d_{ii} = \sum_{j=1}^{m+2n} b_{ij}$. We further compute the normalized Laplacian $L_{sym} = D^{-1/2} L D^{-1/2}$. It should be noted here that both L and L_{sym} are positive semi-definite.

Let the eigenvalues of L_{sym} have the values $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{m+2n}$ with associated eigenvectors $v_1, v_2, \ldots, v_{m+2n}$, our univariate scale being the eigenvector v_2.

The first m components of v_2 are set to be the X vector, and the following n components are set to be the Y vector, solutions of the co-scaling problem.

V CO-HITS

In [8], the authors modify the well-known HITS [16] algorithm and propose the CO-HITS algorithm which is used to rank vertices of a bipartite graph. Even though the adjacency matrix has only positive values in the original HITS paper, the theory still holds for adjacency matrices with signed entries.

Algorithm 1 describes the steps of the CO-HITS algorithm for the co-scaling problem.

Data: Adjacency matrix A
Data: Scale vectors X and Y
Initiate $X^{<0>} = (1, 1, \ldots, 1)$;
Initiate $Y^{<0>} = (1, 1, \ldots, 1)$;
repeat
.....Update X;
.....Update Y;
until X vector converges
Alg 1. Iterative update procedure for CO-HITS

The update functions for X and Y are defined as follows:

$$x_i^{<k>} = \sum_{j=1}^{n} a_{ij} y_j^{<k-1>}$$

$$y_j^{<k>} = \sum_{i=1}^{m} a_{ij} x_i^{<k>}$$

and convergence is achieved when

$$\left\| X^{<k>} - X^{<k-1>} \right\|_2 < \epsilon$$

with ϵ a small positive value.

The drawback of this method is its sensitivity to the vertex degrees. Two vertices with same polarities but different degrees will result with the higher degree vertex having a higher score on the scale.

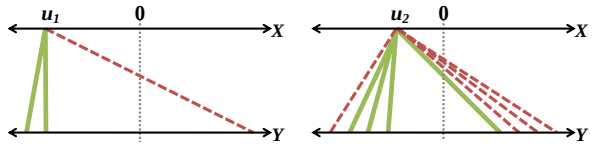


Fig 3. Extremity vs Degree

For example, let us consider two vertices u_1 and u_2 with u_1 having a smaller degree than u_2. However, let u_1 be more polarized than u_2 as shown in Figure 3. In this scenario, the corresponding scale values for u_1 and u_2 should satisfy $|x_1| > |x_2|$. But, this will not be the case with the CO-HITS. This suggests a better algorithm that accounts for the negative impact of degree variation through some normalization mechanism.

VI ALTERNATINGLY NORMALIZED CO-HITS (ANCO-HITS)

According to our problem formulation, the values of the vertices on the scale shall not be sensitive to their degrees, but rather be sensitive to what kind of relations they have with the other set of vertices.

For this purpose, we propose ANCO-HITS algorithm, which introduces a normalization mechanism to address the issue of degree sensitivity of CO-HITS. The proposed method uses the same iteration procedure described in Algorithm 1. The update functions for X and Y are modified such that they are normalized as follows:

$$x_i^{<k>} = \frac{\sum_{j=1}^{n} a_{ij} y_j^{<k-1>}}{\sum_{j=1}^{n} |a_{ij}|}$$

$$y_j^{<k>} = \frac{\sum_{i=1}^{m} a_{ij} x_i^{<k>}}{\sum_{i=1}^{m} |a_{ij}|}$$

Theorem 1. *ANCO-HITS algorithm will converge for any matrix $A \in \mathbb{R}^{m \times n}$, with $|A|$ having non-zero row-sums and column-sums.*

Proof. Let $B \in \mathbb{R}^{m \times m}$ and $C \in \mathbb{R}^{n \times n}$ diagonal matrices with positive entries, where

$$B_{ij} = \begin{cases} \frac{1}{\sum_{j=1}^{n} |a_{ij}|}, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

similarly,

$$C_{ij} = \begin{cases} \frac{1}{\sum_{i=1}^{m} |a_{ij}|}, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

