A Dynamical System on Bipartite Graphs

Kishore Papineni
Google inc.
papineni@google.com

Pratik Worah
Google inc.
pworah@google.com

ABSTRACT

This paper poses a non-linear dynamical system on bipartite graphs and shows its stability under certain conditions. The dynamical system changes the weights on the nodes of the graph in each time step. The underlying weight transformation is non-linear, motivated by information gain in a document retrieval setting. Stability analysis of this problem is therefore more involved than that of PageRank-like algorithms. We show convergence using methods from Lyapunov theory and also provide some examples of how the algorithm performs when ranking keywords and sentences in a set of documents.

KEYWORDS

Ranking sentences, Information gain, Lyupanov theory

ACM Reference Format:
Kishore Papineni and Pratik Worah. 2018. A Dynamical System on Bipartite Graphs. In *The 27th ACM International Conference on Information and Knowledge Management (CIKM '18), October 22–26, 2018, Torino, Italy.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3269206.3269271

1 INTRODUCTION

1.1 Motivation

Suppose we are to summarize a corpus of documents that do not have explicit hyperlinks among them. Arguably, a good summary would contain important concepts from the corpus and arise from important sentences in these documents. Given a collection of words and sentences from a corpus of related documents, our goal then is to associate a weight with each sentence and each word (or phrase) which can be interpreted as their importance in the corpus.

From the viewpoint of natural language understanding, words that occur too frequently are usually not important. For example, words such as 'the' are usually not important and typically will not be the keywords of a document. This is the informal intuition behind Inverse Document Frequency (IDF) [11]. However, IDF is the highest for rarest words; but just as very frequent words in a document are unimportant, very rare words are also not expected to be keywords of a document. Therefore, neither too frequent nor very rare words are going to be keywords in a document, and the sweet-spot lies somewhere in the middle.

The above informal intuition was first described in [5], and was captured in [10] in a mathematical setting, which defined the *information gain* of a word as the Kullback-Leibler divergence of the word's relative frequency of occurence in a document. In other words, the information gain from a given word is

$$g_w := \mathrm{f}_w \cdot (\mathrm{f}_w - 1 - \ln(\mathrm{f}_w)),$$

where f_w denotes the fraction of sentences containing the word w. Note that the gain, as a function of f, is a positve and inverted U-shaped function in $[0, 1]$ (the range of f), with a maximum near 0.2. Of course, there is nothing magical about the constant 0.2 i.e., ideal keywords are not necessarily those that occur with a frequency of 20%. This particular fraction is merely an artifact of using equal weights on all sentences to measure the Kullback-Leibler divergence. Therefore, assigning the correct weights to sentences is important before we can identify keywords. In particular, long sentences with many keywords should intuitively be more important from a natural language understanding and summarization point of view than short sentences which contain mostly filler words. Hence, we are now facing a circular problem: We need to identify important sentences before we can identify important words, but to identify important sentences we need to identify important words.

The circular nature of our problem suggests the following fixed-point type algorithm to compute a weighted version of information gain: Assign a (reasonable) intial weight to all words and sentences. For each word w and each sentence s, compute sum_w and sum_s – the sum of weights of sentences that contain w and the sum of weights of words in s. Update the weight of each w and s as follows:

$$\mathrm{Weight}_w \leftarrow \mathrm{sum}_w \cdot (\mathrm{sum}_w - 1 - \ln(\mathrm{sum}_w)),$$

$$\mathrm{Weight}_s \leftarrow \mathrm{sum}_s \cdot (\mathrm{sum}_s - 1 - \ln(\mathrm{sum}_s)).$$

Repeat the updates until the weights of all vertices do not change significantly between two iterations.

Mathematically, it is not clear under what conditions i.e., for what kinds of corpus and what initial choice of weights, does the above algorithm converge to a "optimal" weight distribution. Intuitively, it is not clear how the important sentences and keywords generated by using such a weighted information gain qualitatively compare with some other conventional approach, or even in some absolute sense.

