Extrapolation to Speed-up Query-dependent Link Analysis Ranking Algorithms

Muhammad Ali Norozi
Department of Computer and Information Science
Norwegian University of Science and Technology
Sem Sælends vei 7-9
NO-7491 Trondheim, Norway
+47 7359 3440
mnorozi@idi.ntnu.no

ABSTRACT
Relevance is a numerical score assigned to a search result, representing how well the results meet the information needs of the user that issued the search query. Several mathematical tools and techniques have been used in research for improving the relevancy ranking models. Advanced concepts in linear algebra, such as the Singular Value Decomposition, and theory of Markov chains have also been employed for innovating relevancy ranking. This study presents the use of Extrapolation technique to speed up the convergence of query-dependent Link Analysis Ranking Algorithms. It contains a novel improvement in algorithms like HITS, SALSA and their descendants (e.g., Exponentiated and Randomized HITS) using the Extrapolation techniques. Using this approach it is possible to accelerate the algorithms in terms of reducing the number of iterations and therefore uncovered a much faster convergence. In the experiments we even got much better results than theoretically predicted. The results present a speedup to the order of 3 – 19 times.

Categories and Subject Descriptors
H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—Link Analysis Ranking

General Terms
Equilibrium, Extrapolation, Algorithms, Linear Algebra

Keywords
Link Analysis Ranking, Information Retrieval, Singular Value Decomposition, Extrapolation, Markov Chain, $L_1$ Norm

1. INTRODUCTION
The apparent ease with which the users click from document to document provides a rich source of information which could be used to understand what and where to find the important documents. The semi-structured and diverse collections of documents are held together by the billions of annotated connections called hyperlinks. Analyzing these myriad interconnections between the documents forms the basis for Link Analysis Ranking (LAR). These analyses will help us identify the proximity and relevance of documents amongst each other. And enables us to find out the social or informational organization of the documents (the sociology of information). We can utilize the contextual exposition of documents to deduce the importance or popularity of the documents in the network, by using the core graph theory concepts and techniques.

The citation structures of the documents contain a wealth of useful but “implicit information”. Through citation structure hundreds and millions of documents can be pulled together into a network of knowledge. Foremost such a structure represents the users’ behaviours and needs. Users on the Web usually discover most relevant and valuable information through the recommendations and references from a good source of information.

One of the main concerns in the link based ranking methods is the convergence to a “good solution” or an equilibrium state. An equilibrium state is a state where system under certain presumptions can declare the set of good results corresponding to user query. Most of the link analysis based ranking models are iterative in nature, they iteratively move towards the required equilibrium state (the good solution). Convergence is a central phenomenon in iterative algorithms [15]. In linear algebra, the iterative methods are employed when direct methods would be prohibitively expensive and in some cases impossible even with best possible computing power to find out the actual solution. Essentially the iterative methods such as “power method” [15] provide an approximation to the true solution starting from a seed value. This work deals with the convergence properties and behaviour of the famous query-dependent LAR algorithms [13, 17, 1, 21].

The major contribution of this work is the improvements primarily in the convergence behaviour of the query dependent LAR algorithms using “careful periodic” applications of extrapolation during iterations. Extrapolations are simple and unique techniques that require little additional infrastructure that needs to be incorporated in the existing LAR algorithms. This study have distinctively applied extrapolation techniques to query-dependent LAR algorithms, which
was not done before. The parameters are manipulated extensively and hence extracted a very novel performance gain due to extrapolation.

In the study by Kamvar et al., [12] they have found an improvement of order 3 at-most due to extrapolation, in PageRank algorithm. By applying extrapolation carefully in the query-dependent algorithms, improvements of order in range (3 − 19) have been discovered in this work, see Table 1 and Appendix in [19].

The document is therefore organized as follows; Sections 2 defines the theoretical background and the preliminaries of the problem at hand. Motivation for this paper is also presented in Section 2.1 and later the novel idea of extrapolation to speed the rate of convergence is proposed in Section 3. In Section 4 the idea proposed in earlier sections are empirically assessed. Section 5 concludes the study with important results and possible future work.