We should note here that both B and C matrices are *symmetric* and *positive definite*. The update rules for x and y vectors can be written in matrix notation as follows:

$$x^{<k>} = BAy^{<k-1>} \quad (3)$$

$$y^{<k>} = CA^T x^{<k>} \quad (4)$$

Therefore,

$$x^{<k>} = (BACA^T)x^{<k-1>} \quad (5)$$

If there exists a vector x^* that the $x^{<t>}$ will converge in direction, it has to satisfy the equation:

$$cx^* = (BACA^T)x^* \quad (6)$$

Even though this is an eigenvalue equation, the eigenvalues may not be real, because the matrix $(BACA^T)$ is not symmetric. But if we multiply each side of the equation with $B^{-1/2}$, which exists since B is positive definite, we will get:

$$cB^{-1/2}x^* = B^{1/2}ACA^TB^{1/2}B^{-1/2}x^* \quad (7)$$

Define $M \in \mathbb{R}^{m \times m}$ to be $M = B^{1/2}ACA^TB^{1/2}$ and $z \in \mathbb{R}^{m \times 1}$ to be $z = B^{-1/2}x^*$, we will get

$$cz = Mz \quad (8)$$

which is again an eigenvalue equation. However, in this case M is a symmetric matrix, and can be shown to be positive semi-definite with z as an eigenvector c as an eigenvalue. The M matrix has a set of m eigenvectors that are all unit vectors and all mutually orthogonal; that is, they form a *basis* for the space $\mathbb{R}^m$.

Let us denote the eigenvalues of the M matrix by $c_1, c_2, \ldots, c_m$ sorted in such a way that $c_1 \geq c_2 \geq \ldots \geq c_m \geq 0$, with the eigenvectors $z_1, z_2, \ldots, z_m$ respectively.

Using Equation (5), we can write a compact form for the k^{th} update iteration of x as follows:

$$x^{<k>} = (BACA^T)^k x^{<0>} \tag{9}$$

We can rewrite the above equation in terms of M matrix

$$x^{<k>} = B^{1/2} M^k B^{-1/2} x^{<0>} \tag{10}$$

Any vector $v \in \mathbb{R}^m$ can be written as a linear combination of the eigenvectors $z_1, z_2, \ldots, z_m$. Therefore,

$$B^{-1/2} x^{<0>} = (a_1 z_1 + a_2 z_2 + \ldots + a_m z_m) \tag{11}$$

$$\begin{aligned} x^{<k>} &= B^{1/2} M^k (a_1 z_1 + a_2 z_2 + \ldots + a_m z_m) \\ &= B^{1/2} (a_1 M^k z_1 + a_2 M^k z_2 + \ldots + a_m M^k z_m) \\ &= B^{1/2} (a_1 c_1^k z_1 + a_2 c_2^k z_2 + \ldots + a_m c_m^k z_m) \end{aligned}$$

As k goes to infinity, the $x^{<k>}$ vector will converge to a multiple of the $B^{1/2} z_1$ vector.

$$\lim_{k \to \infty} \frac{x^{<k>}}{c_1^k} = a_1 B^{1/2} z_1 \tag{12}$$

Similarly, the convergence for the $y^{<k>}$ can be proved in the same fashion:

$$y^{<k>} = C^{1/2} N^k C^{-1/2} y^{<0>} \tag{13}$$

where $N \in \mathbb{R}^{n \times n}$ is $N = C^{1/2} A^T B A C^{1/2}$ and the $y^{<k>}$ vector will converge to a multiple of the $C^{1/2} q_1$ vector, with q_1 being the principal eigenvector of the N matrix. □

VII STRUCTURAL EQUILIBRIUM

The relation phrased as *the enemy of my enemy is my friend* is observed on various networks. This relation in general can be formalized for graphs by constraining any cycle of arbitrary length to have even number of negative edges [26]. In [11], authors relate this constraint with the *multiplicative transitivity* property of an adjacency matrix, which can be measured using a modification of the clustering coefficient introduced in [17].