This paper applies basic linear algebra and Lyapunov theory to prove convergence of the above iterated update algorithm when the underlying word-sentence graph obeys certain eigenvalue constraints. Moreover, we have also implemented the above fixed-point algorithm and ran it on a small corpus of documents to see if there was a natural qualitative difference between the sentences produced with weights and using a conventional method.

Finally, we observe that there is no reason why the algorithm can't be applied to more general settings. For instance, instead of a collection of words and sentences, we can apply it to any container which is a collection of tokens – a corpus of images and the entities within those images; a corpus of videos and the entities within those videos, and so on.

CIKM '18, October 22–26, 2018, Torino, Italy

ACM ISBN 978-1-4503-6014-2/18/10.
https://doi.org/10.1145/3269206.3269271

1.2 Related Work

The fixed-point approach described in the introduction implicitly assumes a graph structure between sentences and words in a document. In the underlying graph, every word and sentence corresponds to a vertex, and a word is linked to a sentence by an undirected edge, if it occurs in the sentence. There is already a significant amount of work which applies fixed-point algorithms over an underlying graph structure to compute a rank or score which has some relevance in the real world. For example, the initial papers [9], [4] in the areas of citation analysis, social networks and analyzing the link structure of the world-wide web, use similar ideas. More importantly, in the area of lexical analysis and ranking of documents there has been significant work using such graph based models, see for example [6], [8] and the book [7].

In [9], the authors work on a graph where the vertices are webpages and the edges reflect the link structure of the web. If $\delta_+(v)$ and $\delta_-(v)$ denote the out-degree and in-degree neighbours of a vertex v, then the score of v (on which its Page-rank is based) is defined as:

$$S(v) := (1-c) + c \cdot \sum_{u \in \delta_-(v)} \frac{S(u)}{|\delta_+(u)|}, \tag{1}$$

where the constant c is around 0.8. The idea is to iterate until the score stabilizes and then use the scores to rank web-pages. Note that the intial scores do not matter too much as the scores converge rapidly [9].

[4] introduced a similar algorithm (HITS), which ranked webpages by implicitly ascribing a bipartitie structure to the graph – vertices correspond to "authority" pages (corresponding to a large δ_-) or hub pages (corresponding to a large δ_+). Therefore, we have two scores for each vertex: an authority score and a hub score – both calculated as a weighted linear sum over their neighbours.

More relevant to us, [8] introduced TextRank an unsupervised procedure to extract and rank keywords and sentences, or more generally text units, from a lexical corpus. Their approach is similar to Page's approach, but now the graph vertices correspond to text units (for example, sentences or phrases) and edges reflect some semantic or syntactic connection between text units. While the algorithm and score used is similar to that in [9], one important difference was that the edges can be weighed and so their counterpart of Equation 1 would be:

$$S(v) := (1-c) + c \cdot \sum_{u \in \delta_-(v)} \frac{W_{uv} \cdot S(u)}{\sum_{w \in \delta_+(u)} W_{uw}}, \tag{2}$$

where W is the weight matrix of the underlying graph and the constant c may be chosen depending upon the exact scenario at hand.

While the viewpoints in each case may be different, the relevant theme, at least for us, remains the same i.e., there is an iterated local computation on a graph that leads to a score for each vertex, and this score is interesting from the perspective of some real world problem. However, in all the above cases the score is computed using a linear function, as in Equations 1 and 2, for example. Linearity together with high connectivity in the graph structure leads to rapid mixing, thus convergence is ensured and is typically very fast. This linearity is in contrast to our situation, where we iteratively update based on a Kullback-Leibler divergence type of function– a non-linear function, and so convergence does not follow from earlier ideas. Therefore, we go back to first principles and analyze convergence of the underlying dynamical system using Lyapunov theory.

2 RESULTS

In this section, we recap the mathematical model for our problem and then provide a proof of Theorem 1 and 3, our main mathematical results. We provide experimental verification in the next section.

