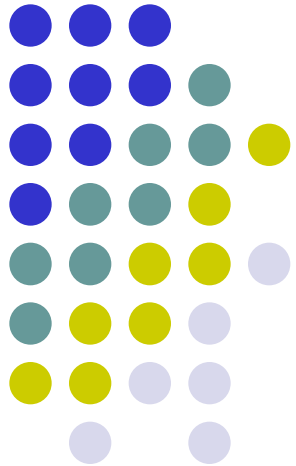
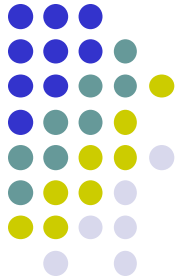
Web Algorithms – Web Search

Part 4: HITS Algorithm

Eng. Fabio Persia, PhD

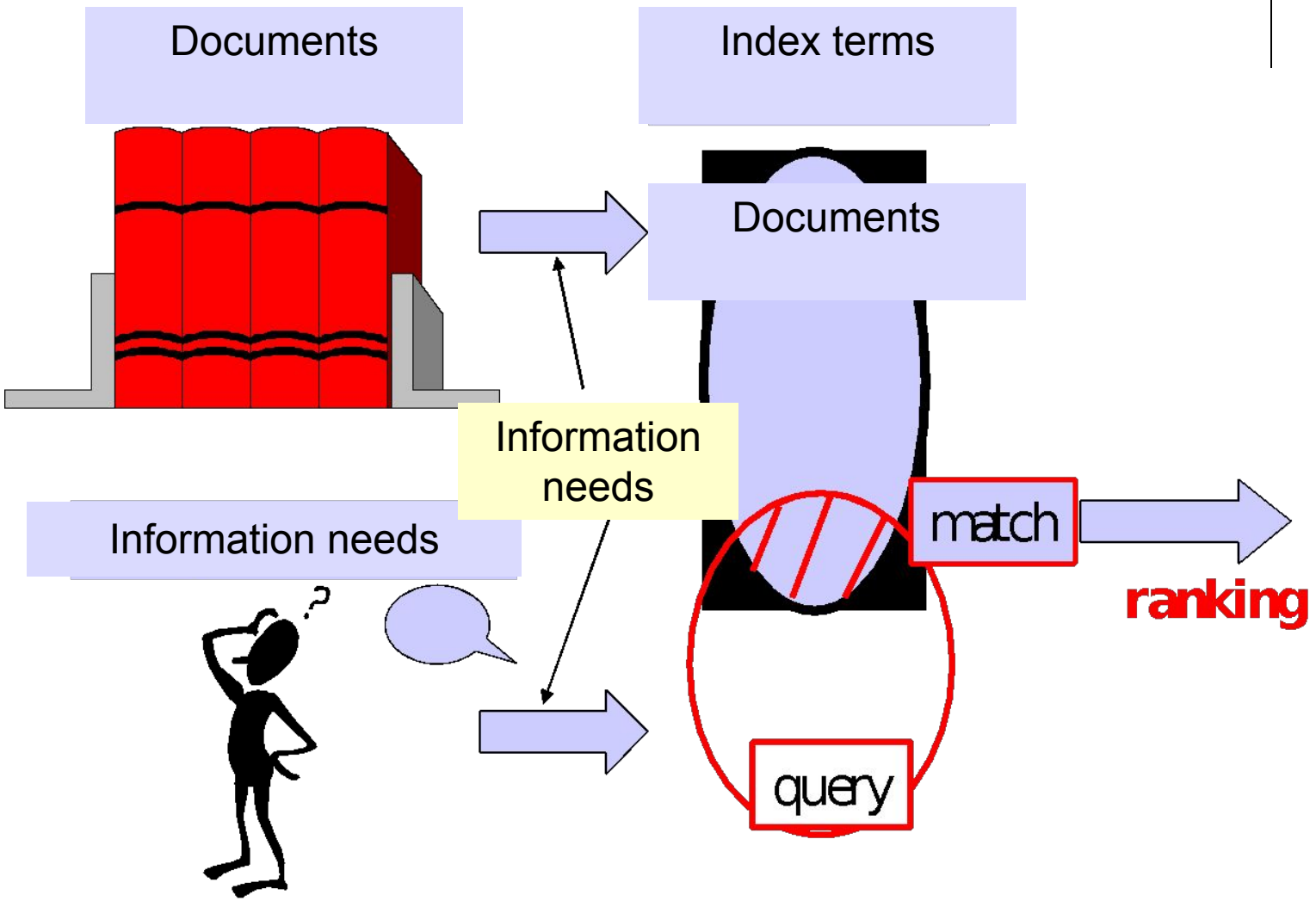


H.I.T.S.