The structural equilibrium(SE) can be measured by checking the consistency of the edges forming cycles of length three. The relative signed clustering coefficient calculates the ratio of balanced cycles among all possible cycles of length three.

$$SE(A) = \frac{\left\| A \circ A^2 \right\|_+}{\left\| \bar{A} \circ \bar{A}^2 \right\|_+}$$

- $\bar{A}$ is the absolute adjacency matrix with $\bar{a}_{ij} = |a_{ij}|$
- $C = A \circ B$ is defined as the Hadamard product (element-wise product) for two matrices, such that $c_{ij} = a_{ij} * b_{ij}$
- $x = \|A\|_+$ is defined as the sum of all matrix elements, such that $x = \sum_i \sum_j a_{ij}$

Bipartite graphs do not have cycles of odd length. Therefore, the structural equilibrium cannot be measured as formalized before. But it can be extended to calculate the ratio of balanced cycles of length four. For this purpose, we define the multiplicative transitivity for bipartite graphs as follows.

A signed bipartite graph exhibits multiplicative transitivity when a path of three edges tend to be completed by a fourth edge having a sign equal to the product of the three edges' signs.

This can be rephrased as *the enemy of my enemy of my enemy is my enemy*, or *the enemy of my friend of my enemy is my friend*, etc. Figure 4 depicts two cycles with odd number of edges (a and c), and two cycles with even number of edges (b and d). By definition, the cycles with odd number of negative edges do not satisfy multiplicative transitivity.

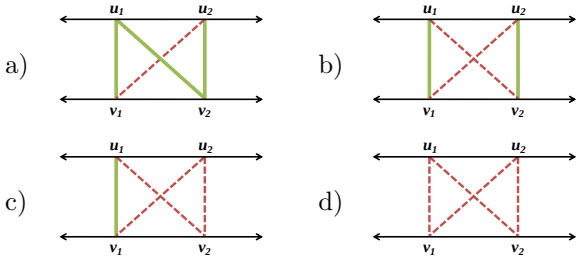


Fig 4. Cycles of four with negative edges

Hence, the corresponding relative signed clustering coefficient can be reformulated for bipartite graphs as follows:

$$SE(A) = \frac{\left\| A \circ AA^T A \right\|_+}{\left\| \bar{A} \circ \bar{A}\bar{A}^T \bar{A} \right\|_+}$$

For our experimental datasets, we report the corresponding SE values.

Table 1: Descriptive summaries of the graphs for each dataset with the partitioning accuracies for each algorithm

	111th US Senate	**111th US House**	**Political Blogosphere**
Vertices in U	64 Democrat 42 Republican Senators	268 Democrat 183 Republican Representatives	13 Liberal 9 Conservative Blogs
Vertices in V	696 Bills	1655 Bills	20 Liberal 14 Conservative People
Graph Density	88.36%	91.23%	39.04%
Str. Eq.	39.47%	39.37%	87.21%
Spectral Clustering	100.00%	99.11%	75.39%
CO-HITS	100.00%	99.56%	98.21%
ANCO-HITS	100.00%	99.56%	98.21%

VIII EXPERIMENTS & EVALUATIONS

To validate our algorithm, we have used two different datasets that are *US Congress* and *political blogosphere*. In addition to real data, we introduced a model to generate synthetic data to analyze the performance of the algorithms for various parameters.

1 US CONGRESS

The US Congress has been collecting data since the very first congress of the US history. This data has been encoded as XML files and publicly shared through the govtrack.us project[2]. From various types of data available at the project site, we collected the *roll call votes* for the 111th US Congress which includes The Senate and The House of Representatives and covers the years 2009-2010.

According to The Library of Congress[3],

> A roll call vote guarantees that every Member's vote is recorded, but only a minority of bills receive a roll call vote.

The 111th Senate has 108[4] senators and the data contains their votes on 696 bills, and The 111th House has 451 representatives and the data contains their votes on 1655 bills.

We extracted the adjacency matrix $A \in \{-1, 0, 1\}^{|U|\times|V|}$, with U vertices representing the congressmen, and the V vertices representing the bills. The values a_{ij} are 1 if the congressman u_i votes 'Yea' for the bill v_j, -1 if the congressman votes 'Nay', and 0 if he did not attend the session.