We are given an undirected bipartite graph $G \equiv (X, Y, W)$, where X and Y denote the vertex sets and W denotes the edge weights i.e., $W_{uv} \in \mathbb{R}$ is the weight of the edge connecting vertices u and v in G. With $|X| = n$ and $|Y| = m$, W is a $(n+m)\times(n+m)$ block offdiagonal symmetric matrix[1]. Let $x_u(t) \in \mathbb{R}^n$ and $y_v(t) \in \mathbb{R}^m$, where $u \in X$ and $v \in Y$, denote the values of the weight distribution at time t on the vertices u and v respectively. Let

$$z(t) := \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}, \tag{3}$$

denote the column vector of all weights, which is the state of the dynamical system that we will induce on the graph.

Following [10], we define the time-invariant gain function $g : \mathbb{R} \to \mathbb{R}$ as follows:

$$g(x) \quad := \quad x \cdot (x - 1 - \ln|x|), x \in \mathbb{R}, \tag{4}$$

define the vector version $\bar{g} : \mathbb{R}^{n+m} \to \mathbb{R}$, as the component-wise application of g to each element in the argument.

At time $t + \Delta t$, we update the weight vectors x and y based on the following update rule:

$$z(t + \Delta t) = \bar{g}(Wz(t)). \tag{5}$$

Note that edge weights are constant, only the vertex weights change over time. We thus have a nonlinear time-invariant dynamical system on the graph. We make a mild assumption that W is full-rank. In our motivating example, this assumption is satisfied when no two tokens appear in exactly the same set of containers. If they do, we can replace by a joint token. When W is non-binary, we can perturb W to make it full-rank. This concludes the description of our underlying model. We can state our main theorem as follows.

Theorem 1. *Assume that there exists $z_e := (x_e, y_e) \in \mathbb{R}^{n+m}$, a positive solution to the system $\bar{g}(Wz_e) = z_e$. Suppose that λ denotes the maximum eigenvalue of the graph G^2 obtained by squaring G i.e., squaring the adjacency matrix of G, and then multiplying the weight of each edge $uv \in G^2$ by $g'(z_e(u)) \cdot g'(z_e(v))$. The algorithm for updating vertex weights given above converges to some stable point z_e if $\lambda < 1$, and the initial value z_0 lies in the basin of attraction of z_e.*

We first sketch the main ideas behind the proof in the scalar case before presenting the actual proof. Consider the following scalar ODE:

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) \quad = \quad \hat{g}(x) := \bar{g}(x) - x(t) \tag{6}$$

$$\quad = \quad x(t)(x(t) - 1 - \ln x(t)) - x(t). \tag{7}$$

The rate of change in weight $x(t)$ in time Δt is equal to the change in $x(t)$ in one update step, and as $\Delta t \to 0$, we get the ODE above.

[1] In our motivating application, W is the adjacency matrix with $w_{uv} \in \{0, 1\}$.

The ODE has two fixed points: (1) $x = 0.158\ldots$, and (2) $x = 3.146\ldots$, with $\hat{g}$.

Our goal is to analyze whether either of the fixed points is locally asymptotically stable and by Lyapunov's criteria such a one dimensional update rule for $x(t)$ would lead to convergence with $x^* \simeq 0.158\ldots$, as long as the initial value $x(0)$ was close to x^*.

We need the following indirect version of Lyapunov's stability criterion for higher dimensions:

Theorem 2 ([3]). *Lyapunov stability criterion: Let $x = 0$ be an equilibrium point for $\dot{x} = f(x)$, where $f : D \to \mathbb{R}^n$ is continuous and differentiable and D is a neighborhood of the origin. Let $\mathrm{J} \equiv \frac{\partial}{\partial x} f(x)|_{x=0}$ denote the Jacobian matrix, then*

- *The origin is asymptotically stable if the real part of all eigenvalues of* J *are negative.*
- *The origin is unstable if the real part for at least one of the eigenvalues is positive.*

Consider the following $n + m$ dimensional ODE:

$$\frac{\mathrm{d}}{\mathrm{dt}} z(t) = \hat{g}(z(t)) := \bar{g}(Wz(t)) - z(t)\,. \tag{8}$$

The above dynamical system captures our update rule. Note that any $z_e \in \mathbb{R}^{n+m}$ that satisfies $\bar{g}(Wz_e) = z_e$ will be a candidate for a stable fixed point.

