




Inspire…Educate…Transform.

4. Link Analysis Algorithms

Dr. Manish Gupta

Sr. Mentor – Academics, INSOFE

Adapted from http://infolab.stanford.edu/~ullman/mining/2009/PageRank.ppt

Course Content

- Collection of three main topics of high recent interest.
  - Search engines (Crawling, Indexing, Ranking)
    - Language Modeling
    - Text Indexing and Crawling
    - Relevance Ranking
    - **Link Analysis Algorithms**
  - Text Processing (NLP, NER, Sentiments)
    - Natural Language Processing
    - Named Entity Recognition
    - Sentiment Analysis
    - Summarization
  - Social networks (Properties, Influence Propagation)
    - Social Network Analysis
    - Influence Propagation in Social Networks

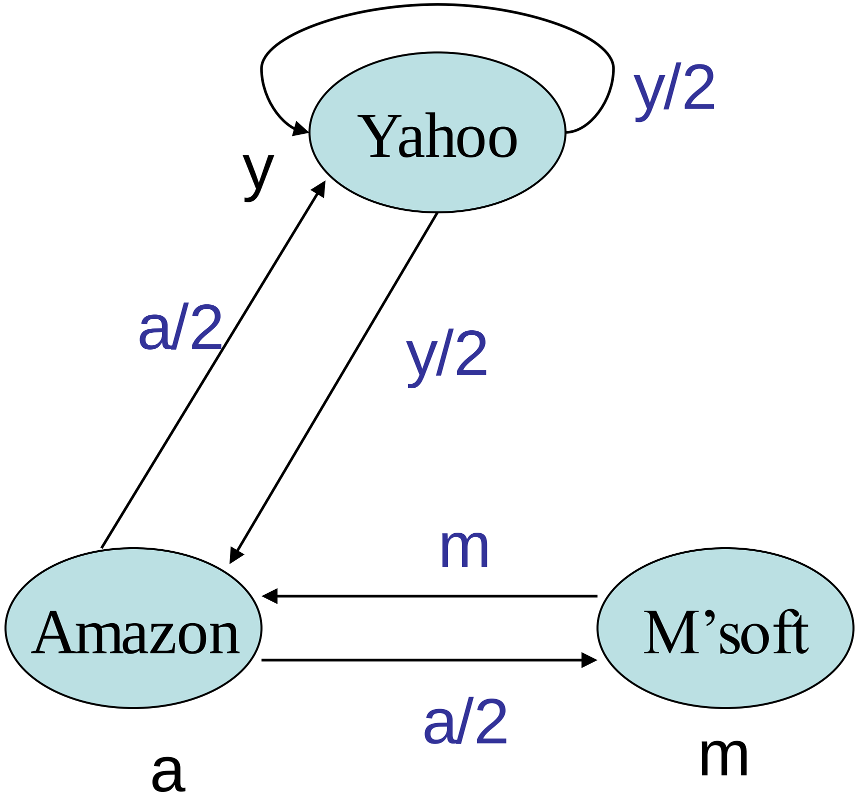
Today’s Agenda

- **PageRank**

Ranking Web Pages

- Web pages are not equally "important"
  - www.joe-schmoe.com vs www.stanford.edu
- Inlinks as votes
  - www.stanford.edu has 23,400 inlinks
  - www.joe-schmoe.com has 1 inlink
- Are all inlinks equal?
  - Recursive question
  - Each link's vote is proportional to the importance of its source page
  - If page P with importance x has n outlinks, each link gets x/n votes
  - Page P's own importance is the sum of the votes on its inlinks

