Legislative Prediction via Random Walks over a Heterogeneous Graph

Jun Wang Kush R. Varshney Aleksandra Mojsilović
Business Analytics and Mathematical Sciences Department
IBM Thomas J. Watson Research Center
1101 Kitchawan Rd., Route 134, Yorktown Heights, NY 10598, USA
{wangjun, krvarshn, aleksand}@us.ibm.com

Abstract

In this article, we propose a random walk-based model to predict legislators' votes on a set of bills. In particular, we first convert roll call data, i.e. the recorded votes and the corresponding deliberative bodies, to a heterogeneous graph, where both the legislators and bills are treated as vertices. Three types of weighted edges are then computed accordingly, representing legislators' social and political relations, bills' semantic similarity, and legislator-bill vote relations. Through performing two-stage random walks over this heterogeneous graph, we can estimate legislative votes on past and future bills. We apply this proposed method on real legislative roll call data of the United States Congress and compare to state-of-the-art approaches. The experimental results demonstrate the superior performance and unique prediction power of the proposed model.

1 Background and Motivation

Humanistic and social studies, including anthropology, criminology, marketing, sociology, and urban planning are increasingly turning to data-driven quantitative methods, informatics, and predictive analytics. Political science is no different. Politics in democracies are centered around votes on bills in legislatures. Voting history, also known as roll call data, is an important historical record that has been studied statistically since the 1920s, if not earlier [27].

Following other political science studies, we focus on the legislature of the federal government of the United States of America, known as the Congress. An important feature of the United States Congress is that legislators are not bound to vote in lockstep with their party. In contrast to parliamentary governments, such as those that follow the Westminster system, party affiliation is not codified in the constitution and thus is only one of many factors that go into determining whether a legislator votes *yea* or *nay*. Congress is a bicameral legislature composed of the Senate with 100 members known as senators, and the House of Representatives with 435 members known as representatives.[1] A session of Congress lasts two years, with the current session being the one hundred twelfth. The composition of Congress changes after every session due to elections. Within a session, the only changes are due to death or resignation.

A bill is a proposed law under consideration by a legislature, that if passed, becomes a law. There are approximately 700 bills voted upon per session in the Senate and approximately 1200 in the House of Representatives. Each bill that comes to a vote in Congress is sponsored by at least one legislator. Other legislators may cosponsor the bill if they coauthored it or if they wish to publicly indicate strong support for it in advance of the vote. Thus, frequent cosponsorship of bills reflects collaboration and similarity in ideology between legislators.

Roll call data can be analyzed to obtain a variety of descriptive statistics, but can also be used in developing predictive models. Legislative prediction leads to a better understanding of government and can also provide actionable insights to political strategists. It is a challenging task to predict the votes of all current legislators on a bill that has not yet been voted upon. One of the representative models in quantitative political science is the ideal point model (IPM), which builds a one-dimensional "political space" and then places each legislator and bill in that space [7]. Realizing the limitations of IPM, such as the low-dimension restriction, researchers from the machine learning and data mining communities have recently proposed some advanced methods, including the ideal point topic model [13], a joint model from the temporal perspective [35], and a multiple kernel learning model [29].

In this article, we propose to leverage both text mining of bills and the social connection between legislators to predict legislative votes. In particular, we develop a novel model based on random walks on a heterogeneous

[1] In this paper, we ignore non-voting delegates in the House of Representatives from territories such as Guam that are not states.

graph (RWHG) to predict the vote links between legislators and bills. In this formulation, the roll call data is represented as a heterogeneous graph, where both legislators and bills are treated as vertices. The legislators are connected based on political relationship, specifically cosponsorship, and the bills are connected based on their semantic similarity in the bag-of-words representation space. The votes, *yea* or *nay*, are treated as directed edges of a bipartite-style legislator-bill graph (refer to Figure 1). Based on this formulation, a two-stage random walk is performed over the heterogeneous graph to iteratively generate vote links. Experimental results on predicting random missing votes and sequentially predicting future votes shows the superior performance of this method over state-of-the-art algorithms.

In the remaining part of this paper, we describe the heterogeneous graph formulation in Section 2. In Section 3, we present the RWHG model to predict vote links. The experimental results on real roll call data from the United States Congress are described in Section 4. We describe our main contributions, related work, conclusions, and directions for future work in Section 5.

2 Heterogeneous Graph Formulation of Roll Call Data

In this section, we use a heterogeneous graph to represent the roll call data, where both the legislators and bills are treated as graph vertices. The legislator vertices are connected based on their social and political relationships, quantified with edge weights. Similarly, the bills are connected based on their estimated semantic similarity. The votes are treated as the links of a bipartite-style legislator-bill graph. Overall, this unique formulation has a heterogeneous-structured graph with two types of vertices and three types of edges.

2.1 Graph Notations We first define the graph notation for the legislators. Assume there are a total of L legislators and denote the set of legislator vertices as $\mathbf{V}_{(x)} = \{x_1, \ldots, x_l, \ldots, x_L\}$ with cardinality $|\mathbf{V}_{(x)}| = L$. These legislators can be connected based on attributes such as party, state, age, gender, and cosponsorship by converting the attributes to a political similarity measure between legislators. In other words, the legislators form a graph $\mathcal{G}_{(x)} = \{\mathbf{V}_{(x)}, \mathbf{E}_{(x)}\}$ independently, with an edge set $\mathbf{E}_{(x)} = \{e_{(x)_{lm}}\} \subset \mathbf{V}_{(x)} \times \mathbf{V}_{(x)}$ $(l, m = 1, \ldots, L)$. The details for estimating political similarity, i.e. the weight of the edges, will be provided in the following subsection.

In addition, we define the set of bills as $\mathbf{V}_{(y)} = \{y_1, \ldots, y_n, \ldots, y_N\}$ with cardinality $|\mathbf{V}_{(y)}| = N$. Given textual content, we reuse the same symbol to represent the standard bag-of-words (BOW) model of bills as $y_n \in \mathbb{R}^B$, where B is the size of the dictionary [14]. Accordingly, the bills form a graph in the semantic space, where the set of vertices $\mathbf{V}_{(y)}$ represents the bills and the set of edges $\mathbf{E}_{(y)} = \{e_{(y)_{nk}}\} \subset \mathbf{V}_{(y)} \times \mathbf{V}_{(y)}$ $(n, k = 1, \ldots, N)$ connects bills based on their semantic similarity. Therefore, we now have the bill graph represented as $\mathcal{G}_{(y)} = \{\mathbf{V}_{(y)}, \mathbf{E}_{(y)}\}$.

The last piece of information we want to leverage into the graph formulation is the initially-given set of votes, i.e. the *yea* or *nay* results for the legislators voting on the bills. Since each vote involves two types of vertices, one legislator and one bill, the vote can be viewed as a special type of directed edge or link across these heterogeneous vertices. This gives the third component of the heterogeneous graph formulation, a bipartite structured vote graph $\mathcal{G}_{(xy)} = \{\mathbf{V}, \mathbf{E}_{(xy)}\}$, where $\mathbf{V} = \mathbf{V}_{(x)} \cup \mathbf{V}_{(y)}$ and $\mathbf{E}_{(xy)} = \{e_{(xy)_{ln}}\} \subset \mathbf{V}_{(x)} \times \mathbf{V}_{(y)}$ $(l = 1, \ldots, L, n = 1, \ldots, N)$.

In summary, the heterogeneous graph $\mathcal{G}$ contains three subgraphs: legislator graph $\mathcal{G}_{(x)}$, bill graph $\mathcal{G}_{(y)}$, and vote graph $\mathcal{G}_{(xy)}$. In a general form, we can write $\mathcal{G}$ as

$$\begin{aligned}
\mathcal{G} &= \{\mathbf{V}, \mathbf{E}\}, \\
\mathbf{V} &= \mathbf{V}_{(x)} \cup \mathbf{V}_{(y)}, \\
\mathbf{E} &= \mathbf{E}_{(x)} \cup \mathbf{E}_{(y)} \cup \mathbf{E}_{(xy)}, \\
\mathbf{E}_{(x)} &\subset \mathbf{V}_{(x)} \times \mathbf{V}_{(x)}, \\
\mathbf{E}_{(y)} &\subset \mathbf{V}_{(y)} \times \mathbf{V}_{(y)}, \\
\mathbf{E}_{(xy)} &\subset \mathbf{V}_{(x)} \times \mathbf{V}_{(y)}.
\end{aligned} \tag{2.1}$$

In other words, graph $\mathcal{G}$ has two types of heterogeneous vertices, i.e. legislators $\mathbf{V}_{(x)}$ and bills $\mathbf{V}_{(y)}$, and three types of edges, legislator political relations $\mathbf{E}_{(x)}$, bill semantic similarity $\mathbf{E}_{(y)}$, and directed vote links $\mathbf{E}_{(xy)}$. In the following subsections, we will detail the estimation of these edge weights and provide some important graph quantities.