Our next task is to characterize the eigenvalues of the Jacobian at fixed point z_e. The uv^{th} entry of the Jacobian matrix at z_e is given by:

$$\frac{\partial}{\partial z_v}\left(g\left(\sum_{w\in N(u)} W_{uv} z_w\right) - z_u\right)_{|z_e}\,, \tag{9}$$

where $N(u)$ denotes the set of neigbours of vertex u. By the chain rule, it equals:

$$\left(\frac{\partial}{\partial z_v} s(u)\right)\left(\frac{\partial}{\partial s} g(s)\right)_{|z_e} - \delta_u(v), \tag{10}$$

where we have used $s(u) := \sum_{w\in N(u)} W_{uv} z_w$ and δ is the kronecker delta function. But, the last expression is equivalent to:

$$W_{uv} \cdot g'(s)_{|z_e} - \delta_u(v). \tag{11}$$

Therefore, the Jacobian about z_e has the form:

$$\mathrm{J}_{|z_e} = \mathrm{D} \cdot \left(\begin{array}{c|c} 0 & \mathrm{M} \\ \hline \mathrm{M}^T & 0 \end{array}\right) - \mathrm{I}, \tag{12}$$

where M is the $n \times m$ block matrix which is the incidence matrix of the bipartitie graph G, I is the $n + m$ dimensional identity matrix, and D is the diagonal matrix with it's u^{th} entry given by:

$$\mathrm{D}_u = g'\left(\sum_{w\in N(u)} W_{uv} z_e(w)\right). \tag{13}$$

Let, H := J + I. If we can place an upper-bound on the real part of the eigenvalues of H so that

$$\mathrm{Re}\,(\lambda_{\max}(\mathrm{H})) < 1 \tag{14}$$

then the eigenvalues of J are simply obtained by translation of the eigenvalues of H, so that $Re(\lambda_{\min}(\mathrm{J})) < 0$ – the fixed point corresponding to z_e is then stable. The condition which ensures Equation 14 will be our condition in Theorem 1.

Let us rewrite D as:

$$\mathrm{D} = \left(\begin{array}{c|c} \mathrm{D}_1 & 0 \\ \hline 0 & \mathrm{D}_2 \end{array}\right), \tag{15}$$

where D_1 and D_2 are $n\times n$ and $m\times m$ diagonal matrices, respectively.

Proposition 1.

$$\mathrm{HH}^T = \left(\begin{array}{c|c} \mathrm{D}_1\mathrm{MM}^T\mathrm{D}_1 & 0 \\ \hline 0 & \mathrm{D}_2\mathrm{M}^T\mathrm{MD}_2 \end{array}\right). \tag{16}$$

The eigenvalues of HH^T are bounded by the maximum of the eigenvalues of $\mathrm{D}_1\mathrm{MM}^T\mathrm{D}_1$ and $\mathrm{D}_2\mathrm{M}^T\mathrm{MD}_2$. It follows from the definition of spectral norm that the absolute value of the eigenvalues of H is upper-bounded by the square-root of the maximum eigenvalue of HH^T. Therefore, the condition that both matrices $\mathrm{D}_1\mathrm{MM}^T\mathrm{D}_1$ and $\mathrm{D}_2\mathrm{M}^T\mathrm{MD}_2$ have all eigenvalues upper-bounded by 1 is sufficient to ensure a stable fixed point.

However, the matrices $\mathrm{D}_1\mathrm{MM}^T\mathrm{D}_1$ and $\mathrm{D}_2\mathrm{M}^T\mathrm{MD}_2$ are simply the adjacency matrix of the graphs on the partite sets X and Y of G, obtained by squaring the weighted bipartite graph G and reweighing every edge uv in the squared graph by $g'(z_e(u))g'(z_e(v))$. Hence, if the eigenvalues of these graphs are all upper-bounded by 1, the statement of Theorem 1 follows.