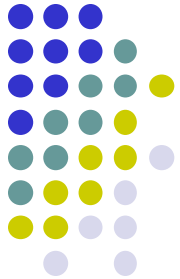


- It stands for ***"Hyperlink Induced Topic Search"***
- It was proposed by J.Kleinberg around 1996
- Is based on the idea that some web pages are about specific topics (are authorities on the subject) and other instead are helpful hubs to find the authoritative pages on the subject
- Unlike the PageRank, for the analysis a graph is chosen dependent on the query
- We see a figure that summarizes the process

H.I.T.S


Documents
Index terms
Documents
Information needs
Information needs
match
ranking
query

H.I.T.S

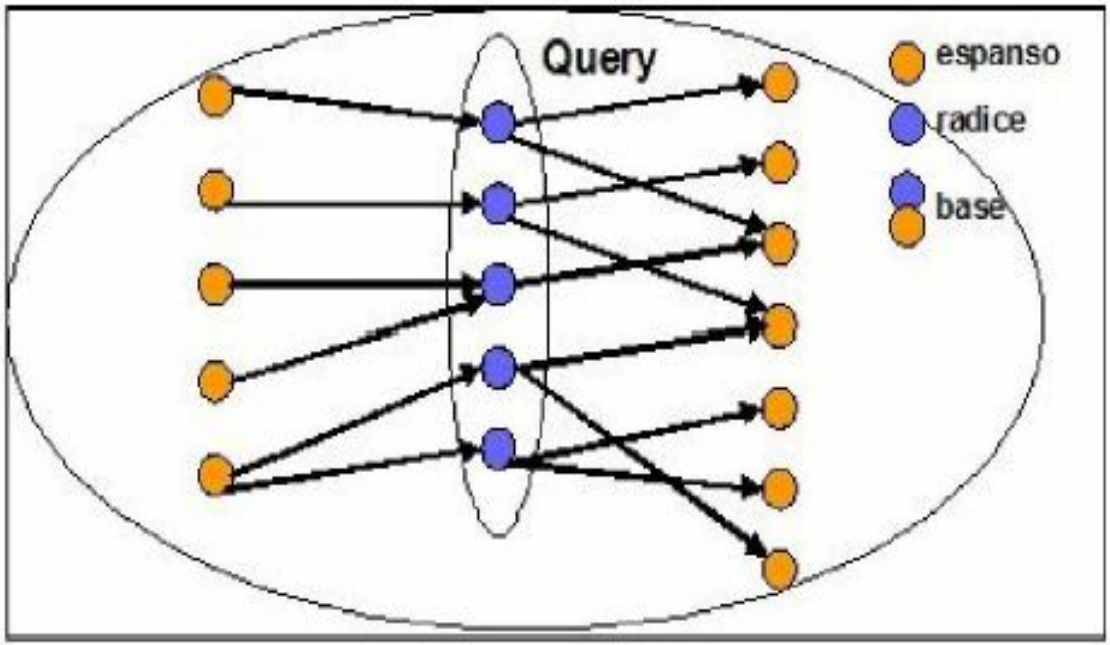


- The query is sent to a standard text-based information retrieval system in order to build the so-called ***root set R*** of nodes of the Web graph

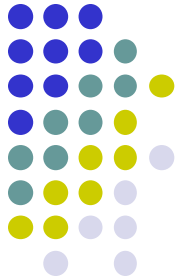
- Are also included all adjacent nodes ***u*** to at least one node $r \in \mathbf{R}$ via an incoming or outgoing arc, i.e. such that $(u,r) \in \mathbf{E}$ *o* $(r,u) \in E$

- The additional nodes compose the ***expanded set*** and the root set ***R***, form the base set V_q

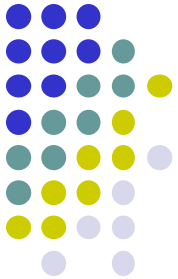
H.I.T.S.