2.2 Legislators' Social and Political Relations Social connections among the members of the House and Senate have been well-studied in fields like social science and political science because they illuminate information for estimating political relevance and revealing the underlying legislative patterns [11]. Different kinds of social connections, such as friendship, family, and acquaintanceship relations, have been identified as important effects on political positions [3]. However, predicting roll call data is about understanding legislators' ideology more than social relationships between them [11, 26]. Therefore, scholars recently proposed to use cosponsorship relations as a more robust and di-

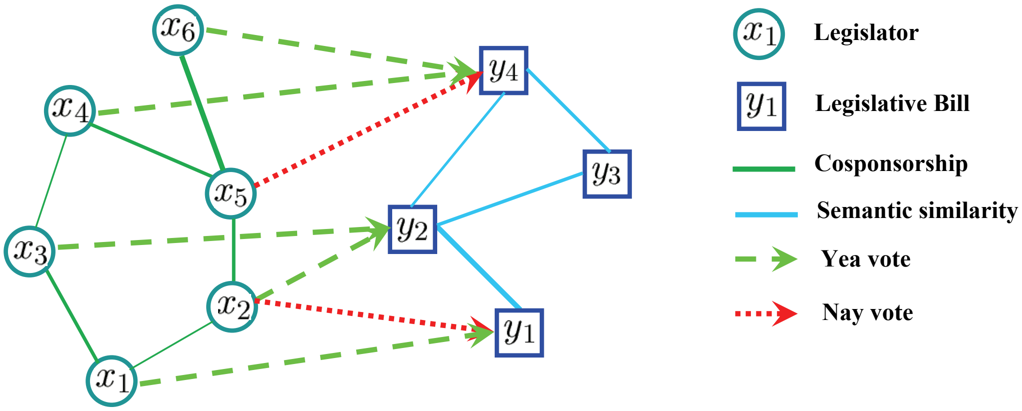


Figure 1: Heterogeneous graph representation of roll call data from the Congress of the United States of America. The heterogeneous graph contains legislator vertices, bill vertices, edges connecting legislators (cosponsorship relation), edges connecting bills (semantic similarity), and directed vote edges from legislators to bills. The thickness of the edges indicates the strength of the corresponding connections.

rect way to understand the voting behavior and political influence of each legislator [11, 12]. Briefly speaking, each piece of legislation is uniquely sponsored by a legislator and publicly cosponsored by a group of legislators. Hence, it is fairly straightforward to reveal the cosponsorship edge between each cosponsor and the corresponding sponsor. In this paper, we are particularly interested in this kind of cosponsorship information and use it to connect the legislators and estimate their political affinity.[2]

More precisely, the pairwise political similarity $w_{(x)_{lm}} \in \mathbb{R}$ refers to the affinity between legislators x_l and x_m, i.e. the weight of the edge $e_{(x)_{lm}}$. Assume that legislators x_l and x_m have a total of c_{lm} cosponsored bills in common and have C_l and C_m individually cosponsored bills. Then the value of $w_{(x)_{lm}}$ computed from the cosponsorship information is

$$w_{(x)_{lm}} = \frac{c_{lm}}{C_l + C_m}. \tag{2.2}$$

In addition, the cosponsorship matrix $\mathbf{W}_{(x)} = \{w_{(x)_{lm}}\}$ presents an intuitive way to estimate the political connectivity of the legislators. Note that the value of $w_{(x)_{lm}}$ represents the normalized cosponsorship weight between the legislator x_l and x_m. For each legislator x_l, the sum of such edge weights of all the connected legislators shows the political popularity of this legislator, which is calculated as

$$d_{(x)_l} = \sum_m w_{(x)_{lm}} = \sum_m \frac{c_{lm}}{C_l + C_m}. \tag{2.3}$$

From the graph formulation perspective, this quantity is exactly the degree of vertex x_l on the graph $\mathcal{G}_{(x)}$. Accordingly, the diagonal degree matrix can be written as $\mathbf{D}_{(x)} = diag\left[d_{(x)_1}, \ldots, d_{(x)_l}, \ldots, d_{(x)_L}\right]$.

2.3 Semantic Similarity of Bills At a high level, the BOW model represents the textual context using the frequency of the words in documents, while omitting grammar and word order. Given the BOW representation of the nth legislative bill as $y_n = \{y_{n1}, \ldots, y_{nb}, \ldots, y_{nB}\}$, the bth element y_{nb} denotes the count of the corresponding bth entry in the dictionary appearing in the bill. For such a histogram-style feature representation, one can use a kernel function over pairs of vertices to compute the weight $w_{(y)_{nk}}$ for the edge $e_{(y)_{nk}}$ in the bill graph. For example, the Gaussian kernel is often applied to modulate semantic similarity between bill y_n and y_k as:

$$w_{(y)_{nk}} = \exp\left[-\frac{dis^2(y_n, y_k)}{2\sigma^2}\right], \tag{2.4}$$

where the function $dis(y_n, y_k)$ evaluates the dissimilarity or distance between bill y_n and y_k, and σ is the kernel bandwidth parameter. Different choices of the distance function $dis(\cdot)$ have been used in previous research, such as ℓ_p $(p = 1, 2)$ distance and χ^2 distance [16, 38, 40]. The kernel function based weighting scheme has the flexibility to adapt to a wide range of data with different priors and distributions. However, the determination of bandwidth σ is fairly heuristic without any theoretic guarantee. Another popular weighting function for histogram-style data representation is cosine similarity,

[2] In this paper, we treat sponsorship and cosponsorship relations equally and use the term "cosponsorship" to represent both.

which is relatively straightforward to compute as

$$w_{(y)_{nk}} = \frac{y_n \cdot y_k}{\|y_n\| \|y_k\|}. \tag{2.5}$$

Similarly, the degree of the bill vertex y_n can be calculated as $d_{(y)_n} = \sum_k w_{(y)_{nk}}$ and the corresponding degree matrix is $\mathbf{D}_{(y)} = diag\left[d_{(y)_1}, \ldots, d_{(y)_n}, \ldots, d_{(y)_N}\right]$.

2.4 Legislator-Bill Vote Links Here we define the vote as a directed edge $e_{(xy)_{ln}}$, which indicates that the legislator x_l has voted on bill y_n. Because there are two types of votes, it is reasonable to set the edge weight $w_{(xy)_{ln}} = 1$ for *yea* and $w_{(xy)_{ln}} = -1$ for *nay*. If the vote does not exist, we set $w_{(xy)_{ln}} = 0$, indicating no vote edge between x_l and y_n. However such straightforward setting of edge weights is infeasible for random walk-based formulations since the edge weight matrix of the graph will eventually be converted to a positive-valued transition probability matrix.