The aforementioned scaling algorithms will scale both the congressmen and the bills. In presence of partisanship[5] in the Congress, the sign of the scale values for the congressmen should correspond to the Democrat and Republican parties, and the magnitude of the scale values should represent the amount of partisanship. The first two columns of Table 1 provide information about this data as well as the partitioning accuracies of the algorithms.

We analyzed the congressmen that have been assigned to be moderate by each algorithm. We observed that the baseline algorithms tend to have the congressmen with less number of votes (i.e. lesser degree) to be moderate regardless of their partisanship. On the other hand, when we queried the names assigned to be most moderates by the ANCO-HITS, for both Democrats and Republicans, we were able to identify a number of supporting articles matching the ANCO-HITS scaling [27–30].

Figure 5 depicts the vote matrices of the 111th US Congress, with rows representing the congressmen and the columns representing the bills. The light green color represents 'Yea' votes, and dark red represents 'Nay' votes. Scaling these graphs leads to a re-ordering of the rows and columns where congressmen and bills are co-clustered together.

[2] `http://www.govtrack.us/data`

[3] `http://thomas.loc.gov/home/rollcallvotes.html`

[4] Normally, each congress has 100 senators (2 from each state), however in many of the congresses, there are unexpected changes on the seats caused by displacements or deaths.

[5] *Partisanship* can be defined as being devoted to or biased in support of a party.

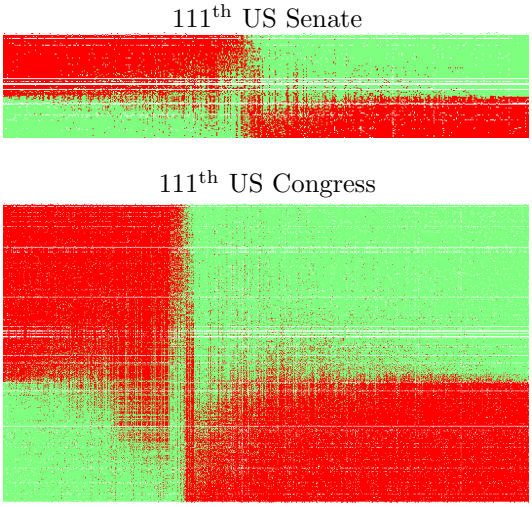


Fig 5. Vote matrix after scaling with ANCO-HITS

Figure 6 represents the bipartite graph of the 111th US Congress data after scaling the entities with ANCO-HITS. The light green colored edges represent 'Yea' votes, and dark red represents 'Nay' votes. Similar to our motivating Figure 1, this figure also shows partisan behavior in the 111th US Congress.

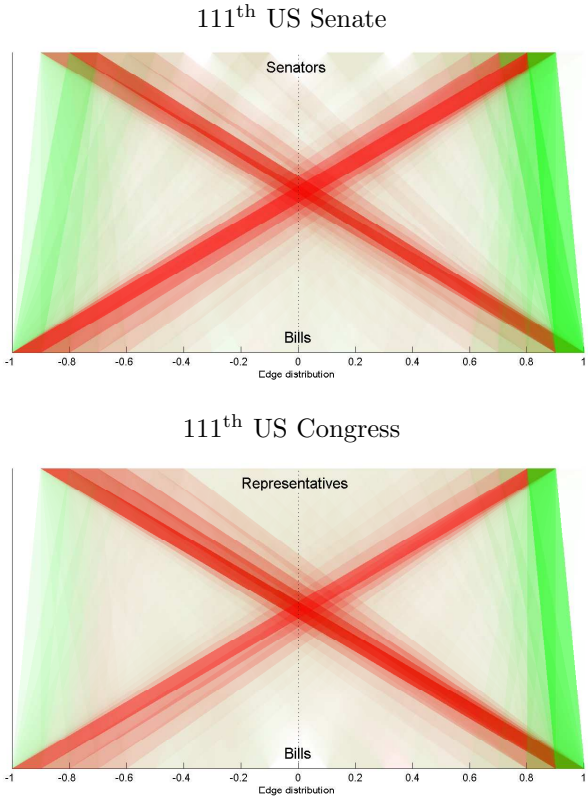


Fig 6. Bipartite graphs after ANCO-HITS

2 POLITICAL BLOGOSPHERE

As Web 2.0 platforms gained popularity, it became easy for web users to be a part of the web and express their opinions, mostly through blogs. Most blogs are maintained by individuals, whereas there are also professional blogs with a group of authors. In this study, we focus on a set of popular political liberal and conservative blogs that have clearly declared positions. These blogs contain discussions about social, political, economic issues and related key individuals. They express positive sentiment towards individuals whom they share ideologies with, and negative sentiment towards others. In these blogs, it is common to see criticism of people within the same camp, or support for people from the other camp.