Simple "Flow" Model



$$y = y/2 + a/2$$
$$a = y/2 + m$$
$$m = a/2$$

CSE 7306c

Solving the Flow Equations

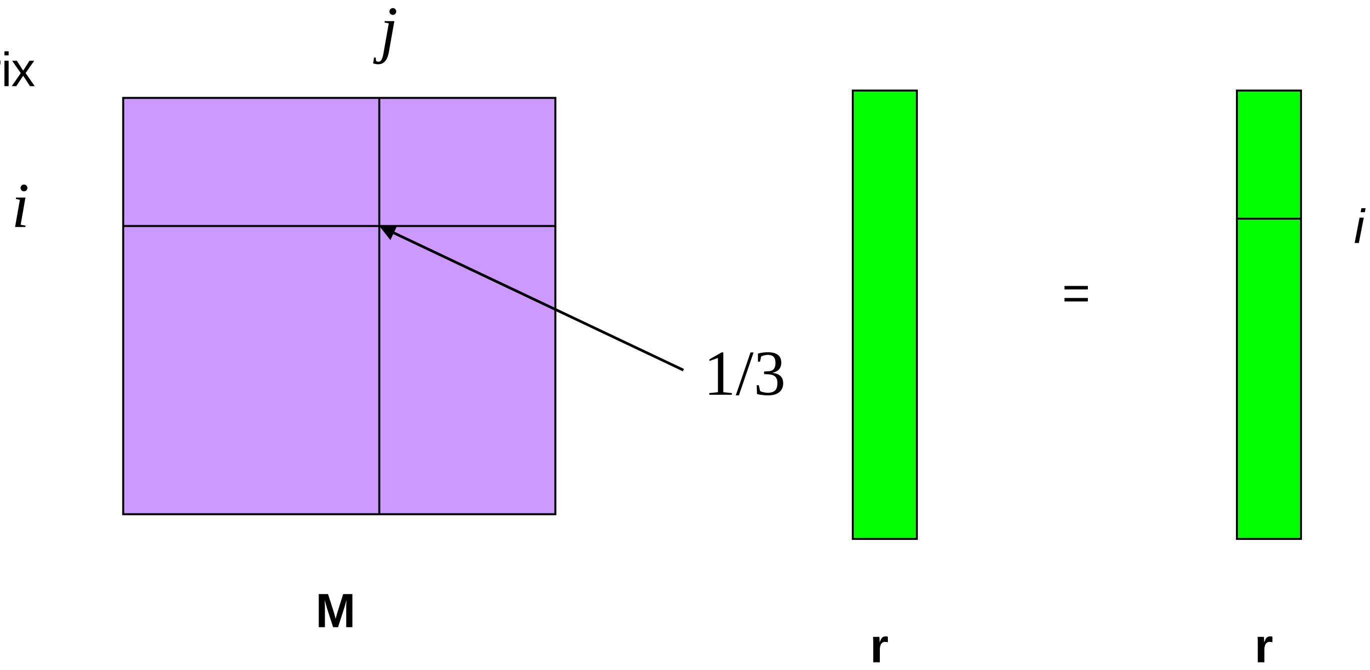
- 3 equations, 3 unknowns, no constants
  - No unique solution
  - All solutions equivalent modulo scale factor
- Additional constraint forces uniqueness
  - $y+a+m = 1$
  - $y = 2/5$, $a = 2/5$, $m = 1/5$
- Gaussian elimination method works for small examples, but we need a better method for large graphs

CSE 7306c

Matrix Formulation

Suppose page j links to 3 pages, including i

M=web linkage matrix
r=rank vector



$$y = y/2 + a/2$$
$$a = y/2 + m$$
$$m = a/2$$

$$\begin{bmatrix} y \\ a \\ m \end{bmatrix} = \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1 \\ 0 & 1/2 & 0 \end{bmatrix} \begin{bmatrix} y \\ a \\ m \end{bmatrix}$$