H.I.T.S



- All edges connecting pages of a same host are eliminated because they were considered navigational or nepotism, getting a set of remaining arches E_q
- Then we construct the graph $G_q=(V_q,E_q)$ for the specific query
- Such a graph is used for computing proper indices like for the prestige and pagerank
- Each page however now has two different associated indices

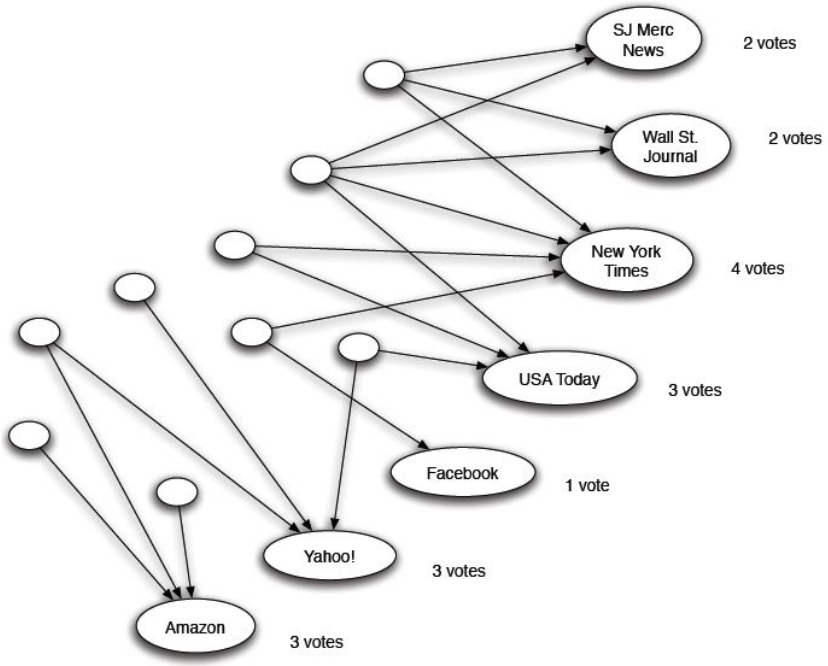


- Two types of popular or relevant pages are considered:
  - authorities, which contain high quality information
    - Newspaper home pages
    - Course home pages
    - Home pages of auto manufacturers
  - hubs, which are comprehensive lists of links to authoritative pages
    - List of newspapers
    - Course bulletin
    - List of US auto manufacturers

- Each page ***u*** is then in a certain way both an hub and an authority, but these properties are quantified

- Idea:
  - Good authorities are “voted” by good hubs
  - Good hubs have outlinks to good authorities
  - Start with all score equal to one
  - Refine iteratively like in previous indices