Hence, we propose to partition the *yea* and *nay* links and treat them separately, which results in two legislator-bill vote graphs, namely *yea* and *nay* graphs with all positive edge weights. More specifically, for either *yea* or *nay* graph, the edge weight is set to one if a vote exists between the corresponding legislator and bill. For non-existent edges, the weights are still set to zero. Finally, we obtain two weight matrices $\mathbf{W}^{yea}_{(xy)} = \{w^{yea}_{(xy)_{ln}}\}$ and $\mathbf{W}^{nay}_{(xy)} = \{w^{nay}_{(xy)_{ln}}\}$ for all the *yea* and *nay* votes, respectively. Accordingly, four types of vertex degree matrices are defined over the bipartite-style legislator-bill vote graph as:

$$\begin{aligned}
& d^{yea}_{(xy)_l} = \sum_n w^{yea}_{(xy)_{ln}}, \quad d^{nay}_{(xy)_l} = \sum_n w^{nay}_{(xy)_{ln}}, \\
& \mathbf{D}^{yea}_{(xy)} = diag\left[d^{yea}_{(xy)_1}, \ldots, d^{yea}_{(xy)_l}, \ldots, d^{yea}_{(xy)_L}\right], \\
& \mathbf{D}^{nay}_{(xy)} = diag\left[d^{nay}_{(xy)_1}, \ldots, d^{nay}_{(xy)_l}, \ldots, d^{nay}_{(xy)_L}\right], \\
& d^{yea}_{(yx)_n} = \sum_l w^{yea}_{(yx)_{ln}}, \quad d^{nay}_{(yx)_n} = \sum_l w^{nay}_{(yx)_{ln}}, \\
& \mathbf{D}^{yea}_{(yx)} = diag\left[d^{yea}_{(yx)_1}, \ldots, d^{yea}_{(yx)_n}, \ldots, d^{yea}_{(yx)_N}\right], \\
& \mathbf{D}^{nay}_{(yx)} = diag\left[d^{nay}_{(yx)_1}, \ldots, d^{nay}_{(yx)_n}, \ldots, d^{nay}_{(yx)_N}\right],
\end{aligned} \tag{2.6}$$

where $d^{yea}_{(x)_l}$ and $d^{nay}_{(x)_l}$ indicate the total numbers of *yea* and *nay* votes from the legislator x_l, and $d^{yea}_{(y)_n}$ and $d^{nay}_{(y)_n}$ are the total number of *yea* and *nay* votes received by the bill y_n. Hence we can define the priors of *yea* and *nay* votes for each legislator and bill based on the given vote links, simply as:

$$\begin{aligned}
& p^{yea}_{(x)_l} = \frac{d^{yea}_{(xy)_l}}{d^{yea}_{(xy)_l} + d^{nay}_{(xy)_l}}, \quad p^{nay}_{(x)_l} = \frac{d^{nay}_{(xy)_l}}{d^{yea}_{(xy)_l} + d^{nay}_{(xy)_l}}, \\
& p^{yea}_{(y)_n} = \frac{d^{yea}_{(yx)_n}}{d^{yea}_{(yx)_n} + d^{nay}_{(yx)_n}}, \quad p^{nay}_{(y)_n} = \frac{d^{nay}_{(yx)_n}}{d^{yea}_{(yx)_n} + d^{nay}_{(yx)_n}},
\end{aligned} \tag{2.7}$$

where $p^{yea}_{(x)_l}$ and $p^{nay}_{(x)_l}$ are the priors that x_l produces *yea* and *nay* votes and $p^{yea}_{(y)_l}$, and $p^{nay}_{(y)_l}$ are the priors that y_n receives *yea* and *nay* votes.

Recall we have the general form of the heterogeneous graph as $\mathcal{G} = \{\mathbf{V}, \mathbf{E}\}$, where $\mathbf{V} = \mathbf{V}_{(x)} \cup \mathbf{V}_{(y)}$ and $\mathbf{E} = \mathbf{E}_{(x)} \cup \mathbf{E}_{(y)} \cup \mathbf{E}_{(xy)}$. Therefore, we can define the edge weight matrix $\mathbf{W}$ for $\mathcal{G}$ in a block-wise form as

$$\mathbf{W} = \left[\begin{array}{cc} \mathbf{W}_{(x)} & \mathbf{W}_{(xy)} \\ \mathbf{0} & \mathbf{W}_{(y)} \end{array}\right], \tag{2.8}$$

where $\mathbf{W}_{(xy)}$ is the weight matrix of the directed legislator-bill vote graph. Since we treat the *yea* and *nay* votes separately, here we decompose $\mathbf{W}$ into $\mathbf{W}^{yea}$ and $\mathbf{W}^{nay}$ accordingly

$$\begin{aligned}
& \mathbf{W}^{yea} = \left[\begin{array}{cc} \mathbf{W}_{(x)} & \mathbf{W}^{yea}_{(xy)} \\ \mathbf{0} & \mathbf{W}_{(y)} \end{array}\right], \\
& \mathbf{W}^{nay} = \left[\begin{array}{cc} \mathbf{W}_{(x)} & \mathbf{W}^{nay}_{(xy)} \\ \mathbf{0} & \mathbf{W}_{(y)} \end{array}\right],
\end{aligned} \tag{2.9}$$

where $\mathbf{W}^{yea}$ and $\mathbf{W}^{nay}$ are the weight matrices of the heterogeneous graphs $\mathcal{G}^{yea}$ and $\mathcal{G}^{nay}$, respectively.

Figure 1 illustrates an example of the heterogeneous graph representation of the roll call data. Based on this formulation, the goal of legislative prediction is to infer the missing edges in $\mathbf{E}_{(xy)}$ given the legislator graph $\mathcal{G}_{(x)}$, bill graph $\mathcal{G}_{(y)}$, and the partially observed vote edges, i.e. *yea* and *nay* vote links. In the following section, we will present a two-stage random walk approach to conduct vote prediction using both $\mathbf{W}^{yea}$ and $\mathbf{W}^{nay}$.

3 Methods and Technical Solutions

Given some observations of the votes, there are two types of legislative prediction tasks: 1) predicting votes missing at random; 2) predicting all votes for new bills. The first task is related to so called within-matrix prediction and the second one to out-of-matrix prediction. To accomplish such challenging prediction tasks, especially predicting the votes for new bills, two major assumptions are made to support our methodology.

1. **Political affinity connects legislative behavior**. Legislators who have strong affinity, e.g. strong cosponsorship relations, tend to vote similarly on a set of bills.

2. **Legislative behavior is consistent among similar bills**. Semantically similar bills tend to receive the same voting results from legislators.

These two assumptions bring two views of predicting the votes along the column and row directions of the weight matrix $\mathbf{W}_{(xy)}$. Along the row direction, the vote $w_{(xy)_{ln}}$ is estimated based on the known votes of the most similar bills from the same legislator, while along the column direction, the prediction of $w_{(xy)_{ln}}$ is accomplished based on the observed votes of the same bill generated by the most similar legislators. In the experimental section, we show empirical analysis to validate these two assumptions.

In the following subsection, we present our method of random walks on a heterogeneous graph (RWHG), which combines the clues from the above assumptions and performs two-stage random walks on two unipartite graphs and across a bipartite graph.

3.1 Random Walks on Unipartite Graphs As discussed earlier, there are three subgraphs in the above formulation, two of which, i.e. legislator cosponsorship graph and bill semantic similarity graph, are unipartite with homogeneous vertices. For each of these two subgraphs, we consider the random walk with restart (RWR) model [25, 34] to derive the steady-state distributions, which indicate the political and semantic relevance among legislators and bills, respectively. Note that during the process of performing RWR, we break the directed vote links and conduct random walks over these two unipartite graphs independently.

For a standard RWR-based relevance model, a random walker starts from the ith vertex and iteratively jumps to its neighbors with transition probabilities $\mathbf{p}_i = \{p_{i1}, \ldots, p_{ij}, \ldots, p_{in}\}$. However, for each transition, the random walker returns to the original vertex i with probability $1 - \alpha$. After achieving the steady-state, the probability of the random walker being at the jth vertex gives the relevance score of vertex j with respect to vertex i. Similarly, if we simultaneously launch n random walkers, one from each vertex of the graph, with transition probabilities $\mathbf{p}_1, \ldots, \mathbf{p}_n$, the final steady-state probability matrix gives the relevance scores between each pair of vertices. Defining the transition probability matrix $\mathbf{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$, one step of RWR from time t to $t+1$ can be formed as

$$\mathbf{R}(t+1) = \alpha \mathbf{P} \mathbf{R}(t) + (1-\alpha)\mathbf{I}, \tag{3.10}$$

where $\mathbf{R}(t)$ and $\mathbf{R}(t+1)$ are the state probability matrices at time t and $t+1$ and the identity matrix $\mathbf{I} \in \mathbb{R}^{n \times n}$ can be treated as the uniform prior for all the vertices. Building on this RWR model, we make a few revisions to adapt to the practical problem of legislative prediction.