r is the principal eigen-vector of M

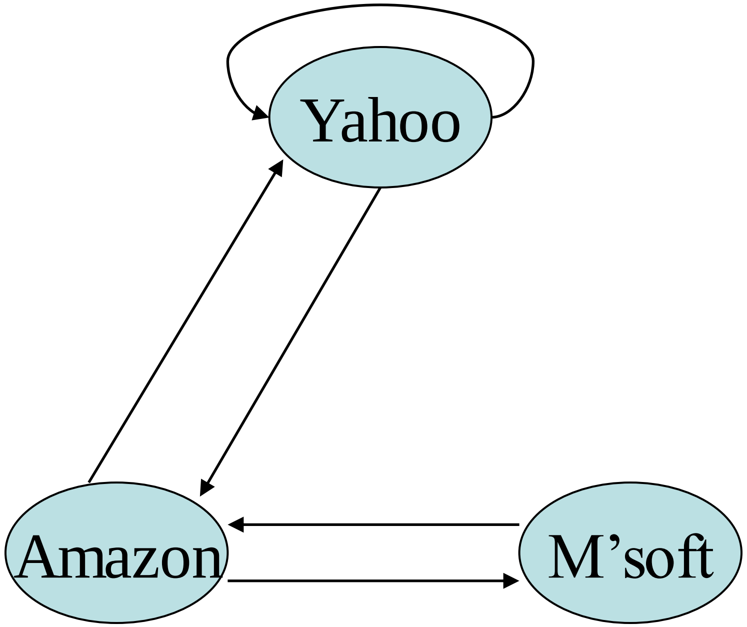
CSE 7306c

Power Iteration Method

- Simple iterative scheme
- Suppose there are N web pages
- Initialize: $\mathbf{r}^0 = [1/N,\ldots.,1/N]^T$
- Iterate: $\mathbf{r}^{k+1} = \mathbf{M}\mathbf{r}^k$
- Stop when $|\mathbf{r}^{k+1} - \mathbf{r}^k|_1 < \varepsilon$
  - $|\mathbf{x}|_1 = \sum_{1 \le i \le N} |x_i|$ is the L_1 norm
  - Can use any other vector norm e.g., Euclidean

CSE 7306c

Power Iteration Example



$$
\begin{array}{c|ccc|}
 & y & a & m \\
\hline
y & 1/2 & 1/2 & 0 \\
a & 1/2 & 0 & 1 \\
m & 0 & 1/2 & 0 \\
\hline
\end{array}
$$

$$
\begin{array}{c} y \\ a \\ m \end{array} = \begin{array}{c} 1/3 \\ 1/3 \\ 1/3 \end{array} \quad \begin{array}{c} 1/3 \\ 1/2 \\ 1/6 \end{array} \quad \begin{array}{c} 5/12 \\ 1/3 \\ 1/4 \end{array} \quad \begin{array}{c} 3/8 \\ 11/24 \\ 1/6 \end{array} \quad \ldots \quad \begin{array}{c} 2/5 \\ 2/5 \\ 1/5 \end{array}
$$

Random Walk Interpretation

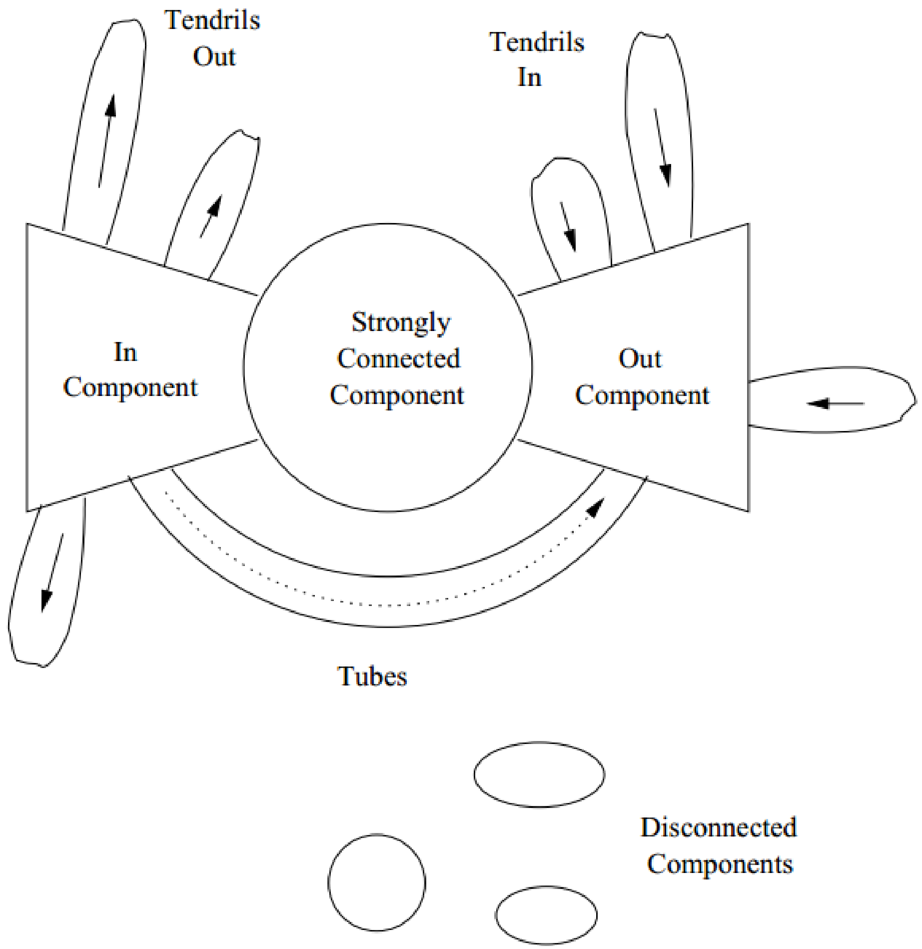
- Imagine a random web surfer
  - At any time t, surfer is on some page P
  - At time t+1, the surfer follows an outlink from P uniformly at random
  - Ends up on some page Q linked from P
  - Process repeats indefinitely
- Let **p**(t) be a vector whose i^{th} component is the probability that the surfer is at page i at time t
  - **p**(t) is a probability distribution on pages