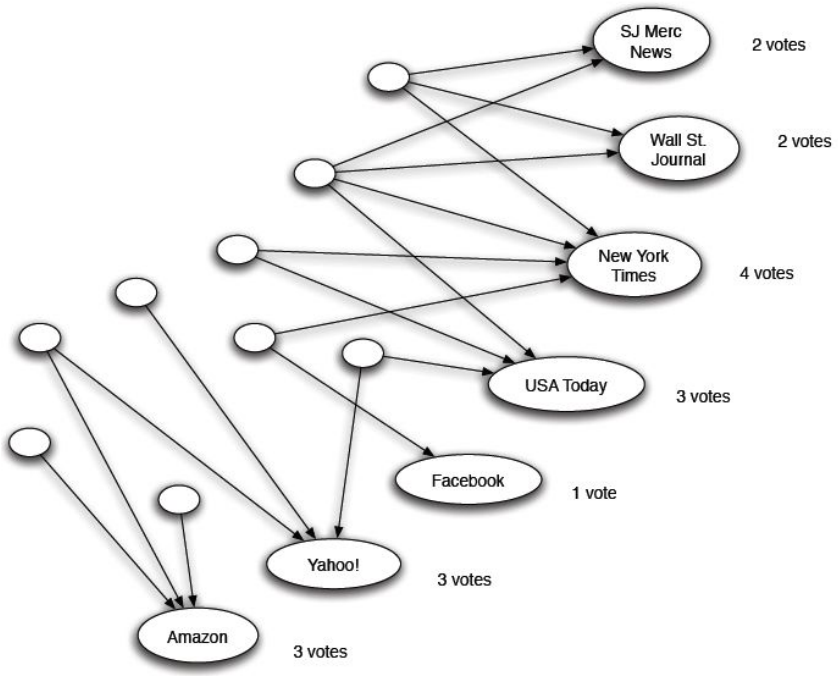
Counting in-links: Authority



Each page starts with **hub** score 1. **Authorities** collect their votes

(Note this is idealized example. In reality graph is not bipartite and each page has both the hub and authority score)

Counting in-links: Authority



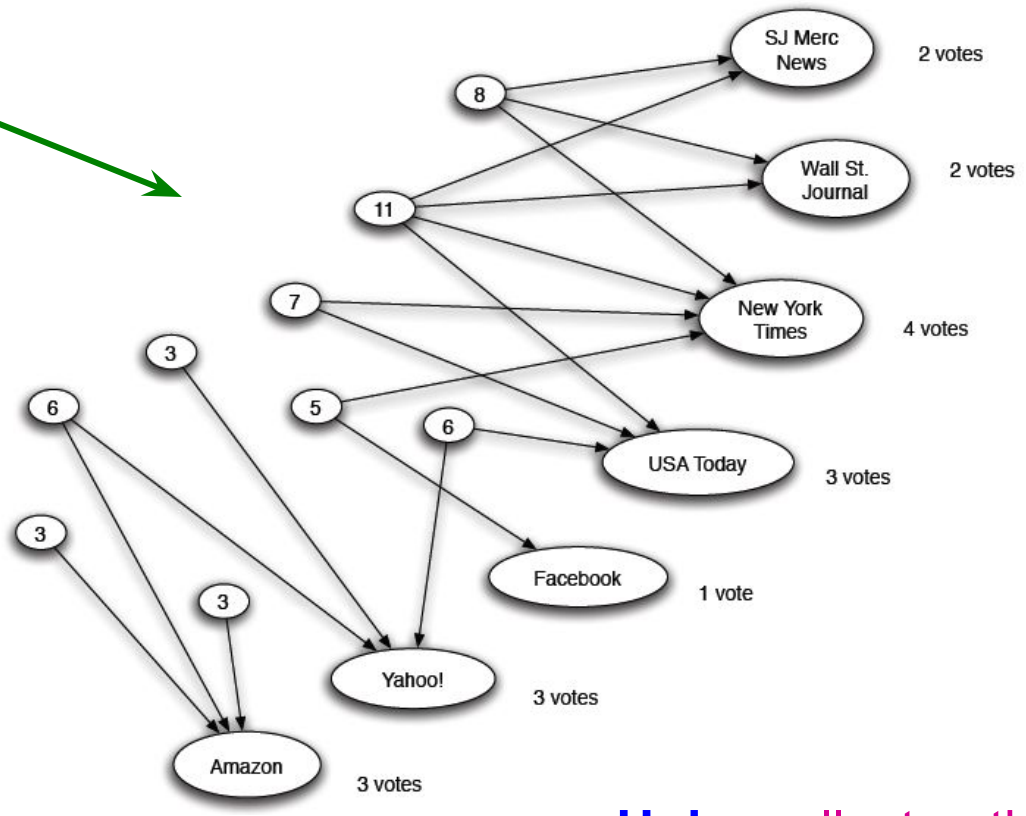
Sum of **hub** scores of nodes pointing to NYT.