2. THEORETICAL BACKGROUND

2.1 Link Analysis Ranking (LAR)

The presence of (hyper-) link information clearly augmented a great deal to the characterization of the informative content present in the documents. LAR approaches are intended to resolve some of the intrinsic weaknesses of the content-based Information Retrieval (IR) models. Through the analyses of network of the documents (due to citation structure) LAR approaches bring in a whole new horizon to information retrieval space. The essence of LAR therefore is that the “overall information” of a hyperlink database of documents is not composed of only static “textual information”, but also another, the “hyper” information.

Link Analysis Ranking is the next step from just content-analyses. It involves analyses and understanding of a very huge and jumbled network(s) of documents. From such a huge and massive network extracting useful information is quite a challenging and difficult task. The challenge is not just because of size of network, but also because of its diversity and unpredictability. The huge network(s) of documents hence forms the core of link analysis ranking.

The resultant hyperlinked graph of the network of document will be given as an input to the LAR algorithms. This graph is encoded in an adjacency matrix $A$, where $A[i,j] = 1$ if there is a link from node $i$ to node $j$ and 0 otherwise (see also [15]). The LAR algorithm iteratively operate on the hyperlinked graph (the adjacency matrix $A$) and returns the $n$-dimensional rank vector $\vec{x}$ with weights computed for each node in the graph, where $x_i$ is the weight of $i^{th}$ node. The weights are actually the probabilities of relevance of each document to the user query. LAR algorithms are thus meant to discover authoritative documents through analyzing the hyperlink graph [21].

The two pioneer LAR algorithms PageRank [20] and HITS [13] are query-independent and query-dependent respectively. They were followed by substantial amount of research [8, 9, 1, 7, 3, 10, 5, 12], to name just a few.

2.2 Extrapolation

Extrapolation is the process of constructing new data points outside a discrete set of known data points. It is similar to the process of Interpolation, which constructs new points between known points, but its results are often less meaningful, and are subject to greater uncertainty. Interpolation is a specific case of curve fitting, in which the function must go exactly through the data points. In case of convergence, Extrapolation techniques can be employed to accelerate the convergence by using the known data points (values from successive iterates) to construct new data points (principal eigenvector(s)). Techniques for accelerating the convergence series are often applied in numerical analysis, where they are used to improve the speed of numerical integration, and other well-known series [16, 15].

3. EXTRAPOLATION TECHNIQUES TO ACCELERATE THE CONVERGENCE

Extrapolation techniques are novel as they offer new approaches of taking into consideration important properties of the iterative method for effectively accelerating the computation of the query dependent family of algorithms. Fast convergence and efficient computational speed in query dependent algorithms are quite crucial, because they operate on query time. For example, for a large matrix representing the Web, it is fairly expensive to compute the operation $\vec{x}^k = A\vec{x}^{k-1}$ several times as $k \to \infty$.

Extrapolation techniques were previously used by Kamvar et al., [12], specifically tailored to the PageRank problem. In this study it is employed to the query-dependent counterparts such as HITS, its improvements and SALSA, and therefore we came up with more in-depth analyses of their convergence behaviours (see Section 4).

Extrapolation techniques in LAR stems from another popular method in numerical linear algebra called fixed point iteration. For a given function $f$ defined on real numbers and a given initial point $x_0$ in the domain of $f$, the fixed point iteration is:

$$x_{k+1} = f(x_k), k = 0, 1, 2, \ldots$$

The series $x_0, x_1, \ldots$ are expected to converge to $x$. If the function $f$ is continuous, then $x$ is a fixed point of $f$, i.e., $x = f(x)$.

Equation (1) is the standard fixed point iteration. Now consider the standard LAR problem, we will get a correspondence with fixed point iteration, i.e: 

$$\vec{x}^{(k)} = A\vec{x}^{(k-1)}$$

Let us consider $f$ in equation (1) as an iterative numerical process, then the intermediate iterates of the linear convergent series, $x_1, x_{k+1}$ and $x_{k+2}$ can be used to extrapolate the fixed point $x$. This three-point extrapolation scheme is well known as Aitken $\Delta^2$ extrapolation [4].

Aitken $\Delta^2$ extrapolation is oldest and most popular extrapolation technique. It forms the basis for other extrapolation techniques. It has also been used to speedup the convergence of power method for faster computation of PageRank [12].