In this experiment, we collected a list of 22 most popular liberal and conservative blogs from the Technorati[6] rankings. For each blog, we fetched the posts for the period of 6 months before the 2008 US presidential elections (May - October, 2008) due to the intensity of the debates and discussions. Table 2 shows the list of blogs with their URLs, political camps and the number of posts for the given period.

We use AlchemyAPI[7] to run a named entity tagger to extract people names from the posts, and entity-level sentiment analysis which provided us with weighted sentiment (positive values indicating support, and negative indicating opposition) for each person. This information was used to synthesize a signed bipartite graph, where the blogs and people correspond to the two sets of vertices U and V. The a_{ij} values of the adjacency matrix A are the cumulative sum of sentiment values for each mention of the person v_j by the blog u_i.

To get a gold standard list of the most influential liberal and conservative people, we used The Telegraph List[8] for 2007. The third column of Table 1 provides information about this data as well as the partitioning accuracies of the algorithms.

[6] http://technorati.com/
[7] http://www.alchemyapi.com/
[8] http://www.telegraph.co.uk/news/uknews/1435447/The-top-US-conservatives-and-liberals.html

3 SYNTHETIC DATA

The actual partitioning information for the real datasets were available, which made it possible to check the partitioning accuracy of the algorithms. However, to thoroughly check the scaling accuracy of the algorithms, we developed a method to generate random bipartite graphs with the following properties:

- The degrees and the scores for the vertices in U and V follow independent probability distribution with varying parameters and shapes.

Following procedure describes the method to generate random graphs.

RandomGraph(m, n, D_{degree}, D_{scale})
Let A, U, V, X and Y be defined as in Definition 1.

1. Create m vertices for U, and n vertices for V
2. Independently assign degrees $1 \leq d(u_i) \leq m$ and $1 \leq d(v_j) \leq n$ with probability distribution D_{degree}
3. Independently assign $-1 \leq x_i, y_j \leq +1$ values with probability distribution D_{scale}.
4. Generate an adjacency list of pairs (u_i, v_j), where each node u_i and v_j occur in the list $d(u_i)$ and $d(v_j)$ times respectively.
5. For each adjacency pair (u_i, v_j), assign the entry in the adjacency matrix a_{ij} with value
   (a) $sgn(x_i) \times sgn(y_j)$, with probability $1-(1-|x_i|)(1-|y_j|)$
   (b) $-sgn(x_i) \times sgn(y_j)$, with probability $(1-|x_i|)(1-|y_j|)$

The difference between the scale obtained for a vertex by executing the scaling algorithm and its scale assigned by the random graph generator algorithm defines the error for that vertex. Table 3 shows the mean error vs vertex degrees plots for each algorithm applied to 12 different synthetic data sets.

In our experiments, the number of vertices of the graph is $m = n = 100$. We used four different distributions for varying polarization. These were perfectly polarized, *Beta*, bimodal and uniform distributions [31]. Perfectly polarized distribution was obtained by mapping all vertices to the extremes of both sides with equal probability. We used three different normal distributions for varying the degree distributions of vertices. Degree distributions were obtained by $\mathcal{N}(\mu = 30, \sigma = 2)$, $\mathcal{N}(\mu = 30, \sigma = 5)$ and $\mathcal{N}(\mu = 30, \sigma = 10)$ in order to evaluate the effect of degree variance on the performance of the algorithms. We also experimented with different μ values of 10, 30, and 50 in order to measure the effect of density variations of the graph on the performance of the algorithms, which did not show any significant impact.