Each page starts with **hub** score 1. **Authorities** collect their votes

(Note this is idealized example. In reality graph is not bipartite and each page has both the hub and authority score)

Expert Quality: Hub

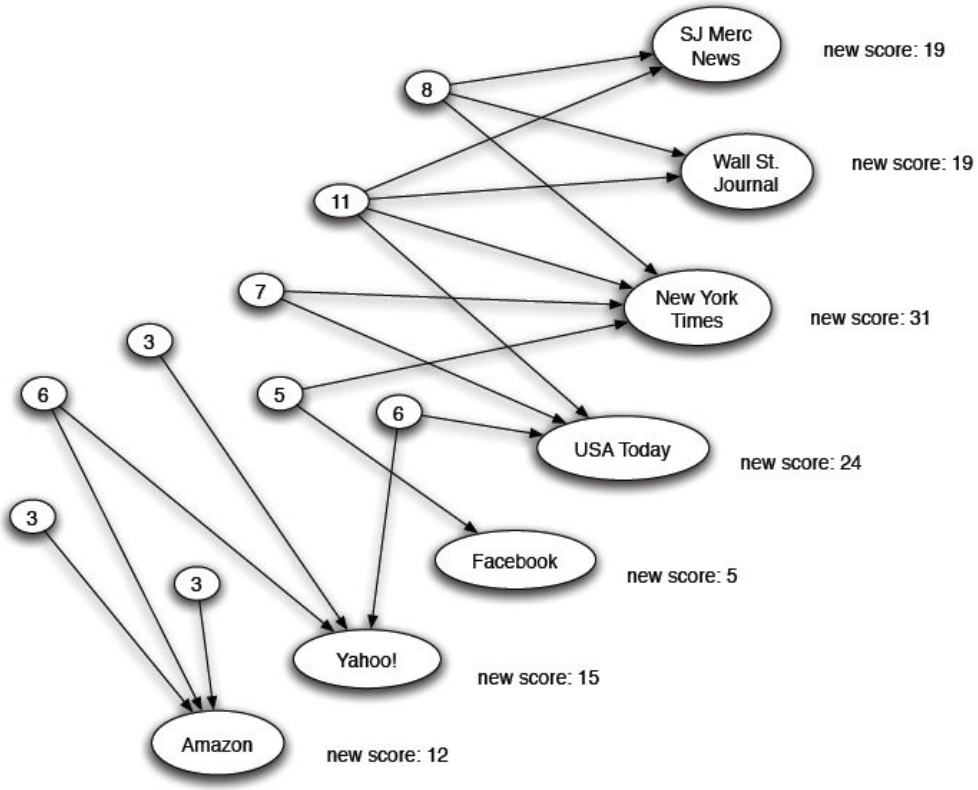
Sum of authority scores of nodes that the node points to.



Hubs collect authority scores

(Note this is idealized example. In reality graph is not bipartite and each page has both the hub and authority score)

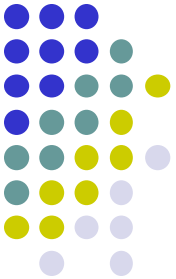
Reweighting



Authorities again collect the **hub** scores

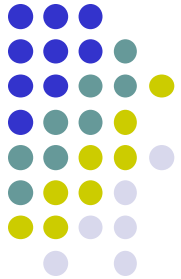
(Note this is idealized example. In reality graph is not bipartite and each page has both the hub and authority score)

Mutually Recursive Definition



- A good hub links to many good authorities
- A good authority is linked from many good hubs
- Model using two scores for each node:
  - Hub score and Authority score
  - Represented as vectors h and a

H.I.T.S.

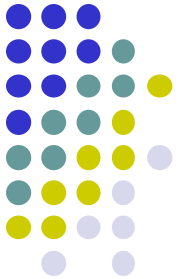


- Thus each page ***u*** has two different measures of merit
  - its authority score ***a[u]***
  - its hub score ***h[u]***

- The scores of all nodes give rise to two vectors ***a*** and ***h*** in which the components of index ***u*** are relative to the scores of node ***u***

- As for the PageRank, the quantitative definitions of hub and authority are recursive

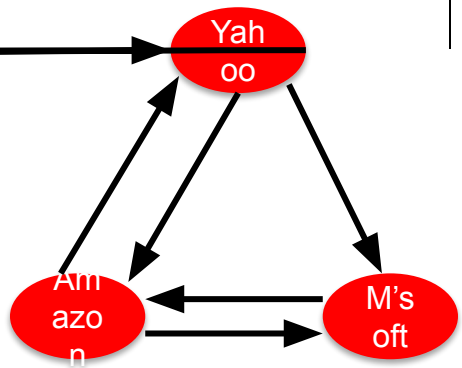
H.I.T.S.