CSE 7306c

The Stationary Distribution

- Where is the surfer at time t+1?
  - Follows a link uniformly at random
  - $\mathbf{p}(t+1) = \mathbf{M}\mathbf{p}(t)$
- Suppose the random walk reaches a state such that $\mathbf{p}(t+1) = \mathbf{M}\mathbf{p}(t) = \mathbf{p}(t)$
  - Then $\mathbf{p}(t)$ is called a stationary distribution for the random walk
- Our rank vector $\mathbf{r}$ satisfies $\mathbf{r} = \mathbf{M}\mathbf{r}$
  - So it is a stationary distribution for the random surfer

Bow-tie Structure of the Web



CSE 7306c

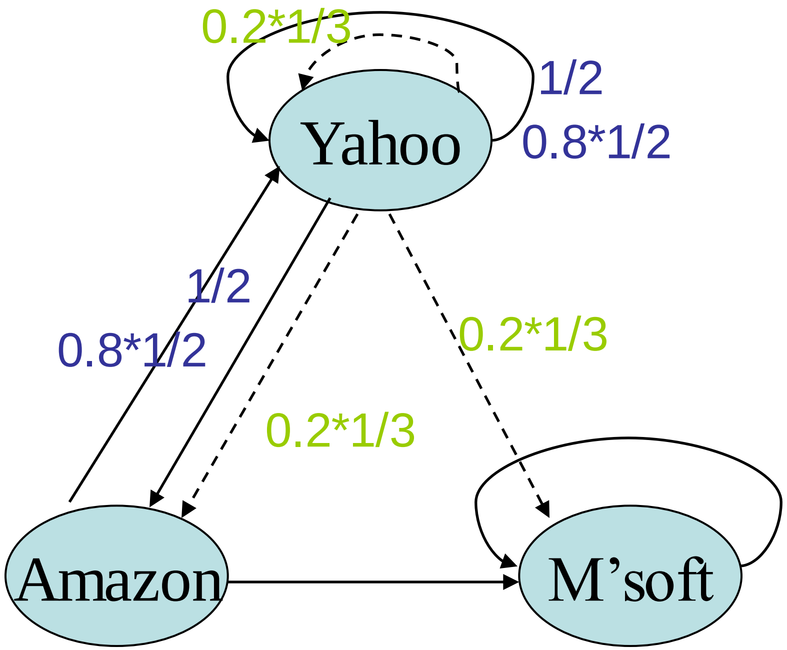
Spider Traps

- A group of pages is a spider trap if there are no links from within the group to outside the group
  - Random surfer gets trapped
- Spider traps violate the conditions needed for the random walk theorem

Random Teleports

- Solution for spider traps
- At each time step, the random surfer has two options:
  - With probability β, follow a link at random
  - With probability $1-\beta$, jump to some page uniformly at random
  - Common values for β are in the range 0.8 to 0.9
- Surfer will teleport out of spider trap within a few time steps

Random Teleports ($\beta = 0.8$)