2.1 Basin of Attraction

The next task is to determine the starting values of z for which we can be assured of convergence. In this case, the conceptually easiest solution is to construct a Lyapunov function, as in the one dimensional case above, and show that $V(z)$ is locally positive definite and $\dot{V}(z)$ is locally negative definite. However, things are more complicated in higher dimensions and such explicit constructions are not easy for non-linear dynamical systems. Therefore, the standard approach is to construct an ellipsoidal Lyapunov function for the linearized system, which is expected to be globally asymptotically stable, and then show it is locally asymptotically stable in a large enough region. Typically, even this problem is difficult but the symmetries of the Jacobian matrix makes it simpler.

Theorem 3. *Given matrix* M *which satisfies the conditions of Theorem 1, the radius of convergence about z_e is non-decreasing in λ, where λ is as in Theorem 1.*

Consider a linear system $\dot{z} = Az$, it is asymptotically stable if the Jacobian J has all eigenvalues with negative real parts. A more direct characterization is that if the linear system is asymptotically stable then for any positive definite matrix Q, there exists a positive definite matrix P such that

$$A^T P + PA = -Q \tag{17}$$

which follows from using $V(z) = z^T Pz$ as the Lyapunov function [2].

The same kind of analysis can be done for non-linear systems i.e., linearize the system about the fixed point (Taylor expansion with higher order terms dropped), compute a Lyapunov function which shows global asymptotic stability, then use it as a candidate for the non-linear system. V already has all the proprties we need except for one: $\dot{V}(z)$ may not be negative definite, but we can show it is locally negative definite and prove a lower bound on the convergence radius.

We know,

$$\frac{\mathrm{d}}{\mathrm{dt}} z(t) = \mathrm{J}_{|z_e} z(t) + \mathrm{h}(z(t)), \tag{18}$$

where h is what remains after substracting $\mathrm{J}_{|z_e}$ from the RHS of Equation 8 i.e. $\bar{g}(z) - z$. In our notation, $A = \mathrm{J}_{|z_e}$, so that Equation 17 reads:

$$\mathrm{J}^T P + P\mathrm{J} = -Q \tag{19}$$

Choosing $Q = \mathrm{I}$ gives the matrix equation:

$$\mathrm{J}^T P + P\mathrm{J} = -\mathrm{I}. \tag{20}$$

Note that, the matrix P in Equation 20 is assumed symmetric, so that the left and right eigenvectors coincide (after transposition). Multiplying the LHS and RHS of Equation 20 by $v_{\max}$ and $v_{\max}^T$, the unit eigenvector corresponding to the maximum eigenvalue $\lambda_{\max}(P)$, we get,

$$\lambda_{\max}(P) \left(v_{\max}^T \mathrm{J}^T v_{\max} + v_{\max}^T \mathrm{J} v_{\max} \right) = -1. \tag{21}$$

By the definition of spectral norm,

$$\max \left(|v_{\max}^T \mathrm{J} v_{\max}|, |v_{\max}^T \mathrm{J}^T v_{\max}| \right) \leq \sqrt{\lambda_{\max}(\mathrm{J}\mathrm{J}^T)}, \tag{22}$$

and so,

$$\sqrt{\lambda_{\max}(\mathrm{J}\mathrm{J}^T)} \geq \frac{1}{2\lambda_{\max}(P)}. \tag{23}$$

Now, it is also known (see [3] or [2]) that

$$\begin{aligned} \dot{V}(z) &= z^T (\mathrm{J}^T P + P\mathrm{J}) z + 2 z^T P \mathrm{h} \\ &\leq -\|z\|^2 + 2\lambda_{\max}(P) \|z\| \cdot \|\mathrm{h}(z)\| \\ &\leq -\left(1 - 2\lambda_{\max}(P) \frac{\|\mathrm{h}(z)\|}{\|z\|} \right) \|z\|^2, \end{aligned}$$