For each polarization and degree distribution we tested the performance of two baseline algorithms and our proposed algorithm. Table 3 presents these experimental results corresponding to 12 scenarios. In this table, columns correspond to the variance in degrees, and rows correspond to the polarization distributions.

In order to better visualize the effect of the degree of the vertex in determining its scaling position we used the mean error as an aggregate score. In the scatter plots, x-axis corresponds to the degree of vertices of a graph, and y-axis corresponds to mean scaling error. There are some error peaks at the boundary degree values due to their low frequencies.

From the table, we can make the following observations:

- Across all polarizations, as the vertex degree variance increases, overall errors for baseline algorithms increase due to their sensitivity to vertex degrees.
- Between the baseline algorithms, spectral clustering consistently outperforms CO-HITS.
- Even though spectral clustering performs almost as good as our proposed ANCO-HITS for bimodal and uniform polarization distributions, when the polarization is high, as in the other two distributions, its performance degrades.
- As polarization increases, from U-shaped to perfect polarization, ANCO-HITS performance also increases. In case of perfect polarization, ANCO-HITS has almost no error.

Overall, in every single case our proposed ANCO-HITS algorithm outperforms the baselines.

Table 2: List of Political Blogs

Blog name	URL	Political Camp	Posts
Huffington Post	http://www.huffingtonpost.com/	Liberal	3959
Daily Kos	http://www.dailykos.com/	Liberal	1957
Boing Boing	http://www.boingboing.net/	Liberal	1576
Crooks and Liars	http://www.crooksandliars.com/	Liberal	1497
Firedoglake	http://www.firedoglake.com/	Liberal	1354
AMERICABlog	http://americablog.com/	Liberal	1297
Think Progress	http://thinkprogress.org/	Liberal	1197
Talking Points Memo	http://www.talkingpointsmemo.com/	Liberal	1081
Wonkette	http://wonkette.com/	Liberal	1064
Balloon Juice	http://www.balloon-juice.com/	Liberal	923
Digby's Hullabaloo	http://digbysblog.blogspot.com/	Liberal	553
Informed Comment	http://www.juancole.com/	Liberal	179
Truthdig	http://www.truthdig.com/	Liberal	159
Hot Air	http://hotair.com/	Conservative	1579
Reason - Hit and Run	http://reason.com/blog	Conservative	1563
Little green footballs	http://littlegreenfootballs.com/	Conservative	787
Atlas shrugs	http://atlasshrugs2000.typepad.com/	Conservative	773
Stop the ACLU	http://www.stoptheaclu.com/	Conservative	741
Wizbangblog	http://wizbangblog.com/	Conservative	621
Michelle Malkin	http://michellemalkin.com/	Conservative	532
Red State	http://www.redstate.com/	Conservative	311
Pajamas media	http://pajamasmedia.com/	Conservative	97

IX CONCLUSIONS & FUTURE WORK

In this paper, we introduced a new problem for scaling and partitioning signed weighted bipartite graphs. We adapted two existing algorithms, and proposed a new algorithm to solve this problem. We used both real data from political blogosphere and US Congress records, as well as synthetic data to evaluate these algorithms. Our experiments showed that our proposed algorithm is very effective and outperforms the two other baselines. The algorithms in source code and the test data is available online at www.PartisanScale.com/paperdata

In real world graphs, it is rarely the case that all the nodes have the same degree. Some bloggers have more extensive coverage than others. Similarly, some senators miss or abstain on more votes than others. Partisanship shall not be affected by the variance in node degrees. A marginal example is the case when a senator supersedes a deceased one through the end of the term. All the baseline algorithms assign this low degree senator to a moderate location, even though he or she can be extremely partisan. ANCO-HITS manages the degree bias by a normalizing scheme, and places this senator to an appropriate location.