For the legislator graph, we note that each legislator has different connectivity with others, resulting in different political influence [11]. Instead of using the uniform prior $\mathbf{I}$, here we propose to use the normalized vertex degree $\mathbf{F}_{(x)} = diag\left[f_{(x)_l}, \ldots, f_{(x)_l}, \ldots, f_{(x)_L}\right]$ as the prior for each legislator, where

$$f_{(x)_l} = \frac{d_{(x)_l}}{\sum_m d_{(x)_m}}, \quad l = 1, \ldots, L. \tag{3.11}$$

The transition probability $\mathbf{P}_{(x)}$ is computed as the row-normalized weight matrix: $\mathbf{P}_{(x)} = \mathbf{D}_{(x)}^{-1}\mathbf{W}_{(x)}$. Then we obtain the following transition equation:

$$\mathbf{R}_{(x)}(t+1) = \alpha \mathbf{P}_{(\mathbf{x})} \mathbf{R}_{(\mathbf{x})}(t) + (1-\alpha)\mathbf{F}_{(x)}, \tag{3.12}$$

The steady-state probability, denoted $\mathbf{R}_{(x)} = \mathbf{R}_{(x)}(\infty) = \{r_{(x)_{lm}}\}$, can be easily derived by solving the above discrete difference equations with $t \to \infty$

$$\mathbf{R}_{(x)} = (1-\alpha)\left(\mathbf{I} - \alpha \mathbf{P}_{(x)}\right)^{-1} \mathbf{F}_{(x)}. \tag{3.13}$$

Each element $r_{(x)_{lm}}$ represents the political relevance score of the legislator x_m with respect to x_l. Note that $r_{(x)_{lm}} \neq r_{(x)_{ml}}$ and $\mathbf{R}_{(x)}$ is asymmetric.

Similarly, for the bill graph, we set the prior $\mathbf{F}_{(y)}$ and transition probability $\mathbf{P}_{(y)}$ as:

$$\begin{aligned} \mathbf{F}_{(y)} &= diag\left[f_{(y)_l}, \ldots, f_{(y)_l}, \ldots, f_{(y)_N}\right], \\ f_{(y)_n} &= \frac{d_{(y)_n}}{\sum_k d_{(y)_k}}, \quad n = 1, \ldots, N, \\ \mathbf{P}_{(y)} &= \mathbf{D}_{(y)}^{-1}\mathbf{W}_{(y)}, \end{aligned} \tag{3.14}$$

and the final relevance matrix $\mathbf{R}_{(y)}$ derived from the steady-sate is computed similar to (3.13):

$$\mathbf{R}_{(y)} = (1-\beta)\left(\mathbf{I} - \beta \mathbf{P}_{(y)}\right)^{-1} \mathbf{F}_{(y)}, \tag{3.15}$$

where $1 - \beta$ is the restart probability for the RWR in $\mathcal{G}_{(y)}$.

In summary, we propose to revise the vertex prior and transition probability to adapt the standard random walk with restart model to our problem. Then through performing RWR over the legislator graph and bill graph independently, we are able to derive two relevance matrices, $\mathbf{R}_{(x)}$ representing the political relevance among legislators and $\mathbf{R}_{(y)}$ representing semantic relevance of bills.

3.2 Random Walks across Bipartite Graph Given the derived relevance matrices $\mathbf{R}_{(x)}$ and $\mathbf{R}_{(y)}$

from the random walk over the legislator cosponsorship graph and the bill semantic similarity graph, the next step is to predict the possible link from a legislator x_l to a bill y_n. From the view of random walk, the goal is to estimate the transition chance of a random walker starting from vertex x_l and transiting to y_n. However, different from the random walk model used in the previous subsection, where the walker only performs random transitions in a unipartite graph, here the random walker has to cross a bipartite graph through the existing vote links. There are two possible paths for a random walker across the vertices of the bipartite graph, i.e. transiting from legislator x_l to bill y_n,

1. **Political relevance-based transition**. Based on the political relevance information, the random walker first performs transition from x_l to x_m, where x_m has an observed vote link $e_{(xy)_{mn}}$. Then the walker can easily transit to y_n through the existing vote link.

2. **Semantic relevance-based transition**. The random walker first transits from x_l to y_k based on the existing vote link $e_{(xy)_{lk}}$. Then the random walk is performed within $\mathcal{G}_{(y)}$, resulting in a jump from y_k to y_n based on the semantic relevance.

Note that the above two transition paths are related to the two assumptions we made earlier in this section. Finally, the estimation of the transition chance involves a heterogeneous graph, across two types of vertices through three types of edges.

Before providing the formulation of the above random walks over a bipartite graph, we first define the transition probability $\mathbf{P}_{(xy)} = \{p_{(xy)_{ln}}\}$ from the given vote links $\mathbf{E}_{(xy)} = \{e_{(xy)_{ln}}\}$ and weights $\mathbf{W}_{(xy)} = \{w_{(xy)_{ln}}\}$. Similar to the unipartite graph-based random walk, the transition probability for the bipartite graph is computed as the row-normalized weight matrix. Since we derive two bipartite graphs $\mathcal{G}^{yea}_{(xy)}$ and $\mathcal{G}^{nay}_{(xy)}$ for the *yea* and *nay* votes separately, here we accordingly have two transition probability matrices $\mathbf{P}^{yea}_{(xy)} = \{p^{yea}_{(xy)_{ln}}\}$ and $\mathbf{P}^{nay}_{(xy)} = \{p^{nay}_{(xy)_{ln}}\}$

$$\begin{aligned}\mathbf{P}^{yea}_{(xy)} &= \mathbf{D}^{yea\ -1}_{(xy)}\mathbf{W}^{yea}_{(xy)}\\ \mathbf{P}^{nay}_{(xy)} &= \mathbf{D}^{nay\ -1}_{(xy)}\mathbf{W}^{nay}_{(xy)}\end{aligned} \tag{3.16}$$

where the elements $p^{yea}_{(xy)_{ln}}$ and $p^{nay}_{(xy)_{ln}}$ represent the transition probability from x_l to y_n over graphs $\mathcal{G}^{yea}_{(xy)}$ and $\mathcal{G}^{nay}_{(xy)}$, respectively.

Now we consider the above transition paths and estimate the transition probabilities $p^{yea}_{(xy)_{ln}}$ and $p^{nay}_{(xy)_{ln}}$ of one step random walk over $\mathcal{G}^{yea}_{(xy)}$ and $\mathcal{G}^{nay}_{(xy)}$ as:

$$\begin{aligned}p^{yea}_{(xy)_{ln}} &= \gamma\sum_m r_{(x)_{lm}} p^{yea}_{(xy)_{mn}}\\ &+ (1-\gamma)\sum_k p^{yea}_{(xy)_{lk}} r_{(y)_{kn}}\\ p^{nay}_{(xy)_{ln}} &= \gamma\sum_m r_{(x)_{lm}} p^{nay}_{(xy)_{mn}}\\ &+ (1-\gamma)\sum_k p^{nay}_{(xy)_{lk}} r_{(y)_{kn}}\end{aligned} \tag{3.17}$$

where the first summation gives the transition probability from x_l to y_n through the political relevance path and the second summation gives the transition probability via the semantic relevance path. The coefficient $0 \le \gamma \le 1$ is the probability that the random walker will take the first transition path. We can further give the matrix form of the above equation, showing the update of transition probability of vote links from time t to time $t+1$,

$$\begin{aligned}\mathbf{P}^{yea}_{(xy)}(t+1) &= \gamma\mathbf{R}_{(x)}\mathbf{P}^{yea}_{(xy)}(t) + (1-\gamma)\mathbf{P}^{yea}_{(xy)}(t)\mathbf{R}_{(y)}\\ \mathbf{P}^{nay}_{(xy)}(t+1) &= \gamma\mathbf{R}_{(x)}\mathbf{P}^{nay}_{(xy)}(t) + (1-\gamma)\mathbf{P}^{nay}_{(xy)}(t)\mathbf{R}_{(y)}\end{aligned} \tag{3.18}$$

Due to the existence of the bipartite graph, the above random walk rule is significantly different than the one over unipartite graph. For instance, for a non-bipartite based random walk, the final distribution when $t \to \infty$ tends to a stationary distribution. However, for a bipartite graph with bipartition $\{\mathbf{V}_{(x)}, \mathbf{V}_{(y)}\}$, the final distribution could oscillate between the prior distributions of $\mathbf{V}_{(x)}$ and $\mathbf{V}_{(y)}$ without achieving a steady-state [21]. Since the formulation of (3.18) involves both transitions within each unipartite graph and across the bipartite graph, the final results are even more complicated to state. In addition, the prediction of new vote links changes the structure of the bipartite graph, and makes efforts to derive the final distribution of such steady-state intractable. Therefore, we develop a new iterative approach to gradually produce the vote prediction results.

