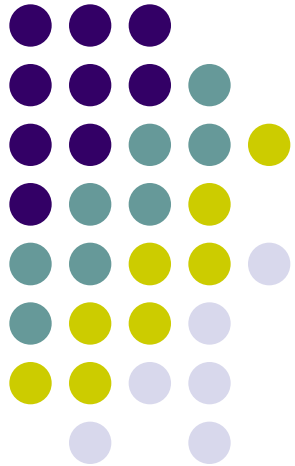
BÀI GiẢNG KHAI PHÁ DỮ LIỆU WEB

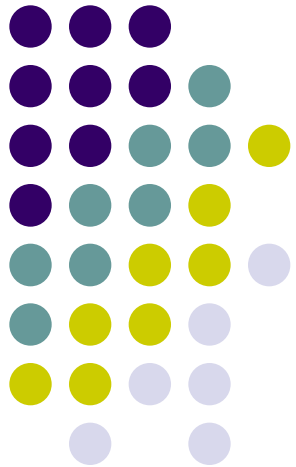
CHƯƠNG 1. GIỚI THIỆU CHUNG

PGS. TS. HÀ QUANG THỤY
HÀ NỘI 10-2010
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ
ĐẠI HỌC QUỐC GIA HÀ NỘI

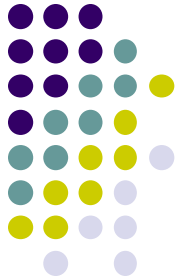


Nội dung

1. Giới thiệu về khai phá text
2. Giới thiệu về khai phá web

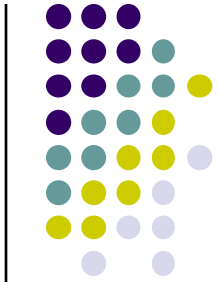


1. Giới thiệu về khai phá text



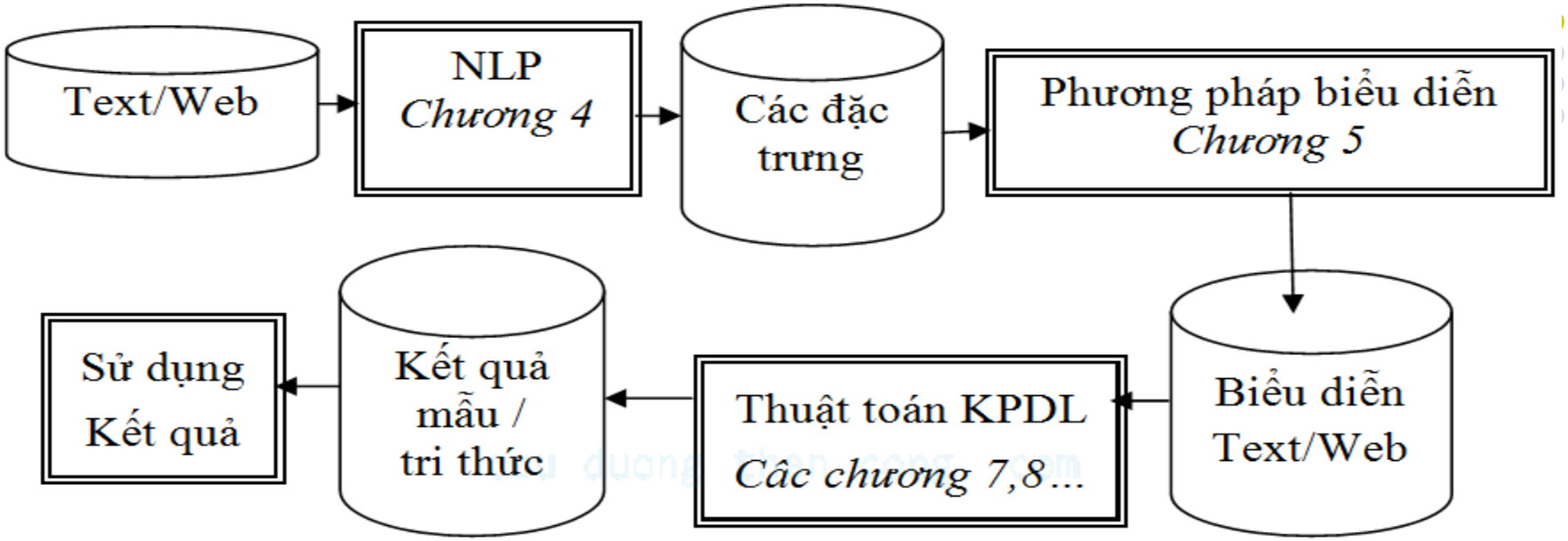
- Khái niệm
- Sự cần thiết của khai phá text
- Đặc trưng của khai phá text
- Các bài toán cơ bản trong khai phá text
- Một ví dụ về bài toán khai phá text
- Xu hướng nghiên cứu khai phá Text

Khái niệm



- Tiếp cận về khái niệm khai phá text
  - Khai phá text là khai phá dữ liệu đối với loại dữ liệu text.
  - Quá trình phát hiện tri thức mới, có giá trị, tiềm ẩn trong tập hợp văn bản
  - Mang tính đa dạng về phát biểu khái niệm khai phá dữ liệu
- Nội dung
  - Khai phá text = Khai phá dữ liệu + Xử lý ngôn ngữ tự nhiên - XLNNTN (Natural Language Processing: NLP)
  - Các bài toán chung về khai phá dữ liệu cho dữ liệu đặc thù
  - Một số bài toán riêng điển hình cho khai phá text
- Mối quan hệ giữa Khai phá Text và XLNNTN
  - XLNNTN cung cấp tài nguyên, công cụ cơ sở cho khai phá Text
  - Khai phá Text mở rộng các bài toán của XLNNTN
  - Đan xen giữa Khai phá Text với XLNNTN

Quy trình khai phá text



- Tuân theo quy trình chung của khai phá dữ liệu
  - Như đã trình bày trong khai phá dữ liệu
- Quy trình tối giản
  - Tiền xử lý
    - Công cụ của Xử lý ngôn ngữ tự nhiên
    - Mô hình cấu trúc văn bản
  - Biểu diễn văn bản
    - Phù hợp với thuật toán
  - Xử lý (khai phá) dữ liệu theo dạng biểu diễn
    - Áp dụng khai phá dữ liệu

Sự cần thiết của khai phá text

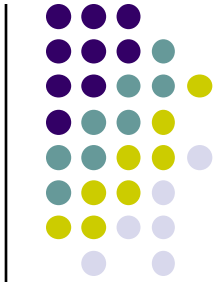
- Text gần gũi nhất với con người
  - Là đối tượng quan trọng nhất chuyển tải thông tin của loài người
  - Phương tiện trình bày tri thức chuyển giao người khác
  - Học chữ là bài toán quan trọng của mỗi con người
- Đặc thù của ngôn ngữ tự nhiên
  - Tính đa nghĩa, đồng nghĩa của đơn vị cú pháp nhỏ nhất là từ
  - Tính cảm ngữ cảnh khi trình bày nội dung văn bản
  - Tính biến động của mỗi ngôn ngữ tự nhiên: bổ sung, thay đổi…
- Sự tăng trưởng của dữ liệu Text
  - Khả năng tạo mới
  - Khả năng lưu trữ

Đặc trưng của khai phá text

Dấu hiệu phân biệt	*Khai phá dữ liệu*	*Khai phá Text*
Đối tượng dữ liệu	Dữ liệu số / phân loại	Văn bản
Cấu trúc đối tượng	CSDL quan hệ	Text dạng tự do: không cấu trúc, nửa cấu trúc
Mục tiêu	Dự báo, đoán nhận	Tìm kiếm thông tin liên quan, hiểu ngữ nghĩa, phân lớp / phân bố
Phương pháp	Học máy: DT, MBR, …	Chỉ số, xử lý mạng nơron, ngôn ngữ, kiến trúc
Kích cỡ thị trường	Trăm nghìn phân tích viên từ công ty lớn và vừa	Hàng triệu người dùng từ hãng và cá nhân
Tình trạng	Quảng bá từ năm 1994	Mới quảng bá từ năm 2000

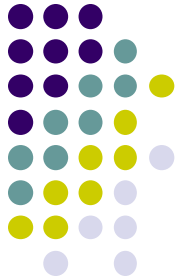
Sergei Ananyan (2001). Text Mining: Applications and Technologies, *Megaputer Intelligence Inc.*. (truy nhập ngày 13/9/2003)

Một số bài toán điển hình trong TM



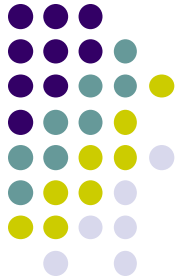
- Biểu diễn Text
  - Là một trong những bài toán quan trọng nhất trong khai phá Text
  - Nghịch lý về "hiệu quả như nhau" trong tìm kiếm Text
  - Tìm biểu diễn phù hợp nhất cho bài toán khai phá text
  - Một lớp hướng mô hình biểu diễn Text: Mô hình sinh Text
  - Nội dung của chương 2.
- Tìm kiếm/thu hồi Text (Text Search/Retrieval)
  - Cho một tập văn bản và một yêu cầu tìm kiếm của người dùng (dạng văn bản / khác).
  - Mục đích: Tìm tập văn bản trong CSDL đáp ứng yêu cầu người dùng
  - Đã tồn tại một CSDL Text: Tìm kiếm full-text trong CSDL này
  - Tìm kiếm trên Internet. Máy tìm kiếm: Nội dung chương 5.

Một số bài toán điển hình trong TM (2)



- Phân lớp văn bản
  - Tương ứng học có giám sát (học có thầy)
  - Cho trước tập lớp và tập ví dụ
  - Mục tiêu : một mô hình phân lớp thực hiện ánh xạ mỗi văn bản vào lớp
  - Ví dụ:
- Phân cụm văn bản
  - Tương ứng hoc không giám sát
  - Cho trước tập văn bản
  - Mục tiêu : tập cụm văn bản và tóm tắt cụm.
  - Ví dụ:
- Phân đoạn văn bản
  - Phân cụm và phân lớp
  - Ví dụ:

Một số bài toán điển hình trong TM (3)

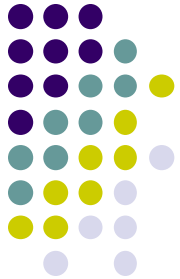


- Phân tích ngữ nghĩa
  - Hiểu văn bản (xem DUC: Document Understanding Conferences và TAC: Text Analysis Conferences)
  - Ngữ nghĩa của các thành phần trong văn bản
  - Phát hiện quan hệ thực thể trong văn bản
  - Taxonomy, ontology, web ngữ nghĩa (semantic Web)
  - Roxana Girju [Gij08] liệt kê một số danh sách quan hệ ngữ nghĩa, trong đó có danh sách 22 quan hệ do chính tác giả tổng hợp:

HYPERNYMY (IS-A)	PART-WHOLE (MERONYMY)	CAUSE	POSSESSION
KINSHIP	MAKE/PRODUCE	INSTRUMENT	TEMPORAL
LOCATION/SPACE	PURPOSE	SOURCE/FROM	EXPERIENCER
TOPIC	MANNER	MEANS	GENT
THEME	PROPERTY	BENEFICIARY	MEASURE
TYPE	DEPICTIONDEPICTED.		

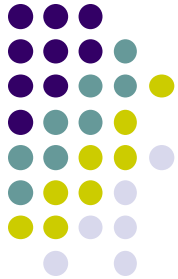
[Gir08] Roxana Girju (2008). Semantic Relation Extraction and its Applications, *ESSLLI 2008*: Invited Tutorial, Hamburg, Germany, August 2008

Một số bài toán điển hình trong TM (4)



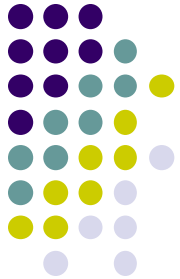
- Trích chọn đặc trưng
  - Phát hiện/lưu trữ từ khóa (term), đặc trưng (feature), cụm từ mang nghĩa
  - Đặc trưng chưa định trước: xác định đồng thời với phân tích nội dung
  - Phân biệt trích chọn đặc trưng (feature extraction) với chọn lựa đặc trưng (feature selection)
  - Phân tích văn bản để phát hiện tần số xuất hiện
- Tóm tắt văn bản
  - Document Abstract/Summarization
  - Xây dựng một văn bản thu gọn hơn (tỷ lệ/số lượng từ/câu) song vẫn giữ được ngữ nghĩa
  - Abstract (rút trích câu) /Summarization (xây dựng câu)
  - Xây dựng tự động mục lục văn bản
  - Tóm tắt đơn văn bản/ tóm tắt đa văn bản
  - Quan hệ chặt chẽ với “hiểu văn bản”

Một số bài toán điển hình trong TM (5)



- Xây dựng ontology
  - Kho ngữ liệu về một/một nhóm lĩnh vực
  - Phục vụ, nâng cao chất lượng các bài toán ngữ nghĩa
  - Tập khái niệm, lớp khái niệm, quan hệ giữa chúng
  - Biểu diễn hình học dạng đồ thị
  - Dạng đặc biệt: Taxonomy
  - Ví dụ: WordNet, TreeBank
- Kế thừa nguyên bản (Textual Entailment)
  - "Văn bản T kế thừa giả thiết nguyên bản H" nếu tính chân thực của H có thể được suy diễn từ T.
  - "Ý nghĩa" của T tiềm ẩn trong H: trình bày nào đó của H có thể phù hợp trình bày nào đó của T (mức độ chi tiết hay trừu tượng)
- Dẫn đường văn bản (Text focusing)
  - Tích hợp xử lý văn bản với cơ sở tri thức cho phép kết nối trực tiếp tri thức trong quá trình xử lý văn bản
  - Dẫn dắt các văn bản theo tri thức đã được kết nối

Một số bài toán điển hình trong TM (6)



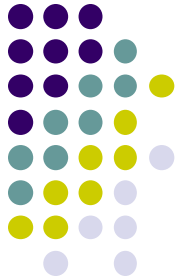
- Khai phá quan điểm
  - Là chủ đề thời sự hiện nay
  - Đối tượng: không là sự vật/ hiện tượng mà là tình cảm thái độ
  - Ứng dụng: tiếp thị (quan hệ khách hàng), điều tra xã hội học…
  - Một số ví dụ
- Khai phá Text trong lĩnh vực cụ thể
  - Y Sinh học: Quan hệ tương tác protein – protein, gene – bệnh
  - Các lĩnh vực khoa học khác:

Một số bài toán ví dụ

- Ví dụ 1
  - Nêu bài toán: Nhằm mục đích quán lý, một công ty Nhật Bản muốn xây dựng một hệ thống "quản lý" các nội dung đã được máy in của công ty in ra.
  - Đặt vấn đề:
    - Xây dựng hệ thống quản lý văn bản với thuộc tính in văn bản. Do một số lý do, đây không phải là điều công ty muốn.
    - Quản lý mọi nội dung được in ra: Dữ liệu nguồn chỉ có thể là dòng dữ liệu đi qua máy in của công ty. Cần xây dựng hệ thống có các năng lực (1) lấy được dòng dữ liệu Text đi tới các máy in; (2) Tổ chức lại hệ thống các văn bản được in ra để thuận tiện cho việc quản lý.
  - Giải pháp:
    - Thu nhận dữ liệu: Xây dựng luồng xử lý dòng dữ liệu vào máy in, một bản đưa ra máy in và một bản đưa vào thành phần xử lý tiếp theo.
    - Tổ chức hệ thống văn bản: Tiền xử lý dữ liệu; phân lớp đã cấp (trong đó có phân cụm)

Nguồn: từ một học viên công tác tại FSOFT làm việc với Nhật Bản

Một số bài toán ví dụ (2)



- Ví dụ 2. Bài toán của Rich Caruana & cộng sự
  - Bài toán: Cho trước một tập (khoảng 300000) công trình nghiên cứu khoa học (bài đăng tạp chí, báo cáo hội nghị, luận án Tiến sỹ) đã được công bố.

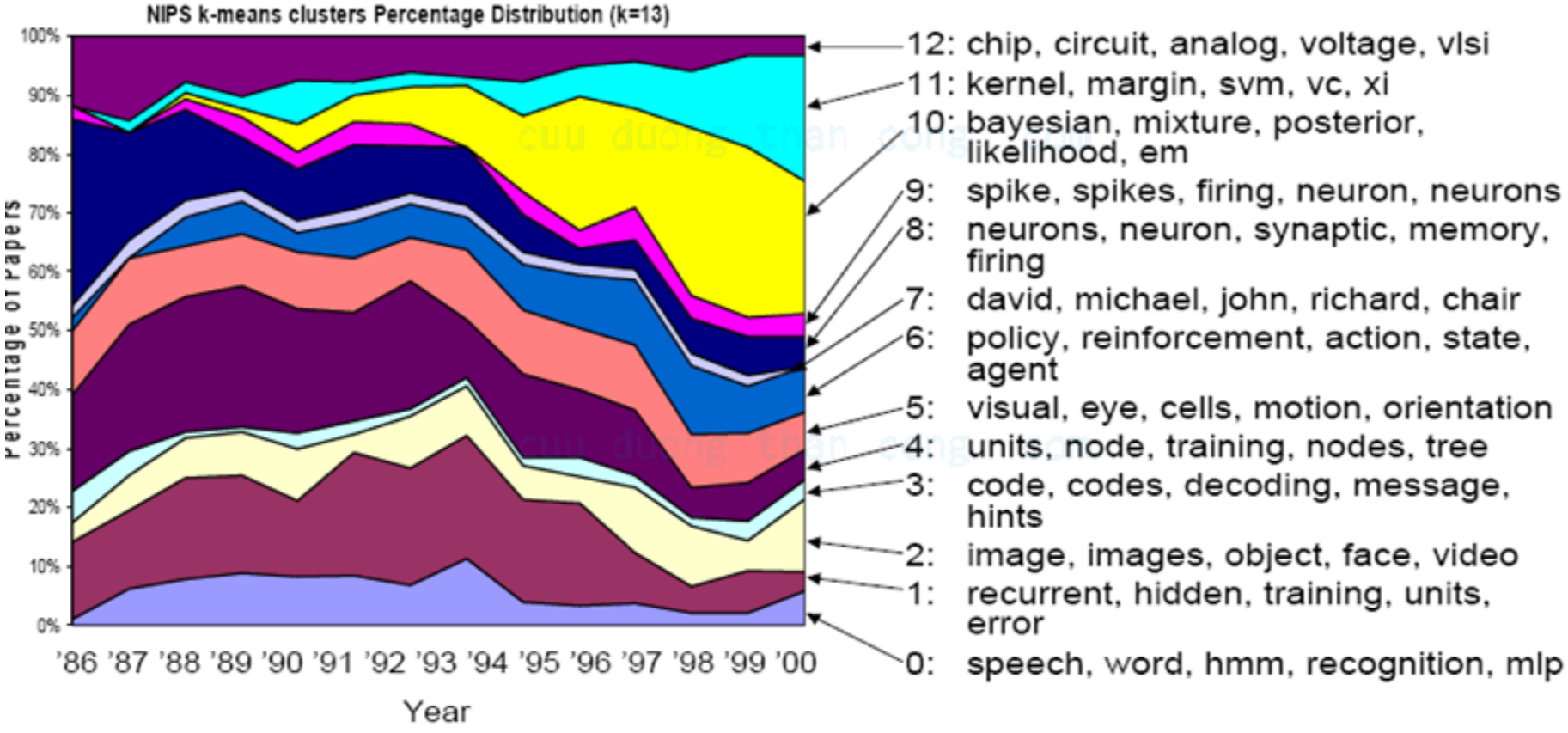
    Từ nội dung văn bản của mỗi công trình nghiên cứu, chúng ta nhận được tên tác giả (các tác giả), các tài liệu tham khảo, nơi công bố (tên tạp chí, hội nghị, hội thảo …).
  - Yêu cầu: Chỉ dùng *nội dung*, *năm XB* và *tên các tác giả* của tài liệu, tìm ra:
    - Tìm ra diễn biến theo thời gian của các chủ đề khoa học theo một số tiêu chí như tỷ lệ các tài liệu theo các chủ đề, các chủ đề nổi bật mới, thời điểm một chủ đề cụ thể đat đỉnh cao nhất, chủ đề nào đang tàn lụi... và theo đó, tìm ra được các chủ đề có vai trò chủ chốt.
    - Nhận biết được các tài liệu có uy thế là tài liệu giới thiệu các ý tưởng mới và có chỉ số ảnh hưởng lớn
    - Nhận biết được tác giả có uy thế là tác giả có ảnh hưởng lớn đối với sự phát triển của các chủ đề.

[CJG06] Rich Caruana, Thorsten Joachims, Johannes Gehrke, Benyah Shaparenko (2006). Patterns and Key Players in Document Collections, *KDD Challenge 2005.*

Một số bài toán ví dụ

- Ví dụ 2. Một kết quả [CJG06]
  - Phân cụm tài liệu và gán nhãn cụm (bằng các từ khóa điển hình trong cụm)
  - Biểu diễn hình học theo thời gian

Temporal Cluster Histograms: Results



Các khách hàng sử dụng sản phẩm TextAnalyst của Megaputer

TextAnalyst

Customer base: 300+ installations

Sample customers

- **Ask Jeeves** (USA)
- **IMS Health** (USA)
- **The Gallup Organization** (USA)
- **Centers for Disease Control** (USA)
- **Best Buy** (USA)
- **France Telecom** (France)
- **Skila.com** (USA)
- **US Navy** (USA)
- **Dow Chemical** (USA)
- **Clontech** (USA)
- **Pfizer** (USA)
- **TRW** (USA)
- **McKinsey & Company** (USA)
- **Liberty Mutual** (USA)
- **Logicon** (USA)
- **Net Shepherd** (Canada)
- **Dept of Environmental Protection** (Australia)
- **KPN Research** (Netherlands)
- **Talkie.com** (USA)
- **NICE Systems** (Israel)

Nguồn: *http://www.megaputer.com/company/index.php3*

Ví dụ về Dự án Khai phá Text

Briefs

TEMIS, Leader in Text Mining in Europe, raises 3.6 million euros.

Paris, November 9th 2004 - TEMIS, provider of corporate Text Mining solutions, has just completed a 3.6 million euro round of financing with ACE Management and Crédit Lyonnais Private Equity (CLPE).

TEMIS develops and markets software solutions for Text Mining. The software transforms free text into usable data, enabling either retrieval of relevant data contained in a document or automatic document classification by topic or by recipient.

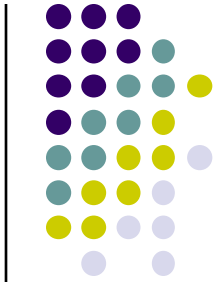
At a time when information flow in organizations is constantly increasing, TEMIS' software plays a crucial role in processing this information. It enables decisive productivity gains, in particular in the fields of Competitive Intelligence, analysis of scientific documents, and in customer relationship management.

TEMIS doubled its revenue between 2002 and 2003 and is set to achieve a similar performance for 2004. Its customers include major companies such as Novartis, IPSEN, Total, PSA Peugeot-Citroën, DaimlerChrysler, TIM-Telecom Italia Mobile, etc.

For more information, visit http://www.temis-group.com/

Nguồn: *http://www.megaputer.com/company/index.php3*

Nghiên cứu về khai khá Text

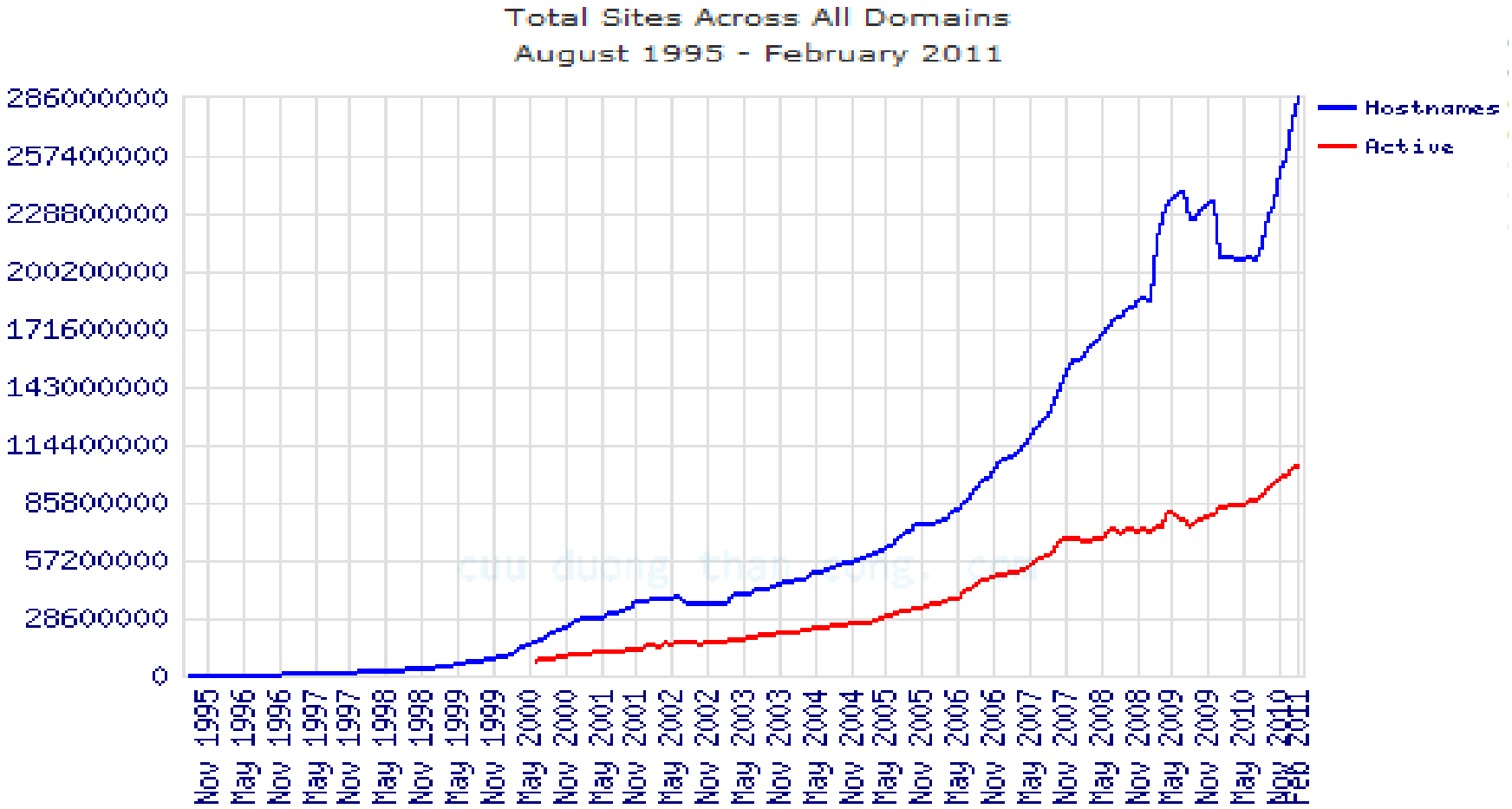


- Theo thống kê từ Google Scholar về số bài viết:
  - Với cụm từ “Text Mining”:
    - Ở tiêu đề: 2.800 bài (khoảng)
    - Ở mọi nơi: 33.000 bài (khoảng)
  - Với cụm từ “Text Analysis”:
    - Ở tiêu đề: 1.680 bài (khoảng)
    - Ở mọi nơi: 43.300 bài (khoảng)

- Nơi công bố tài liệu về Khai phá Text
  - Thường đi kèm với XLNNTN.
  - The ACL Anthology Network Corpus: http://aclweb.org/ anthology-new/. ACL: “The Association for Computational Linguistics is THE international scientific and professional society for people working on problems involving natural language and computation”.
  - DUC (Document Understanding Conferences: http://duc.nist.gov/ : 2001-2007) và TAC (Text Analysis Conferences: http://www.nist.gov/tac/about/index.html: 2008-nay)
  - Mọi hội nghị, tạp chí khoa học liên quan
  - Kdnuggets: http://www.kdnuggets.com/

2. Sự cần thiết của khai phá Web

- Web cũng rất gần gũi với con người
  - Tạo ra môi trường của xã hội ảo
  - Một phần quan trọng chuyển tải thông tin của loài người từ Web
  - Phương tiện chuyển giao tri thức
- Đặc thù của khai phá Text và Web
  - Web có bán cấu trúc
  - Kết nối không gian thời gian
  - Mở rộng giao lưu: diễn đàn, blog…
- Sự tăng trưởng của dữ liệu Web
  - Tương tự như dữ liệu Text
  - Dữ liệu đa phương tiện



- Hình minh họa sự tăng trưởng của Web
  - http://news.netcraft.com/archives/category/web-server-survey/ *(02/2011)*
- Khái niệm
  - Khai phá Web = Khai phá Text + WWW
  - Trích chọn mẫu mới, hữu ích, hiểu được, tiềm ẩn trong Web

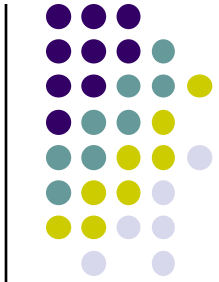
Các loại khai phá Web

- Khai phá nội dung Web
  - Khai phá nội dung trang web
  - Tối ưu hiệu quả trả về (hạng, phân cụm)
- Khai phá cấu trúc web: Độ liên quan + cách tổ chức và liên kết
- Khai phá sử dụng web
  - Phân tích các mẫu truy cập (General Access Pattern Tracking)
  - Phân tích các xu hướng cá nhân (Customizied Usage tracking)

Các chủ đề của khai phá Web

- Tìm kiếm và thu hồi: Thu hồi và tính hạng
- Phân tích đồ thị Web và Khai phá cấu trúc Web
- Phân cụm Web và Phân lớp Web
- Trích rút thông tin, Quảng cáo và tối ưu hóa Web
- Lọc cộng tác và lọc nội dung
- Phân tích web log và Khai phá sử dụng web
- Mạng xã hội trên Web
- Web ngữ nghĩa
- Khai phá quan điểm trên Web
- Các vấn đề về hệ thống Web