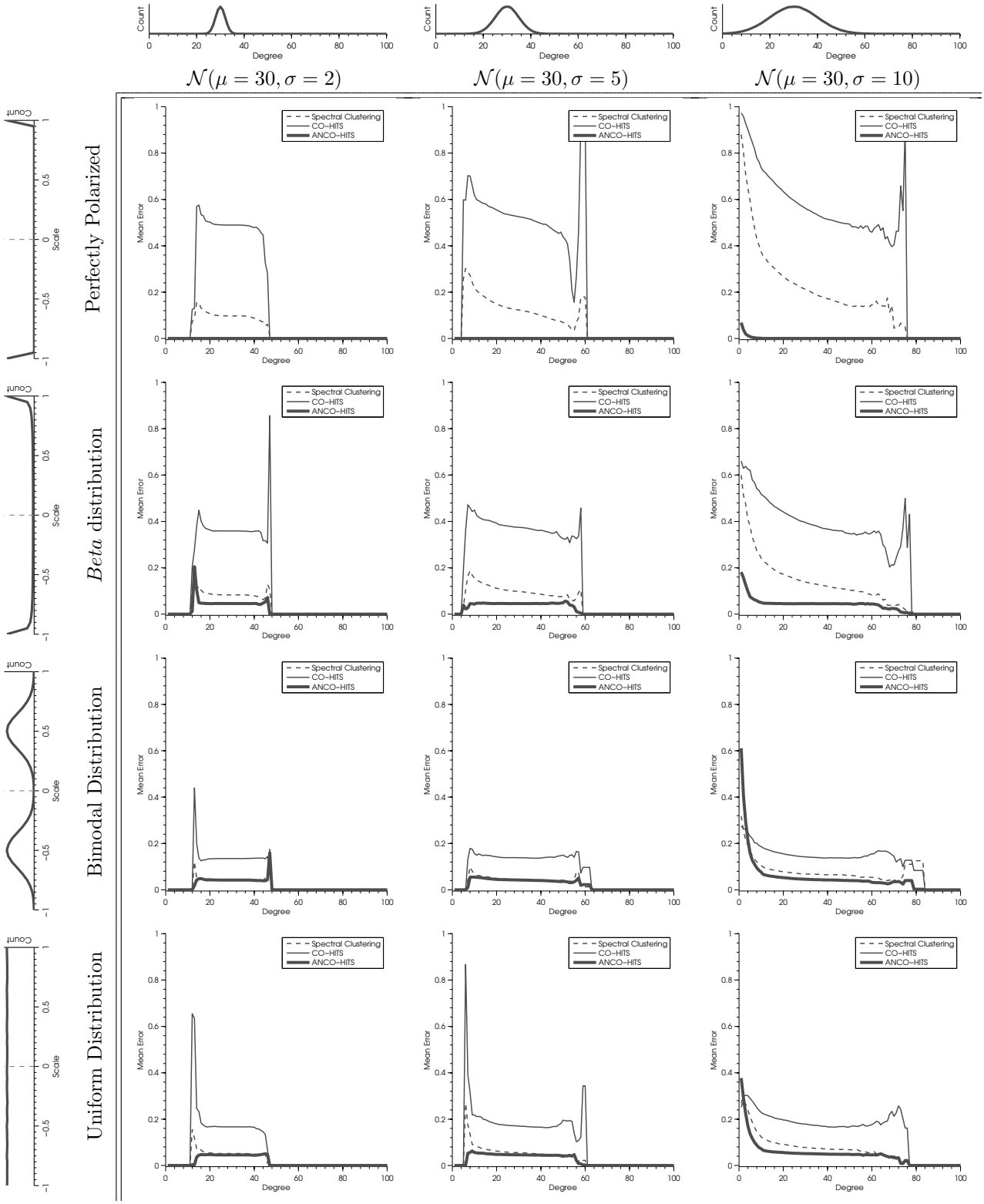
In order to analyze longitudinal voting patterns for the US Congresses, we downloaded all voting records since the 1st US Congress, executed ANCO-HITS, and produced an interactive visualization system as described in [32]. This system is accessible online at www.PartisanScale.com

Our future work involves developing techniques for detecting and presenting both friendly and unfriendly *neighborhoods* of a blog or an issue, and their agreements and disagreements. We also plan to incorporate longitudinal analysis to detect trends and trajectories over time.

ACKNOWLEDGMENTS

We would like to thank Dananjayan Thirumalai for helping us collect the political blogosphere data. This research was supported in part by US DOD Minerva Research Initiative grant N00014-09-1-0815.