3.3 Iterative Prediction of Vote Links

Note that we construct two legislator-bill vote graphs $\mathcal{G}^{yea}$ and $\mathcal{G}^{nay}$ using *yea* and *nay* votes separately, as described in Section 2.4. Accordingly, two separate random walks should be performed over these two graphs using the rules in (3.18). For one step of random walk from time t to $t+1$, we can derive two new transition matrices $\mathbf{P}^{yea}_{(xy)}(t+1)$ and $\mathbf{P}^{nay}_{(xy)}(t+1)$, where the elements $p^{yea}_{(xy)_{ln}}$, $p^{nay}_{(xy)_{ln}}$ measure the chance of *yea* and *nay* result

Algorithm 1 Iterative Vote Link Prediction through Random Walk over a Heterogeneous Graph

```
Initialization:
Construct legislator graph $\mathcal{G}_{(x)} = \{\mathbf{V}_{(x)}, \mathbf{E}_{(x)}, \mathbf{W}_{(x)}\}$, bill graph $\mathcal{G}_{(y)} = \{\mathbf{V}_{(y)}, \mathbf{E}_{(y)}, \mathbf{W}_{(y)}\}$, and vote graphs $\mathcal{G}^{yea}_{(xy)} = \{\mathbf{V}_{(x)} \cup \mathbf{V}_{(y)}, \mathbf{E}^{yea}_{(xy)}, \mathbf{W}^{yea}_{(xy)}\}$ and $\mathcal{G}^{nay}_{(xy)} = \{\mathbf{V}_{(x)} \cup \mathbf{V}_{(x)}, \mathbf{E}^{nay}_{(xy)}, \mathbf{W}^{nay}_{(xy)}\}$;
Compute transition probabilities $\mathbf{P}_{(x)}$ and $\mathbf{P}_{(y)}$;
Derive steady-state distributions $\mathbf{R}_{(x)}$ and $\mathbf{R}_{(y)}$;
iteration counter $t = 0$;
Compute the initial bipartite graph transition probability matrices $\mathbf{P}^{yea}_{(xy)}(t)$ and $\mathbf{P}^{nay}_{(xy)}(t)$ using (3.16);
repeat
    Compute the bipartite graph transition probability matrices $\mathbf{P}^{yea}_{(xy)}(t+1)$ and $\mathbf{P}^{nay}_{(xy)}(t+1)$ using (3.18);
    Estimate the posterior probabilities $P^{yea}_{ln}$ and $P^{nay}_{ln}$ for each possible vote link using (3.19);
    For each possible prediction vote link, estimate the mutual information $I(x_l, y_n)$ of the legislator and bill vertices (3.20):
    Update vote link $\mathbf{W}^{yea}_{(xy)}$ or $\mathbf{W}^{yea}_{(xy)}$ from the prediction with maximum mutual information, as in (3.22);
    Update iteration counter: $t = t + 1$;
    Compute $\mathbf{P}^{yea}_{(xy)}(t)$ or $\mathbf{P}^{nay}_{(xy)}(t)$ with the new vote graphs;
until All missing vote links are completed
Output:
The complete vote links $\mathbf{W}^{yea}_{(xy)}$ and $\mathbf{W}^{nay}_{(xy)}$.
```

for x_l voting on y_n. In the proposed iterative procedure of vote prediction, we gradually complete the vote links in a greedy way, in which only the most confident prediction are used to create the new vote links. To achieve this, we normalize the transition probabilities over the two graphs to derive the posterior probability $P_{ln} = \{P^{yea}_{ln}, P^{nay}_{ln}\}$, representing the probability for x_l to vote y_n with *yea* and *nay*, respectively:

$$P^{yea}_{ln} = \frac{p^{yea}_{(xy)_{ln}}}{p^{yea}_{(xy)_{ln}} + p^{nay}_{(xy)_{ln}}} \qquad (3.19)$$

$$P^{nay}_{ln} = \frac{p^{nay}_{(xy)_{ln}}}{p^{yea}_{(xy)_{ln}} + p^{nay}_{(xy)_{ln}}}$$

Then the uncertainty of the prediction of vote from x_l to y_n is measured by the mutual information $I(x_l, y_n)$ as:

$$I(x_l, y_n) = \sum_{x_l} \sum_{y_n} \frac{p(x_l, y_n)}{p(x_l)p(y_n)}. \qquad (3.20)$$

where $p(x_l, y_n)$ is the joint probability distribution and the marginal probabilities are $p(x_l), p(y_n)$. In our formulation, $p(x_l, y_n)$ refers to the probability of the *yea* and *nay* vote edge as shown in (3.19). The value $p(x_l)$ is interpreted as the probability of x_l giving *yea* and *nay* votes and $p(y_n)$ is the probability of y_n receiving *yea* and *nay* votes, which are estimated as the priors in (2.7). Therefore, the mutual information $I(x_l, y_n)$ is approximately computed as

$$I(x_l, y_n) \approx P^{yea}_{ln} \log \frac{P^{yea}_{ln}}{p^{yea}_{(x)_l} p^{yea}_{(y)_n}} + P^{nay}_{ln} \log \frac{P^{nay}_{ln}}{p^{nay}_{(x)_l} p^{nay}_{(y)_n}}. \qquad (3.21)$$

For each possible prediction of vote links, the above mutual information is computed and the one associated with maximum value is chosen. Then a new vote link is accordingly generated to $\mathcal{G}^{yea}_{(xy)}$ or $\mathcal{G}^{nay}_{(xy)}$ based on which one has higher posterior probability. We can write this one steep greedy assignment for vote link prediction as:

$$\begin{aligned} (3.22) \qquad & (l^*, n^*) = \arg\max_{l,n} I(x_l, y_n) \\ & P^{yea}_{l^*n^*} > P^{nay}_{l^*n^*} \Rightarrow w^{yea}_{(xy)_{l^*n^*}} = 1, \\ & P^{yea}_{l^*n^*} < P^{nay}_{l^*n^*} \Rightarrow w^{nay}_{(xy)_{l^*n^*}} = 1, \end{aligned}$$

Finally, with the updated $\mathcal{G}^{yea}_{(xy)}$ or $\mathcal{G}^{nay}_{(xy)}$, the corresponding transition matrix $\mathbf{P}^{yea}_{(xy)}$ or $\mathbf{P}^{nay}_{(xy)}$ is recomputed for the next random walk step. Through iteratively performing this random walk, where in each step the bipartite vote graph is updated with more links, we can gradually complete all the missing vote links. This heterogeneous graph-based random walk method for predicting legislative votes is summarized in Algorithm 1.

Our proposed algorithm shares features with algorithms for learning the structure of Markov random fields (MRFs), a problem in which roll call data has also been analyzed [2]. MRFs and random walk models are intimately related, as recently discussed in [8]. Greedy algorithms for structure learning use comparison tests of information-theoretic quantities such as mutual information and entropy to find the edges connecting MRF vertices like our proposed method for finding vote edges [6, 23, 32]. These greedy algorithms are optimal in a Kullback–Leibler divergence-sense for acyclic graphs such as tree-structured graphs. However, such methods are not intended for heterogeneous graphs or missing data as encountered in our legislative prediction model. In future work, we would like to further investigate the theoretical connections between the proposed RWHG model and comparison test formulations of MRF learning.