Một số đặc điểm của khai phá Web



- Web quá lớn để tổ chức thành kho dữ liệu
  - Tăng kích cỡ DW chậm hơn nhiều tốc độ phát triển Web
- Độ phức tạp của trang Web là rất lớn
  - Các kiểu tổ chức
  - Các kiểu dữ liệu
- Web: nguồn tài nguyên thông tin có độ thay đổi cao
  - Tăng nhiều và mất nhiều
- Web phục vụ một cộng đồng người rộng lớn và đa dạng
  - Phản ánh toàn bộ thế giới
- Chỉ phần rất nhỏ thông tin trên Web là thực sự hữu ích
  - Đối với toàn bộ và từng cá nhân
- Khai phá Web có lợi thế: bán cấu trúc, giàu thông tin (thẻ, liên kết, file log)

Nghiên cứu về khai khá Web

- Theo thống kê từ Google Scholar về số bài viết:
  - Với cụm từ “Web Mining”:
    - Ở tiêu đề: 2.680 bài (khoảng)
    - Ở mọi nơi: 20.000 bài (khoảng)
  - Với cụm từ “Text Analysis”:
    - Ở tiêu đề: 240 bài (khoảng)
    - Ở mọi nơi: 4.300 bài (khoảng)
  - Với cụm từ “Search Engine”:
    - Ở tiêu đề: 6.260 bài (khoảng)
    - Ở mọi nơi: 414.000 bài (khoảng)
  - Với cụm từ “Image Search”:
    - Ở tiêu đề: 890 bài (khoảng)
    - Ở mọi nơi: 15.800 bài (khoảng)
- Nơi công bố tài liệu về Khai phá Web
  - Đi kèm với XLNNTN và khai phá Text
  - Kdnuggets: http://www.kdnuggets.com/
  - Mọi hội nghị, tạp chí khoa học liên quan

BÀI GiẢNG KHAI PHÁ DỮ LIỆU WEB

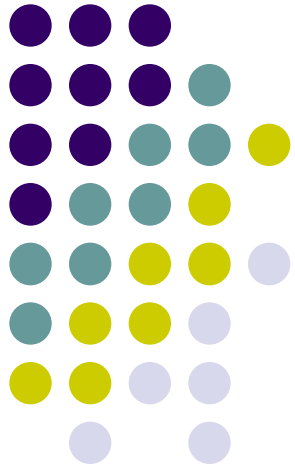
CHƯƠNG 2. KHAI PHÁ SỬ DỤNG WEB VÀ KHAI PHÁ CẤU TRÚC WEB

PGS. TS. HÀ QUANG THỤY

HÀ NỘI 10-2010

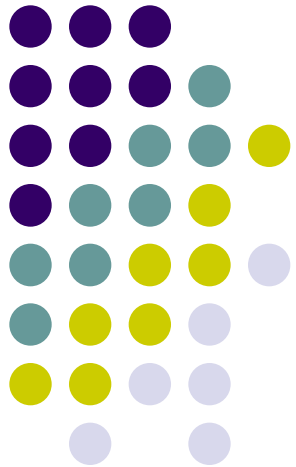
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

ĐẠI HỌC QUỐC GIA HÀ NỘI



Nội dung

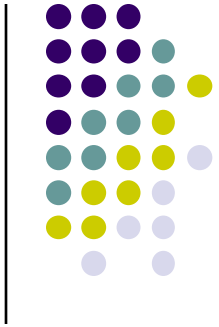
1. Khai phá sử dụng Web
2. Khai phá cấu trúc web



1. Khai phá sử dụng Web

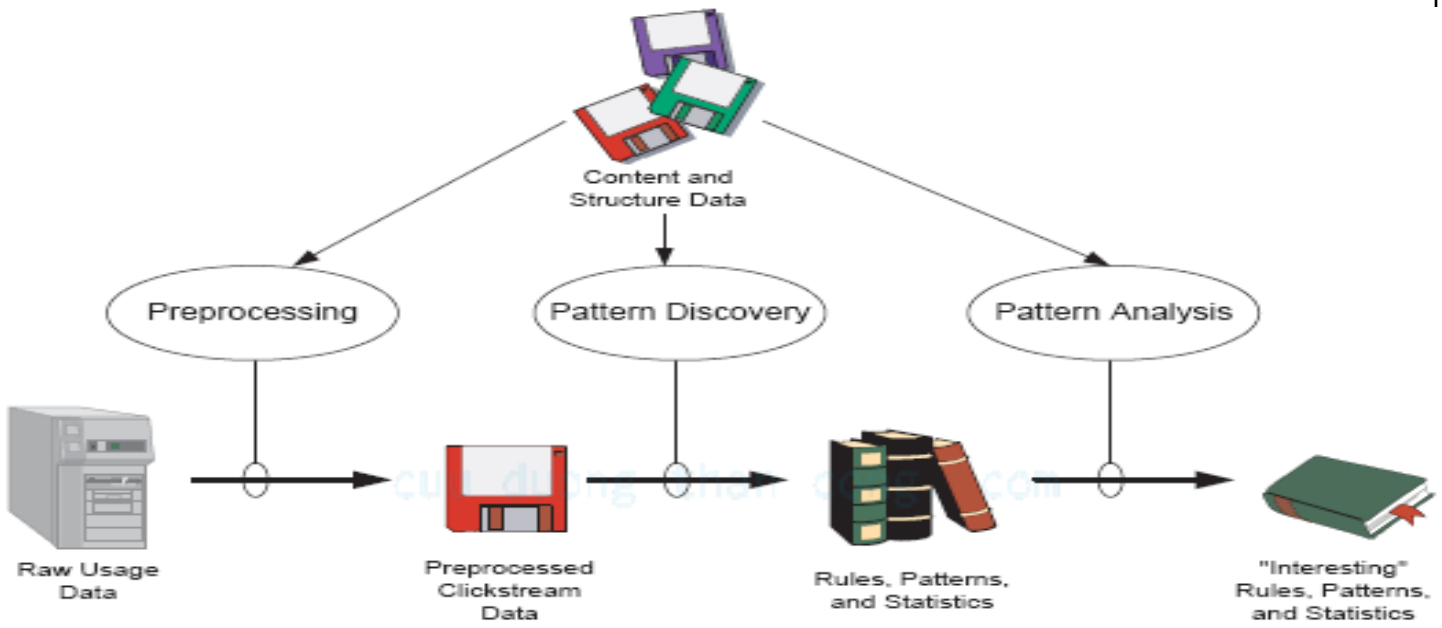
- Giới thiệu chung
- Phân tích mẫu truy nhập Web
  - Mang tính thói quen có tính cộng đồng
  - Khai phá mẫu truy nhập theo luật kết hợp
- Khai phá xu hướng sử dụng
  - Cá nhân hóa
  - Các hệ tư vấn

1.a. Giới thiệu chung



- Nguồn dữ liệu
  - Các logfile (máy chủ, máy khách, máy trung gian)
  - CSDL khách hàng
- Mô hình dữ liệu
  - *Thực thể*: người sử dụng, khung nhìn trang web, file trang Web, trình duyệt, phục vụ web, phục vụ nội dung, phiên người sử dụng, phiên phục vụ, dãy các sự kiện liên quan (episode).
- Tiền xử lý dữ liệu
  - Loại: cấu trúc, nội dung
  - Bài toán: xử lý văn bản, rút gọn đặc trưng, mô hình dữ liệu.
- Phát hiện mẫu
  - Mẫu quan hệ: thống kê, luật kết hợp, luật chuỗi, phân cụm, phân lớp, mô hình phụ thuộc
  - Đại chúng và cá nhân hóa

1.a. Một quy trình khai phá sử dụng Web

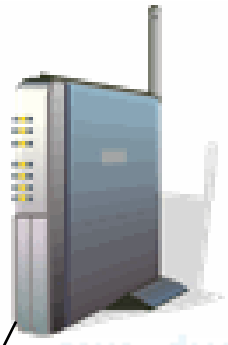


Quá trình khai phá sử dụng Web [Coo00]

- Input: Dữ liệu sử dụng Web
- Output: Các luật, mẫu, thống kê hấp dẫn
- Các bước chủ yếu:
  - Tiền xử lý dữ liệu
  - Khám phá mẫu
  - Phân tích mẫu

Sơ đồ ghi dữ liệu vào logfile

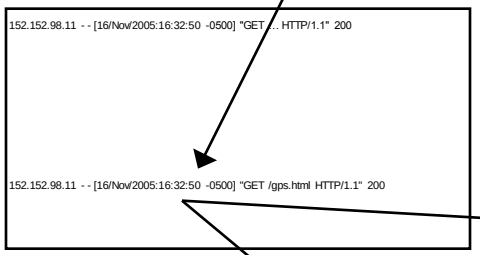
KDnuggets.com
Server



Page contents

http://www.kdnuggets.com/jobs

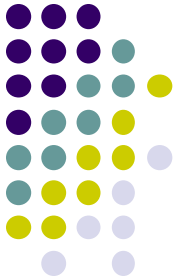
Web server log




```
152.152.98.11 - - [16/Nov/2005:16:32:50 -0500] "GET /jobs/ HTTP/1.1" 200  …
```

- Thông tin truy nhập người dùng
  - Server tổ chức ghi nhận vào logfile
  - Hỗ trợ quản lý điều hành
  - Tài nguyên Khai phá dữ liệu, nâng cao hiệu năng hệ thống

Một dòng ví dụ trong weblog



152.152.98.11 - - [16/Nov/2005:16:32:50 -0500] "GET /jobs/ HTTP/1.1" 200 15140 "http://www.google.com/search?q=salary+for+data+mining&hl=en&lr=&start=10&sa=N" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)“

152.152.98.11 Địa chỉ của hotname

- - Tên và login của người dùng từ xa: thường là “-”

[16/Nov/2005:16:32:50 -0500] Ngày và giờ truy nhập.

Giờ GMT: (+|-)HH00 US UST: -500

"GET /jobs/ HTTP/1.1" Phương thức lấy thông tin, URL liên quan tới tên miền; giao thức

200 Trạng thái **200 – OK (hầu hết, đạt được)** | 206 – truy nhập bộ phận – chuyển hướng vĩnh viến (truy nhập tới/ tiến trình định hướng lại /tiến trình/)| 302 – định hướng tạm thời| **304 – không thay đổi** | **404 – không thấy**|…

15140 Dung lượng tải về máy khách | “-” nếu trạng thái 304

"http://www.google.com/search?q=salary+for+data+mining&hl=en&lr=&start=10&sa=N" URL của người thăm (ở đây là từ Google)

"Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)“ đại lý của người dùng

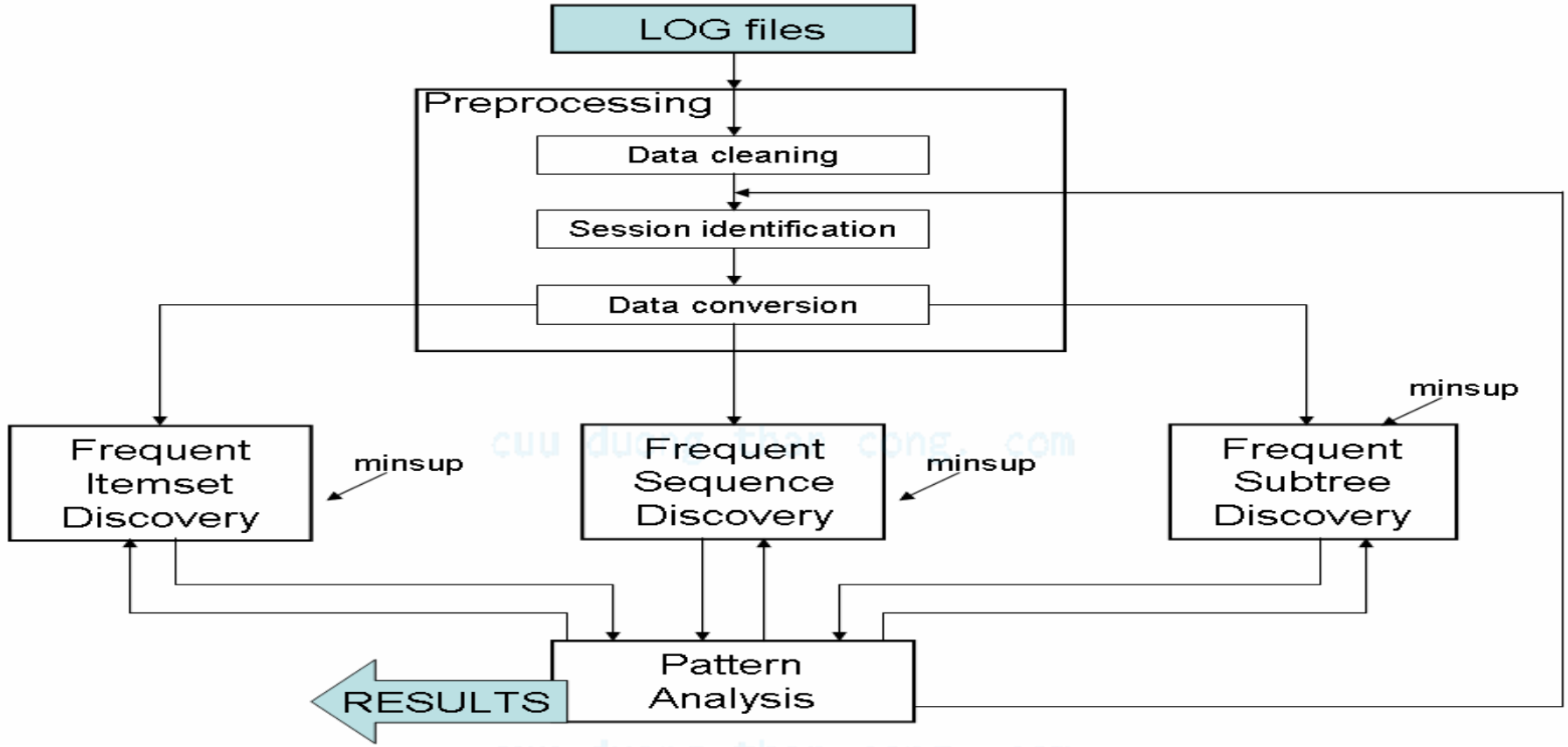
Một ví dụ về log files

AnonID	Query	QueryTime	Rank	ClickURL
479	family guy movie references	2006-03-03 22:37:46	1	http://www.familyguyfiles.com
479	top grossing movies of all time	2006-03-03 22:42:42	1	http://movieweb.com
479	top grossing movies of all time	2006-03-03 22:42:42	2	http://www.imdb.com
479	car decals	2006-03-03 23:20:12	4	http://www.decaljunky.com
479	car decals	2006-03-03 23:20:12	1	http://www.modernimage.net
479	car decals	2006-03-03 23:20:12	5	http://www.webdecal.com
479	car window decals	2006-03-03 23:24:05	9	http://www.customautotrim.com
479	car window sponsor decals	2006-03-03 23:27:17	3	http://www.streetglo.net
479	bose	2006-03-03 23:30:11	1	http://www.bose.com
479	bose car decal	2006-03-03 23:31:48	1	http://stickers.signprint.co.uk
479	bose car decal	2006-03-03 23:31:48	1	http://stickers.signprint.co.uk
479	bose car decal	2006-03-03 23:31:48	7	http://www.motorcitydecals.com
479	chicago the mix	2006-03-04 22:11:31	1	http://www.wtmx.com
479	chicago the drive	2006-03-04 22:14:51	2	http://www.wdrv.com

```
q         = cars
URL       = www.google.com/search?q=cars
IP        = 72.14.253.103
Cookie    = PREF=ID=03b1d4f329293203:LD=en:NR=10…
Browser   = Firefox/2.0.0.4;Windows NT 5.1
Time      = 25 Mar 2007 10:15:32
```

Một phần query log của AOL (*trên*) và Cấu trúc log của Google (*dưới*)

1.b. Phân tích mẫu truy nhập



- Phân tích mẫu từ logfile
  - Tìm tập mục phổ biến, dãy phổ biến, cây con phổ biến
  - Phân tích mẫu phổ biến tìm được

[IV06] Renáta Iváncsy, István Vajk (2006). Frequent Pattern Mining in Web Log Data, *Acta Polytechnica Hungarica*, **3**(1):77-90.

1.b. Ví dụ về mẫu phổ biến sử dụng Web

opinion & misc & travel	-->	on-air	90.26%
news & misc & business & bbs	-->	frontpage	90.24%
living & business & sports & bbs	-->	frontpage	90.00%
news & misc & business & sports	-->	frontpage	89.68%
news & tech & living & business & sports	-->	frontpage	89.00%
news & living & business & bbs	-->	frontpage	88.01%
frontpage & tech & living & business & sports	-->	news	87.87%
frontpage & opinion & living & sports	-->	news	87.81%
frontpage & tech & opinion & living	-->	news	87.60%
frontpage & tech & on-air & business & sports	-->	news	87.59%
news & misc & sports & bbs	-->	frontpage	87.56%
news & tech & on-air & business & sports	-->	frontpage	87.43%
news & living & business & sports	-->	frontpage	87.18%
news & business & sports & bbs	-->	frontpage	86.70%
misc & living & travel	-->	on-air	86.56%
tech & living & sports & bbs	-->	frontpage	86.52%
tech & business & sports & bbs	-->	frontpage	86.40%
news & misc & living & business	-->	frontpage	86.22%
on-air & business & sports & bbs	-->	frontpage	86.22%
news & tech & misc & bbs	-->	frontpage	86.18%
on-air & misc & business & sports	-->	frontpage	86.16%
tech & misc & travel	-->	on-air	86.09%
tech & living & business & sports	-->	frontpage	86.08%
news & living & sports & bbs	-->	frontpage	85.99%
misc & business & sports	-->	frontpage	85.79%
frontpage & tech & opinion & sports	-->	news	85.78%
news & opinion & living & sports	-->	frontpage	85.69%
misc & business & travel	-->	on-air	85.65%
news & tech & misc & business	-->	frontpage	85.63%
misc & business & bbs	-->	frontpage	85.57%
tech & living & sports & bbs	-->	news	85.49%
local & misc & business & sports	-->	frontpage	85.43%
news & opinion & business & bbs	-->	frontpage	85.32%
news & misc & living & sports	-->	frontpage	85.19%
news & on-air & business & sports	-->	frontpage	85.01%

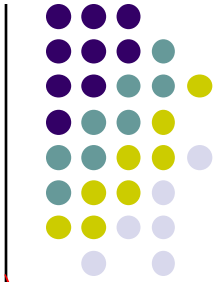
(a)

misc → local	2.07%
frontpage → frontpage → sports	2.02%
local → frontpage	1.83%
on-air → misc → on-air	1.72%
on-air → frontpage	1.69%
on-air → news	1.51%
news → frontpage → news	1.49%
local → news	1.46%
frontpage → frontpage → business	1.35%
news → sports	1.33%
news → bbs	1.23%
health → local	1.16%
misc → frontpage → frontpage	1.16%
on-air → local	1.15%
misc → on-air → misc	1.15%
frontpage → frontpage → living	1.14%
local → frontpage → frontpage	1.13%
health → misc	1.12%
misc → on-air → on-air	1.10%
local → misc → local	1.09%
misc → news	1.06%
news → living	1.06%
on-air → misc → on-air → misc	1.00%

(b)

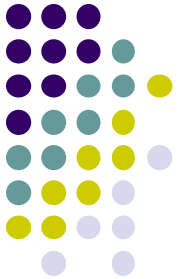
[IV06] Renáta Iváncsy, István Vajk (2006). Frequent Pattern Mining in Web Log Data, *Acta Polytechnica Hungarica*, 3(1):77-90, 2006

1.b. Ví dụ về mẫu kết hợp



- Một số ví dụ về *"luật kết hợp" (associate rule)*
  - "98% khách hàng *mà* mua tạp chí thể thao *thì đều* mua các tạp chí về ôtô" ⇨ sự ***kết hợp*** giữa *"tạp chí thể thao"* với *"tạp chí về ôtô"*
  - "60% khách hàng *mà* mua bia tại siêu thị *thì đều* mua bỉm trẻ em" ⇨ sự ***kết hợp*** giữa *"bia"* với *"bỉm trẻ em"*
  - "Có tới 70% người truy nhập Web *vào địa chỉ Url1* thì *cũng vào địa chỉ Url2* trong một phiên truy nhập web" ⇨ sự ***kết hợp*** giữa *"Url 1"* với *"Url 2"*. Khai phá dữ liệu sử dụng Web (lấy dữ liệu từ file log của các site, chẳng hạn được MS cung cấp). Các Url có gắn với nhãn "lớp" là các đặc trưng thì có luật kết hợp liên quan giữa các lớp Url này.
- Khái niệm cơ sở về luật kết hợp

Khai phá luật kết hợp: Cơ sở

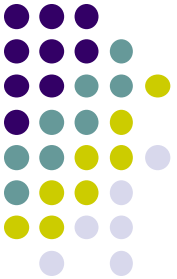


Cơ sở dữ liệu giao dịch (transaction database)

- Tập toàn bộ các mục I = {i_1, i_2, …, i_k}: "tất cả các mặt hàng".
- *Giao dịch*: danh sách các mặt hàng (mục: item) trong một phiếu mua hàng của khách hàng. Giao dịch T là một tập mục.

  Một giao dịch T là một tập con của I: T I. Mỗi giao dịch T có một định danh là T_{ID}.
- A là một tập mục A I và T là một giao dịch: Gọi T chứa A nếu A T.

Khai phá luật kết hợp: cơ sở



Luật kết hợp

- Gọi A B là một "luật kết hợp" nếu A I, B I và A B= .
- Luật kết hợp A B có độ hỗ trợ (support) *s* trong CSDL giao dịch D nếu trong D có s% các giao dịch T chứa AB: chính là xác suất P(AB).

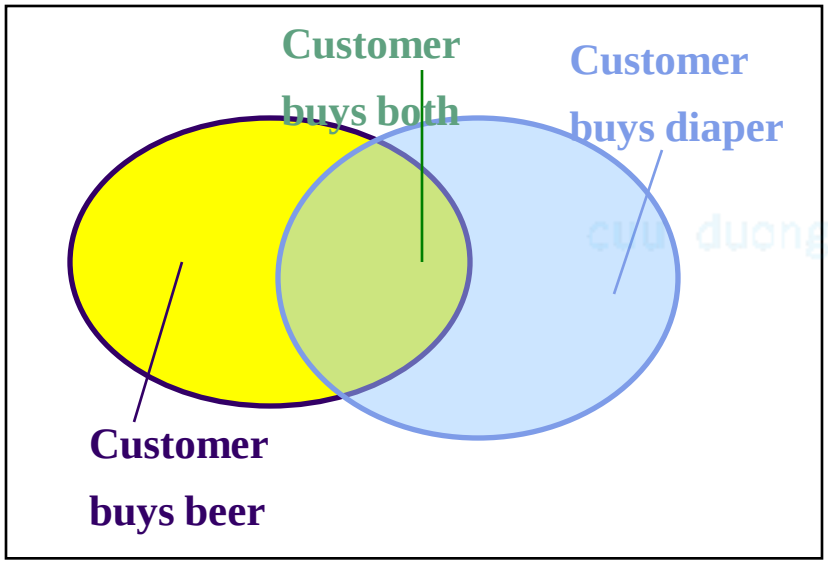
  Tập mục A có P(A) s>0 (với s cho trước) được gọi là *tập phổ biến (frequent set)*.
- Luật kết hợp A B có độ tin cậy (confidence) *c* trong CSDL D nếu như trong D có c% các giao dịch T chứa A thì cũng chứa B: chính là xác suất P(B|A).

  Support (A B) = P(A B) : 1 s (A B) 0

  Confidence (A B) = P(B|A) : 1 c (A B) 0
- Luật A B được gọi là đảm bảo độ hỗ trợ s trong D nếu s(A B) s. Luật A B được gọi là đảm bảo độ tin cậy c trong D nếu c(A B) c. Tập mạnh.

Ví dụ: Mẫu phổ biến và luật kết hợp

Transaction-id	Items bought
10	A, B, C
20	A, C
30	A, D
40	B, E, F



- Tập mục I={i_1, …, i_k}. CSDL giao dịch D = {d I}
- A, B I, A B= : $A \rightarrow B$ là luật kết hợp
- Bài toán tìm luật kết hợp.

  Cho trước độ hỗ trợ tối thiểu s>0, độ tin cậy tối thiểu c>0. Hãy tìm mọi luật kết hợp mạnh $X \rightarrow Y$.

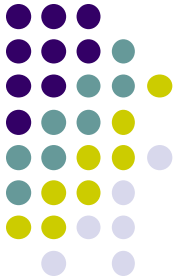
  Giả sử min_support = 50%, min_conf = 50%:

  $A \rightarrow C$ (50%, 66.7%)

  $C \rightarrow A$ (50%, 100%)

- Hãy trình bày các nhận xét về khái niệm luật kết hợp với khái niệm phụ thuộc hàm.
- Các tính chất Armstrong ở đây.

Một ví dụ tìm luật kết hợp



Transaction-id	Items bought
10	A, B, C
20	A, C
30	A, D
40	B, E, F

Min. support 50%

Min. confidence 50%

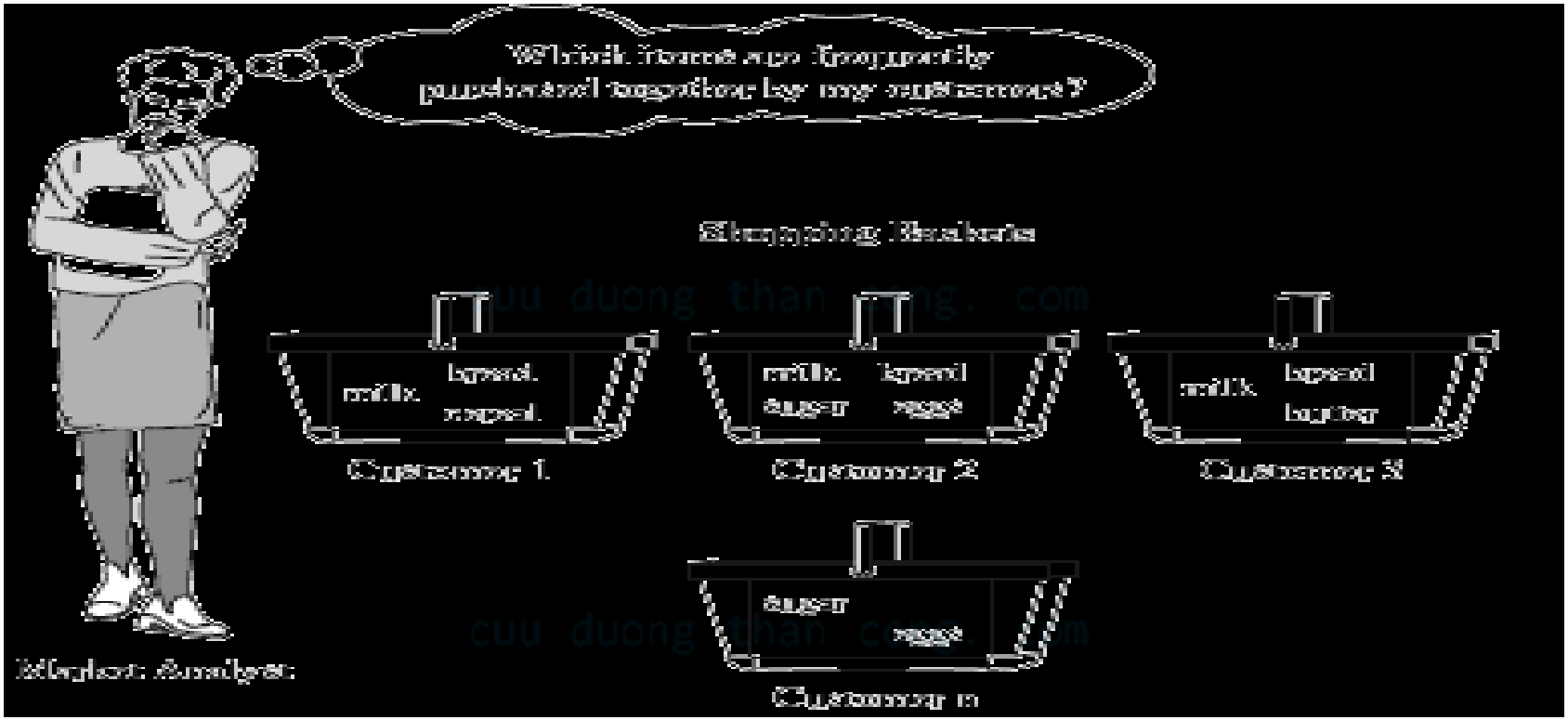
Frequent pattern	Support
{A}	75%
{B}	50%
{C}	50%
{A, C}	50%

For rule *A* *C*:

support = support({*A*} {*C*}) = 50%

confidence = support({*A*} {*C*})/support({*A*}) = 66.6%

Khai niệm khai phá kết hợp

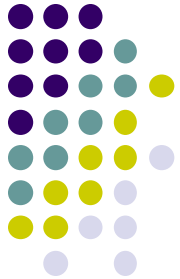


$$computer \Rightarrow antivirus\_software \ [support = 2\%, confidence = 60\%]$$

Khai phá luật kết hợp

- Khai phá luật kết hợp:
  - Tìm tất cả mẫu phổ biến, kết hợp, tương quan, hoặc cấu trú nhan-quả trong tập các mục hoặc đối tượng trong CSDL quan hệ hoặc các kho chứa thông tin khác.
  - Mẫu phổ biến (Frequent pattern): là mẫu (tập mục, dãy mục…) mà xuất hiện phổ biến trong 1 CSDL [AIS93]
- Động lực: tìm mẫu chính quy (regularities pattern) trong DL
  - Các mặt hàng nào được mua cùng nhau? — Bia và bỉm (diapers)?!
  - Mặt hàng nào sẽ được mua sau khi mua một PC ?
  - Kiểu DNA nào nhạy cảm với thuộc mới này?
  - Có khả năng tự động phân lớp Web hay không ?

Mẫu phổ biến và khai phá luật kết hợp là một bài toán bản chất của khai phá DL



- Nền tảng của nhiều bài toán KPDL bản chất
  - Kết hợp, tương quan, nhân quả
  - Mẫu tuần tự, kết hợp thời gian hoặc vòng, chu kỳ bộ phận, kết hợp không gian và đa phương tiện
  - Phân lớp kết hợp, phân tích cụm, khối tảng băng, tích tụ (nén dữ liệu ngữ nghĩa)
- Ứng dụng rộng rãi
  - Phân tích DL bóng rổ, tiếp thị chéo (cross-marketing), thiết kế catalog, phân tích chiến dịch bán hàng
  - Phân tích Web log (click stream), Phân tích chuỗi DNA v.v.