- The authority score for a page is proportional to the sum of the hub scores of pages that link to it
- Conversely, its hub score is proportional to the authority scores of the pages to which it points.
- Basically $$a[u] = \sum_{v:(v,u)\in E} h[v] \qquad h[u] = \sum_{v:(u,v)\in E} a[v]$$
- In matrix terms: $\boldsymbol{a = E^T h}$ e $\boldsymbol{h = E a}$

Example of HITS

$$E = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad E^T = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

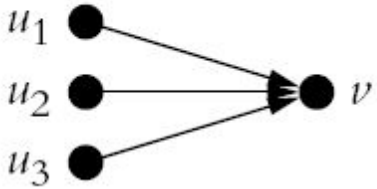


h(yahoo)	=	.58	.80	.80	.79	. . .	.788
h(amazon)	=	.58	.53	.53	.57	. . .	.577
h(m'soft)	=	.58	.27	.27	.23	. . .	.211

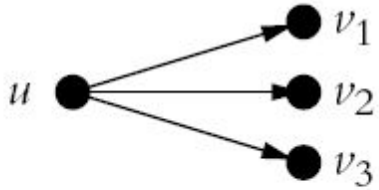
a(yahoo)	=	.58	.58	.62	.62	. . .	.628
a(amazon)	=	.58	.58	.49	.49	. . .	.459
a(m'soft)	=	.58	.58	.62	.62	. . .	.628

H.I.T.S.

- We see an example in the figure below

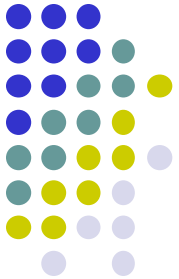


$$a(v) = h(u_1) + h(u_2) + h(u_3)$$



$$h(u) = a(v_1) + a(v_2) + a(v_3)$$

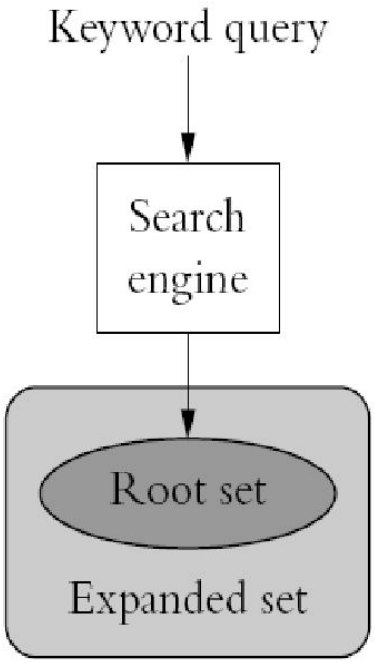
H.I.T.S.

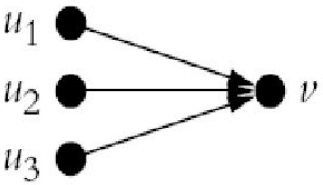


- Again, you can use the method of power iterations to solve this system of equations iteratively
- Initially for any ***u***, ***h[u]=1*** e ***a[u]=1***
- At each iteration, to prevent overflow problems, the vectors a and h are normalized using the norm L_1
- The converged values of ***a*** and ***h*** correspond to the principal eigenvectors of $E^T E$ and $E E^T$
- Typically, executions with many thousands of nodes and links "converge" in a number of iterations from 20 to 30

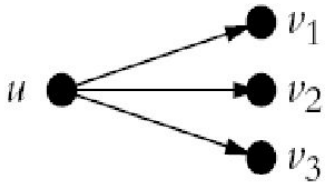
H.I.T.S.

- We summarize the HITS algorithm in the following figure:





$$a(v) = h(u_1) + h(u_2) + h(u_3)$$



$$h(u) = a(v_1) + a(v_2) + a(v_3)$$

$\vec{a} \leftarrow (1,\ldots,1)^T, \vec{h} \leftarrow (1,\ldots,1)^T$

while $\vec{h}$ and $\vec{a}$ change "significantly" **do**

$\vec{h} \leftarrow E\vec{a}$

$\ell_h \leftarrow \|\vec{h}\|_1 = \Sigma_w h[w]$

$h \leftarrow h/\ell_h$

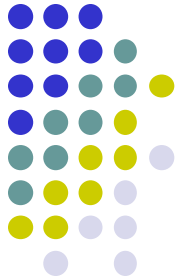
$\vec{a} \leftarrow E^T h_0 = E^T E\vec{a}_0$

$\ell_a \leftarrow \|\vec{a}\|_1 = \Sigma_w a[w]$

$\vec{a} \leftarrow \vec{a}/\ell_a$

end while

H.I.T.S.