4 Empirical Evaluation

4.1 Data and Material For the roll call data, a subset of bills and votes from [13] is used in our experiments. In particular, we use the data from the two most recent congressional sessions, i.e. the 110th (January 2007 to December 2008) and 111th (January 2009 to December 2010). It contains a total of 1585 bills, 631 unique legislators who have at least one valid vote, and 638,955 *yea* or *nay* votes. Table 1 shows the statistics of the selected roll call data for each congressional session.

Bill cosponsorship information dating back to the 93rd session of Congress is accessible from the Library of Congress' Thomas database.[3] We use a version of the data curated by Fowler et al. [11, 12].[4] The pair of legislators in the data with the highest number of cosponsored bills is Edolphus Towns and Major Owens, both African-American representatives of the Democratic party from New York City. In predicting the votes of the 110th and 111th sessions, we only use cosponsorship data up to the 109th session (December 2006) to avoid overfitting.

To obtain the BOW representation of the bills, significant n-grams ($1 \leq n \leq 5$) are first extracted to construct the vocabulary. As described in [13], only the n-grams which occur in more than 0.2% and less than 15% bills are included. Finally, each legislative bill is represented as a 4743 vector, where each element indicates the counts of the corresponding n-gram appearing in the bill. For constructing the semantic similarity graph of bills, we simply use cosine function to compute the edge weights. The k-nearest neighbor approach with $k = 6$ is applied to sparsify both the legislator cosponsorship graph and the bill graph [16].

4.2 Analysis of Cosponsorship and Semantic Similarity As discussed in Section 3, two major assumptions of RWHG are the correlation between legislator cosponsorship and their voting behavior, and the correlation between semantic similarity of bills and vote results. The correlation between cosponsorship and voting behavior of legislators contributes to the first term in the random walk equation (3.22), and the correlation between the semantics of bills and votes is related to the second term in the same equation. In the following, we will validate these two assumptions by empirical analysis.

Our analysis is performed on the roll call data set from two congressional sessions, as described above. We define the similarity of vote behavior between legislators x_l and x_m as the normalized correlation, i.e. cosine angle

$$sim(x_l, x_m) = \frac{\sum_k w_{(xy)_{lk}} \cdot w_{(xy)_{mk}}}{\sqrt{\sum_k w^2_{(xy)_{lk}}} \cdot \sqrt{\sum_k w^2_{(xy)_{mk}}}}, \tag{4.23}$$

where $w_{(xy)_{lk}} = w^{yea}_{(xy)_{lk}} - w^{nay}_{(xy)_{lk}}$ and $w_{(xy)_{mk}} = w^{yea}_{(xy)_{mk}} - w^{nay}_{(xy)_{mk}}$ combine the edges from both *yea* and *nay* votes. Similarly, the similarity of the voting results received by bills y_n and y_k can be defined as

$$sim(y_n, y_k) = \frac{\sum_l w_{(xy)_{ln}} \cdot w_{(xy)_{lk}}}{\sqrt{\sum_l w^2_{(xy)_{ln}}} \cdot \sqrt{\sum_l w^2_{(xy)_{lk}}}}. \tag{4.24}$$

Session	bills	legislators	*yea*	*nay*
110	745	549	248,077	48,543
111	845	553	298,106	44,229

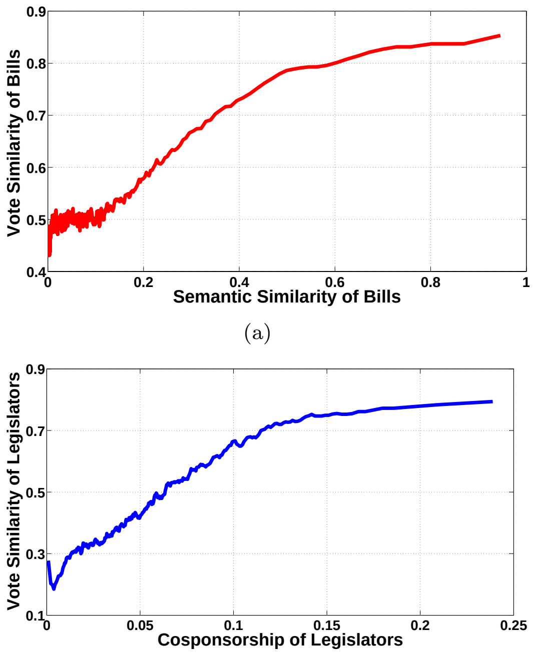
Table 1: The number of legislators, bills, *yea*, and *nay* votes in the 110th and 111th congressional sessions.

Based on these definitions, we can provide the curves of semantic similarity $w_{(y)_{nk}}$ versus the values of $sim(y_n, y_k)$ (Figure 2(a)), and cosponsorship $w_{(x)_{lm}}$ versus the values of $sim(x_l, x_m)$ (Figure 2(b)). These figures validate the effectiveness of the two assumptions: similar bills tend to receive similar votes, and legislators with high cosponsorship tend to vote similarly. In addition, Figure 2(a) presents higher values of vote similarity of bills measured as $sim(y_n, y_k) > 0.8$ when $w_{(y)_{nk}} > 0.6$, while $sim(x_l, x_m) < 0.8$ for most cases in Figure 2(b). This indicates that the semantic similarity of bills is more correlated with the votes than the cosponsorship among legislators. This observation will be further confirmed by the prediction experiments in Section 4.4, where the prediction solely based on the bill similarity achieves higher accuracy than that solely based on legislator cosponsorship.

4.3 Political Influence and Affinity Another analysis afforded by the proposed random walk model is the identification of the most influential legislators and the pair of legislators with the greatest political affinity. The most influential legislators are those whose steady-state probability $r_{(x)_{ll}}$ is highest. The top ten influential legislators in the two sessions are given in Table 2. The pair of most-correlated legislators indicated by the maximum political relevance score $r_{(x)_{lm}}$ are Russell Feingold and John Kerry in the 110th session, and Bob Graham and John Kerry in the 111th session.

[3] http://thomas.loc.gov

[4] http://jhfowler.ucsd.edu/cosponsorship.htm



(a)

(b)

Figure 2: The curves indicate: a) the semantic similarity and the similarity of the received votes of bills; b) the cosponsorship and the similarity of voting behavior of legislators.

John Kerry was one of the most influential personalities in Washington, even running for president in 2004. Feingold and Kerry were together the leading proponents of troop withdrawal from Iraq, and Bob Graham was mentioned as a possible running mate for Kerry in the 2004 presidential election. Dodd, Levin, Cochran, Lautenberg, and Warner are all long-serving leaders in the Senate from both parties. Marchant, Westmoreland, Franks, Foxx, Conaway, Barrett, and McHenry are all Southern Republicans that have not served in the House for an extended period; the American Conservative Union rates all of these representatives as being more than 95% conservative. These results make sense from the political perspective, but also suggest different dynamics in the Senate and House: influence through senior leadership in the Senate, and influence by 'young guns' in the House, and should be investigated further by political scientists.

110th Session	111th Session
Kenny Marchant	Kenny Marchant
John Kerry	John Kerry
Christopher Dodd	Carl Levin
Carl Levin	Christopher Dodd
Thad Cochran	Lynn Westmoreland
Frank Lautenberg	J. Gresham Barrett
Virginia Foxx	Patrick McHenry
Mike Conaway	Frank Lautenberg
Lynn Westmoreland	Virginia Foxx
John Warner	Trent Franks

Table 2: Legislators with the largest steady-state probability $r_{(x)_{ll}}$.

4.4 Experimental Results of Predicting Votes

Similar to the experimental setting in [13], we perform vote prediction at the session level (a 2-year period). In particular, we design two types of experiments: a) prediction for random missing votes, and b) sequential prediction. For a) type experiments, we randomly partition the roll call data from each session into 6 folds and use standard cross-validation to compute the average prediction accuracy. For b) type experiments, we use the votes from the first 21 months to predict the votes in the remaining 3 months of each session. The final reported accuracy is the average accuracy in the two sessions. We set the same parameter $\gamma = 6/7$ for both experiments.