Apriori: Một tiếp cận sinh ứng viên và kiểm tra

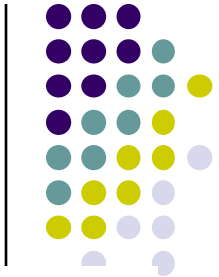
- Khái quát: Khai phá luật kết hợp gồm hai bước:
  - Tìm mọi tập mục phổ biến: theo min-sup
  - Sinh luật mạnh từ tập mục phổ biến
- Mọi tập con của tập mục phổ biến cũng là tập mục phổ biến
  - Nếu {*bia*, *bỉm*, *hạnh nhân*} là phổ biến thì {*bia*, *bỉm*} cũng vậy: Mọi giao dịch chứa {*bia*, *bỉm*, *hạnh nhân*} cũng chứa {*bia*, *bỉm*}.
- Nguyên lý tỉa Apriori: Với mọi tập mục không phổ biến thì mọi tập bao không cần phải sinh ra/kiểm tra!
- Phương pháp:
  - Sinh các tập mục ứng viên dài (k+1) từ các tập mục phổ biến có độ dài k (Độ dài tập mục là số phần tử của nó),
  - Kiểm tra các tập ứng viên theo CSDL
- Các nghiên cứu hiệu năng chứng tỏ tính hiệu quả và khả năng mở rộng của thuật toán
- Agrawal & Srikant 1994, Mannila, và cộng sự 1994

Thuật toán Apriori

- Trên cơ sở tính chất (nguyên lý tỉa) Apriori, thuật toán hoạt động theo quy tắc quy hoạch động
  - Từ các tập $F_i = \{c_i | \ c_i$ tập phổ biến, $|c_i| = i\}$ gồm mọi tập mục phổ biến có độ dài i với 1 i k,
  - đi tìm tập F_{k+1} gồm mọi tập mục phổ biến có độ dài k+1.
- Trong thuật toán:

  các tên mục $i_1, i_2, \ldots i_n$ (n = |I|) *được sắp xếp theo một thứ tự cố định*: thường được đánh chỉ số 1, 2, ..., n.

Thuật toán Apriori



Thuật toán Apriori [WKQ08]:

Input: - Cơ sở dữ liệu giao dịch D = {t| t giao dịch}

- Độ hỗ trợ tối thiểu minsup > 0

Output: - Tập hợp tất cả các tập phổ biến.

0: mincount = minsup * |D|;

1. F_1 = {các tập phổ biến có độ dài 1}

2. for (k=1; $F_k \neq \emptyset$; k++) do begin

3. C_{k+1} = apriori-gen (F_k); // sinh mọi ứng viên độ dài k+1

4. for t $\in$ D do begin

5. C_t = {c $\in$ C_{k+1} | c $\subseteq$ t}; //mọi ứng viên chứa trong t

6. for c $\in$ C_t do

7. c.count ++;

8. end

9. F_{k+1} = {c $\in$ C_{k+1} | c.count $\geq$ mincount} ;

10. end

11. Answer $\cup_k F_k$;

Thuật toán: Thủ tục con Apriori-gen

Trong mỗi bước k, thuật toán Apriori đều phải duyệt CSDL D.

Khởi động, duyệt D để có được F_1.

Các bước k sau đó, duyệt D để tính số lượng giao dịch t thoả từng ứng viên c của C_{k+1}: mỗi giao dịch t chỉ xem xét một lần cho mọi ứng viên c thuộc C_{k+1}.

Thủ tục con Apriori-gen sinh tập phổ biến: *tư tưởng*

Bước nối: Sinh các tập mục R_{k+1} là ứng viên tập phổ biến có độ dài k+1 bằng cách kết hợp hai tập phổ biến P_k và Q_k có độ dài k và trùng nhau ở k-1 mục đầu tiên:

$$R_{k+1} = P_k \cup Q_k = \{i_1, i_2, \ldots, i_{k-1}, i_k, i_{k'}\} \text{ với}$$

$$P_k = \{i_1, i_2, \ldots, i_{k-1}, i_k\} \text{ và } Q_k = \{i_1, i_2, \ldots, i_{k-1}, i_{k'}\}$$

trong đó $i_1 \le i_2 \le \ldots \le i_{k-1} \le i_k \le i_{k'}$.

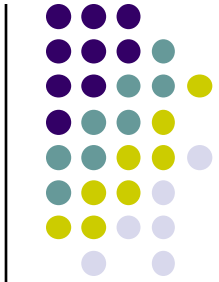
Bước tỉa: Giữ lại tất cả các R_{k+1} thỏa tính chất Apriori ($\forall X \subseteq R_{k+1}$ và $|X|=k \Rightarrow X \in F_k$), nghĩa là đã loại (tia) bớt đi mọi ứng viên R_{k+1} không đáp ứng tính chất này.

Thủ tục con Apriori-gen

(1) for mọi tập mục phổ biến $l_1 \in L_k$
(2) for mọi tập mục phổ biến $l_2 \in L_k$
(3) if $(l_1[1]=l_2[1]) \wedge (l_1[2]=l_2[2]) \wedge \ldots \wedge (l_1[k-1]=l_2[k-1]) \wedge (l_1[k]<l_2[k])$
then {
$c = l_1 \Leftrightarrow l_2$; // join step: generate candidates
// c = $\{l_1[1], l_1[2], \ldots, l_1[k-1], l_1[k], l_2[k]\}$
(5) if has_infrequent_subset(c, L_k) then
(6) delete c; // bước tỉa: bỏ ứng viên không đúng
else add c to C_{k+1};
(8) }
(9) return Ck;

procedure has_infrequent_subset(c: tập ứng viên độ dài $k+1$;
L_k: tập các tập mục phổ biến độ dài k); // tri thức đã có
(1) for mỗi tập con s độ dài k *của* c
(2) if $s \notin L_k$ then
(3) return TRUE;
(4) return FALSE;

Một ví dụ thuật toán Apriori (s=0.5)



Database TDB

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

C_1

1st scan

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

L_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

C_2

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

2nd scan

C_2

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

L_2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

C_3

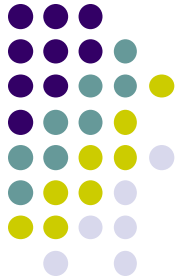
Itemset
{B, C, E}

3rd scan

L_3

Itemset	sup
{B, C, E}	2

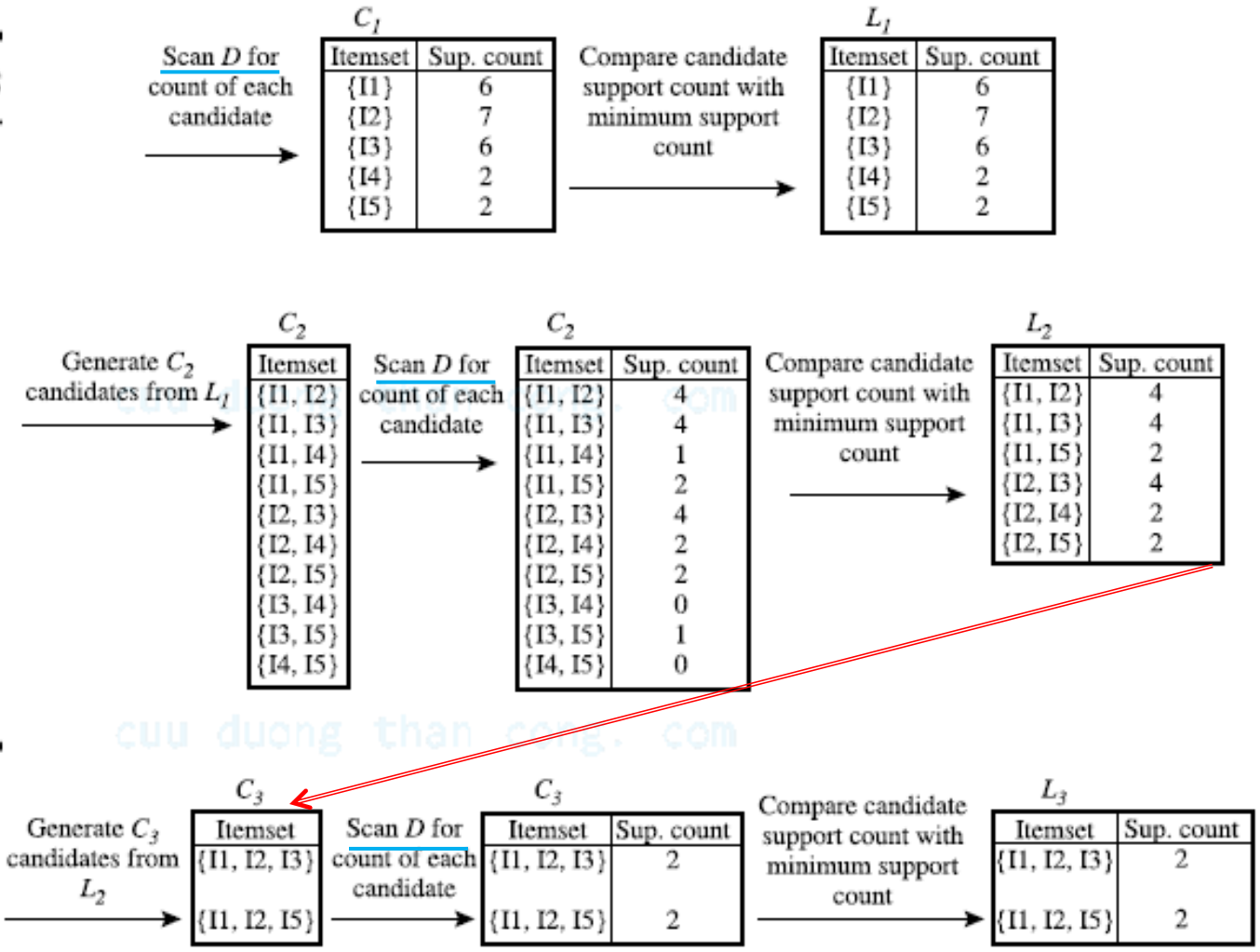
Chi tiết quan trọng của Apriori



- Cách thức sinh các ứng viên:
  - Bước 1: Tự kết nối L_k
  - Step 2: Cắt tỉa
- Cách thức đếm hỗ trợ cho mỗi ứng viên.
- Ví dụ thủ tục con sinh ứng viên
  - L_3={*abc, abd, acd, ace, bcd*}
  - Tự kết nối: L_3*L_3
    - *abcd* từ *abc* và *abd*
    - *acde* từ *acd* và *ace*
  - Tỉa:
    - *acde* là bỏ đi vì *ade* không thuộc L_3
  - C_4={*abcd*}

Ví dụ: D, min_sup*|D| = 2 (C_4 =)

TID	List of item_IDs
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3



C_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

C_2

Itemset
{I1, I2}
{I1, I3}
{I1, I4}
{I1, I5}
{I2, I3}
{I2, I4}
{I2, I5}
{I3, I4}
{I3, I5}
{I4, I5}

C_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I4}	1
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2
{I3, I4}	0
{I3, I5}	1
{I4, I5}	0

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

C_3

Itemset
{I1, I2, I3}
{I1, I2, I5}

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Sinh luật kết hợp

TID	List of item_IDs
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Việc sinh luật kết hợp gồm hai bước

- Với mỗi tập phổ biến W tìm được hãy sinh ra mọi tập con thực sự X khác rỗng của nó.
- Với mỗi tập phổ biến W và tập con X khác rỗng thực sự của nó: sinh luật X (W – X) nếu P(W-X|X) c.

Như ví dụ đã nêu có L3 = {{I1, I2, I3}, {I1, I2, I5}}

Với độ tin cậy tối thiểu 70%, xét tập mục phổ biến {I1, I2, I5} có 3 luật như dưới đây: Duyệt CSDL ?

$$I1 \wedge I2 \Rightarrow I5, \quad confidence = 2/4 = 50\%$$
$$I1 \wedge I5 \Rightarrow I2, \quad confidence = 2/2 = 100\%$$
$$I2 \wedge I5 \Rightarrow I1, \quad confidence = 2/2 = 100\%$$
$$I1 \Rightarrow I2 \wedge I5, \quad confidence = 2/6 = 33\%$$
$$I2 \Rightarrow I1 \wedge I5, \quad confidence = 2/7 = 29\%$$
$$I5 \Rightarrow I1 \wedge I2, \quad confidence = 2/2 = 100\%$$

1.b. Luật kết hợp và luật dãy sử dụng Web

opinion & misc & travel	-->	on-air	90.26%
news & misc & business & bbs	-->	frontpage	90.24%
living & business & sports & bbs	-->	frontpage	90.00%
news & misc & business & sports	-->	frontpage	89.68%
news & tech & living & business & sports	-->	frontpage	89.00%
news & living & business & bbs	-->	frontpage	88.01%
frontpage & tech & living & business & sports	-->	news	87.87%
frontpage & opinion & living & sports	-->	news	87.81%
frontpage & tech & opinion & living	-->	news	87.80%
frontpage & tech & on-air & business & sports	-->	news	87.59%
news & misc & sports & bbs	-->	frontpage	87.56%
news & tech & on-air & business & sports	-->	frontpage	87.43%
news & living & business & sports	-->	frontpage	87.18%
news & business & sports & bbs	-->	frontpage	86.70%
misc & living & travel	-->	on-air	86.55%
tech & living & sports & bbs	-->	frontpage	86.52%
tech & business & sports & bbs	-->	frontpage	86.40%
news & misc & living & business	-->	frontpage	86.22%
on-air & business & sports & bbs	-->	frontpage	86.22%
news & tech & misc & bbs	-->	frontpage	86.18%
on-air & misc & business & sports	-->	frontpage	86.16%
tech & misc & travel	-->	on-air	86.09%
tech & living & business & sports	-->	frontpage	86.08%
news & living & sports & bbs	-->	frontpage	85.99%
misc & business & sports	-->	frontpage	85.79%
frontpage & tech & opinion & sports	-->	news	85.78%
news & opinion & living & sports	-->	frontpage	85.69%
misc & business & travel	-->	on-air	85.65%
news & tech & misc & business	-->	frontpage	85.63%
misc & business & bbs	-->	frontpage	85.57%
tech & living & sports & bbs	-->	news	85.49%
local & misc & business & sports	-->	frontpage	85.43%
news & opinion & business & bbs	-->	frontpage	85.32%
news & misc & living & sports	-->	frontpage	85.19%
news & on-air & business & sports	-->	frontpage	85.01%

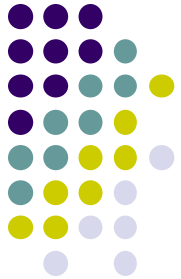
(a)

misc → local	2.07%
frontpage → frontpage → sports	2.02%
local → frontpage	1.83%
on-air → misc → on-air	1.72%
on-air → frontpage	1.69%
on-air → news	1.51%
news → frontpage → news	1.49%
local → news	1.46%
frontpage → frontpage → business	1.35%
news → sports	1.33%
news → bbs	1.23%
health → local	1.16%
misc → frontpage → frontpage	1.16%
on-air → local	1.15%
misc → on-air → misc	1.15%
frontpage → frontpage → living	1.14%
local → frontpage → frontpage	1.13%
health → misc	1.12%
misc → on-air → on-air	1.10%
local → misc → local	1.09%
misc → news	1.06%
news → living	1.06%
on-air → misc → on-air → misc	1.00%

(b)

- Các loại mẫu điển hình: xu hướng chung của mọi người
  - Luật kết hợp
  - Luật dãy
  - Cây con phổ biến

1.c. Nghiên cứu về luật kết hợp

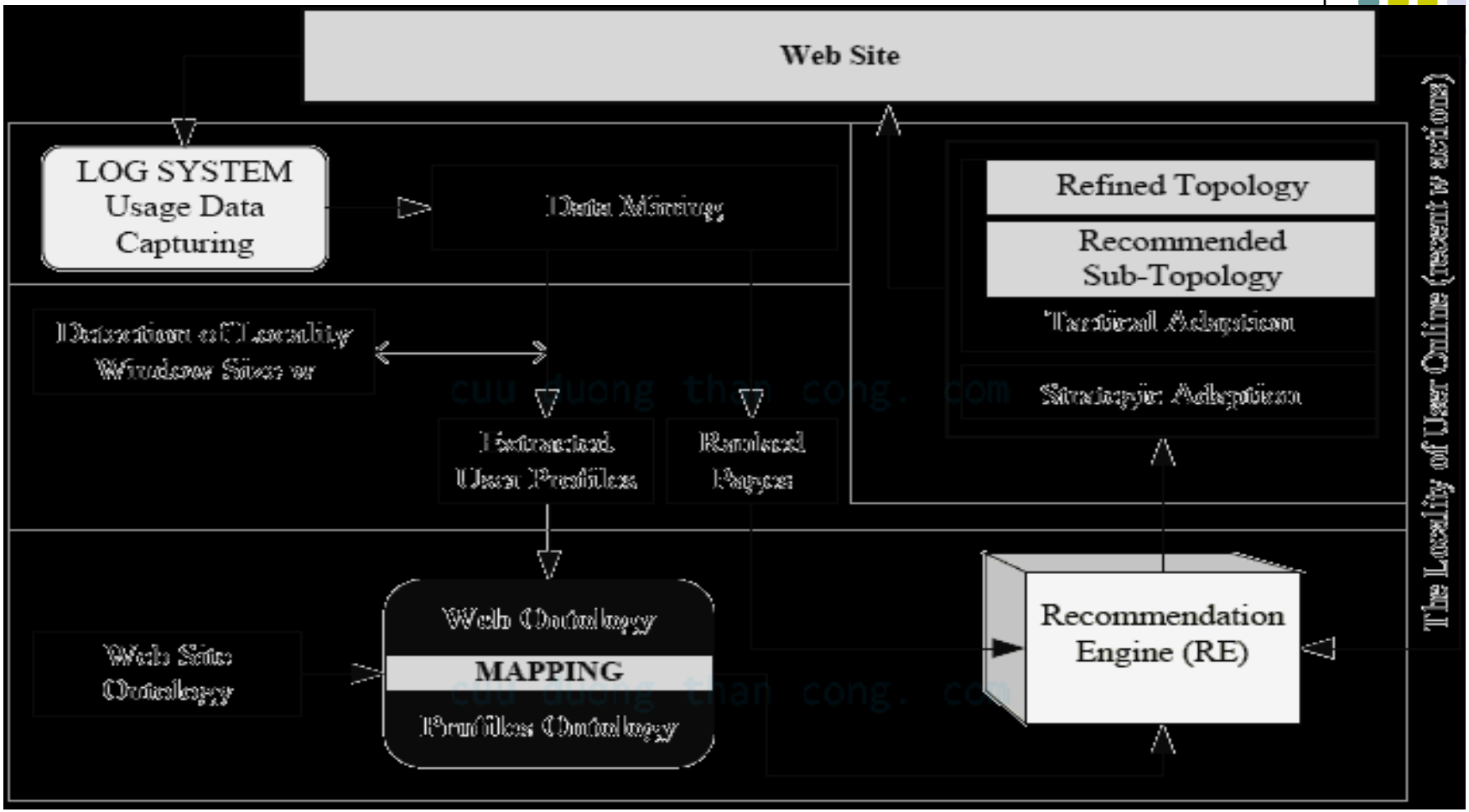


- Thống kê từ Google Scholar về số bài viết:
  - Với cụm từ “Association Rule”:
    - Ở tiêu đề: 2.060 bài (khoảng)
      1.000 bài (2006 – nay)
    - Ở mọi nơi: 27.400 bài (khoảng)
  - Với cụm từ “Apriori Algorithm”:
    - Ở tiêu đề: 350 bài (khoảng)
      219 bài (2006 – nay)
    - Ở mọi nơi: 8.820 bài (khoảng)
  - Với cụm từ “Sequential Pattern”:
    - Ở tiêu đề: 590 bài (khoảng)
      270 bài (2006 – nay)
    - Ở mọi nơi: 15.700 bài (khoảng)

1.c. Khai phá xu hướng cá nhân

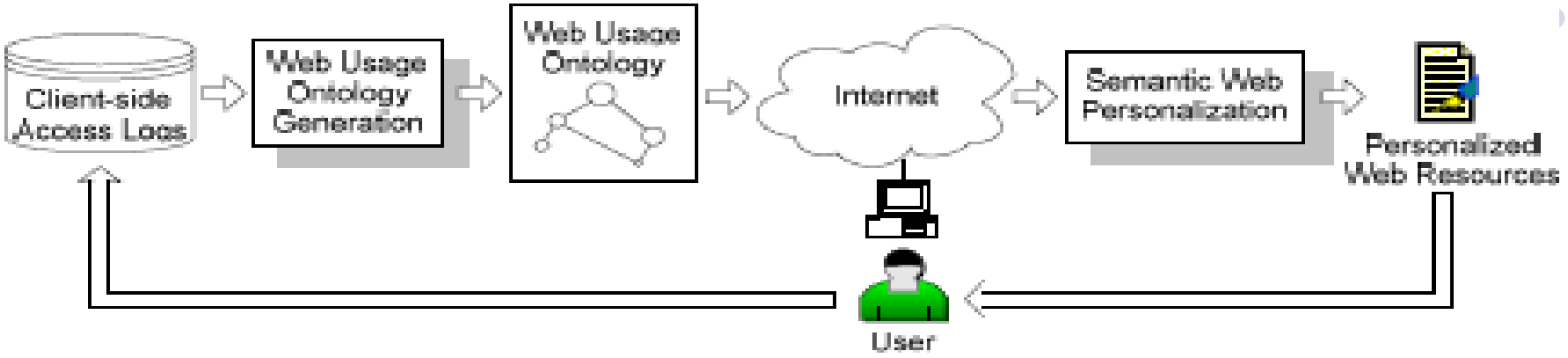
- Giới thiệu
  - “Cá nhân hóa”: Thông tin cá nhân và tư vấn cá nhân hóa
  - Thông tin cá nhân: CSDL quản lý; Máy khách
  - Ngữ cảnh làm việc của cá nhân
- Một số hình thức
  - Khai phá xu hướng cá nhân từ thông tin máy khách
  - Hệ tư vấn
- Hệ tư vấn
  - Recommendation Systems
  - Lọc cộng tác, lọc nội dung, lọc kết hợp
  - Hội thảo dành riêng: các năm 2007, 2009, 2010
  - http://recsys.acm.org/2010/

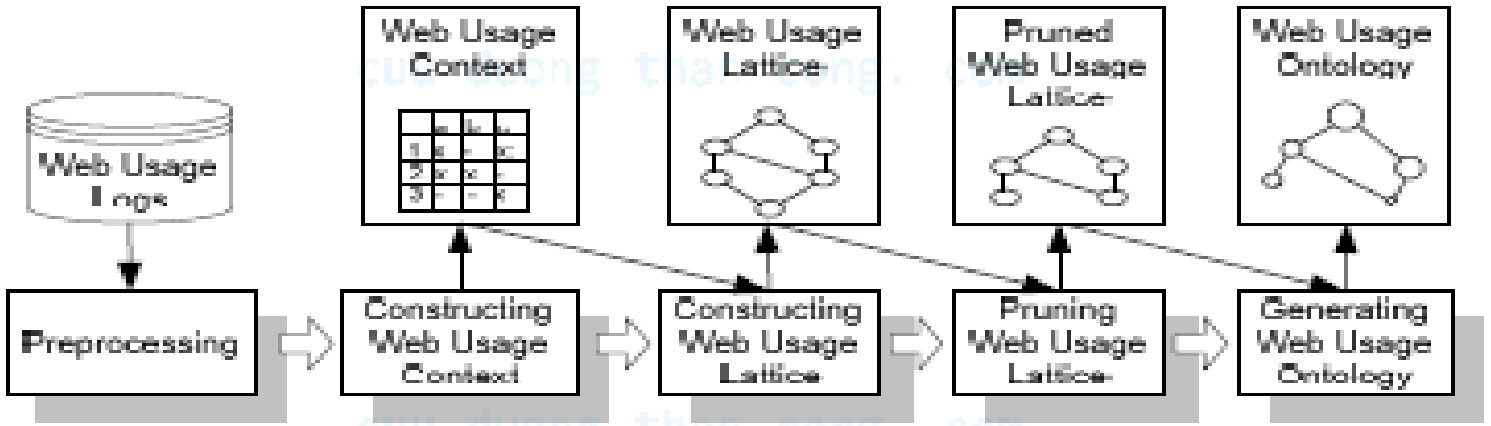
1.c. Sinh tư vấn dựa theo tiểu sử người dùng



[RK07] Tarmo Robal, Ahto Kalja (2007). Applying User Profile Ontology for Mining Web Site Adaptation Recommendations, *ADBIS Research Communications 2007*

1.c. Khai phá sử dụng Web



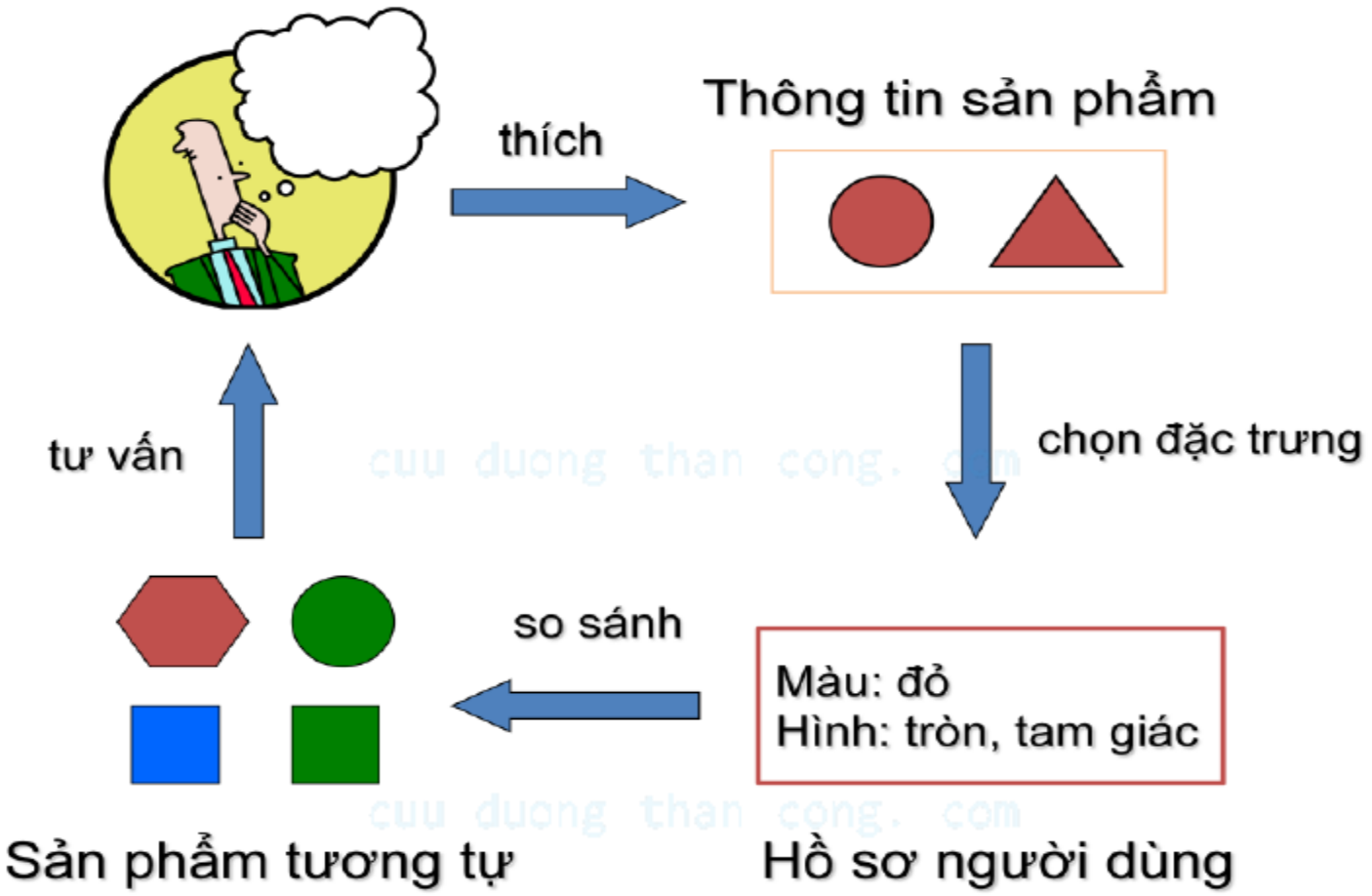


Hệ thống khai phá sử dụng Web tư vấn hướng cá nhân

- *Kiến trúc hệ thống (trên)*
- *và sinh ontology sử dụng Web (dưới)*

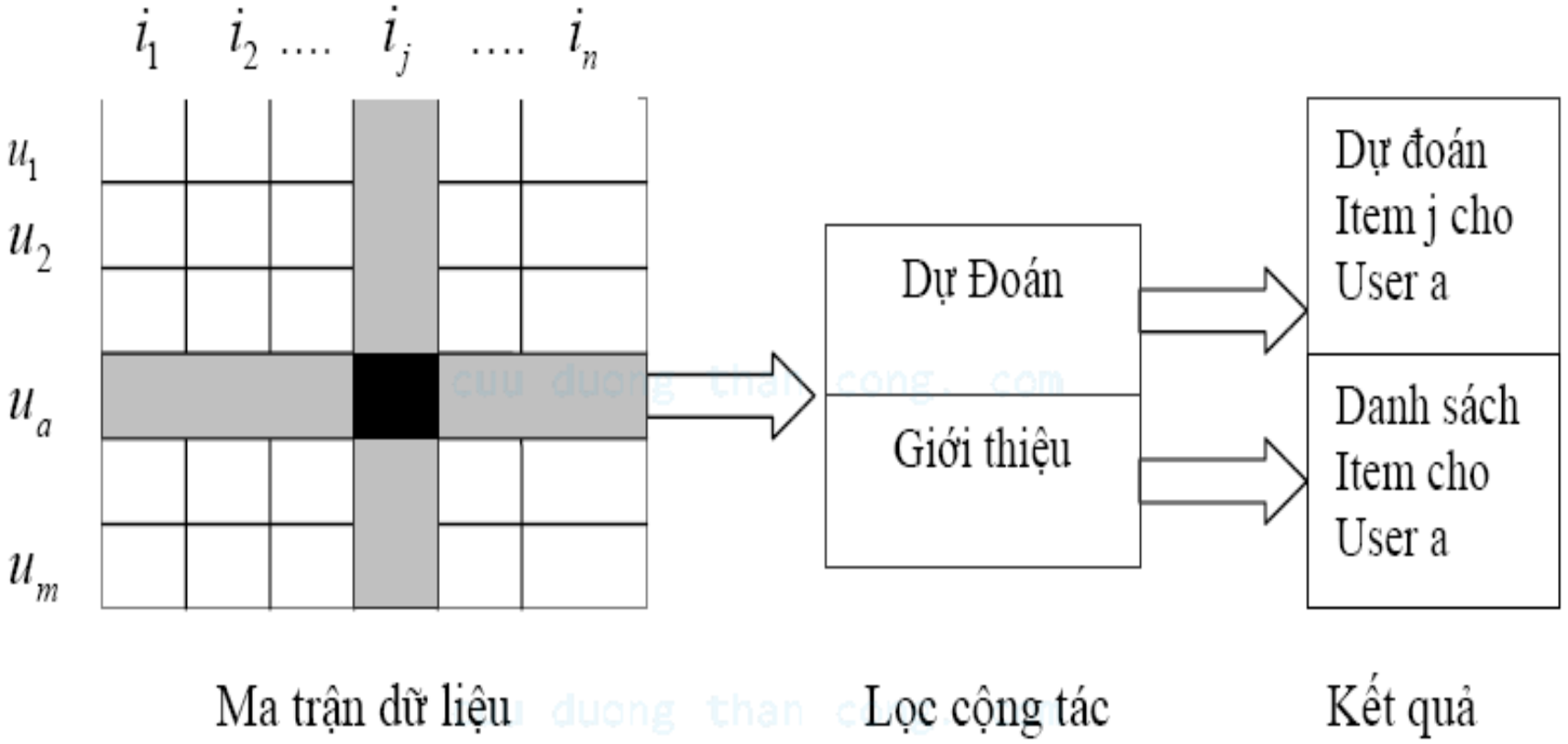
Baoyao Zhou, Siu Cheung Hui, Alvis C. M. Fong (2005). Web Usage Mining for Semantic Web Personalization, *Workshop on Personalization on the Semantic Web*, 66–72, Edinburgh, UK, 2005.