In LAR, Aitken acceleration computes the principal eigenvector of the Markov matrix in one step, under the assumption that the power iteration estimate $\vec{x}^{(k-2)}$ can be expressed as the linear combination of the first two eigenvectors, $\vec{u}_1$ and $\vec{u}_2$.

$$\vec{x}^{(k-2)} = \alpha_1 \vec{u}_1 + \alpha_2 \vec{u}_2$$

where $\vec{u}_1$ is the principal eigenvector and $\vec{u}_2$ is the second eigenvector of Markov matrix.
Algorithm 1 The HubAvg Algorithm

1: func $a^k \leftarrow \text{HubAvg}(A, a^{(0)}, \epsilon)$;
2: A : adjacency matrix formed from the base-set $S_0$
3: $A \leftarrow A_j/\text{RowNorm}_j \forall j$
4: $a^{(0)}$: set the seed values of the authority vector
5: $h^{(0)}$: set the seed values of the hub vector
6: while not converged do
7: $a^k \leftarrow A^T A a^{k-1}$
8: $h^k \leftarrow A A^T h^{k-1}$
9: Periodically:
10: $a^k \leftarrow \text{QuadraticExtrapolation} (a^{k-3}, a^{k-2}, a^{k-1}, a^k)$
11: $a^k \leftarrow a^k$ Normalize
12: $h^k \leftarrow h^k$ Normalize
13: $k \leftarrow k + 1$
14: {Compute the convergence}
15: end while

Equation (3) shows that from the nonprincipal eigenvectors (the values of $x^{(k-2)}$ from successive iterates), we can extrapolate the value of the principal eigenvector $\vec{u}_1$. The previous values calculated in the successive iterations could be used to extrapolate the new value (the new data point outside the known data points), the principal eigenvector. This way we could accelerate the rate of convergence of the already convergent series produced by the query-dependent LAR algorithm.

The Extrapolation step when applied periodically, enables us to subtract off the estimates of the nonprincipal eigenvectors from the current iterates $x^{(k)}$. For the derivation of Aitken acceleration and the empirical proof that it can extrapolate the principal eigenvector for power method see [12, 4]. Aitken extrapolation technique is crucial primarily because the subsequent extrapolation techniques build upon the ideas advocated in this technique. It serves to provide a general premise for extrapolation. It is therefore essential to have a sound appreciation of this technique to comprehend the newer more sophisticated techniques of extrapolation used for accelerating convergence.

In a nutshell we use the priori knowledge (which we acquire from the prior iterates of an LAR algorithm) as a basis to extrapolate the new and better value (the principal eigenvector). We use the assumption that the new iterate(s) can be expressed as a linear combination of the last few iterates. With some changes to this basic assumption various extrapolation techniques can be formulated (for example, Quadratic Extrapolation assumes that last three iterates $x^{(k-3)}, x^{(k-2)}$ and $x^{(k-1)}$ together with current iterate $x^{(k)}$ can be used to express the new and improved iterate value, see Equation 4). In case of Aitken Extrapolation we are using three-points $x^{(k-3)}, x^{(k-1)}$ and $x^{(k)}$ to extrapolate the next point $\vec{u}_1$.

$$x^{(k-3)} = \alpha_1 \vec{u}_1 + \alpha_2 \vec{u}_2 + \alpha_3 \vec{u}_3 \quad (4)$$

The extrapolation methods are different from standard fast eigensolvers, which mostly relies on the matrix factorization and/or matrix inversion. The extrapolation methods that we study here rely upon the fact that the principal (first) eigenvalue of the Markov matrix is, $\lambda_1 = 1$ [19], in order to find an approximation to the principal eigenvector. This information can be used to compute the estimates of the nonprincipal eigenvectors during the iterations.

Algorithm 2 Quadratic Extrapolation

1: func $a^k \leftarrow \text{QuadraticExtrapolation}(a^{k-3}, a^{k-2}, a^{k-1}, a^k)$
2: $y^{k-2} \leftarrow a^{k-2} - a^{k-3}$
3: $y^{k-1} \leftarrow a^{k-1} - a^{k-3}$
4: $y^k \leftarrow y^k - a^{k-3}$
5: $Y \leftarrow (y^{k-2} y^{k-1})$
6: $\gamma_3 \leftarrow 1$
7: $\gamma_2 \leftarrow -Y^T y^k$
8: $\gamma_0 \leftarrow (\gamma_1 + \gamma_2 + \gamma_3)$
9: $\beta_0 \leftarrow \gamma_1 + \gamma_2 + \gamma_3$
10: $\beta_1 \leftarrow \gamma_2 + \gamma_3$
11: $\beta_2 \leftarrow \gamma_3$
12: $a^k \leftarrow \beta_0 * a^{k-2} + \beta_1 * a^{k-1} + \beta_2 * a^k$

Through the estimates computed during the successive iterates of power method, we expect to extrapolate the value of the principal eigenvector. Specifically, we ignore the non-dominant eigenvectors corresponding to the negligibly small values of non-dominant eigenvalues ($< 1$ or $\approx 0$).