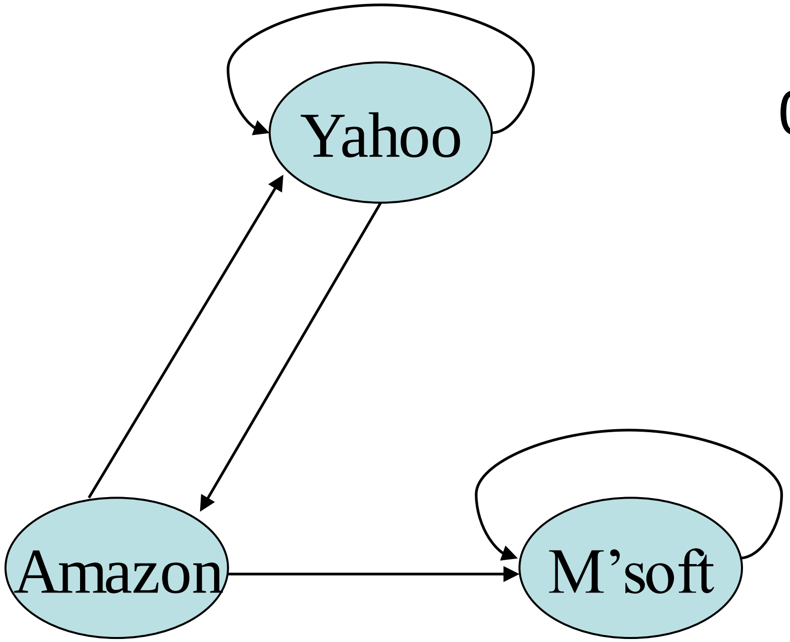
$$\begin{array}{cc} & y \\ \begin{array}{c} y \\ a \\ m \end{array} & \boxed{\begin{array}{c} 1/2 \\ 1/2 \\ 0 \end{array}} \end{array} \quad 0.8* \begin{array}{c} y \\ \boxed{\begin{array}{c} 1/2 \\ 1/2 \\ 0 \end{array}} \end{array} \quad + 0.2* \begin{array}{c} y \\ \boxed{\begin{array}{c} 1/3 \\ 1/3 \\ 1/3 \end{array}} \end{array}$$

$$0.8 \boxed{\begin{array}{ccc} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 0 \\ 0 & 1/2 & 1 \end{array}} \quad + 0.2 \boxed{\begin{array}{ccc} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{array}}$$

$$\begin{array}{cc} \begin{array}{c} y \\ a \\ m \end{array} & \boxed{\begin{array}{ccc} 7/15 & 7/15 & 1/15 \\ 7/15 & 1/15 & 1/15 \\ 1/15 & 7/15 & 13/15 \end{array}} \end{array}$$

CSE 7306c

Random Teleports ($\beta = 0.8$)



$$0.8 \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 0 \\ 0 & 1/2 & 1 \end{bmatrix} + 0.2 \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix}$$

$$\begin{array}{c} y \\ a \\ m \end{array} \begin{bmatrix} 7/15 & 7/15 & 1/15 \\ 7/15 & 1/15 & 1/15 \\ 1/15 & 7/15 & 13/15 \end{bmatrix}$$

$$\begin{array}{c} y \\ a \\ m \end{array} = \begin{array}{c} 1 \\ 1 \\ 1 \end{array} \quad \begin{array}{c} 1.00 \\ 0.60 \\ 1.40 \end{array} \quad \begin{array}{c} 0.84 \\ 0.60 \\ 1.56 \end{array} \quad \begin{array}{c} 0.776 \\ 0.536 \\ 1.688 \end{array} \quad \ldots \quad \begin{array}{c} 7/11 \\ 5/11 \\ 21/11 \end{array}$$

The PageRank vector **r** is the stationary distribution of the random walk with teleports

CSE 7306c

Dealing with Dead-ends

- Pages with no outlinks are "dead ends" for the random surfer
- Teleport
  - Follow random teleport links with probability 1.0 from dead-ends
  - Adjust matrix accordingly
- Prune and propagate
  - Preprocess the graph to eliminate dead-ends
  - Might require multiple passes
  - Compute PageRank on reduced graph
  - Approximate values for deadends by propagating values from reduced graph