1.c. Hệ thống tư vấn: lọc nội dung



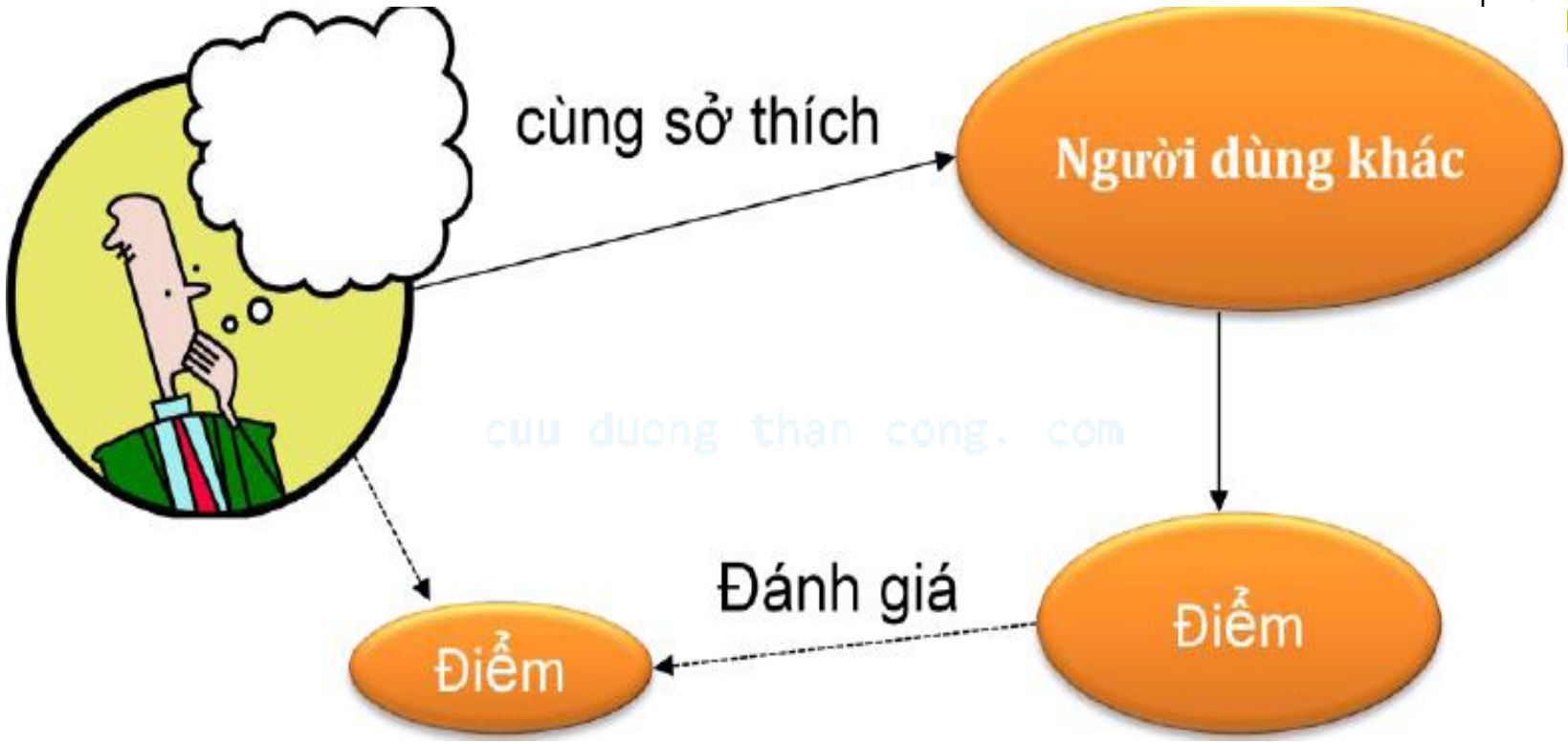
Lấy nội dung thuộc tính các sản phẩm người dùng đã ưa thích để dự đoán sản phẩm ưa thích tiếp theo

1.c. Hệ thống tư vấn: lọc cộng tác

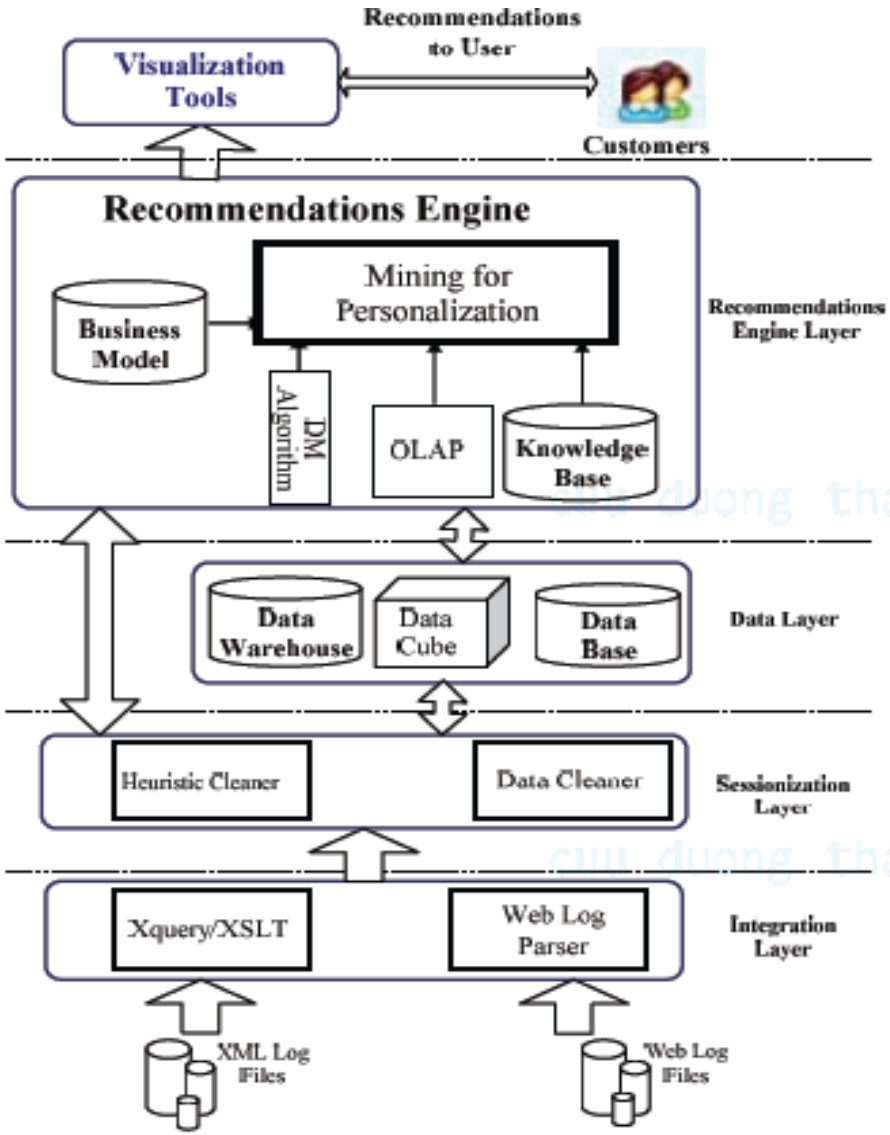


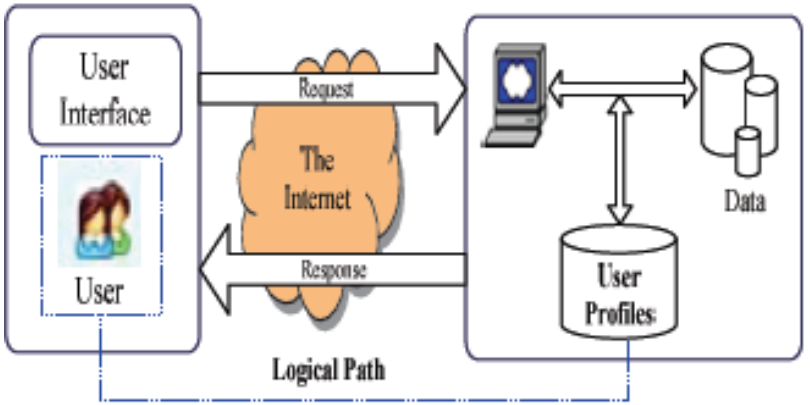
Quan hệ người dùng – sản phẩm: nhóm người dùng "tương tự nhau" và khi có một người dùng trong "thích" thì các người khác cũng "thích" tương tự

1.c. Hệ thống tư vấn: lọc cộng tác



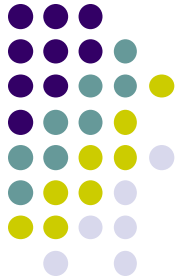
1.c. Hệ thống tư vấn: lọc cộng tác





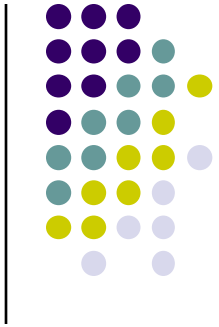
Jinhua Sun, Yanqi Xie (2009). A Web Data Mining Framework for E-commerce Recommender Systems, *Computational Intelligence and Software Engineering, 2009. CiSE 2009.*

Nghiên cứu về khai khá sử dụng Web



- Thống kê từ Google Scholar về số bài viết:
  - Với cụm từ “Web Usage Mining”:
    - Ở tiêu đề: 860 bài (khoảng)
      280 bài (2006 – nay)
    - Ở mọi nơi: 171.000 bài (khoảng)
  - Với cụm từ “Web Log Mining”:
    - Ở tiêu đề: 340 bài (khoảng)
      140 bài (2006 – nay)
    - Ở mọi nơi: 137.000 bài (khoảng)
  - Với cụm từ “Recommendation System”:
    - Ở tiêu đề: 1.750 bài (khoảng)
      750 bài (2006 – nay)
    - Ở mọi nơi: 1.760.000 bài (khoảng)

2. Khai phá cấu trúc Web



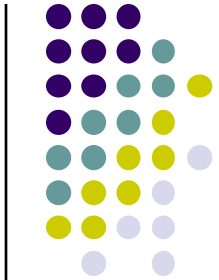
- Hai bài toán điển hình
  - Khai phá liên kết Web
  - Khai phá cấu trúc trang Web
- Khai phá liên kết Web
  - Mỗi trang Web là một đỉnh
  - Liên kết các trang Web hình thành các cung
  - Đồ thị có hướng hoặc vô hướng
  - Web phản ánh xã hội: đồ thị Web là một loại mạng xã hội
  - Hạng trang Web, một bài toán điển hình: tính "độ quan trọng" của một trang Web (một nút trên đồ thị Web)
  - Khai phá liên kết Web: Phân lớp trang web dựa theo liên kết, Phân tích cụm dựa theo liên kết, Kiểu liên kết; Độ mạnh liên kết;

2. Khai phá liên kết Web

- Phân lớp Web dựa theo liên kết
  - Khai thác thông tin liên kết cho phân lớp Web
- Phân cụm Web dựa theo liên kết
  - Tìm ra sự xuất hiện tự nhiên các lớp con: dữ liệu là liên kết
- Phân tích kiểu liên kết
  - Dự báo về sự tồn tại của liên kết
  - Dự báo mục đích của liên kết
- Phân tích độ mạnh liên kết
  - Độ mạnh của cung và đỉnh (hạng trang)
- Phân tích số lượng liên kết
  - Dự báo số lượng liên kết giữa các đối tượng

Miguel Gomes da Costa Júnior, Zhiguo Gong (2006). Web Structure Mining: An Introduction, *the 2005 IEEE International Conference on Information Acquisition*: 590-595

2. Khai phá cấu trúc trang Web

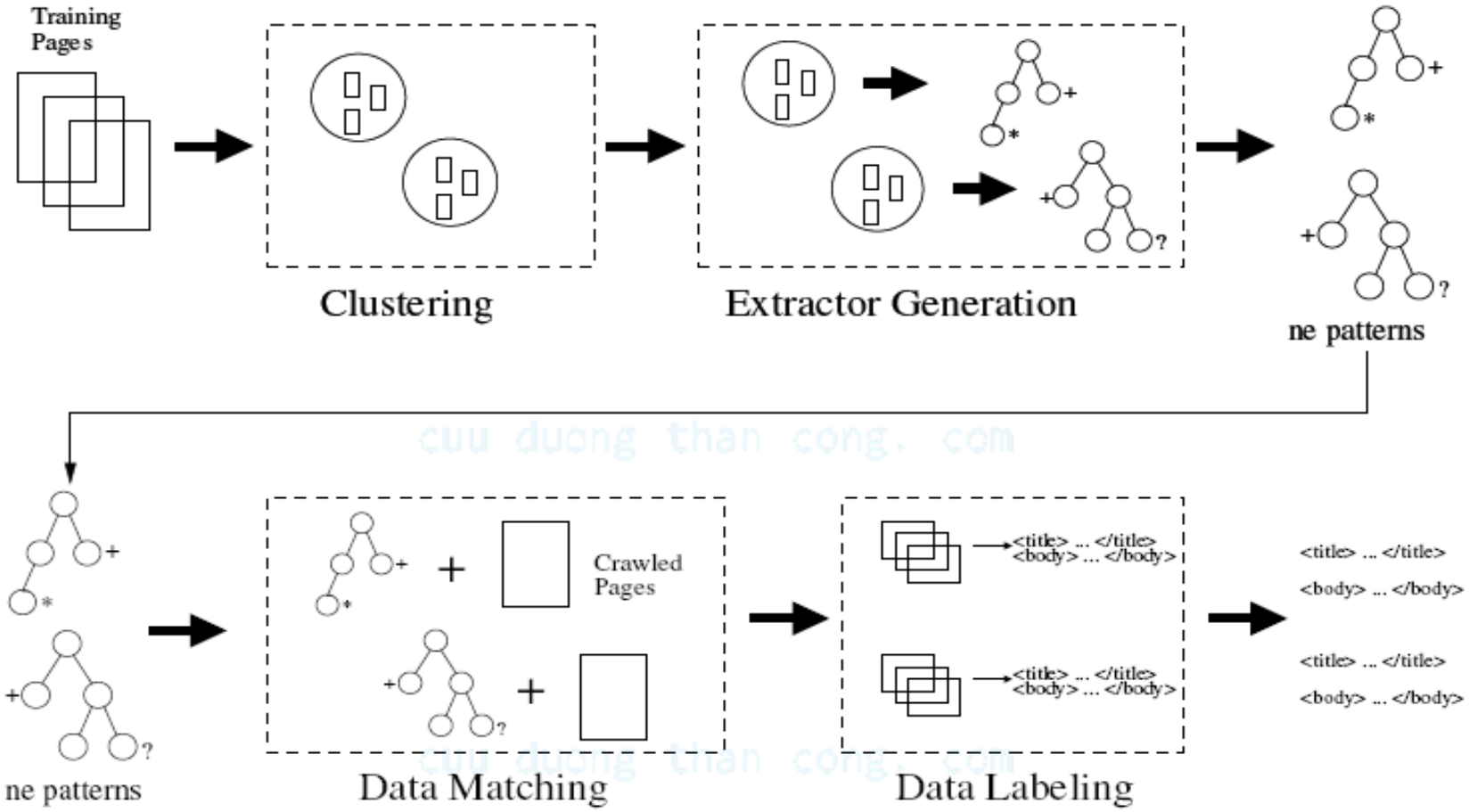


- Cấu trúc trang Web
  - Trang Web được viết theo ngôn ngữ trình bày Web: chẳng hạn HTML, XML
  - Trang web được tổ chức dưới dạng hình cây
  - Cấu trúc trình bày nội dung trang web
- Phân tích cấu trúc trang Web
  - Tìm các mẫu cấu trúc trang Web
  - Kết hợp với khai phá nội dung Web

2. Khai phá cấu trúc trang báo điện tử

Davi de Castro Reis, Paulo B. Golgher, Altigran S. da Silva, Alberto H. F. Laender (2004). Automatic Web News Extraction Using Tree Edit Distance, *Proceedings of the Thirteenth International World Wide Web Conference:* 502-601, ACM Press, New York, NY, May 2004, ISBN 1581139128

2. Khai phá cấu trúc trang báo điện tử



Davi de Castro Reis, Paulo B. Golgher, Altigran S. da Silva, Alberto H. F. Laender (2004). Automatic Web News Extraction Using Tree Edit Distance, *Proceedings of the Thirteenth International World Wide Web Conference:* 502-601, ACM Press, New York, NY, May 2004, ISBN 1581139128

2. Áp dụng: báo điện tử Việt Nam

NGHIÊN CỨU CÔNG NGHỆ KHAI PHÁ DỮ LIỆU VĂN BẢN, ÁP DỤNG CHO CÁC TRANG TIN TỨC TRÊN CÁC THIẾT BỊ CẦM TAY (PDAs & SMARTPHONEs)

Sinh viên thực hiện: KS. Vũ Ngọc Anh

Giáo viên hướng dẫn: TS. Hà Quang Thụy

Lớp: K9T3

Wednesday, November 10, 2010 Kênh tin tức điện tử cho PDAs & Smartphones Trang 1

2. Áp dụng: báo điện tử Việt Nam

Vũ Ngọc Anh (2006). Kênh tin tức điện tử cho PDAs & Smartp, *Luận văn Thạc sỹ*, Trường ĐHCN-ĐHQGHN

2. Áp dụng: báo điện tử Việt Nam

- http://vietbao.vn/Vi-tinh-Vien-thong/12-san-pham-vao-vong-chung-khao-Tri-tue-Viet-Nam/20641855/217/ ; Thứ sáu, 08 Tháng mười hai 2006, 02:31 GMT+7
  - "4. Vienews - kênh báo điện tử trên thiết bị điện thoại di động thông minh (Vũ Ngọc Anh, Hà Duyên Hòa - *Hà Nội*): Sản phẩm hỗ trợ thiết bị di động cầm tay đọc báo điện tử qua môi trường Internet không dây".
- http://www.tapchibcvt.gov.vn/vi-vn/dacsan/2006/8/17521.bcvt ; 7:58, 02/01/2007
  - **7. Giải Ba:** Sản phẩm đoạt giải: **"Các kênh báo điện tử trên thiết bị điện thoại di động thông minh"** của Hà Duyên Hoá (Hà Nội).

BÀI GIẢNG KHAI PHÁ DỮ LIỆU WEB

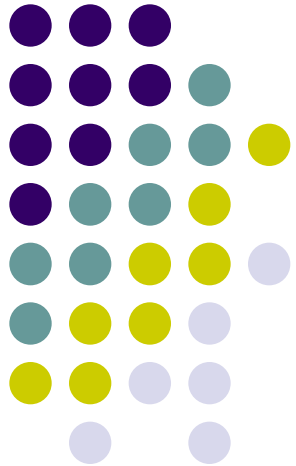
CHƯƠNG 3. MỘT SỐ KIẾN THỨC TOÁN HỌC BỔ TRỢ
CHƯƠNG 4. MỘT SỐ BÀI TOÁN XỬ LÝ NGÔN NGỮ TỰ NHIÊN NỀN TẢNG

PGS. TS. HÀ QUANG THỤY

HÀ NỘI 10-2010

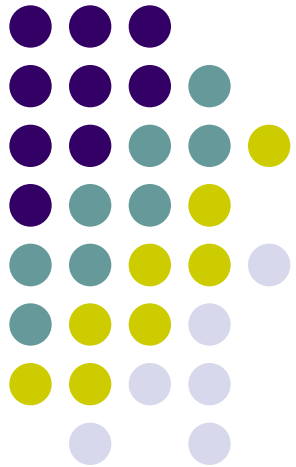
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

ĐẠI HỌC QUỐC GIA HÀ NỘI

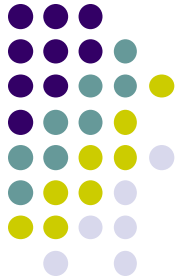


Nội dung

1. Một số kiến thức Toán học bổ trợ
2. Một số bài toán xử lý ngôn ngữ tự nhiên nền tảng

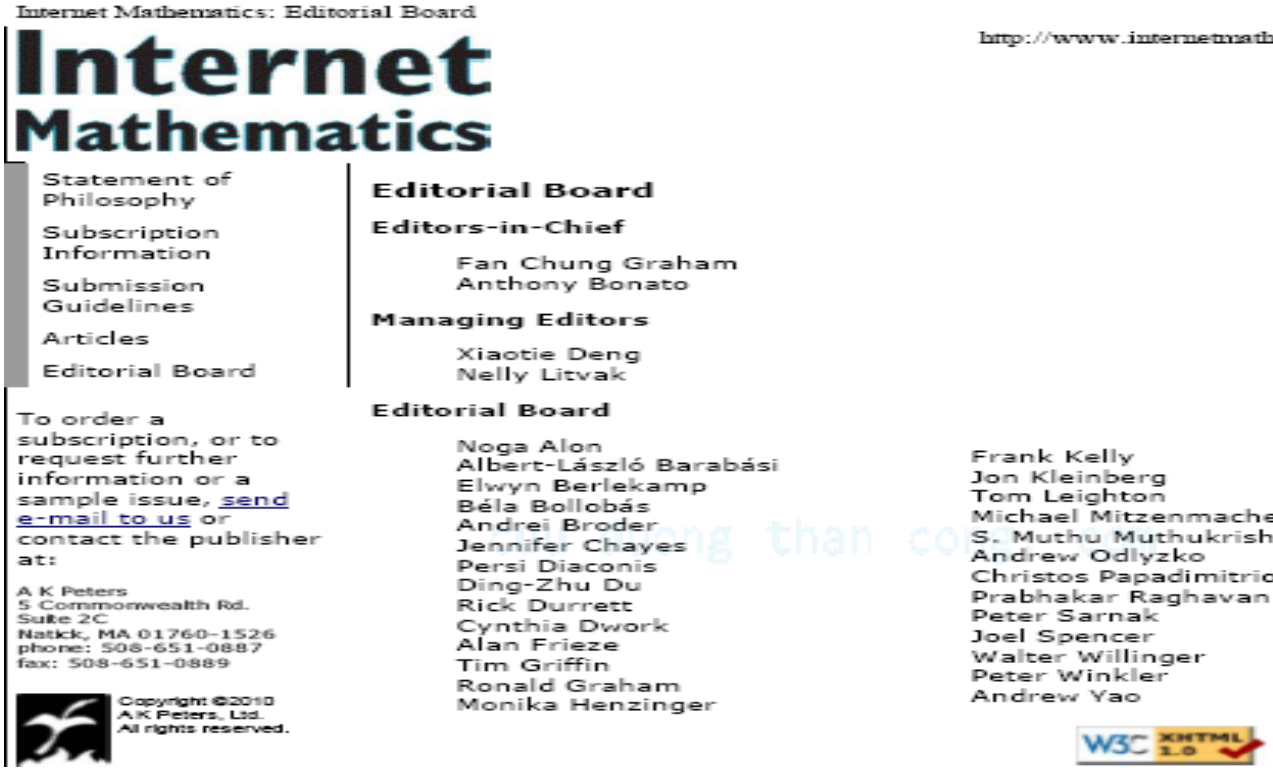


C3. Một số kiến thức Toán học bổ trợ



- Toán học Internet
  - Ra đời một lĩnh vực mới: Internet Mathematics
  - Cộng đồng Toán học Internet: Internet Mathematics Community
- Đối tượng và các chủ đề
  - Đối tượng: *Mạng phức tạp trên Internet và Web*: đồ thị Web, đồ thị Internet, mạng xã hội trực tuyến (Facebook, LinkedIn, và Twitter…), mạng sinh học trên Web…
  - Các chủ đề thuộc khai phá và mô hình hóa web (cơ sở lý thuyết và ứng dụng thực tiễn) trong môi trường mạng phức tạp.
- Tạp chí Internet Mathematics
  - http://www.internetmathematics.org/ (2/2011 - *xem trang sau*)
  - Đồng Trưởng ban biên tập:
    - Fan Chung Graham (http://www.math.ucsd.edu/~fan/). DBLP: 137 bài báo
    - Anthony Bonato (http://www.math.ryerson.ca/~abonato/). DBLP: 35 bài báo
  - Công bố bài báo chất lượng cao về mạng phức

Tạp chí Internet Mathematics



Internet Mathematics: Editorial Board

http://www.internetmathematics.org/board.htm

Internet Mathematics

Statement of Philosophy
Subscription Information
Submission Guidelines
Articles
Editorial Board

To order a subscription, or to request further information or a sample issue, send e-mail to us or contact the publisher at:

A K Peters
5 Commonwealth Rd.
Suite 2C
Natick, MA 01760-1526
phone: 508-651-0887
fax: 508-651-0889

Editorial Board

Editors-in-Chief

Fan Chung Graham
Anthony Bonato

Managing Editors

Xiaotie Deng
Nelly Litvak

Editorial Board

Noga Alon
Albert-László Barabási
Elwyn Berlekamp
Béla Bollobás
Andrei Broder
Jennifer Chayes
Persi Diaconis
Ding-Zhu Du
Rick Durrett
Cynthia Dwork
Alan Frieze
Tim Griffin
Ronald Graham
Monika Henzinger

Frank Kelly
Jon Kleinberg
Tom Leighton
Michael Mitzenmacher
S. Muthu Muthukrishnan
Andrew Odlyzko
Christos Papadimitriou
Prabhakar Raghavan
Peter Sarnak
Joel Spencer
Walter Willinger
Peter Winkler
Andrew Yao

W3C XHTML 1.0

- Ban biên tập tạp chí: Bổ sung một số chuyên gia

  Jennifer Tour Chayes http://research.microsoft.com/en-us/um/people/jchayes/. "She is the co-author of over 100 scientific papers and the co-inventor of more than 25 patents"

  Rick Durrett http://www.math.duke.edu/~rtd/ .

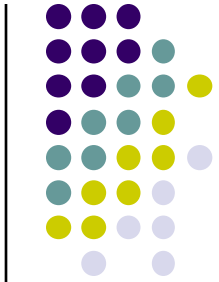
  Andrew Tomkins http://www.tomkinshome.com/andrew/paperlist. DBLP: 88 bài báo

- Một số biên tập viên được lưu ý

  Ronald L. Graham (http://www.math.ucsd.edu/~ronspubs/). DBLP:116 bài báo. Nhiều giải thưởng

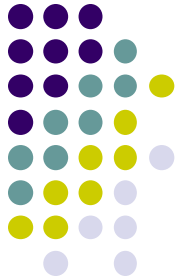
  Frank Kelly (http://www.statslab.cam.ac.uk/~frank/)

Một số nội dung Toán học bổ trợ



- Mô hình đồ thị
  - Một số kiến thức cơ sở
  - Đồ thị ngẫu nhiên
  - Mạng xã hội
- Học máy xác suất Bayes
  - Một số kiến thức cơ sở
  - Học máy xác suất Bayes
  - Ước lượng giá trị tham số
- Thuật toán Viterbi
  - Lý thuyết quyết định hỗn hợp
  - Nội dung thuật toán

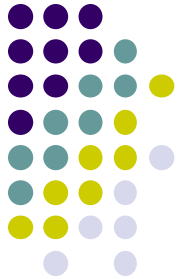
Đồ thị Web và đồ thị ngẫu nhiên



- Đồ thị Web
  - Web có cấu trúc đồ thị
    - Đồ thị Web: nút ⇔ trang Web, liên kết ngoài ⇔ cung (có hướng, vô hướng).
    - Bản thân trang Web cũng có tính cấu trúc cây (đồ thị)
  - Một vài bài toán đồ thị Web
    - Biểu diễn nội dung, cấu trúc
    - Tính hạng các đối tượng trong đồ thị Web: tính hạng trang, tính hạng cung..