Algorithm 1 and 2 depicts the apparent elegance of extrapolation on improvement of HITS algorithm, the HubAvg algorithm [21, 19]. In Algorithm 1 we are periodically applying the extrapolation step (lines 9 and 10). Also observe in Algorithm 2, we are only using prior knowledge (the values of intermediate iterates) to extrapolate the new and improved value (the expected principal eigenvector). The most expensive operation in Algorithm 2 is the solution of the overdetermined system of linear equation (line 7).

What is crucial here is to identify theoretically and empirically that the extrapolation methods accelerate the convergence, and the computed value is actually the principal eigenvector. In [19] we have provided theoretical proof for the importance of Quadratic Extrapolation and its capability to extrapolate the principal eigenvector of the Markov matrix. In Section 4 we have provided experimental evidences of the capabilities of Extrapolation to improve the convergent sequence in query-dependent LAR algorithms.

The assumption in equation 4 is not in contrast to reality rather it is used to form the much stronger relation later in the derivation. Of course, the matrix $A$ can have more than 3 eigenvectors. In [19] a relation based on this assertion has been formulated (Algorithm 2 is written based on that assertion). Empirically the Quadratic extrapolation derived from this assumption provides much better rate of convergence than the original algorithms (see Figure 2).

4. EXPERIMENTAL EVALUATIONS

In this section the focus is on the empirical evaluations of the extrapolation technique discussed in the Section 3. We will specifically observe the effectiveness extrapolation on the query-dependent LAR algorithms, such as; HITS, SALSA and their improvements. For brevity, only the results of HITS and HubAvg algorithm has been discussed here, but for more comprehensive understanding see [19]. Specifically, we will observe the peculiarities of Extrapolation techniques in improving the rate of convergence.

We have primarily relied on the dataset used in [1]. The dataset is gathered using the prescriptions of Kleinberg [13] (as described in the section below) and is stored in an inverted file format, see Figure 1. The effectiveness of the find-
corresponding to each query given
A
input to the LAR algorithms. The choices of
in Table 1, can be given as an input to the LAR algorithms. The application of a
single extrapolation step is considered to be equivalent to 0.5
times or less the cost of an iteration of power method (32% of
cost of an iteration [9]), e.g., Figure 2(b) extrapolation step is applied 6 times only while the improvement is surprising
(the original algorithm stabilizes after 779 iterations while
extrapolated version in just 37 iterations). Hence the effects
in number of iterations is interpreted as almost equivalently
to the improvements in time, e.g., 2 times improvement in
number of iterations is treated as 1.8 – 2.0 times speedup in
the wall-clock time.

Extrapolation techniques are therefore very effective, the
extrapolated algorithms in our experiments yielded a net
speedup of over 3 (see Table 1), the speedup could be even
more significant in practice; for example we even got a speed-
dup of 19 on our dataset depending on careful application of
extrapolation step (see Figure 2(a) for query “basketball”).

Table 1 provides a comprehensive overview of all the res-
ults for each of 34 queries that we used. In the table:
 itr, is the number of iterations
 ext, is the number of times extrapolation applied
 E, is the Extrapolated version of the algorithms, and
 N refers to normal version

The extrapolation columns in the Table 1 indicate the
best performance in terms of number of iterations that we
got as a result of tweaking the parameters. Overall we have
applied extrapolation steps 8 times on an average to get
rapid convergences. So, the overhead of net application of
extrapolation is very less, almost 3 – 4 iterations of LAR
algorithm. On average the algorithms converge after just
46 as a result of extrapolation in comparison to the average
170 iterations of the original algorithms. A net average
speedup of order 5.78 in all the algorithms presents a very
good reason for the usefulness of extrapolation techniques.
Note that Extrapolation technique can also be applied
in conjunction with other acceleration techniques, such as
(a) Query “computational complexity”  
(b) Query “basketball”  