Table 3: Synthetic data performances



References

[1] D. Drezner and H. Farrell, "The power and politics of blogs," *Public Choice*, vol. 134, pp. 15–30, 2008.

[2] T. Mullen and R. Malouf, "A preliminary investigation into sentiment analysis of informal political discourse," in *AAAI symposium on computational approaches to analysing weblogs (AAAI-CAAW)*, 2006, pp. 159–162.

[3] R. Malouf and T. Mullen, *Graph-based user classification for informal online political discourse*, 2007.

[4] M. Thomas, B. Pang, and L. Lee, "Get out the vote: Determining support or opposition from congressional floor-debate transcripts," in *In Proceedings of EMNLP*, 2006, pp. 327–335.

[5] M. Bansal, C. Cardie, and L. Lee, "The power of negative thinking: Exploiting label disagreement in the min-cut classification framework," *Proceedings of COLING: Companion volume: Posters*, pp. 13–16, 2008.

[6] W. Lin and A. Hauptmann, "Are these documents written from different perspectives?: a test of different perspectives based on statistical distribution divergence," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2006, pp. 1057–1064.

[7] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 u.s. election: divided they blog," in *Proceedings of the 3rd international workshop on Link discovery*, ser. LinkKDD '05. New York, NY, USA: ACM, 2005, pp. 36–43.

[8] H. Deng, M. Lyu, and I. King, "A generalized co-hits algorithm and its application to bipartite graphs," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp. 239–248.

[9] M. Rege, M. Dong, and F. Fotouhi, "Co-clustering documents and words using bipartite isoperimetric graph partitioning," in *Data Mining, 2006. ICDM'06. Sixth International Conference on*. IEEE, 2006, pp. 532–541.

[10] H. Zha, X. He, C. Ding, H. Simon, and M. Gu, "Bipartite graph partitioning and data clustering," in *Proceedings of the tenth international conference on Information and knowledge management*. ACM, 2001, pp. 25–32.

[11] J. Kunegis, A. Lommatzsch, and C. Bauckhage, "The slashdot zoo: mining a social network with negative edges," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 741–750.

[12] U. V. Luxburg, "A tutorial on spectral clustering," 2007.

[13] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14: Proceeding of the 2001 Conference*, 2001, pp. 849–856.

[14] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888 –905, aug 2000.

[15] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." 1999.

[16] J. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[17] D. Watts and S. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[18] I. Dhillon, J. Fan, and Y. Guan, "Efficient clustering of very large document collections," *Data mining for scientific and engineering applications*, pp. 357–381, 2001.

[19] N. Slonim and N. Tishby, "Document clustering using word clusters via the information bottleneck method," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2000, pp. 208–215.

[20] P. Berkhin, "Survey of clustering data mining techniques," *Grouping Multidimensional Data: Recent Advances in Clustering*, pp. 25–71, 2006.

[21] I. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: spectral clustering and normalized cuts," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 551–556.

[22] J. Hartigan and M. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[23] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. De Luca, and S. Albayrak, "Spectral analysis of signed graphs for clustering, prediction and visualization," in *Proc SDM*. Citeseer, 2010.

[24] I. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001, pp. 269–274.

[25] X. Fern and C. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 36.

[26] P. Hage and F. Harary, "Structural models in anthropology." 1983.

[27] S. T. Dennis, "Senate moderates look for more influence," http://www.rollcall.com/issues/56_90/-203808-1.html.

[28] J. Newton-Small, "Can ben nelson get a bipartisan stimulus win," http://www.time.com/time/politics/article/0,8599,1877535,00.html.

[29] "Factions in the republican party (united states)," http://en.wikipedia.org/wiki/Factions_in_the_Republican_Party_(United_States).

[30] "Blue dog coalition," http://ross.house.gov/BlueDog/Members/.

[31] V. Rohatgi and A. Saleh, *An introduction to probability and statistics*. Wiley-India, 2008.

[32] S. Gokalp and H. Davulcu, "Partisan scale," in *Proceedings of the 21st international conference companion on World Wide Web*. ACM, 2012, pp. 349–352.