  Nghiên cứu về đồ thị Web (xem trang sau)
- Đồ thị ngẫu nhiên
  - Tính ngẫu nhiên trong khai phá Web
    - WWW có tính ngẫu nhiên: mới, chỉnh sửa, loại bỏ
    - Hoạt động con người trên Web cũng có tính ngẫu nhiên
  - Là nội dung nghiên cứu thời sự

Bibliography Webgraph Papers
Dragomir R. Radev, 03/4/2010

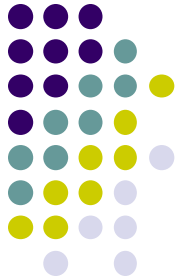


Toàn bộ	2007	2008	2009	To 04/10	2007-10
1542	127	61	36	13	237

- So many webgraph research papers.
- Some previous versions of "Bibliography Webgraph Papers" by Dragomir R. Radev
- 1601: http://clair.si.umich.edu/~radev/webgraph/webgraph-bib.html

5/2005	5/2007	5/2008	1/2009	8/2009	4/2010	11/2010
496	1212	1361	1457	1471	1542	1601

Lý thuyết về đồ thị lớn



Đồ thị lớn

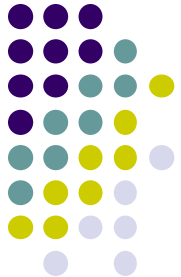
- Số đỉnh lên tới hàng tỷ
- Biểu diễn cung chính xác không còn là quan trọng

Cơ sở lý thuyết trong nghiên cứu đồ thị lớn

- Khả năng là lý thuyết sinh đồ thị
- Bất biến tới một số thay đổi nhỏ trong định nghĩa
- Phải có năng lực chứng minh các định lý cơ bản

[Hop07] John E. Hopcroft (2007). Future Directions in Computer Science, *http://www.cs.cornell.edu/jeh/China%202007.ppt*

Đồ thị ngẫu nhiên: Mô hình Erdös-Renyi

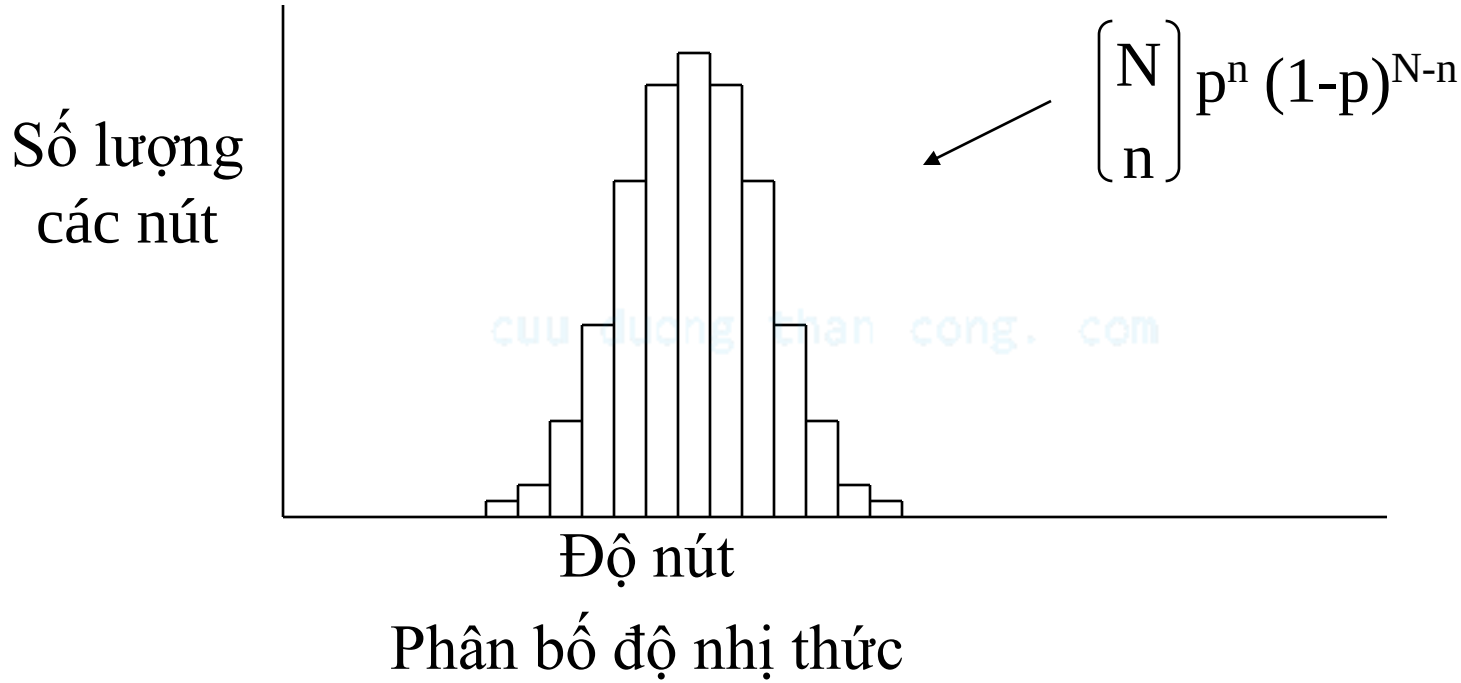


- Đồ thị ngẫu nhiên: có thể mô hình mạng thế giới thực.
- Định nghĩa: có hai định nghĩa
  - Chọn ngẫu nhiên: $G_{n, N}$ được chọn ngẫu nhiên từ $_{n, N}$ = {mọi đồ thị có n đỉnh và N cung}' các phần tử trong $_{n, N}$ là đồng khả năng được chọn với xác suất 1/((^{n} $_{2}$)/N);
  - Quá trình hình thành các cung trong $G_{n, N}$ là ngẫu nhiên: mỗi cạnh xuất hiện với xác suất p, sự xuất hiện hay vắng mặt hai cạnh là độp lập nhau.

[ER61] P. Erdös, A. Rényi (1961). On the evolution of random graphs, *Théorie de L'Information:* 343-347, 1961.

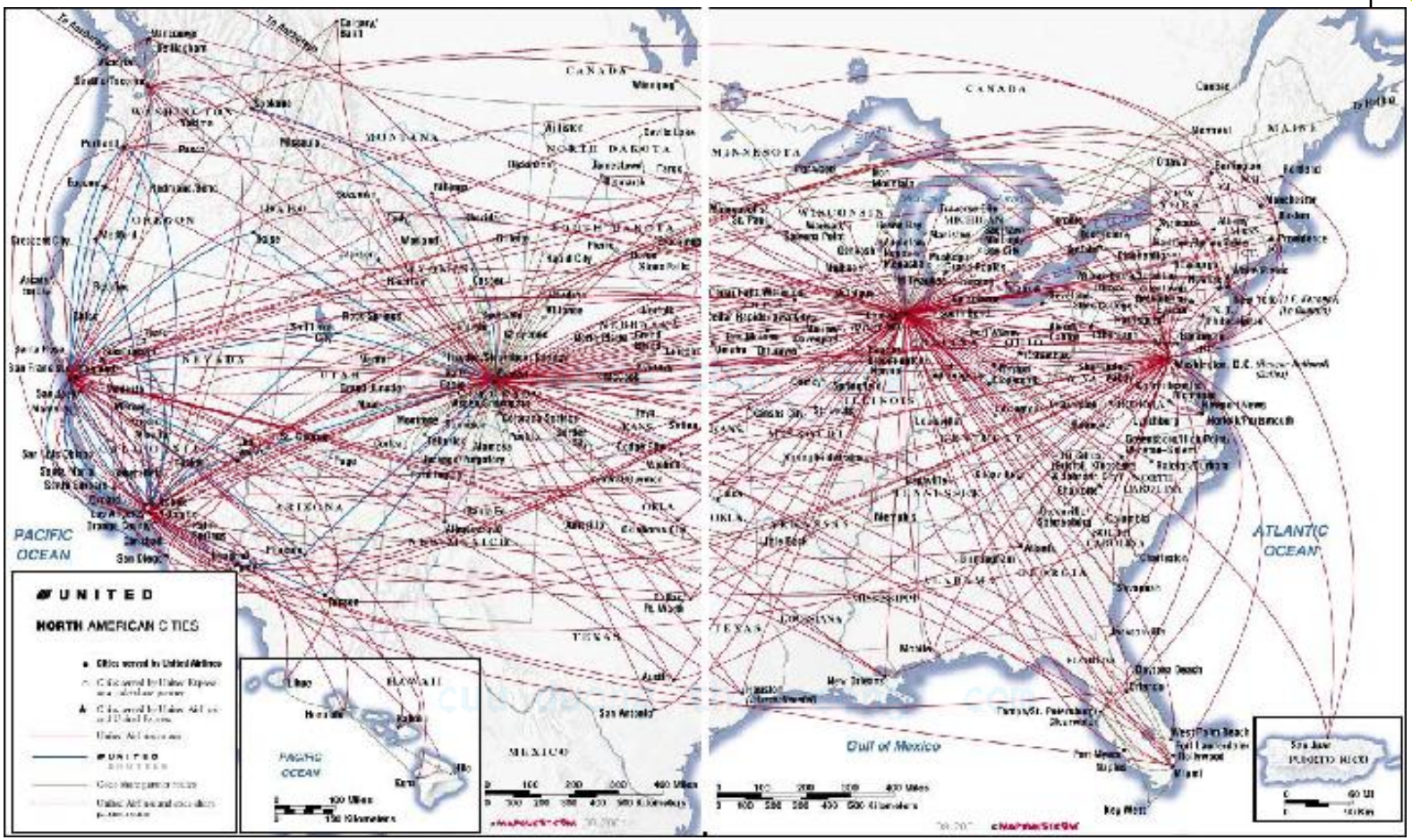
Đồ thị ngẫu nhiên: Mô hình Erdös-Renyi

- Đặt tên: Paul Erdős và Alfréd Rényi
- Là một trong hai mô hình sinh các đồ thị ngẫu nhiên
- Chứa tập các nút mà mỗi nút trong mỗi tập đó có xác suất như nhau, độc lập với các cung khác
- n nút: Mỗi bộ n^2 cung tiềm năng được biểu diễn với xác xuất độc lập



Phân bố độ nhị thức

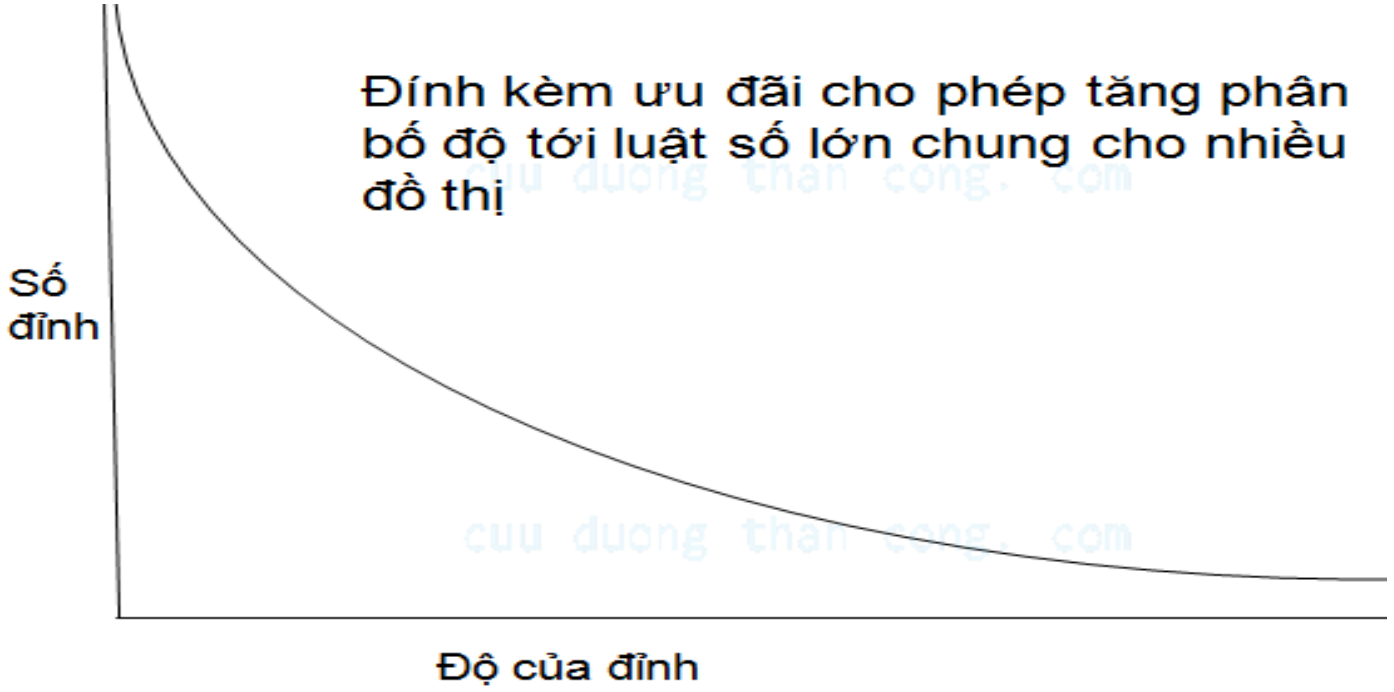
Đồ thị ngẫu nhiên



[Hop07] John E. Hopcroft (2007). Future Directions in Computer Science, http://www.cs.cornell.edu/jeh/China%202007.ppt

Mô hình sinh đồ thị

- Các nút và cung được bổ sung sau mỗi đơn vị thời gian
- Quy tắc xác định nơi cung xuất hiện (nơi đặt cung mới)
  - Xác suất đồng nhất
  - Đính kèm ưu đãi – đưa đến phân bố theo luật số lớn

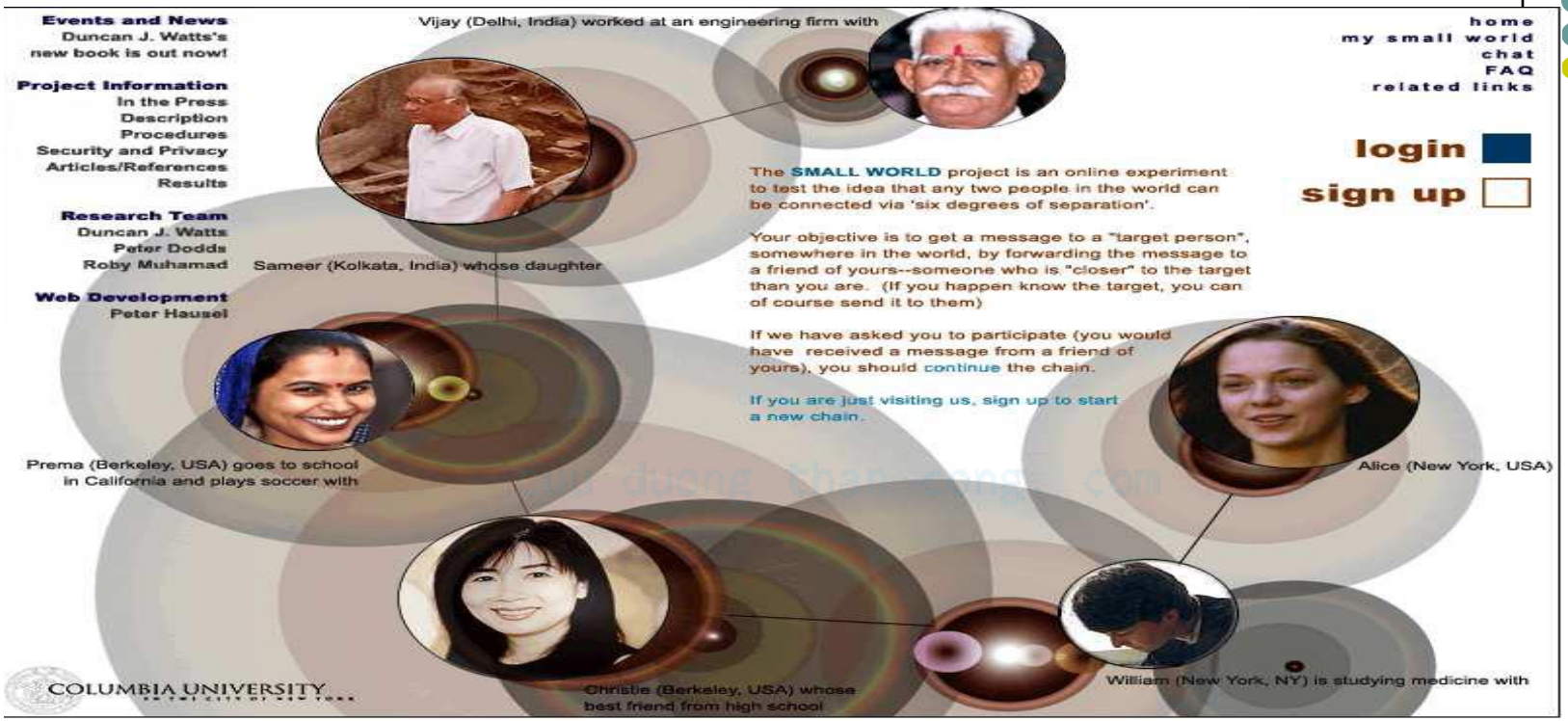


[Hop07] John E. Hopcroft (2007). Future Directions in Computer Science, *http://www.cs.cornell.edu/jeh/China%202007.ppt*

Mạng xã hội

- Mạng xã hội
  - Internet, Web là một xã hội ảo
    - Nhiều hoạt động (đặc biệt là hoạt động thông tin) trong thế giới thực được thi hành
    - “Thế giới phẳng”, “toàn cầu hóa” và “bản địa hóa”
  - Khái niệm
    - Mạng xã hội là mạng của một nhóm người có hoạt động và các mối quan hệ gắn kết họ với nhau.
    - Mạng xã hội là một kiểu của mạng phức tạp
  - Một số ví dụ mạng xã hội trên Internet
    - Diễn đàn, Blog, Mạng e-mail, mạng xã hội chuyên đề
    - Một số ví dụ khác (trang bên)
  - Nghiên cứu mạng xã hội
    - Vấn đề nghiên cứu thời sự.
    - Kết hợp nhiều lĩnh vực, chẳng hạn như CNTT + Xã hội học

Mạng xã hội: ví dụ



http://www.uvm.edu/~pdodds/teaching/courses/2008-01UVM-295/docs/2008-01UVM-295smallworldnetworks-slides-handout.pdf

Social Networks: Properties

- The small-world property
  - Almost any pair of people in the world can be connected together by a short chain of intermediate acquaintances, usually about six lengths.

[TM69] Jeffrey Travers, Stanley Milgram (1969). An Experimental Study of the Small World Problem, *Sociometry*, **32**(4): 425-443, Dec., 1969.

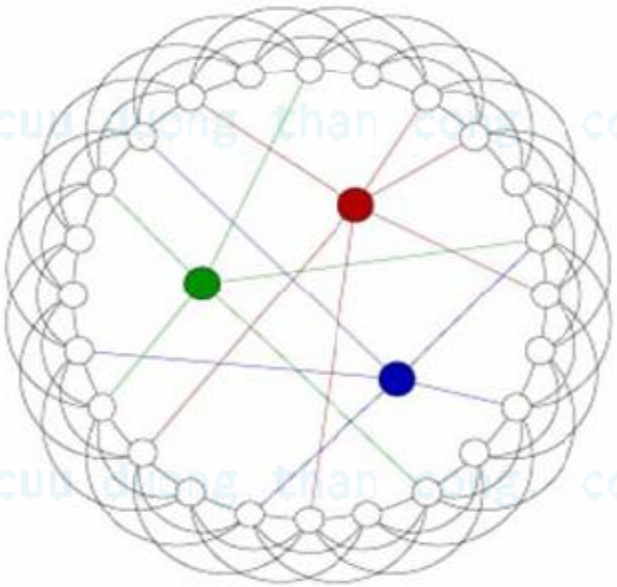
- Power-law degree distributions / the scale – free property
  - Social network's nodes (also edges) are distributed under the power-law degree

- Network transitivity
  - Structure and dynamics of the network influenced by nodes with the large number of connectings (*using to detect communities in a social network*!)
- **Community structure**
  - Networks are divided into communities in which the nodes in the same community closed links, and links communities liquid
  - A community in social networks as an **"interest group"** in the real world. http://en.wikipedia.org/wiki/Interest_group_(disambiguation) as meaning of **"nhóm lợi ích"** in Vietnamese. See also **"Advocacy group"**, **"Lobby group"**. 5P&5C marketing model: **People ⇨ Customer approach** (Product ⇨ Consumer desire; Price ⇨ Cost; Place ⇨ Convenience; Promotion ⇨ Communication)
  - Flexible community structure: one community structure for one case.

Social Networks: Properties

Small-world Networks

Almost any pair of people in the world can be connected to one another by a short chain of intermediate acquaintances, of typical length about six.



Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

P.O.Boykin and V.Roychowdhury's research

The quantitative definition of the clustering coefficient

$$C_i = \frac{2E_i}{k_i(k_i - 1)} \qquad (1)$$

- C_i is clutering coefficient of node i in email networks.
- k_i is degree of node i (or node i has k_i neighbors)
- E_i is the number of links between k_i neighbors of i.

Comments

- Can't calculate in case $K_i = 1$.
- If $E_i = 0$ then $C_i = 0$, independent of k_i

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

Proposed method

Clustering coefficient

$$C_i = \frac{2(E_i + 1)}{k_i(k_i - 1) + 1} \qquad (2)$$

Comments

- Formula (2) can't discriminate between users who sent e-mails from users who received e-mails.
- For this reason, we consider the email network graph as directed graph.

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

Proposed method

New clustering coefficient formula

$$C_i = a \times \frac{2(E_i + 1)}{S_i(S_i - 1) + 1} + b \times R_i \qquad (3)$$

Formulas of E_i, S_i, R_i

$$E_i = \sum_{j=1}^{Edge} (1 + w_i \times 0.05) \qquad (4)$$

$$S_i = \sum_{j=1}^{Send} (1 + w_i \times 0.05) \qquad (5)$$

$$R_i = \sum_{j=1}^{Recieve} (1 + w_i \times 0.05) \qquad (6)$$

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

Experimental Data

- The e-mail network studied here is constructed from log files of the VNUH e-mail server.
- Logs over of a period of one week (from March, 28th to April, 4th, 2006.)
- Consists of **19876** users (1149 in-users, 18727 out-users).
- Total of exchanged messages in this time is **88842** messages

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

Email Network Graph

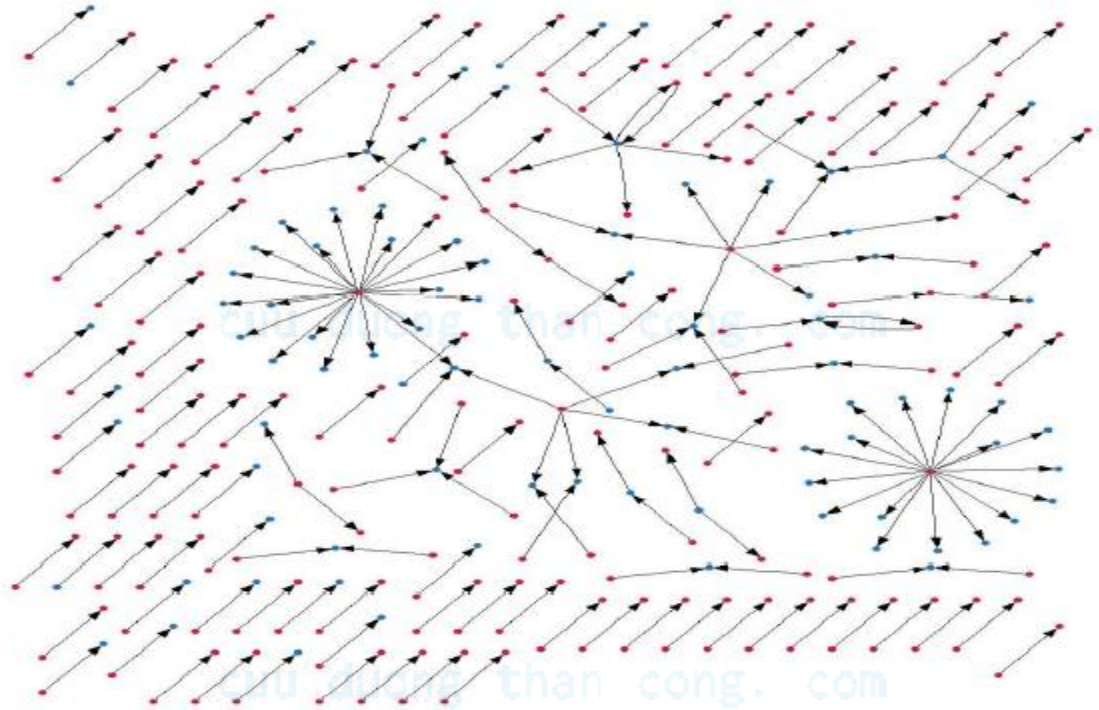


Figure: Email network graph is constructed from log files of VNUH in an hour

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

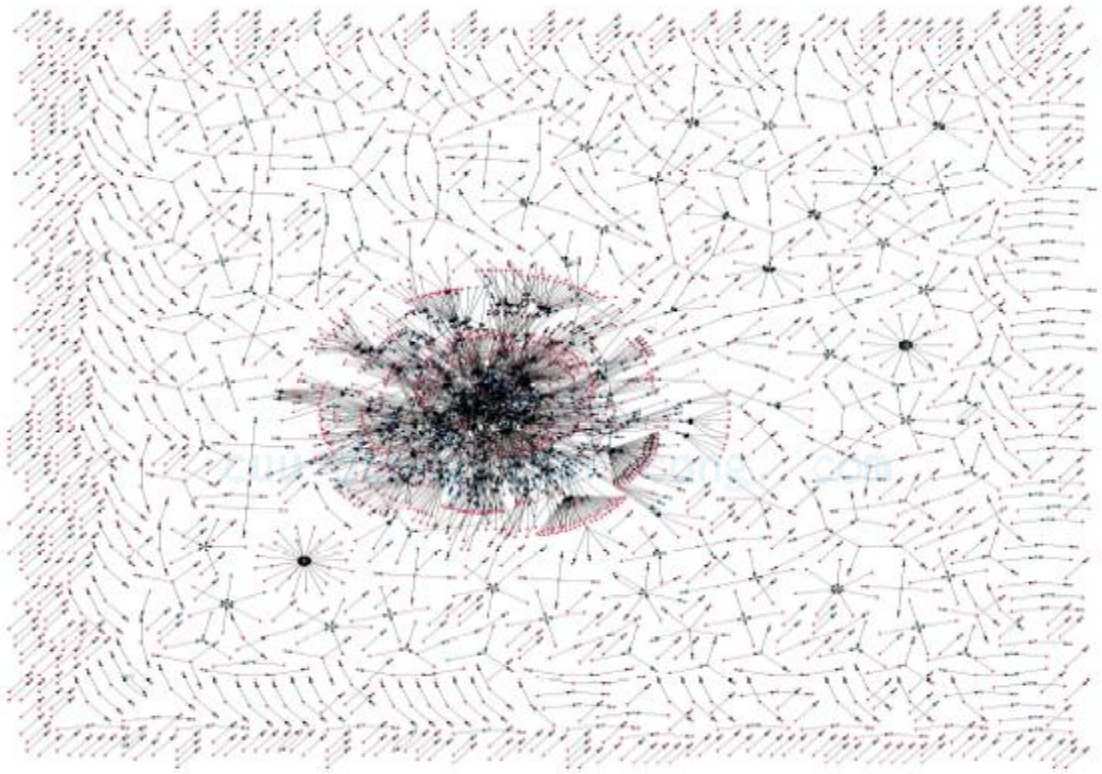


Figure: Email network graph is constructed from log files of VNUH in a week

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

E-mail Networks

Result

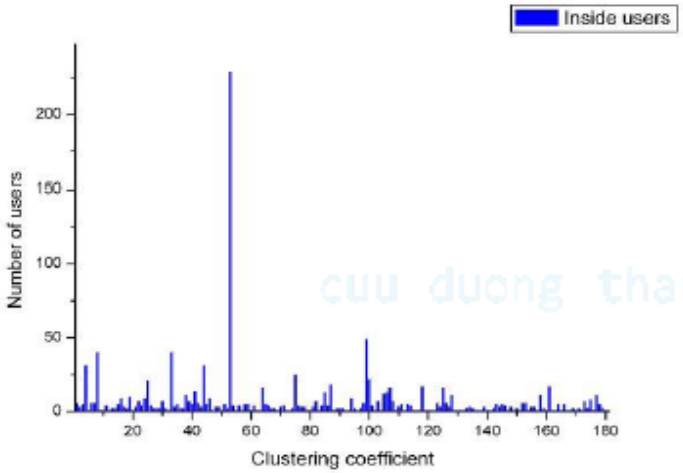


Figure: Clustering coefficient distribution diagram of in-users

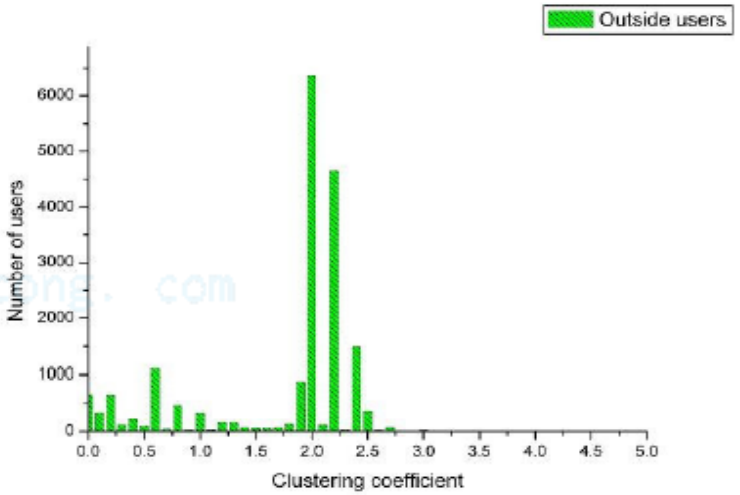
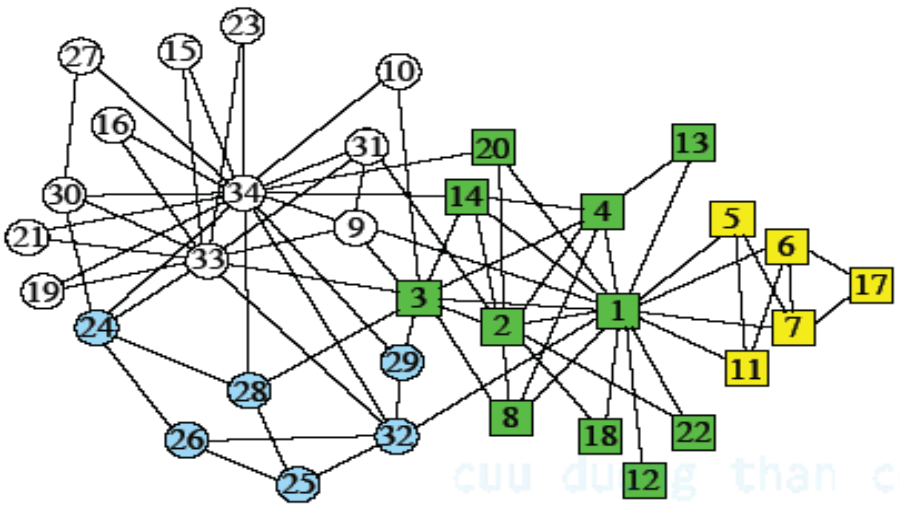


Figure: Clustering coefficient distribution diagram of out-users

Lan N. Bui, Anh Q. Tran, Thuy Q. Ha (2006). User authentic Rating based on Email Networks, *ICMOCCA2006*: 144-148, Seoul, Korea & *International Journal of Natural Sciences and Technology*, **1**(2): 173-180, 2006.