We compared with the methods used in [13], including a simple *yea* model predicting all votes as *yea*, two text based regression models, i.e. ridge regression (LARS) and Lasso (L2), and the ideal point topic model (IPTM).[5] In addition, we designed two baseline methods using nearest neighbor (NN) method. Briefly speaking, to estimate the vote $w_{(xy)_{ln}}$ from x_l on y_n, we average results of the known votes from the k most similar legislators on y_l or the known votes from x_l on the k ($k = 6$ in our experiments) most similar bills. The prediction equations of $w_{(xy)_{ln}}$ are written as:

$$w_{(xy)_{ln}} = \text{sgn} \sum_{x_m \in \mathcal{N}(x_l)} w_{(xy)_{mn}} \tag{4.25}$$

$$w_{(xy)_{ln}} = \text{sgn} \sum_{y_k \in \mathcal{N}(y_n)} w_{(xy)_{lk}} \tag{4.26}$$

where $\mathcal{N}(x_l)$ and $\mathcal{N}(y_n)$ refer to the k-nearest neighbors of x_l and y_n in $\mathbf{V}_{(x)}$ and $\mathbf{V}_{(y)}$, respectively. Eq. (4.25) is named "NN_Legislator" since it essentially performs nearest neighbor search on the legislator space and (4.26) is "NN_Bill" since it estimates the votes using the existing votes from the most similar bills.

[5] Under similar experimental settings, we compare with the methods using the same parameters or the best performance reported in the literature.

Method	(a)-Accuracy %	(b)-Accuracy %
yea	85.48	86.98
LARS	82.2	NA[7]
L2	89.7	88.1
IPTM	88.7	87.0
NN_Legislator	85.42	NA
NN_Bill	89.06	88.35
RWHG	**91.10**	**90.36**

Table 3: The accuracy of a) type experiments: the prediction on random missing votes, and b) type experiments: sequential prediction. The compared methods include *yea* model, ridge regression (LARS), Lasso (L2), Ideal Point Topic Model (IPTM) [13], k-Nearest Neighbor of Legislators (NN_Legislator), k-Nearest Neighbor of Bills (NN_Bill), and the proposed random walk over heterogeneous graph (RWHG).

Table 3 shows the experimental results on the two prediction tasks. In both tasks, the proposed RWHG method achieved the best performance. For instance, the baseline *yea* model generates 85.48% accurate votes while RWHG can predict 91.10% of the votes, which means RWHG can correctly predict around $42,500$ more votes in the two-session period. The NN_Bill method has strong prediction performance indicating that the voting behavior is fairly consistent across these two sessions for the legislators and they tend to give the same votes for similar bills. However, solely considering cosponsorship does not provide satisfactory performance since NN_Legislator produces fair performance in predicting random missing votes. [6] It indicates that a strong cosponsorship relationship between legislators does not guarantee a similar political position across all legislative issues. This also further confirms our argument about the unique feature of the United States Congress discussed earlier in Section 1, where the legislators are not bound to vote in lockstep with either the party they belong to, or with political and social relations. However, the experiments clearly show that the proposed RWHG method can synergistically combine the cosponsorship information with bill semantics to achieve the unique strength for predicting legislative votes.

5 Significance and Conclusions

This paper is dedicated to developing a new random walk model on a heterogeneous graph for predicting legislative roll call data. We first summarize the main contribution of this paper and then introduce the related work.

5.1 Main Contributions The main contributions of this paper include the following:

1. **Unique heterogeneous graph formulation**. We proposed a unique formulation of heterogeneous graph for legislative prediction. Specifically, we treat both the legislators and the bills as vertices and accordingly generate three subgraphs. Two unipartite subgraphs with homogeneous vertices and edges connecting legislators and bills separately using the cosponsorship information for legislators and semantic similarity for bills. A special bipartite graph containing directed edges between legislators and bills represents the vote information. In particular, we treat *yea* and *nay* votes separately and construct two independent vote graphs accordingly.

2. **Two-stage random walk with iterative vote prediction**: Based on the above heterogeneous graph formulation, a two-stage random walk model is proposed to perform vote link prediction. In the first stage, two independent random walks are launched over the legislator cosponsorship graph and the bill semantic similarity graph to derive the relevance between vertices when achieving steady-state. In the second stage, we perform random walk across the directed vote links and iteratively generate new vote links from the most confident predictions.

To our best knowledge, the heterogeneous graph formulation of the legislative roll call data has not been proposed in any previous literature in either quantitative political science or data mining fields. The two-stage random walk model over heterogeneous graph is also significantly different from existing random walk models. Besides the previous work in political science area introduced in Section 1, we summarize the related work from technical perspective in below.

5.2 Related Work Motivated by the real data generated in different domains, heterogeneous graph-based formulations have attracted much attention in the past years. For instance, ranking and classifying the vertices of a heterogeneous graph are two well-formulated problems from real applications, such as author and docu-

[6] Since there is no existing votes for the bills in the sequential prediction task, NN_Legislator doest not generate meaningful results.

[7] LARS model does not accomplish the sequential prediction task due to some computational issues [13].

ments co-ranking [39] and identifying research communities from bibliographic data [17]. A very recent study performs both random walk and propagation over a heterogeneous graph to improve topic modeling of documents [10]. Compared to these existing approaches, we formulate the vote prediction as a link prediction problem via random walk over a heterogeneous graph, instead of either vertex ranking or classification.

Although link prediction has been widely studied for social networks and the Internet, most of the existing methods are developed for graphs with either homogeneous vertices or edges [19, 15, 20, 33]. Recently, Sun et al. proposed to learn optimal weights to combine heterogeneous topological features and then build a logistic regression model to predict co-authorship links [31].

As a special case of a finite Markov chain, random walk on graphs arises in many domains. Besides the standard random walk [21] and the random walk with restart model [25, 34], various random walk models have been developed for graphs with homogeneous vertices or bipartite graphs with homogeneous edges, including the well-known PageRank algorithm [4], multiple random walks [9], and random walk on bipartite graph [30]. Another related bi-relational graph-based random walk formulation for the application of image annotation was recently proposed [36]. In [36], the authors deal with the label prediction problem in a semi-supervised learning paradigm and the random walk is performed over the entire bi-relational graph, instead of our two-stage random walk across directed vote edges.

In the above heterogeneous graph formulation, two unipartite graphs can be viewed as relational data among legislators and bills. Therefore, multiple relational learning is another related topic. However, most relational learning methods tend to recover the underlying clusters of the data [24], or reveal and visualize the low-dimensional structures of the data [28, 18, 5].

Finally, if we treat the legislators as users, the bills as items, and the votes as binary preference scores, then the prediction of votes can be viewed as a special case of recommendation problems [1]. The use of cosponsorship graph between legislators then becomes a typical user-based collaborative filtering model, which predicts the votes of a legislator by collecting votes information from other legislators. Similarly, the use of bill semantic relations for vote prediction can be categorized as a item-based collaborative filtering problem. Jointly learning the user and item patterns has shown significant effectiveness in real applications [22, 37].

5.3 Conclusions and Future Work In this article, we proposed a novel method for the application of legislative prediction through random walks on a heterogeneous graph. We performed vote prediction tasks on real roll call data from the 110th and 111th congressional sessions. The experimental results clearly show the superior performance of the proposed RWHG method, compared with the state of the art. In addition, we conducted an empirical study and demonstrated that the semantic information of bills provides relatively stronger clues for predicting the votes than the cosponsorship information between legislators. Note that the BOW-based semantic similarity could be enhanced based on advanced topic modeling algorithms; it was our intent to consider a new prediction model rather than dwell on the fine points of text analysis. We could also use additional features in defining legislator similarity. Finally, it is interesting to see that the most influential legislators and the most strongly bound legislator pairs discovered by the proposed model can be well interpreted and supported by the political reality.

The proposed method addresses legislative prediction, but it is indeed a general formulation and can be extended to other applications. Therefore, one of the directions of our future work is to extend this method to other domains, such as recommendation applications. Another possible future direction is to enrich this method for temporal analysis of a longer range of roll call data [35].

References

[1] G. Adomavicius and A. Tuzhilin, *Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions*, IEEE Trans. Knowl. Data Eng. **17** (2005), no. 6, 734–749.