Today’s Agenda

- PageRank
- **Topic-Specific PageRank**

CSE 7306c

Topic-sensitive PageRank

- $r = \beta Mr + (1-\beta)p$
- **Conventional PageRank:** **p** is a uniform vector with values **1/N**
- Topic-sensitive PageRank uses a **non-uniform** personalization vector **p**
- Not simply a post-processing step of the PageRank computation
- Personalization vector **p** introduces bias in all iterations of the iterative computation of the PageRank vector

CSE 7306c

Personalization Vector

CSE 7306c

Topic-sensitive PageRank: Overall Approach

- Preprocessing
  - Fix a set of **k** topics
  - For each topic c_j compute the PageRank scores of page **u** wrt to the **j**-th topic: $r(u,j)$
- Query-time processing:
  - For query q compute the total score of page **u** wrt **q** as $score(u,q) = \Sigma_{j=1 \ldots k} Pr(c_j|q)\, r(u,j)$

CSE 7306c

Topic-sensitive PageRank: Preprocessing

- Create **k** different biased PageRank vectors using some pre-defined set of k categories **($c_1, \ldots, c_k$)**
  - E.g., Open Directory (DMOZ)'s 16 top level categories like sports, medicine, etc.
- T_j: set of URLs in the **j**-th category
- Use non-uniform personalization vector p_j such that:

$$p_j(v) = \begin{cases} \dfrac{1}{T_j} \ldots . v \in T_j \\ 0 \ldots \text{otherwise} \end{cases}$$

CSE 7306c

Topic-sensitive PageRank: Query Processing

- **$\text{score}(u,q) = \Sigma_{j=1\ldots k} \Pr(c_j|q)\, r(u,j)$**

$$\Pr(c_j \mid q) = \frac{\Pr(c_j)\Pr(q \mid c_j)}{\Pr(q)} \propto \Pr(c_j)\prod_i \Pr(q_i \mid c_j)$$

- How can we compute **$P(c_j)$**?
  - Can be fixed as uniform
  - Can be biased to a particular set of categories if we have that information about the user
- How can we compute **$\Pr(q|c_j)$**?
  - Estimated using n-gram model from set of URLs in j-th category.

CSE 7306c

Take-away Messages

- Linkage on the web is an important phenomena
- Links contain a lot of knowledge
- Knowledge in the link structure can be used to rank webpages
  - Using PageRank
  - Using Topic Sensitive PageRank

Further Reading

- A nice summary of link analysis algorithms from John Kleinberg
  - http://dl.acm.org/citation.cfm?id=345982
- Chapter 5: "Link Analysis" from Mining of Massive Datasets
  - http://infolab.stanford.edu/~ullman/mmds.html
- Chapter 7 (Social Network Analysis) from Mining the Web
  - http://www.cse.iitb.ac.in/soumen/mining-the-web/
- J.M. Kleinberg: Authoritative Sources in a Hyperlinked Environment, JACM 46(5), 1999
- S Brin, L. Page: The Anatomy of a Large-Scale Hypertextual Web Search Engine, WWW 1998
- M. Najork, H. Zaragoza, M. Taylor: HITS on the Web: How does it Compare?, SIGIR 2007
- R. Lempel, S. Moran: SALSA: The Stochastic Approach for Link-Structure Analysis, ACM TOIS 19(2), 2001.
- G. Jeh, J. Widom: SimRank: a Measure of Structural-Context Similarity, KDD 2002
- Taher Haveliwala: Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search, IEEE Trans. on Knowledge and Data Engineering, 2003.
- G. Jeh, J. Widom: Scaling personalized web search, WWW 2003.
- D. Fogaras, B. Racz, K. Csalogany, A. Benczur: Towards Scaling Fully Personalized PageRank: Algorithms, Lower Bounds, and Experiments, Internet Mathematics 2(3): 333-358, 2006.

International School of Engineering

Plot 63/A, 1st Floor, Road # 13, Film Nagar, Jubilee Hills, Hyderabad - 500 033

For Individuals:	+91-9502334561/63 or 040-65743991
For Corporates:	+91-9618483483
Web:	http://www.insofe.edu.in
Facebook:	https://www.facebook.com/insofe
Twitter:	https://twitter.com/Insofeedu
YouTube:	http://www.youtube.com/InsofeVideos
SlideShare:	http://www.slideshare.net/INSOFE
LinkedIn:	http://www.linkedin.com/company/international-school-of-engineering