Mạng XH và cộng đồng [For10]

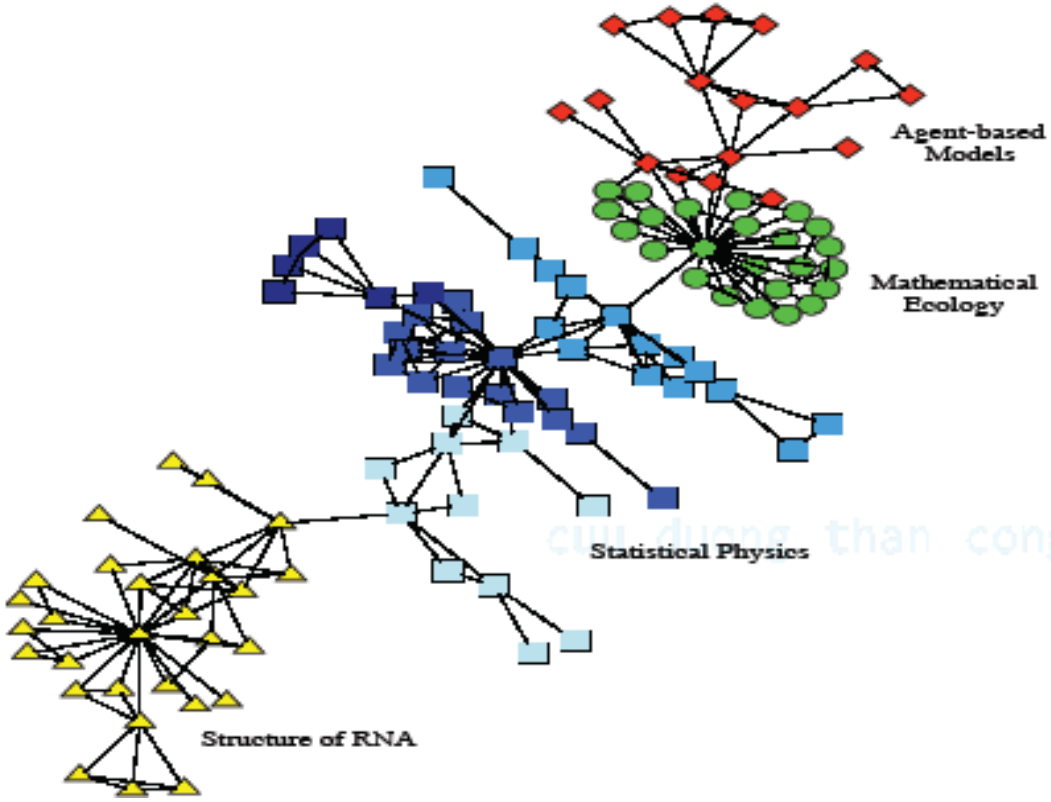


- Câu lạc bộ karate của Zachary (được quan sát trong 3 năm), một kiểm chứng chuẩn cho phát hiện cộng đồng. Các màu sắc tương ứng với phân hoạch tốt nhất tìm được bằng cách tối ưu các mô đun của Newman và Girvan.

- Đồ thị gồm 34 đỉnh thành viên của câu lạc bộ. Cạnh nối các cá nhân có tương tác bên ngoài các hoạt động của câu lạc bộ. Theo quan sát, có xung đột giữa chủ tịch câu lạc bộ và người hướng dẫn dẫn đến sự phân hoạch câu lạc bộ thành hai nhóm riêng biệt, tương ứng ủng hộ người hướng dẫn và chủ tịch (chỉ dẫn hình vuông và hình tròn). Câu hỏi đặt ra là liệu từ cấu trúc mạng ban đầu có thể suy luận các thành phần của hai nhóm.
- Nhìn vào hình, có thể phân biệt hai tập hợp, một tập quanh các đỉnh 33 và 34 (*34 là chủ tịch*), tập còn lại quanh đỉnh 1 (*người hướng dẫn*).
- Cũng có một số đỉnh nằm giữa hai cấu trúc chính, chẳng hạn như 3, 9, 10; đỉnh như vậy thường không phân loại được theo phương thức phát hiện cộng đồng.

[For10] Santo Fortunato (2010), Community detection in graphs, *Technical Report*, Complex Networks and Systems Lagrange Laboratory, ISI Foundation, Torino, ITALY.

Mạng XH và cộng đồng [For10]

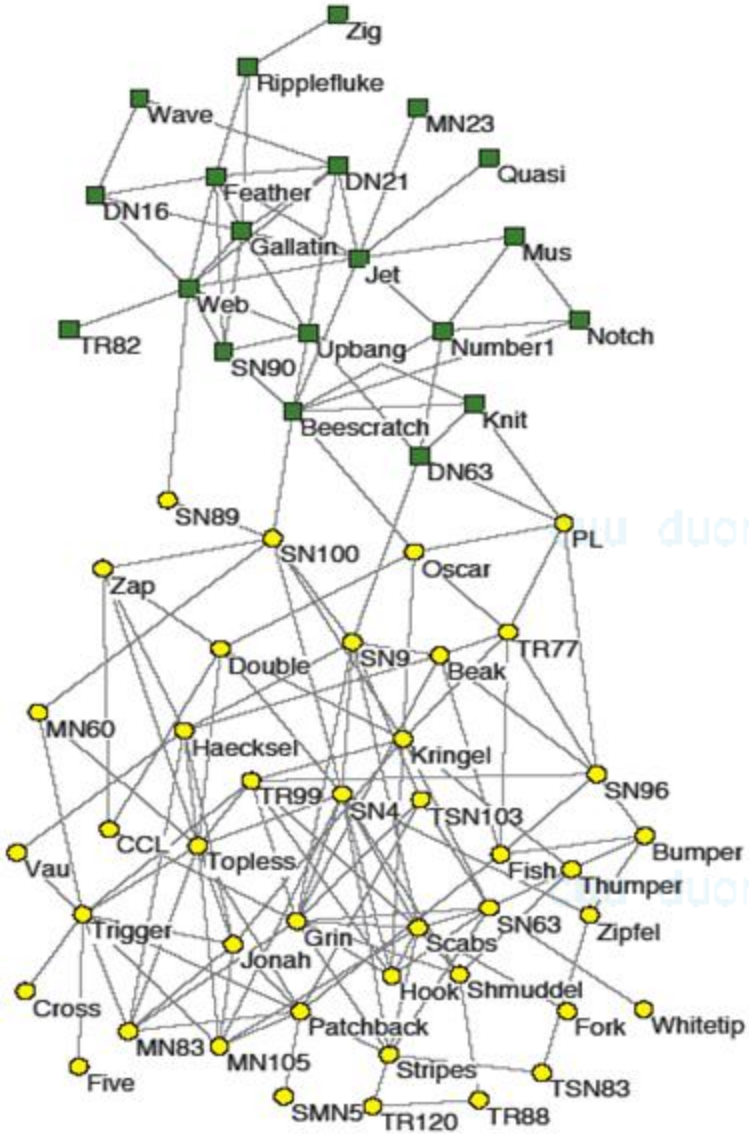


- Mạng hợp tác giữa mạng các nhà khoa học làm việc tại học viện Santa Fe (SFI). Các màu chỉ dẫn cộng đồng ở mức độ cao thu được theo thuật toán của Girvan và Newman (mục VA) và tương ứng khá chặt chẽ với các đơn vị nghiên cứu của học viện. Phân chia nhỏ hơn tương ứng với các nhóm nghiên cứu nhỏ hơn, xoay quanh các lãnh đạo dự án.

Đồ thị hiện có 118 đỉnh (các nhà khoa học đại diện cho cư dân tại SFI và cộng tác viên của họ). Các cạnh nối các nhà khoa học đã cùng công bố ít nhất một bài báo. Trực quan cho phép phân biệt được các nhóm chuyên ngành. Trong mạng này, khi quan sát nhiều nhóm, là tác giả của một bài báo thì tất cả cùng liên kết với nhau. Có chỉ một số ít các kết nối giữa hầu hết các nhóm.

[For10] Santo Fortunato (2010), Community detection in graphs, *Technical Report*, Complex Networks and Systems Lagrange Laboratory, ISI Foundation, Torino, ITALY.

Mạng XH và cộng đồng [For10]



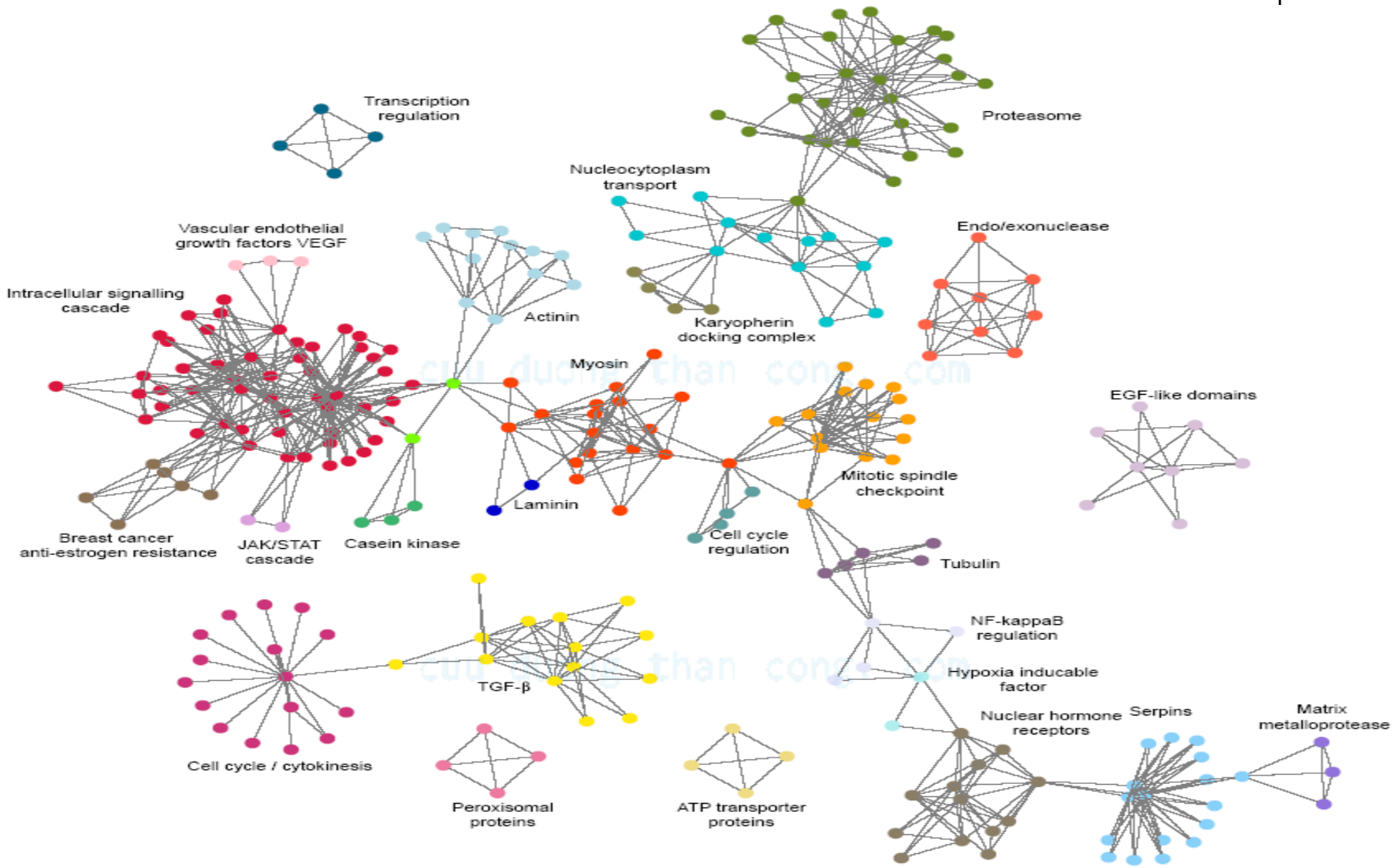
Mạng Lusseau các cá heo mũi to. Những màu nhãn cộng đồng được xác định qua tối ưu hóa một phiên bản mô đun của Newman và Girvan, theo đề xuất của Arenas và cộng sự. Phân hoạch phù hợp với các lớp sinh học của cá heo được Lusseau đề xuất.

Hiện có 62 cá heo, các cạnh nối các cá heo được nhìn thấy thường xuyên hơn so với dự kiến. Tập cá heo bị tách thành hai nhóm sau khi cá heo một trái nơi dành cho một số thời gian (hình vuông và hình tròn trong Hình vẽ). Các nhóm như vậy là khá cố kết, với một vài cụm (clique) nội bộ, và dễ dàng định danh: chỉ có sáu cạnh nối các đỉnh của nhóm khác nhau.

Do mạng này phân lớp tự nhiên cá heo Lusseau, cũng như câu lạc bộ karate của Zachary, thường được dùng để kiểm tra thực nghiệm thuật toán phát hiện cộng đồng

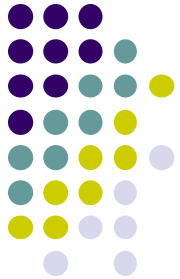
[For10] Santo Fortunato (2010), Community detection in graphs, *Technical Report*, Complex Networks and Systems Lagrange Laboratory, ISI Foundation, Torino, ITALY.

Mạng XH và cộng đồng: tương tác protein - protein [For10]



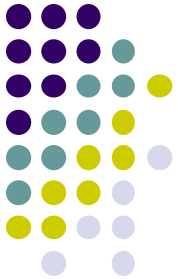
[For10] Santo Fortunato (2010), Community detection in graphs, *Technical Report*, Complex Networks and Systems Lagrange Laboratory, ISI Foundation, Torino, ITALY.

Học máy xác suất Bayes



- Một số kiến thức cơ sở về lý thuyết xác suất
  - Không gian đo được
  - Không gian xác suất
  - Sigma – trường
  - Hệ thống động
  - Quá trình ngẫu nhiên thời gian rời rạc
  - Kỳ vọng
  - Entropy
  - Trong tài liệu
- Nhắc thêm về học máy xác suất
  - …

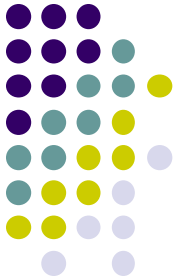
Học máy xác suất Bayes



- Mô hình tần số
  - Tiến hành thử nghiệm lặp đi lặp lại
  - Tỷ số xuất hiện trên toàn bộ số lần thử
- Mô hình xác suất
  - Xác suất có điều kiện: sự kiện e với tri thức nền là $P(e|\ )$
  - Tri thức nền là sự xuất hiện của tài liệu (trái) hoặc sự xuất hiện của tài liệu mới.
  - Xác suất tiên nghiệm và xác suất hậu nghiệm.

$$P(e|D) \quad \frac{P(D|e)P(e)}{P(D)} \qquad P(e|D,D_2) \quad \frac{P(D_2|e,D)P(e|D)}{P(D_2|D)}$$

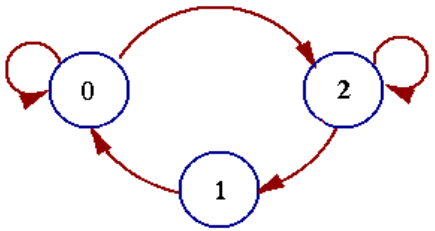
Ước lượng giá trị tham số



- Hai bài toán
  - Lựa chon mô hình hay dạng hàm: Dựa trên tri thức miền đã có
  - Mỗi mô hình/hàm có bộ tham số tương ứng
  - Cần xác định bộ tham số này
- Xác định tham số
  - Thường theo ước lượng
  - Ước lượng tham số mô hình
  - Ước lượng tham số cho trường hợp cụ thể

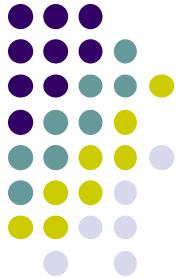
Thuật toán Viterbi

- Thuật toán Viterbi
  - Mô hình máy trạng thái hữu hạn
    - xác định tham số mô hình phù hợp tập ví dụ học
  - Lý thuyết quyết định hỗn hợp
  - Bài toán giải mã
    - Đã có mô hình máy trạng thái hữu hạn
    - Tìm dãy trạng thái phù hợp nhất với trường hợp cụ thể
  - Nội dung thuật toán
    - Xem trong giáo trình



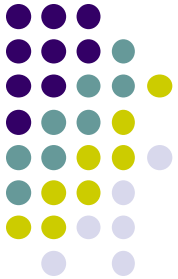
Input: $Z=z_1,z_2,\ldots,z_n$ // dãy quan sát đầu vào
Output: Đường đi ngắn nhất tương ứng với dãy quan sát đầu vào
Khởi tạo:
$k \leftarrow 1$ // chỉ số lặp
$S(c_1) \leftarrow c_1$
$L(c_1) \leftarrow 0$ // biến chứa tổng độ dài, khởi tạo là 0
Đệ quy:
repeat
For $\forall$ bộ chuyển $t_k = (c_k, c_{k+1})$
$L(c_k, c_{k+1}) \leftarrow L(c_k) + l\,[t_k = (c_k, c_{k+1})]$ theo $\forall\ c_k$
Tìm $L(c_{k+1}) = \min L(c_k, c_{k+1})$
For mỗi c_{k+1}
lưu $L(c_{k+1})$ và vết $S(c_{k+1})$ tương ứng
$k \leftarrow k+1$
until $k=n$

C4. Một số bài toán xử lý tiếng Việt



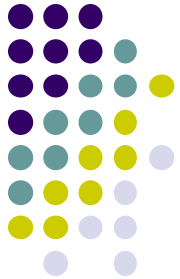
- Lĩnh vực xử lý ngôn ngữ tự nhiên
  - Xử lý ngôn ngữ tự nhiên (tự động hóa)
  - Ra đời khoảng nhứng năm 1950
  - Ngày càng phát triển
- Phân loại
  - Xử lý
    - Cơ bản
    - Ứng dụng
  - Tài nguyên
    - Cơ bản
    - Mức cao

Bài toán tách câu



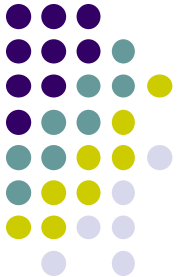
- Đây là bài toán khá đơn giản
- Khái niệm
  - Chuỗi ký tự kết thúc bằng dấu chấm, chấm hỏi, chấm than
  - Vẫn có trường hợp ngoại lệ (khoảng 10%)
    - Các dấu trên không kết thúc câu (trong nháy kép)
    - Một số dấu khác kết thúc câu
- Một số trường hợp
  - Dựa theo kinh nghiệm
  - Mô hình ME (Le Hong Phuong & Ho Tuong Vinh)
  - Xem giáo trình

Bài toán tách từ



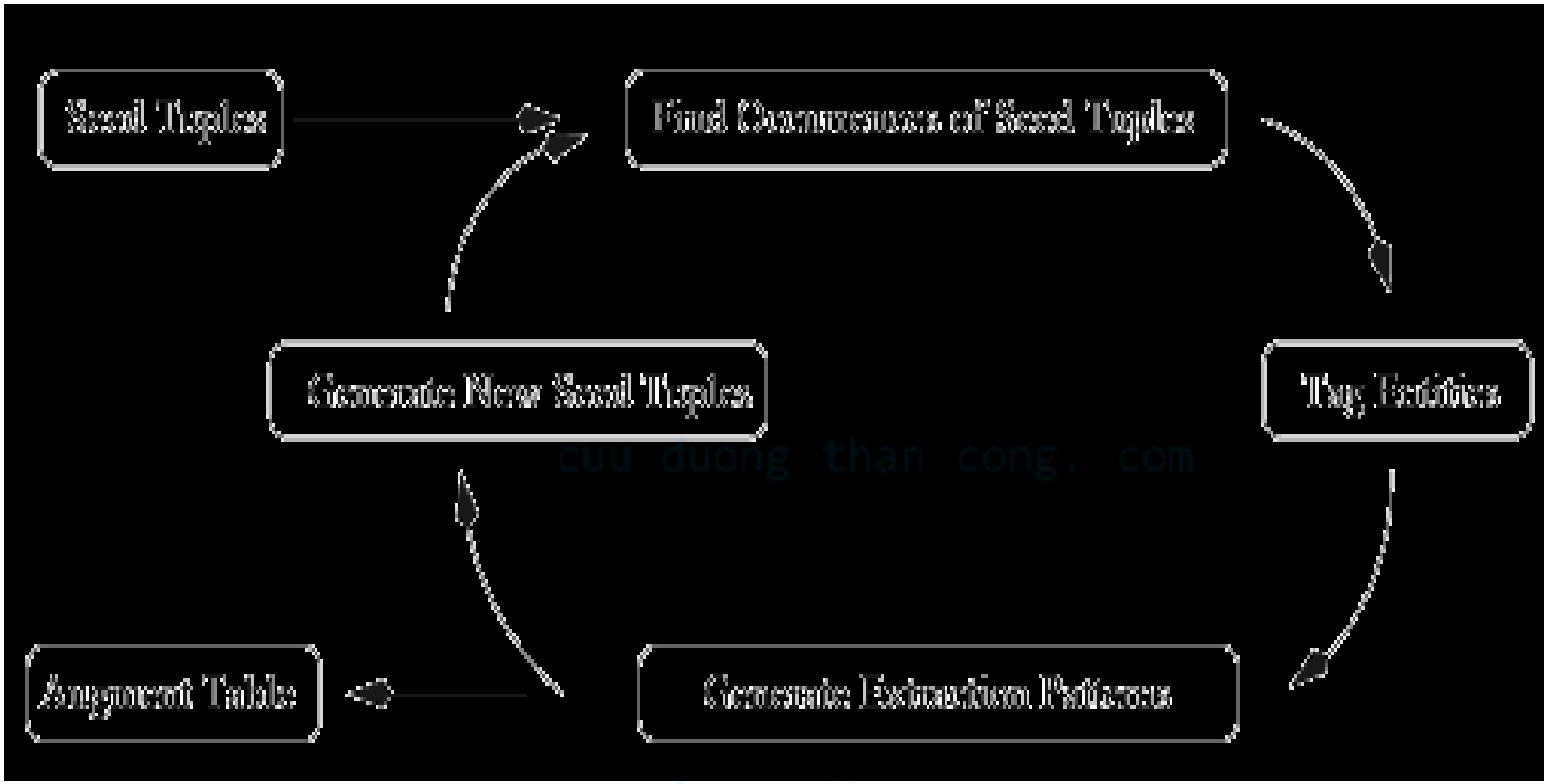
- Đây là bài toán rất cơ bản, luôn thời sự
  - Từ vẫn phát triển bổ sung, thay đổi
  - Ngăn cách hiển, nhập nhằng, mờ
  - “Ông già đi nhanh quá” | “Học sinh học sinh học” …
- Khái niệm
  - Cho một câu hãy xác định các từ trong câu
  - “Phù hợp ngữ cảnh”
- Một số phương pháp
  - Khớp tôi đa
  - Máy trạng thái hữu hạn có trọng số (WSFT)
    - Trường ngẫu nhiên có điều kiện
  - Xem giáo trình

Phát hiện quan hệ ngữ nghĩa



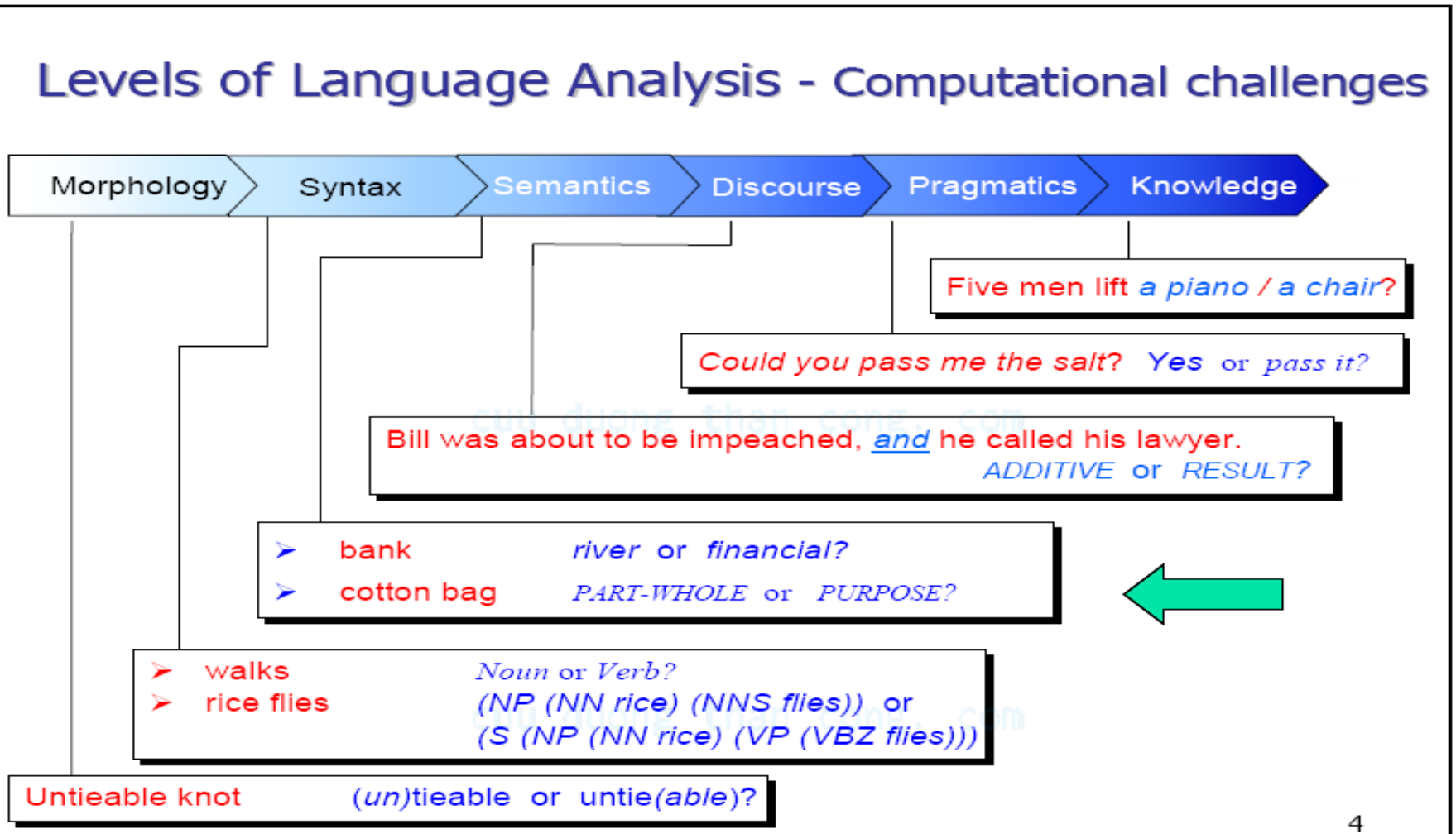
- Là bài toán cơ bản
  - Quan hệ ngữ nghĩa giữa các đối tượng ngữ pháp
  - Một số quan hệ ngữ nghĩa: theo cách tiếp cận
- Khái niệm
  - Cho một tập các văn bản
  - Tìm ra các đối tượng ngữ pháp và các quan hệ giữa chúng
- Một số phương pháp
  - DIPRE
  - SNOWBALL
  - Xem giáo trình

Phương pháp Snowball



Eugene Agichtein, Luis Gravano (2000). Snowball: extracting relations from large plain-text collections, *ACM DL 2000*: 85-94

Phát hiện quan hệ ngữ nghĩa



Các mức: Hình vị, Cú pháp, Ngữ nghĩa, Diễn ngôn, Phát ngôn (?), Tri thức

Roxana Girju (2008). Semantic Relations:Discovery and Applications

Quan hệ ngữ nghĩa: Ngôn ngữ học

Lists of Semantic Relations: Approaches in Linguistics (10)

(Levi 1978):

- Two syntactic processes are used:
  - predicate nominalization:
    - those involving nominalizations, i.e., compounds whose heads are nouns derived from a verb, and whose modifiers are interpreted as arguments of the related verb
    - E.g.: "x such that x plans cities" => city planner;
  - predicate deletion:
    - List of relations: ***cause, have, make, use, be, in, for, from, about***
    - E.g.: "field mouse" derived from "a mouse which is in the field" ("in" deletion);
    - Deleted predicates represent the only semantic relations which can underlie NCs not formed through predicate nominalization;

42

Roxana Girju (2008). Semantic Relations:Discovery and Applications

Quan hệ ngữ nghĩa: XLNNTN

Semantic Relation	Definition/ Example
HYPERNYMY (IS-A)	an entity/event/state is a subclass of another; (*daisy flower; large company, such as Microsoft*)
PART-WHOLE (MERONYMY)	an entity/event/state is a part of another entity/event/state; (*door knob; the door of the car*);
CAUSE	an event/state makes another event/state to take place; (*malaria mosquitos; "death by hunger"; "The earthquake generated a big Tsunami"*);
POSSESSION	an animate entity possesses (owns) another entity; (*family estate; the girl has a new car.*)
KINSHIP	an animated entity related by blood, marriage, adoption or strong affinity to another animated entity; (*boy's sister; Mary has a daughter*)
MAKE/PRODUCE	an animated entity creates or manufactures another entity; (*honey bees; GM makes cars*)
INSTRUMENT	an entity used in an event as instrument; (*pump drainage; He broke the box with a hammer.*)
TEMPORAL	time associated with an event; (*5-O' clock tea; the store opens at 9 am*)
LOCATION/ SPACE	spacial relation between two entities or between an event and an entity; (*field mouse; I left the keys in the car*)
PURPOSE	a state/activity intended to result from another state/event; (*migraine drug; He was quiet in order not to disturb her.*)
SOURCE/FROM	place where an entity comes from; (*olive oil*)

Roxana Girju (2008). Semantic Relations:Discovery and Applications

Quan hệ ngữ nghĩa: XLNNTN

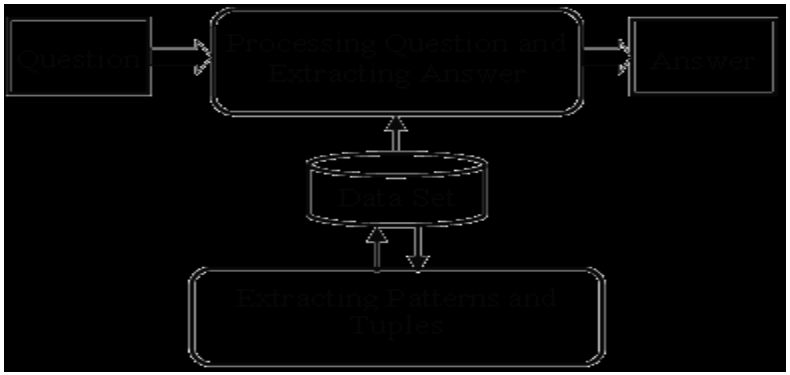
EXPERIENCER	an animated entity experiencing a state/feeling; (*desire for chocolate; Mary's fear*.)
TOPIC	an object specializing another object; (*they argued about politics*)
MANNER	a way in which an event is performed or takes place; (*hard-working immigrants; performance with passion*)
MEANS	the means by which an event is performed or takes place; (*bus service; I go to school by bus.*)

AGENT	the doer of an action; (the investigation of the police)
THEME	the entity acted upon in an action/event (music lover)
PROPERTY	characteristic or quality of an entity/event/state; (*red rose; the juice has a funny color.)*
BENEFICIARY	an animated entity that benefits from the state resulting from an event; (*customer service; I wrote Mary a letter*.)