where $\|\cdot\|$ denotes ℓ_2 norm. The RHS is negative as long as

$$2\lambda_{\max}(P) \frac{\|\mathrm{h}(z)\|}{\|z\|} \leq 1, \tag{24}$$

which is equivalent, by Equation 23, to

$$\frac{\|\mathrm{h}(z)\|}{\|z\|} \leq \frac{1}{2\lambda_{\max}(P)} \tag{25}$$

$$\leq \sqrt{\lambda_{\max}(\mathrm{J}\mathrm{J}^T)}. \tag{26}$$

As $\lambda_{\max}(\mathrm{J}\mathrm{J}^T)$ increases $\dot{V}$ is at least going to remain locally negative definite around z_e. Recall from Proposition 1 and the discussion that followed that $\lambda_{\max}(\mathrm{J}\mathrm{J}^T)$ is upper-bounded by the largest eigenvalue of the reweighed square of G i.e., λ ($\equiv \lambda_{\max}(\mathrm{D}W^2\mathrm{D}^T)$). Therfore, as λ increases, the time derivative of the candidate Lyapunov function either remains negative or may become negative. Hence, the statement of Theorem 3 follows.

3 EXPERIMENTAL RESULTS

We present some qualitative comparisons with the MMR algorithm [1] below. We believe the top-ranked sentences for the reflection algorithm have more information content. Further experimental results will appear in a full paper.

Our comparison in the table simply displays the top-ranked sentence, for each algorithm, when both algorithms are run on the concatenated text in the set of top twenty documents obtained using google web-search for some arbitrary queries related to monarch butterflies.

Query	Weighted Gain	MMR
monarch butterfly parasites	Protozoan parasites such as Ophryocystis elektroscirrha and a microsporidian Nosema species have also been identified in wild and captive monarchs (McLaughlin and Myers 1970, Leong et al. 1992;1997, Altizer and Oberhauser 1999, O. Taylor, personal communication).	Monarchs have many natural enemies - predators, parasitoids, and parasites can harm monarch eggs, larvae, pupae, and adults.
monarch butterfly migration	While the practice of transferring monarchs from place to place is generally not condoned by scientists, some reciprocal transfers of tagged monarchs have demonstrated that monarchs from east of the Rocky Mountains will migrate south if transferred west, in the range of the western population (rather than SW).	The Monarch butterfly migrates for 2 reasons.
monarch butterfly adult	The four stages of the monarch butterfly life cycle are the egg, the larvae (caterpillar), the pupa (chrysalis), and the adult butterfly.	The King of Butterflies – The Monarch Butterfly
monarch butterfly climate	Aside from the ecological significance of these migrations – monarch butterflies are the only insects known to migrate to warmer climates more than 2,500 miles away – the butterflies' five-month layover in Mexico before returning to the United States has become one of the region's main tourist attractions and economic drivers.	Given that monarchs largely depend on the genus Asclepias as larval host plants, the effects of climate change on monarch northward migrations will most likely be mediated by climate change effects on Asclepias.
monarch butterfly captive rearing	Moreover, these subtle variation appear to have biological significance; monarchs with darker shades of orange (approaching red) show higher flight ability in captive settings [13], and a recent study provided evidence that the degree of black pigment is related to migration distance in wild-caught monarchs [14].	Chrysoperla rufilabris (Green Lacewing) larvae is a common inhabitant of milkweed species and a voracious predator of monarch eggs.

Table 1: Some example sentences

REFERENCES

[1] Jaime Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. *21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (1998).

[2] Emilio Frazzoli and Munther Dahleh. 2011. MIT Open Courseware. *https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-241j-dynamic-systems-and-control-spring-2011* (2011).

[3] Hassan Khalil. 2002. Nonlinear Systems. *Prentice Hall, 3rd edition* (2002).

[4] Jon Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *J. ACM* (1999).

[5] Hans Peter Luhn. 1958. The Automatic Creation of Literature Abstracts. *IBM journal of Research and Development* (1958).

[6] Rada Mihalcea. 2004. Graph based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization. (2004).

[7] Rada Mihalcea and Dragomir Radev. 2011. Graph-based Natural Language Processing and Information Retrieval. *Cambridge University Press* (2011).

[8] Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Texts. *Conference on Empirical Methods in Natural Language Processing* (2004).

[9] Larry Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1998. The PageRank Citation Ranking: Bringing Order to the Web. *Stanford University Technical Report* (1998).

[10] Kishore Papineni. 2001. Why Inverse Document Frequency? *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies* (2001).

[11] Karen Sparck-Jones. 1973. Index Term Weighting. *Information Storage and Retrieval, 9* (1973).