Figure 2: The convergence graphs. The spikes in each graph shows the point where extrapolation step is applied

Table 1: Results of the experiments with Extrapolation

<table>
<thead>
<tr>
<th>Queries</th>
<th>HITS</th>
<th></th>
<th></th>
<th>HubAvg</th>
<th></th>
<th>HubAvg</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>abortion</td>
<td>itr</td>
<td>12</td>
<td>17</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>3</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>affirmative</td>
<td>itr</td>
<td>397</td>
<td>2529</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>action</td>
<td>ext</td>
<td>4</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>alcohol</td>
<td>itr</td>
<td>21</td>
<td>38</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>2</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>amusement</td>
<td>itr</td>
<td>26</td>
<td>57</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>parks</td>
<td>ext</td>
<td>3</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>architecture</td>
<td>itr</td>
<td>45</td>
<td>98</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>4</td>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>armstrong</td>
<td>itr</td>
<td>30</td>
<td>68</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>5</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>automobile</td>
<td>itr</td>
<td>50</td>
<td>175</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>industries</td>
<td>ext</td>
<td>9</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>basketball</td>
<td>itr</td>
<td>20</td>
<td>27</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>3</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>blues</td>
<td>itr</td>
<td>29</td>
<td>52</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>6</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cheese</td>
<td>itr</td>
<td>17</td>
<td>24</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>3</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>classical</td>
<td>itr</td>
<td>43</td>
<td>160</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>guitar</td>
<td>ext</td>
<td>19</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>complexity</td>
<td>itr</td>
<td>21</td>
<td>32</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>4</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>computational</td>
<td>itr</td>
<td>53</td>
<td>144</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>complexity</td>
<td>ext</td>
<td>5</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>computational</td>
<td>itr</td>
<td>28</td>
<td>58</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>geometry</td>
<td>ext</td>
<td>6</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>death penalty</td>
<td>itr</td>
<td>12</td>
<td>18</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>1</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>genetic</td>
<td>itr</td>
<td>26</td>
<td>40</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>3</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>geometry</td>
<td>itr</td>
<td>25</td>
<td>45</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>5</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Median</td>
<td>itr</td>
<td>29.0</td>
<td>59.0</td>
<td>E</td>
<td>N</td>
<td>E</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ext</td>
<td>4.0</td>
<td>7.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Average: 
itr 50.2 191.7 67.8 247.1
BlockRank [11], or other iterative algorithms e.g., Gauss-Seidel, Successive Over Relaxation, Conjugate Gradient or any other methods [6, 14]. When used in conjunction with any other methods, we might expect more insights about the effectiveness of Extrapolation, both in terms of time and convergence.

We have also had a limited evaluation of the hybrid implementation of extrapolation technique, see [19] for details.

5. CONCLUSIONS AND FUTURE WORK

The speedup in convergence due to extrapolation came first as a surprise. There were some interesting observations that came out as a result of the experiments. It is observed that a careful application of extrapolation can improve convergence inevitably. Therefore, it matters how many times during iterations you apply extrapolation step to gain the required acceleration. The importance of extrapolation in accelerating the convergence is hence remarkable. We have also tested hybrid extrapolation technique, where the effects of extrapolation based on different techniques (AitkenΔ2, Power [9] and Quadratic extrapolation [19]) applied together have been observed in different settings.

Quadratic Extrapolation in query dependent LAR algorithm improves convergence much better than the original rate of convergence of the algorithm, based on the empirical results. In the slow convergent series, the Quadratic Extrapolation is proved to be an effective technique. For example, if the second eigenvalue of the Markov matrix is close to 1, i.e., λ2 → 1, theoretically and empirically the convergence of power method tends to slow down [2, 15]. In such a situation the slow converging sequence of Power method can be accelerated radically by Quadratic Extrapolation (see Section 4). As a result of this study, it is possible to observe the convergence graph much more closely from another perspective. For example, use any other convergence measure instead of just $L_1$ norm. The question is; from the convergence behaviours, is it possible to automate when, where and how many times extrapolation should be applied to gain certain acceleration? Also the more active use of induced ordering [8] to measure convergence together with extrapolation could be a possible future work. Hybrid approach of extrapolation could also be further observed to formulate a better framework for extrapolation, which could possibly be used for personalization too, apart from just accelerating the convergences.

6. REFERENCES