MEASURE	an entity expressing quantity of another entity/event; (*70-km distance; The jacket costs $60; a cup of sugar*)
TYPE	a word/concept is a type of word/concept; (*member state; framework law*)
DEPICTION-DEPICTED	an entity is represented in another; (*the picture of the girl*)

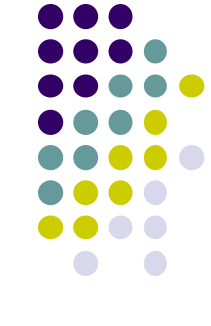
Roxana Girju (2008). Semantic Relations:Discovery and Applications

Phát hiện quan hệ ngữ nghĩa



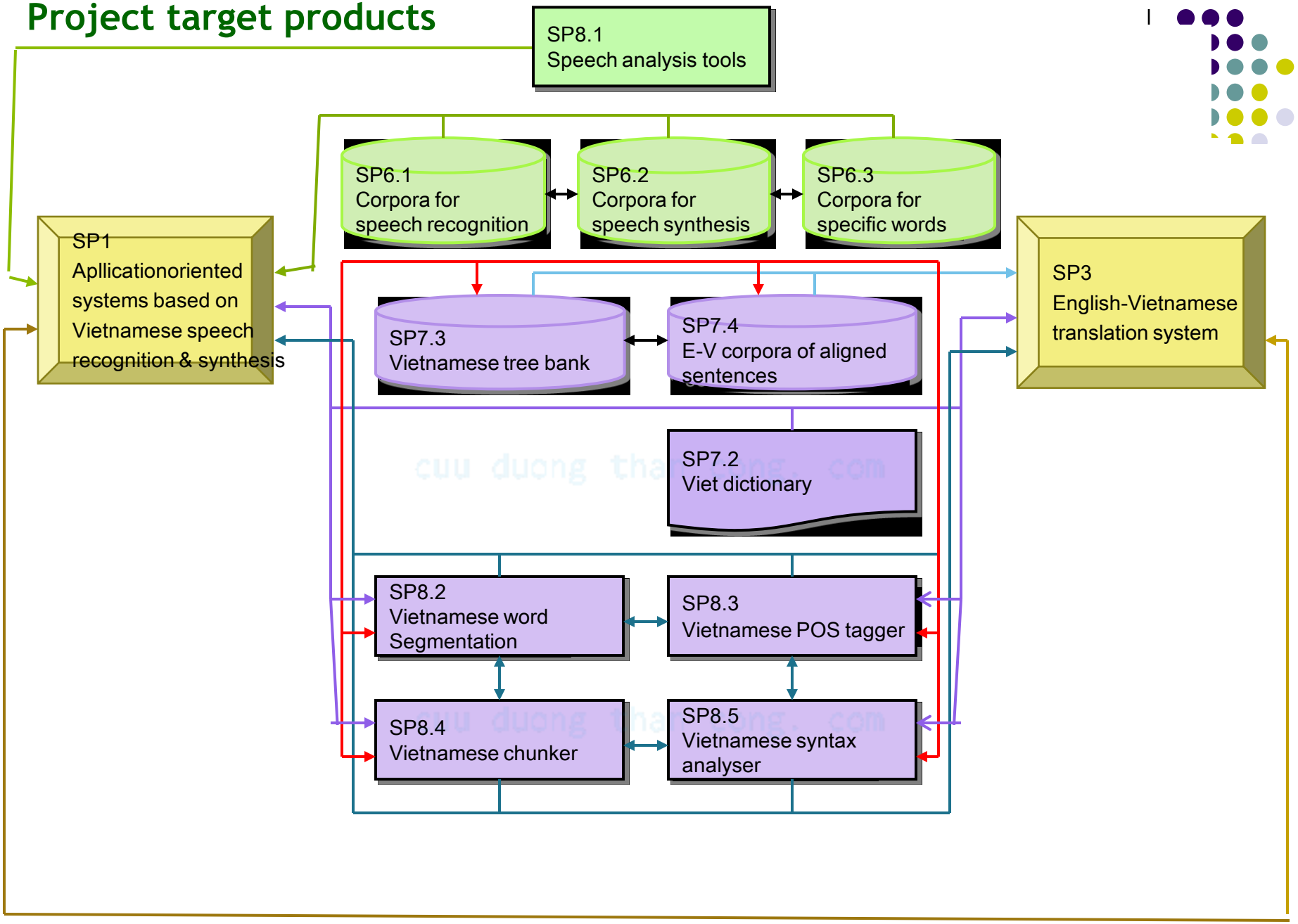
Vu Tran, Vinh Nguyen, Uyen Pham, Oanh Tran and Quang Thuy Ha (2009). An Experimental Study of Vietnamese Question Answering System, *International Conference on Asian Language Processing (IALP 2009)*: 152-155, Dec 7-9, 2009, Singapore, http://www.computer.org/portal/web/csdl/doi/10.1109/IALP.2009.39

Một số công cụ nguồn mở



- Chuyển từ trang Web sang văn bản
  - Bộ phân tích HTML (http://jexpert.us), Tác giả: Jose Solorzano
  - Một số công cụ tinh chế cho tiếng Việt (html2text.php, text2telex.php http://203.113.130.205/~cuongnv/thesis/code/tools.tar.gz). Tác giả: Nguyễn Việt Cường
- Một số bộ công cụ xử lý
  - Nhóm KPLD phát triển (PXHiếu, NCTú, NTTrang)
    - Bộ công cụ xử lý Text trên Java: **JtextPro** (http://jtextpro.sourceforge.net/) và **JwebPro** http://jwebpro.sourceforge.net/)
    - Phần mềm phân đoạn từ tiếng Việt: **JvnSegmenter** (http://jvnsegmenter.sourceforge.net/)
  - Sản phẩm tài nguyên và công cụ của Đề tài "*Nghiên cứu phát triển một số sản phẩm thiết yếu về xử lý tiếng nói và văn bản tiếng Việt*" mã số KC.01.01/06-10 do PGS, TS. Lương Chi Mai chủ trì.
    - http://vlsp.vietlp.org:8080/demo/?page=home
  - Một số tiện ích liên quan: http://vnlp.net/blog/?p=229 và http://vnlp.net/blog/wp-content/uploads/2010/08/Toolkits.pptx

Project target products



Chủ trì đề tài KC.01.01/06-10: Prof. Luong Chi Mai (IOIT), Prof. Ho Tu Bao (JAIST, IOIT)

BÀI GIẢNG KHAI PHÁ DỮ LIỆU WEB

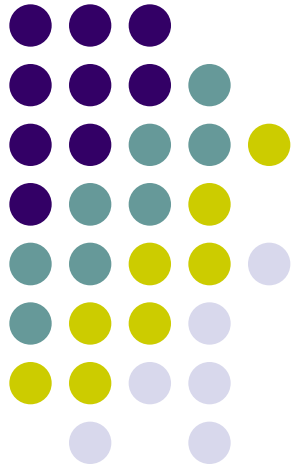
CHƯƠNG 5. BIỂU DIỄN WEB

PGS. TS. HÀ QUANG THỤY

HÀ NỘI 02-2011

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

ĐẠI HỌC QUỐC GIA HÀ NỘI



Nội dung

Giới thiệu

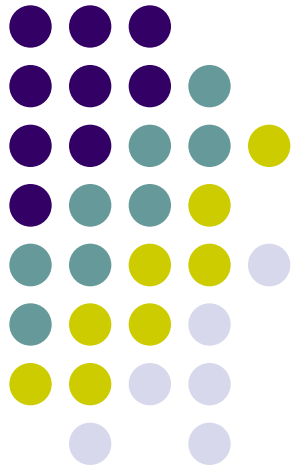
Phân tích văn bản

Biểu diễn Text

Lựa chọn đặc trưng

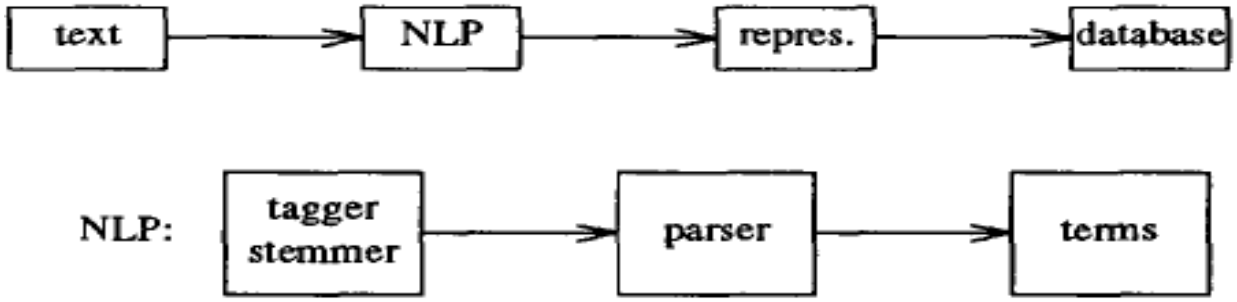
Thu gọn đặc trưng

Biểu diễn Web



Giới thiệu

- Biểu diễn văn bản
  - Là bước cần thiết đầu tiên trong xử lý văn bản
  - Phù hợp đầu vào của thuật toán khai phá dữ liệu
  - Tác động tới chất lượng kết quả của thuật toán KHDL
  - Thuật ngữ tiếng Anh: (document/text) (representation/indexing)
- Phạm vi tác động của một phương pháp biểu diễn văn bản
  - Không tồn tại phương pháp biểu diễn lý tưởng
  - Tồn tại một số phương pháp biểu diễn phổ biến
  - Chọn phương pháp biểu diễn phù hợp miền ứng dụng
- Một sơ đồ sơ lược: Tomek Strzalkowski: Document Representation in Natural Language Text Retrieval, *HLT 1994*: 364-369



Nghiên cứu về biểu diễn văn bản

- Nghiên cứu biểu diễn văn bản (Text + Web)
  - Luôn là nội dung nghiên cứu thời sự
  - Biểu diễn Web bổ sung một số yếu tố cho biểu diễn Text
- Số công trình liên quan
  - "Document representation"
    - mọi nơi: 8000 bài; tiêu đề: 200 (60 bài từ 2006-nay)
  - "Document indexing"
    - mọi nơi: 5200 bài; tiêu đề: 220 (60 bài từ 2006-nay)
  - "Text representation"
    - mọi nơi: 9200 bài; tiêu đề: 240 (60 bài từ 2006-nay)
  - "Text indexing"
    - mọi nơi: 6800 bài; tiêu đề: 210 (60 bài từ 2006-nay)

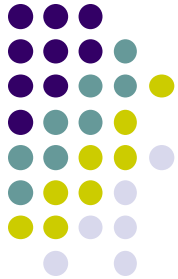
Ghi chú: các bài "ở mọi nơi" phần đông thuộc vào các bài toán xử lý văn bản bao gồm bước trình bày văn bản

Nghiên cứu về biểu diễn văn bản (2)

Research paper reference	Document Representation	Feature Selection	Learning algorithm
Apté et al. [6]	bag-of-words (freq)	stop list+ frequency	Decision Rules
Armstrong et al. [7]	bag-of-words	informativity	TFIDF Winnow, WordStat
Balabanović et al. [9]	bag-of-words (freq)	stop list+stemming+ keep 10 best words	TFIDF
Bartell et al. [11]	bag-of-words (freq)	latent semantic indexing using SVD	—
Berry et al. [12] Foltz and Dumais [28]	bag-of-words(freq)	latent semantic indexing using SVD	TFIDF
Cohen [21]	bag-of-words	infrequent words pruned	Decision Rules ILP
Joachims [40]	bag-of-words (freq)	infrequent words+ informativity	TFIDF, PrTFIDF, Naive Bayes
Lam et al. [60]	bag-of-words (freq)	mutual info.	Bayesian Network
Lewis et al. [66]	bag-of-words	log likelihood ratio	logistic regression with Naive Bayes
Maes [69]	bag-of-words+ header info.	mail/news header + selecting keywords	Memory-Based reasoning
Pazzani et al. [83, 84]	bag-of-words	stop list+ informativity	TFIDF, Naive Bayes, Nearest Neighbor, Neural Network, Decision Trees
Sorensen and Mc Elligott [97, 25]	n-gram graph (only bigrams)	weighting graph edges	connectionist combined with Genetic Algorithms
Yang [100]	bag-of-words	stop list	adapted k-Nearest Neighbor

Dunja Mladenic' (1998). Machine Learning on Non-homogeneous, Distributed Text Data. *PhD. Thesis*, University of Ljubljana, Slovenia.

Phân tích văn bản



- Mục đích biểu diễn văn bản (Keen, 1977 [Lew91])
  - Từ được chọn liên quan tới chủ đề người dùng quan tâm
  - Gắn kết các từ, các chủ đề liên quan để phân biệt được từ ở các lĩnh vực khác nhau
  - Dự đoán được độ liên quan của từ với yêu cầu người dùng, với lĩnh vực và chuyên ngành cụ thể
- Môi trường biểu diễn văn bản (đánh chỉ số)
  - Thủ công / từ động hóa. Thủ công vẫn có hỗ trợ của công cụ máy tinh và phần mềm
  - Điều khiển: chọn lọc từ làm đặc trưng (feature) biểu diễn) / không điều khiển: mọi từ đều được chọn.
  - Từ điển dùng để đánh chỉ số. Từ đơn và tổ hợp từ.

Luật Zipt

- Luật Zipt
  - **Cho dãy dữ liệu được xếp hạng** x_1 x_2 … x_n

thì hạng tuân theo công thức

$$x_{(r)} = \frac{C}{r^{\alpha}}$$

C là hằng số, gần 1; kỳ vọng dạng loga

$$E(\log x_{(r)}) = c - \alpha \log(r)$$

  - Dạng hàm mật độ:

$$p(x) = \frac{C^{1/\alpha}}{\alpha n} \frac{1}{x^{(1/\alpha)+1}} = \frac{A}{x^{\beta}}$$

- Một số dạng khác
  - Phân phối Yule

$$x_{(r)} = \frac{C}{r^{\alpha} B^{r}}$$

  - Mô hình thống kê

$$E(\log x_{(r)}) = c - \alpha \log(r) - be^{\log(r)}$$

c=log(C), b= log(B)

  - Biến thể loga-chuẩn

$$E(\log x_{(r)}) = c - \alpha \log(r) - b(\log(r))^2$$

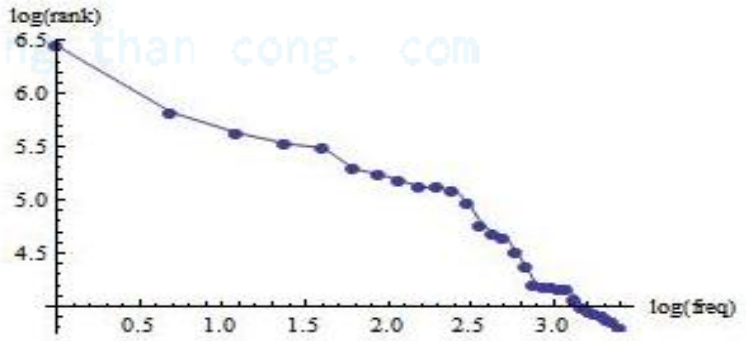
  - Phân phối Weibull với 0< <1

$$E(\log x_{(r)}) = c - \alpha \log(r) - be^{\beta \log(r)}$$

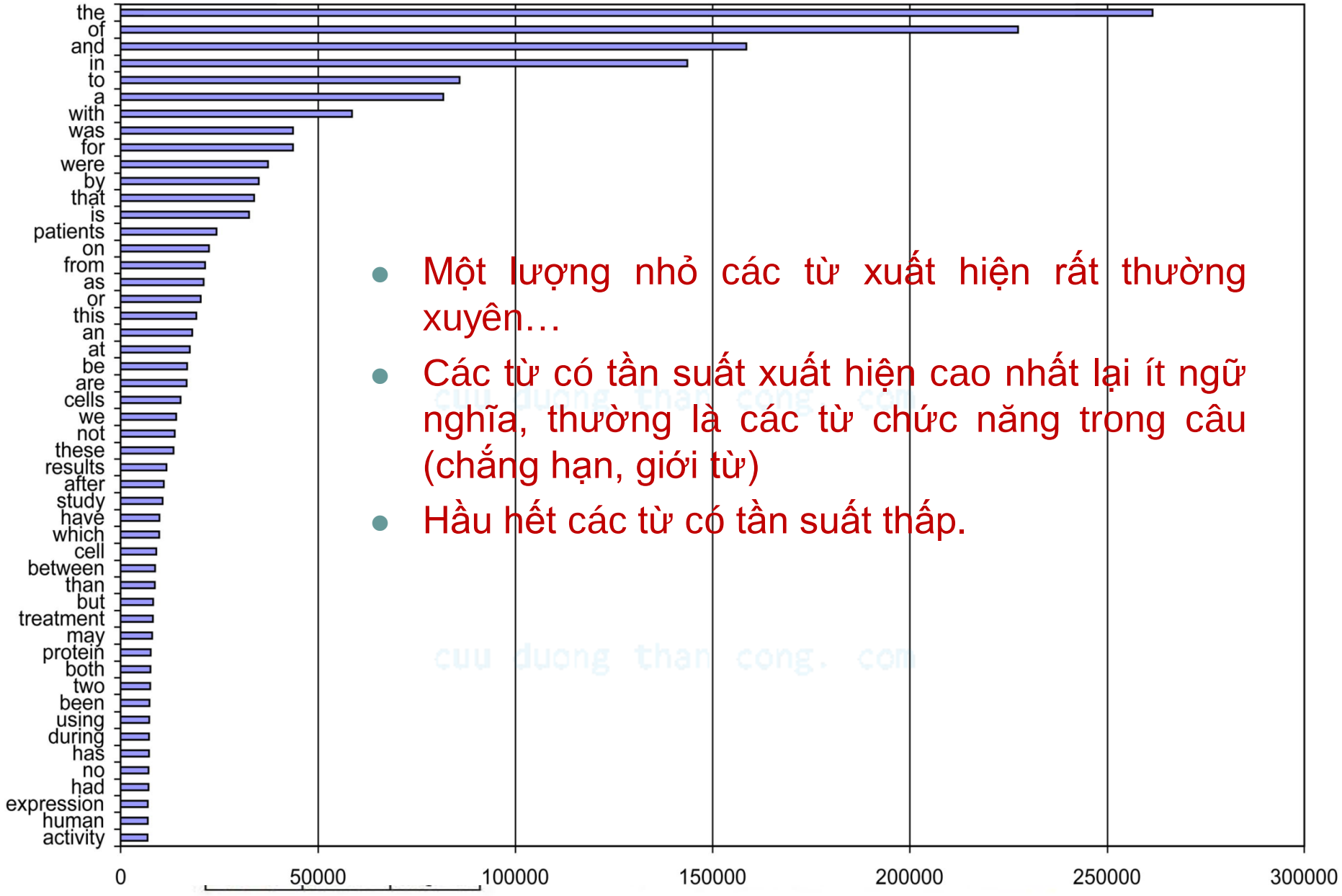
Luật Zipt trong phân tích văn bản

- Trọng số của từ trong biểu diễn văn bản (Luhn, 1958)
  - Dấu hiệu nhấn mạnh: một biểu hiện của độ quan trọng
    - thường viết lặp lại các từ nhất định khi phát triển ý tưởng
    - hoặc trình bày các lập luận,
    - phân tích các khía cạnh của chủ đề. …
  - Các từ có tần suất xuất hiện cao nhất lại ít ngữ nghĩa. Từ xuất hiện trung bình lại có độ liên quan cao.
- Luật Zipt
  - Là một quan sát hiện tượng mà không phải là luật thực sự: xem hình vẽ "Alice ở xứ sở mặt trời"
  - $r_t * f_t = K$ (hằng số): r_t : độ quan trọng của từ t; f_t: tần số xuất hiện từ t. Có thể logarith

the 632	and 338	a 278
to 252	she 242	of 199
it 189	i 178	was 167
alice 167	in 163	said 144
you 118	her 108	that 105
as 91	at 79	with 67
s 66	had 65	all 64
on 64	little 59	out 54
down 52	this 51	t 50
for 48	but 47	they 45



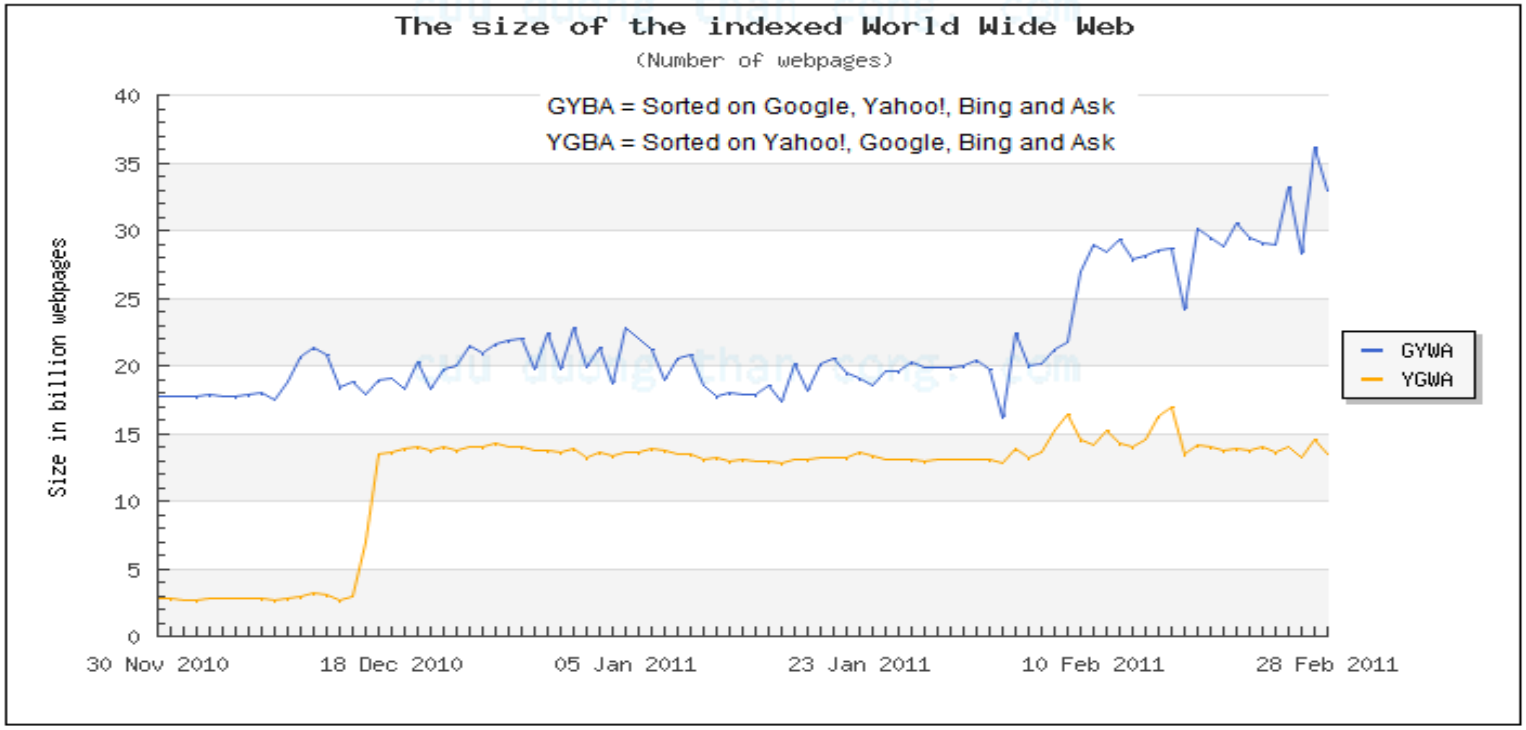
Luật Zipt trong tiếng Anh



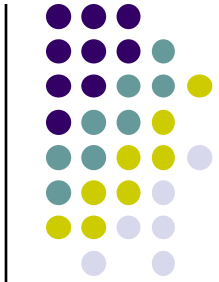
- Một lượng nhỏ các từ xuất hiện rất thường xuyên…
- Các từ có tần suất xuất hiện cao nhất lại ít ngữ nghĩa, thường là các từ chức năng trong câu (chẳng hạn, giới từ)
- Hầu hết các từ có tần suất thấp.

Luật Zipt: ước lượng trang web được chỉ số

- Ước lượng tối thiểu lượng trang web chỉ số hóa
  - http://www.worldwidewebsize.com/
  - Luật Zipt: từ kho ngữ liệu DMOZ có hơn 1 triệu trang web
  - Dùng luật Zipt để ước tính lượng trang web chỉ số hóa.
  - Mỗi ngày: 50 từ (đều ở đoạn logarith luật Zipt) gửi tới 4 máy tìm kiếm Google, Bing, Yahoo Search và Ask.
  - Trừ bớt phần giao ước tính giữa các công cụ tìm kiếm: làm già
  - Thứ tự trừ bớt phần giao → tổng (được làm non)



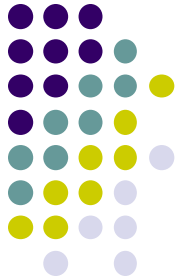
Các mẫu luật Zipt khác



- Dân số thành phố
  - Dân số thành phố trong một quốc gia: có = 1. Đã xác nhận ở 20 quốc gia.
  - Có thể mở rộng sang: dân cư khu đô thị, vùng lãnh thổ
- Lượt thăm trang web và mẫu giao vận Internet khác
  - Số lượt truy nhập trang web/tháng
  - Các hành vi giao vận Internet khác
- Quy mô công ty và một số số liêu kinh tế khác
  - Xếp hạng công ty theo: số nhân viên, lợi nhuận, thị trường
  - Các hành vi giao vận Internet khác
- …

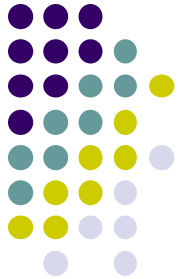
[Li02] Wentian Li (2002). Zipf's Law Everywhere, *Glottometrics* ***5*** *(2002)*: 14-21

Phương pháp lựa chọn từ Luhn58



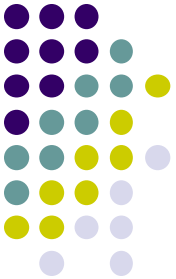
- Bài toán
  - Input: Cho một tập văn bản: có thể coi tất cả các văn bản trong miền ứng dụng; ngưỡng trên, ngưỡng dưới dương.
  - Output: Tập từ được dùng để biểu diễn văn bản trong tập
- Giải pháp
  - Tính tần số xuất hiện mỗi từ đơn nhất trong từng văn bản
  - Tính tần số xuất hiện của các từ trong tập toàn bộ văn bản
  - Sắp xếp các từ theo tần số giảm dần
  - Loại bỏ các từ có tần số xuất hiện vượt quá ngưỡng trên hoặc nhỏ thua ngưỡng dưới.
  - Các từ còn lại được dùng để biểu diễn văn bản
  - "Từ" được mở rộng thành "đặc trưng": n-gram, chủ đề..
- Lưu ý
  - *Chọn ngưỡng*: ngưỡng cố định, ngưỡng được điều khiển
  - Liên hệ vấn đề chọn lựa đặc trưng (mục sau).