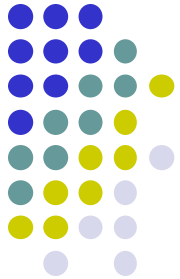


- In summary, the main steps of the HITS algorithm are
  - **submitting queries to an IR system based on texts and get the whole root**
  - **expand along the root of a unit radius for obtaining the expanded graph**
  - **run the power iterations simultaneously on the hub scores and authority**
  - **return hubs and authority with the highest scores**

- The whole process is generically called ***topic distillation***

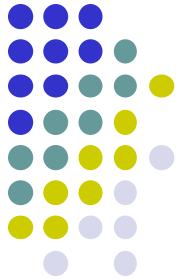
- Several studies have shown that the hub is more useful than the authority, because they provide a useful starting point for users for the exploration for a topic

H.I.T.S.: pro and cons



- The HITS algorithm cannot pre-calculate the hub scores and authority because the graph G_q can be calculated only after the query q is known.
- This represents both a strength and weakness
- Clearly, giving authority through connectivity makes more sense when restricted to a subgraph of the Web that is relevant to a query; then, we expect HITS, once the scores are calculated, to need less gimmicks to get the ranking of pages than PageRank
- On the other hand, with respect to PageRank it must perform the calculation of eigenvectors to each query
- The key distinction between HITS and PageRank is the modeling of the hub.
- PageRank has no concept of a hub, but that does not seem to be a big handicap in searches, probably because major hub on the Web soon accumulate incoming links and therefore high prestige, becoming also good authoritative pages

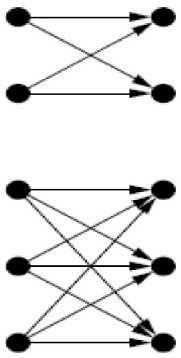
H.I.T.S.: problems



- HITS is affected by what is called the TKC(Tightly Knit Community) effect : When it takes over a small collection of pages (in relation to a given topic) is connected so that each hub page has a link to each authority page (bipartite cores)
- A tightly-knit community is a small set of highly interconnected sites: that community determines high scores in the links analysis algorithms, even if the sites in TKC are not authoritative on the subject, or are only relevant to one aspect
- This is due to a mutual reinforcement of the nodes in a bipartite graph: the authority (right) make up the scores of the hub (right), which in turn increases again the scores of the authorities, and the cycle repeats endlessly
- The TKC effect could be exploited by spammers in order to increase the weight of their pages

H.I.T.S.: problems

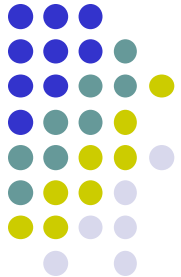
- A slightly larger TKC (or bipartite graph) can completely hide other TKC (or bipartite graphs), in the sense that their scores become negligible (compared to the initial TKC)
- In fact, ignoring for the moment the normalization of vectors, we have the following situation



Iteration	h_{small}	a_{small}	h_{large}	a_{large}
0	1	0	1	0
1a	1	2	1	3
1h	4	2	9	3
2a	4	8	9	27
2h	16	8	81	27

- In other words, the relationship between the authority scores of the two graphs at iteration *i* is $a_{large}/a_{small}=(3/2)^{2i-1}$, which grows indefinitely to growing of the *i*
- As a result, by normalizing the principal eigenvector the smaller bipartite would have no representation

H.I.T.S.: problems



- The expansion step of the base set is a mechanism to increase the recall but it affects the precision
- In fact, you may have contamination problems with the topic specified by the query:
  - generalization: related topics nodes fall into the expanded set
  - drift: totally different topics nodes can fall into the expanded set (such as Firefox browser homepage or Adobe homepage)
- As a result, the authority and hub scores can not be related to the topic of interest
- Another issue of HITS is that small perturbations (changes) of the Web graph can have dramatic effects on the hub scores and authority