[2] O. Banerjee, L. El Ghaoui, and A. d'Aspremont, *Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data*, J. Mach. Learn. Res. **9** (2008), 485–516.

[3] P. A. Beck, R. J. Dalton, S. Greene, and R. Huckfeldt, *The social calculus of voting: Interpersonal, media, and organizational influences on presidential choices*, Am. Polit. Sci. Rev. **96** (2002), no. 1, 57–73.

[4] S. Brin and L. Page, *The anatomy of a large-scale hypertextual web search engine*, Proc. Int. Conf. World Wide Web (Brisbane), Apr. 1998, pp. 107–117.

[5] M. Cammarano, X. L. Dong, B. Chan, J. Klingner, J. Talbot, A. Halevy, and P. Hanrahan, *Visualization of heterogeneous data*, IEEE Trans. Vis. Comput. Graphics **13** (2007), no. 6, 1200–1207.

[6] C. K. Chow and C. N. Liu, *Approximating discrete probability distributions with dependence trees*, IEEE Trans. Inf. Theory **IT-14** (1968), no. 3, 462–467.

[7] J. Clinton, S. Jackman, and D. Rivers, *The statistical analysis of roll call data*, Am. Polit. Sci. Rev. **98** (2004), no. 2, 355–370.

[8] W. W. Cohen, *Graph walks and graphical models*, Tech. Report CMU-ML-10-102, School Comp. Sci., Carnegie Mellon Univ., 2010.

[9] C. Cooper, A. Frieze, and T. Radzik, *Multiple random walks in random regular graphs*, SIAM J. Discrete Math. **23** (2009), no. 4, 1738–1761.

[10] H. Deng, J. Han, B. Zhao, Y. Yu, and C. X. Lin, *Probabilistic topic models with biased propagation on heterogeneous information networks*, Proc. ACM SIGKDD Conf. Knowl. Disc. Data Min. (San Diego, CA), Aug. 2011, pp. 1271–1279.

[11] J. H. Fowler, *Connecting the Congress: A study of cosponsorship networks*, Polit. Anal. **14** (2006), no. 4, 456–487.

[12] ______, *Legislative cosponsorship networks in the US House and Senate*, Soc. Networks **28** (2006), no. 4, 454–465.

[13] S. M. Gerrish and D. M. Blei, *Predicting legislative roll calls from text*, Proc. Int. Conf. Mach. Learn. (Bellevue, WA), Jun.–Jul. 2011, pp. 489–496.

[14] Z. S. Harris, *Distributional structure*, Word **10** (1954), no. 2–3, 146–162.

[15] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki, *Link prediction using supervised learning*, Proc. SIAM Conf. Data Min. Workshops (Bethesda, MD), Apr. 2006.

[16] T. Jebara, J. Wang, and S. F. Chang, *Graph construction and b-matching for semi-supervised learning*, Proc. Int. Conf. Mach. Learn. (Montreal), Jun. 2009, pp. 441–448.

[17] M. Ji, J. Han, and M. Danilevsky, *Ranking-based classification of heterogeneous information networks*, Proc. ACM SIGKDD Conf. Knowl. Disc. Data Min. (San Diego, CA), Aug. 2011, pp. 1298–1306.

[18] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada, and N. Ueda, *Learning systems of concepts with an infinite relational model*, Proc. Nat. Conf. Artif. Int. (Boston, MA), vol. 1, Jul. 2006, pp. 381–386.

[19] D. Liben-Nowell and J. Kleinberg, *The link-prediction problem for social networks*, J. Am. Soc. Inf. Sci. Tech. **58** (2007), no. 7, 1019–1031.

[20] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla, *New perspectives and methods in link prediction*, Proc. ACM SIGKDD Conf. Knowl. Disc. Data Min. (Arlington, VA), Jul. 2010, pp. 243–252.

[21] L. Lovász, *Random walks on graphs: A survey*, Combinatorics, Paul Erdos is Eighty **2** (1993), no. 1, 1–46.

[22] P. Melville, R. J. Mooney, and R. Nagarajan, *Content-boosted collaborative filtering for improved recommendations*, Proc. Nat. Conf. Artif. Int. (Edmonton), Jul.–Aug. 2002, pp. 187–192.

[23] P. Netrapalli, S. Banerjee, S. Sanghavi, and S. Shakkottai, *Greedy learning of Markov network structure*, Proc. Allerton Conf. Commun. Control Comput. (Monticello, IL), Sep.–Oct. 2010, pp. 1295–1302.

[24] J. Neville, M. Adler, and D. Jensen, *Clustering relational data using attribute and link information*, Proc. IJCAI Text Min. Link Anal. Workshop (Acapulco), Aug. 2003.

[25] J. Y. Pan, H. J. Yang, C. Faloutsos, and P. Duygulu, *Automatic multimedia cross-modal correlation discovery*, Proc. ACM SIGKDD Conf. Knowl. Disc. Data Min. (Seattle, WA), Aug. 2004, pp. 653–658.

[26] K. T. Poole and H. Rosenthal, *Patterns of congressional voting*, Am. J. Polit. Sci. **35** (1991), no. 1, 228–278.

[27] S. A. Rice, *The political vote as a frequency distribution of opinion*, J. Am. Stat. Assoc. **19** (1925), no. 145, 70–75.

[28] M. Roland and H. Geoffrey, *Multiple relational embedding*, Adv. Neural Inf. Process. Syst. 17, MIT Press, Cambridge, MA, 2005, pp. 913–920.

[29] D. Sheldon, *Graphical multi-task learning*, NIPS Structured Input Structured Output Workshop (Whistler), Dec. 2008.

[30] J. Sun, H. Qu, D. Chakrabarti, and C. Faloutsos, *Neighborhood formation and anomaly detection in bipartite graphs*, Proc. IEEE Int. Conf. Data Min. (Houston, TX), Nov. 2005.

[31] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, *Co-author relationship prediction in heterogeneous bibliographic networks*, Proc. Int. Conf. Adv. Soc. Netw. Anal. Min. (Kaohsiung), Jul. 2011, pp. 121–128.

[32] V. Y. F. Tan, A. Anandkumar, and A. S. Willsky, *Learning high-dimensional Markov forest distributions: Analysis of error rates*, J. Mach. Learn. Res. **12** (2011), 1617–1653.

[33] B. Taskar, M.-F. Wong, P. Abbeel, and D. Koller, *Link prediction in relational data*, Adv. Neural Inf. Process. Syst. 16, MIT Press, Cambridge, MA, 2004.

[34] H. Tong, C. Faloutsos, and J.-Y. Pan, *Fast random walk with restart and its applications*, Proceedings of the Sixth International Conference on Data Mining (Washington, DC), 2006, pp. 613–622.

[35] E. Wang, D. Liu, J. Silva, D. Dunson, and L. Carin, *Joint analysis of time-evolving binary matrices and associated documents*, Adv. Neural Inf. Process. Syst. 23, MIT Press, Cambridge, MA, 2011, pp. 2370–2378.

[36] H. Wang, H. Huang, and C. Ding, *Image annotation using bi-relational graph of images and semantic labels*, Proc. IEEE Conf. Comp. Vis. Pattern Recogn. (Colorado Springs, CO), Jun. 2011, pp. 793–800.

[37] J. Wang, A. P. de Vries, and M. J. T. Reinders, *Unifying user-based and item-based collaborative filtering approaches by similarity fusion*, Proc. ACM SIGIR Conf. Res. Dev. Inf. Retrieval (Seattle, WA), Aug. 2006, pp. 501–508.

[38] J. Wang, T. Jebara, and S.-F. Chang, *Graph transduction via alternating minimization*, Proc. Int. Conf. Mach. Learn. (Helsinki), Jul. 2008, pp. 1144–1151.

[39] D. Zhou, S. A. Orshanskiy, H. Zha, and C. L. Giles, *Co-ranking authors and documents in a heterogeneous network*, Proc. IEEE Int. Conf. Data Min. (Omaha, NE), Oct. 2007, pp. 739–744.

[40] X. Zhu, *Semi-supervised learning literature survey*, Tech. Report 1530, Dept. Comp. Sci., Univ. Wisconsin, 2005.