Phương pháp đánh trọng số của từ



- Bài toán
  - *Input*: Cho một tập văn bản miền ứng dụng D và tập từ được chọn biểu diễn văn bản V (sau bước trước đây).
  - *Output*: Đánh trọng số từ trong mỗi văn bản ⇨ Xây dựng ma trận $\{w_{i,j}\}$ là trọng số của từ w_i V trong văn bản d_j D.
- Giải pháp
  - Một số phương pháp điển hình
  - Boolean
  - dựa theo tần số xuất hiện từ khóa
  - Dựa theo nghịch đảo tần số xuất hiện trong các văn bản
- Phương pháp Boolean
  - Đơn giản: trọng số là xuất hiện/ không xuất hiện
  - $w_{i,j} = 1$ nếu w_i xuất hiện trong văn bản d_j, ngược lại $w_{i,j} = 0$.

Các phương pháp đánh trọng số của từ theo tần số



- Dạng đơn giản: TF
  - $w_{i,j} = f_{i,j}$: trong đó $f_{i,j}$ là số lần từ khóa w_i xuất hiện trong văn bản d_j
- Một số phiên bản khác của dạng đơn giản
  - Cân đối số lần xuất hiện các từ khóa: giảm chênh lệch số lần xuất hiện
  - Giảm theo hàm căn $w_{i,j} = \sqrt{tf_{ij}}$
  - Tránh giá trị "0" và giảm theo hàm loga: $w_{i,j} = 1+\log(f_{i,j})$
- Nghịch đảo tần số xuất hiện trong tập văn bản: IDF
  - Từ xuất hiện trong nhiều văn bản thì trọng số trong 1 văn bản sẽ thấp
  - $w_i = \log(\frac{m}{df_i})$ $\log(m)$ $\log(df_i)$

Trong đó m = |D|, df_i là |d D: w_i xuất hiện trong d}

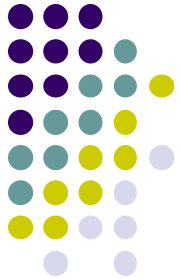
Phương pháp TFIDF

- Tích hợp TF và IDF
  - Dạng đơn giản: $w_{i,j} = f_{i,j} * df_i / m$
  - Dạng căn chỉnh theo hàm loga

$$w_{i,j} \quad \begin{array}{ll} (1 \quad \log(tf_{ij}))\log(\frac{m}{df_i}) & : tf_{ij} \quad 0 \\ 0 & : tf_{ij} \quad 0 \end{array}$$

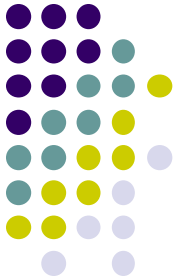
  - Ngoài ra, có một số dạng tích hợp trung gian khác

Mô hình biểu diễn văn bản



- Bài toán
  - *Input*: Cho tập văn bản miền ứng dụng D = $\{d_j\}$, tập đặc trưng được chọn biểu diễn văn bản V = $\{w_i\}$, ma trân trọng số W = $(w_{i,j})$.
  - *Output*: Tìm biểu diễn của các văn bản d_j D.
- Một số mô hình
  - Mô hình Boolean
  - Mô hình không gian vector
  - Mô hình túi các từ (Mô hình xác suất)
  - Các mô hình khác
- Mô hình Boolean
  - Tập các từ thuộc V mà xuất hiện trong văn bản

Mô hình không gian vector



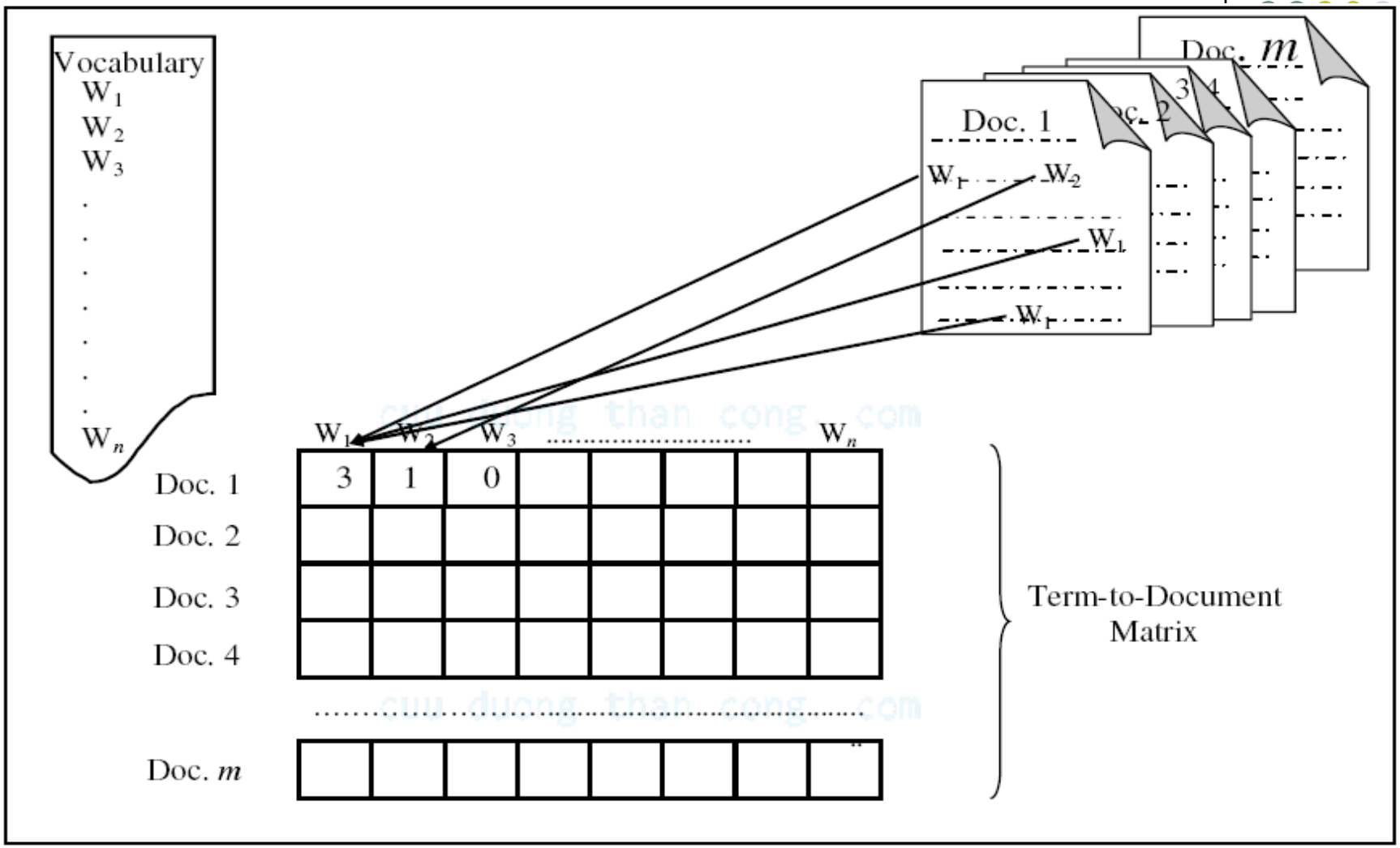
- Nội dung chính
  - Ánh xạ tập tài liệu vào không gian vector n =|V| chiều.
  - Mỗi tài liệu được ánh xạ thành 1 vector

$$d_i \Leftrightarrow (w_{i1}, w_{i2}, \ldots, w_{in})$$

- Độ đo tương tự nội dung văn bản
  - Chuẩn hóa vector: đưa về độ dài 1
    - Độ "tương tự nội dung" giữa hai văn bản ⇔ độ tương tự giữa hai vector
    - Một số phương án sơ khai "các thành phần giống nhau", "nghịch đảo khoảng cách", ..
  - Phổ biến là tính độ đo cosin của góc giữa hai vector: không yêu cầu chuẩn hóa

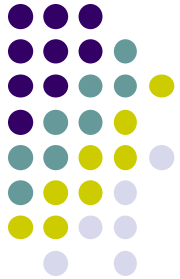
$$sim(d_1, d_2) \quad \frac{(v_1, v_2)}{\|v_1\|\|v_2\|} \quad \frac{\sum_{i=1}^{n} w_{1i} * w_{12}}{\sqrt{\sum_{i=1}^{n} w_{1i}^2} * \sqrt{\sum_{i=1}^{n} w_{2i}^2}}$$

Mô hình không gian vector



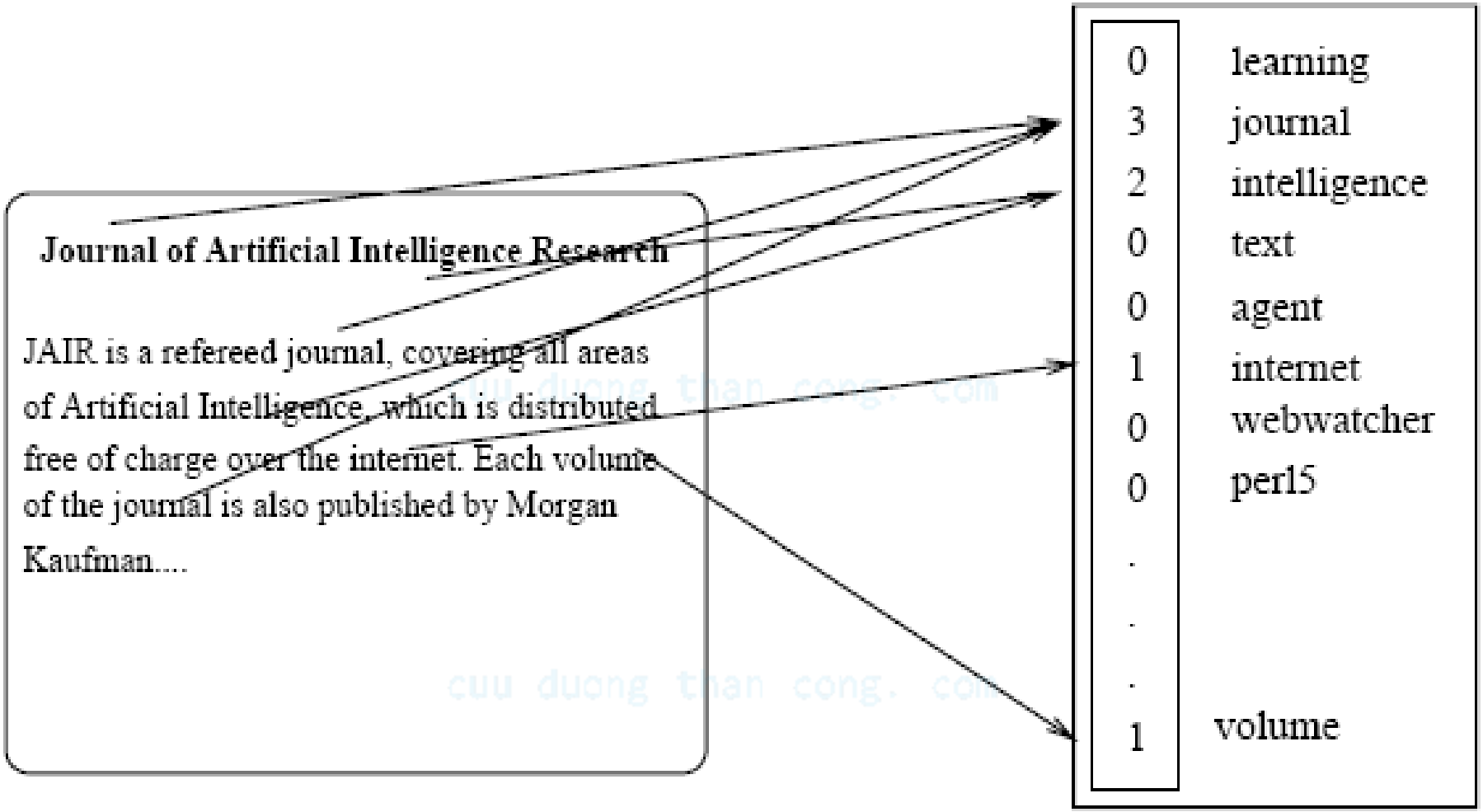
Khaled Shaban (2006). A semantic graph model for text representation and matching in document mining, *PhD Thesis*, University of Waterloo, Canada

Mô hình xác suất



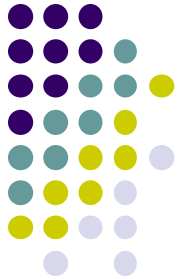
- Giả thiết chính
  - Mô hình xác suất: cặp (Y, P) với Y là tập quan sát được và P là mô hình xác suất trên Y (có thể coi Y là quan sát được các từ/đặc trưng trên văn bản).
  - Các từ xuất hiện trong văn bản thể hiện nội dung văn bản
  - Sự xuất hiện của các từ là độc lập lẫn nhau và độc lập ngữ cảnh
  - Dạng đơn giản: chỉ liệt kê từ, dạng chi tiết: liệt kê từ và số lần xuất hiện
  - *Lưu ý*: Các giả thiết về tính độc lập không hoàn toàn đúng (độc lập lẫn nhau, độc lập ngữ cảnh) song mô hình thi hành hiệu quả trong nhiều trường hợp.
- Độ đo tương tự nội dung văn bản
  - So sánh hai túi từ

Mô hình túi từ (bag-of-word)



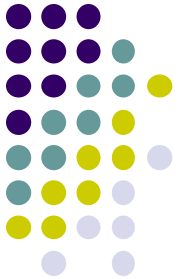
Dunja Mladenic' (1998). Machine Learning on Non-homogeneous, Distributed Text Data. *PhD. Thesis*, University of Ljubljana, Slovenia.

Mô hình biểu diễn LSI và theo phân cụm



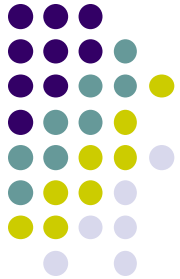
- Giới thiệu
  - Tồn tại nhiều phương pháp biểu diễn khác
  - Tồn tại nhiều phiên bản cho một phương pháp
  - Gần đây có một số phương pháp mới
  - Hai phương pháp phổ biến: LSI và theo phân cụm
  - Lưu ý: Giá phải trả khi *tiền xử lý dữ liệu*
- Mô hình phân cụm
  - Phân cụm các từ trong miền ứng dụng: ma trận trọng số
  - Thay thếtừ bằng cụm chứa nó
- Mô hình biểu diễn LSI
  - LSI: Latent Semantic Indexing biểu diễn ngữ nghĩa ẩn
    - Nâng mức ngữ nghĩa (trừu tượng) của đặc trưng
    - Rút gọn tập đặc trưng, giảm số chiều không gian biểu diễn
    - Không gian từ khóa ⇨ không gian khái niệm (chủ đề).
  - Phương pháp chuyển đổi
    - Ma trận trọng số ⇨ ma trận hạng nhỏ hơn
    - Phép biến đổi đó Từ khóa ⇨ khái niệm. Thay thế biểu diễn.

Lựa chọn từ trong biểu diễn văn bản



- Loại bỏ từ dừng
  - Những từ được coi là không mạng nghĩa
  - Có sẵn trong ngôn ngữ
- Đưa về từ gốc
  - Các ngôn ngữ có biến dạng từ: Anh, Nga…
  - Thay từ biến dạng về dạng gốc
- Chon đặc trưng n-gram
  - Các âm tiết liền nhau n-gram
    - Uni-gram: chỉ chứa một âm tiết
    - Bigram: chứa không quá 2 âm tiết
    - Trigram: chứa không quá 2 âm tiết
    - N-gram: Thường không quá 4 gram
  - Một số đặc trưng
    - Chính xác hơn về ngữ nghĩa
    - Tăng số lượng đặc trưng
    - Tăng độ phức tạp tính toán

Một số đô đo cho lựa chọn đặc trưng



- Giới thiệu chung
  - Lựa chọn đặc trưng: lợi thế chính xác, lợi thể tốc độ hoặc cả hai
  - Các độ đo giúp khẳng định lợi thế
- Phân nhóm độ đo
  - Hai nhóm: theo tần số và theo lý thuyết thông tin
- Một số độ đo điển hình
  - Xem hai trang sau

Một số đô đo cho lựa chọn đặc trưng

$P(t_k, c_i)$ kí hiệu là xác suất của từ t_k có trong chủ đề c_i và $P(t_k, \bar{c}_i)$ là xác suất của từ t_k không có trong chủ đề c_i.

1. DIA (Darmstadt Indexing Approach – Tiếp cận đánh chỉ số Darmstadt): Được đề xuất bởi Fuhn và đồng nghiệp [FHK91].

$$f(t_k, c_i) = z(t_k, c_i) = P(c_i|t_k)$$

2. Độ đo IG (Information Gain).

$$f(t_k, c_i) = IG(t_k, c_i) = \sum_{c \in \{c_i, \bar{c}_i\}} \sum_{t \in \{t_k, \bar{t}_k\}} P(t, c) \cdot \log \frac{P(t, c)}{P(t) \cdot P(c)}$$

3. Độ đo thông tin tương hỗ (mutual information).

$$f(t_k, c_i) = \log \frac{P(t_k, c_i)}{P(t_k) \cdot P(c_i)}$$

4. Độ đo Khi -bình phương (Chi-square).

$$f(t_k, c_i) = \chi^2(t_k, c_i) = \frac{|Tr| \cdot [P(t_k, c_i) \cdot P(\bar{t}_k, \bar{c}_i) - P(t_k, \bar{c}_i \cdot P(\bar{t}_k, c_i))]^2}{P(t_k) \cdot P(\bar{t}_k) \cdot P(c_i) \cdot P(\bar{c}_i)}$$

5. Độ đo liên quan (Relevancy score).

$$f(t_k, c_i) = RS(t_k, c_i) = \log \frac{P(t_k|\bar{c}_i) + d}{P(\bar{t}_k|\bar{c}_i) + d}$$

6. Tỷ lệ dư (Odd Ratio).

$$f(t_k, c_i) = OR(t_k, c_i) = \frac{P(t_k|c_i) \cdot (1 - P(t_k|\bar{c}_i))}{(1 - P(t_k|c_i)) \cdot P(t_k, |\bar{c}_i)}$$

Một số đô đo cho toàn bộ các lớp

Các độ đo trên là tính cho từng lớp. Độ đo cho toàn bộ các lớp trong tập hợp có thể được tính theo nhiều cách khác nhau,

$$f(t_k) = \sum_{i=1}^{|C|} f(t_k, c_i)$$

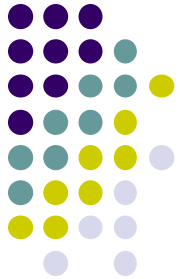
hoặc

$$f(t_k) = \sum_{i=1}^{|C|} P(c_i) f(t_k, c_i)$$

hoặc

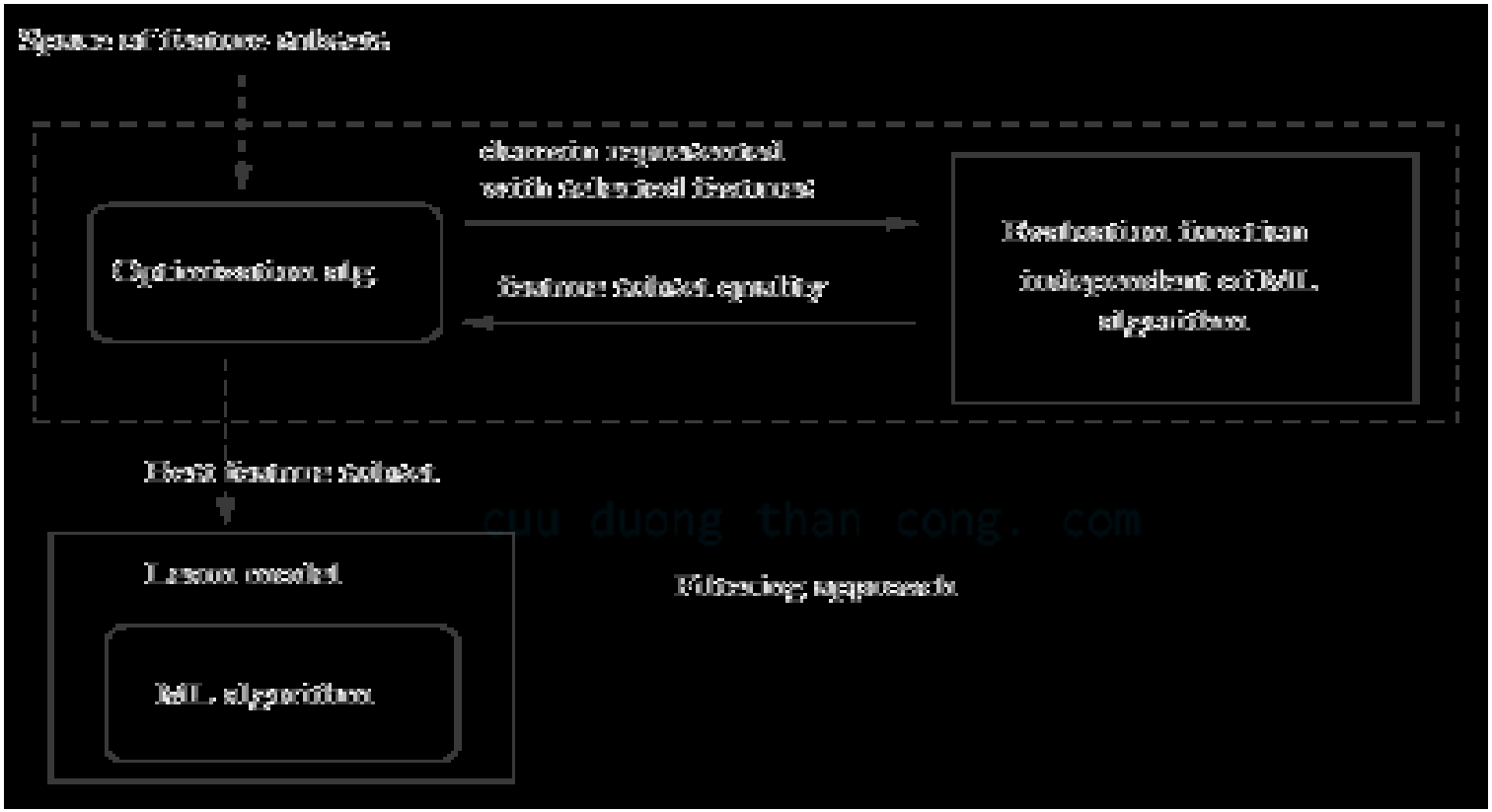
$$f(t_k) = \max_{i=1}^{|C|} f(t_k, c_i).$$

Thu gọn đặc trưng



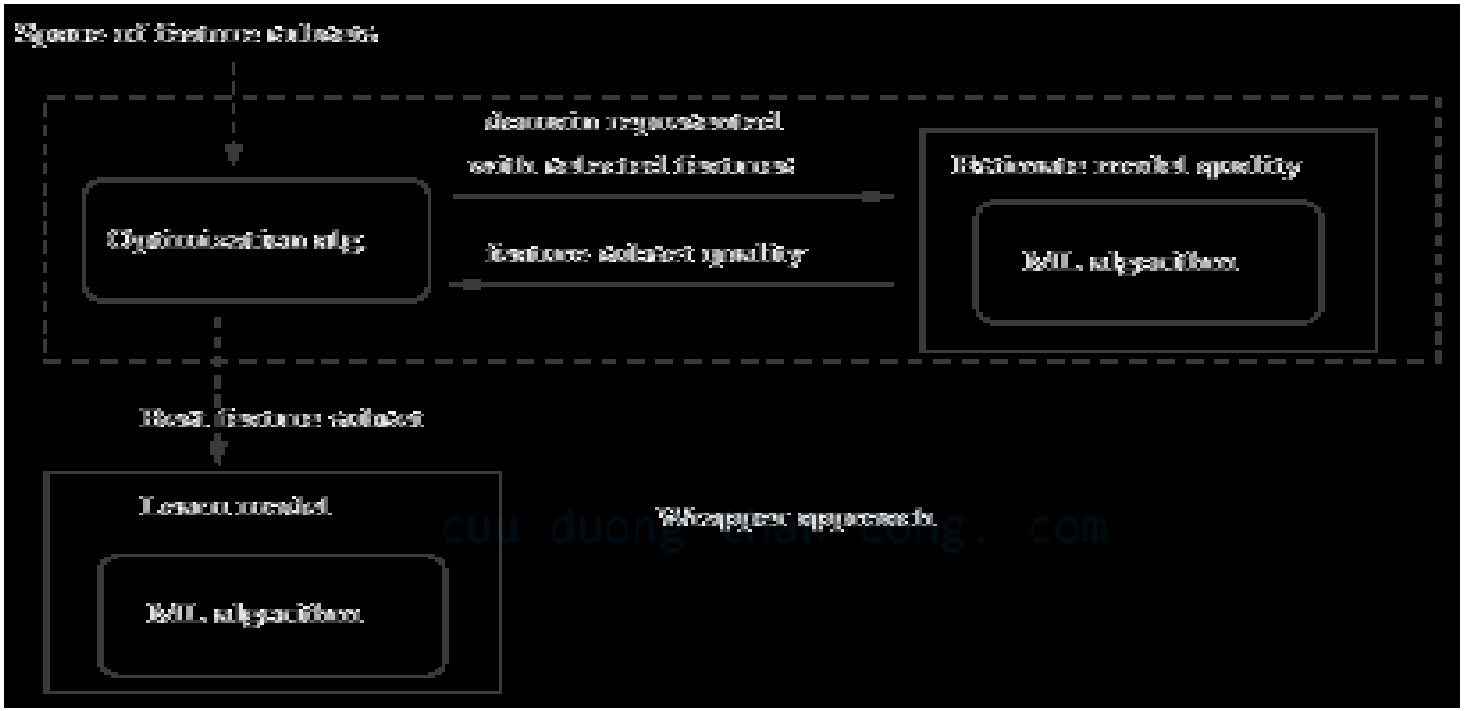
- Giới thiệu chung
  - "Tối ưu hóa" chọn tập đặc trưng
    - Số lượng đặc trưng nhỏ hơn
    - Hy vọng tăng tốc độ thi hành
    - Tăng cường chất lượng khai phá văn bản. ? Giảm đặc trưng đi là tăng chất lượng: có các đặc trưng "nhiễu"
    - Hoặc cả hai mục tiêu trên
- Hai tiếp cận điển hình
  - Tiếp cận lọc
  - Tiếp cận bao gói
- Với dữ liệu văn bản
  - Tập đặc trưng: thường theo mô hình vector
  - Tính giá trị của từng đặc trưng giữ lại các đặc trưng được coi là "tốt".

Tiếp cận tổng quát: lọc



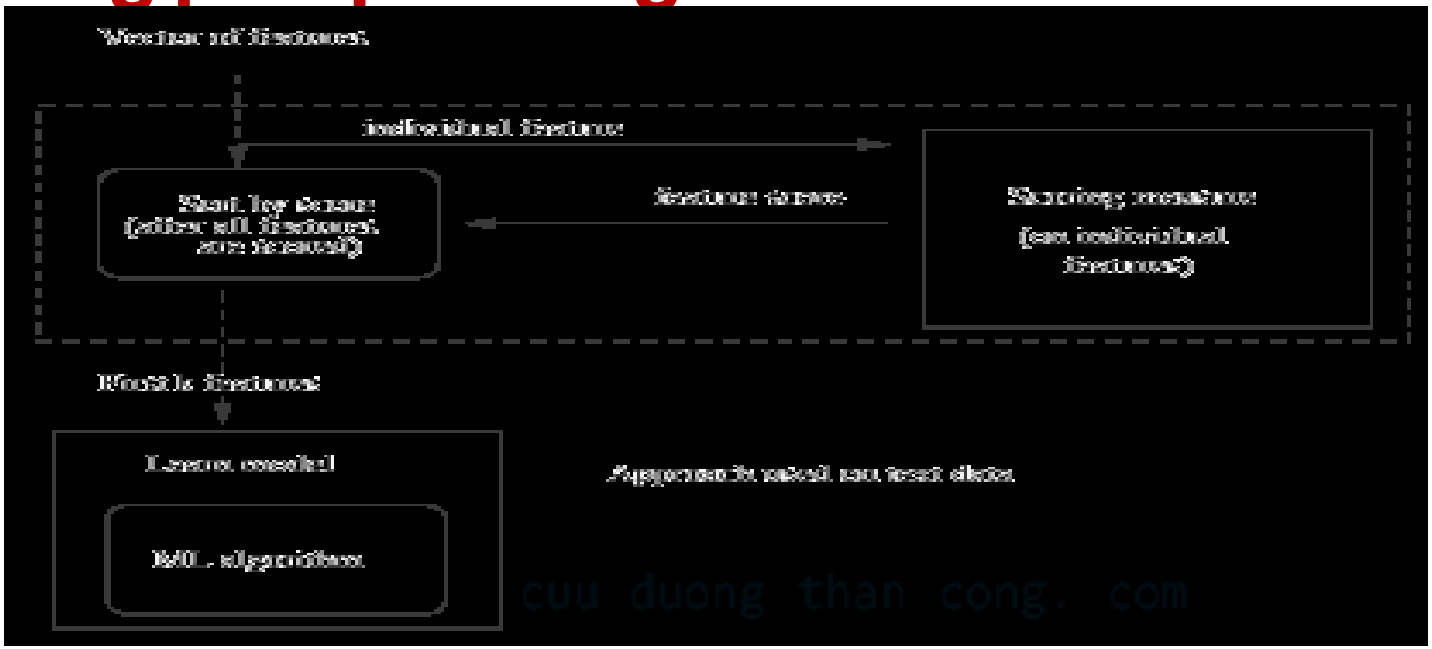
- Tiếp cận lọc
  - Đầu vào: Không gian tập các tập đặc trưng
  - Đầu ra: Tập con đặc trưng tốt nhất
  - Phương pháp
    - Dò tìm "cải tiến" bộ đặc trưng: Thuật toán tối ưu hóa
    - Đánh giá chất lượng mô hình: độc lập với thuật toán học máy

Tiếp cận bao gói tổng quát



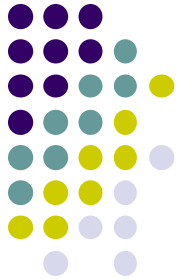
- Tiếp cận bao gói
  - Đầu vào: Không gian tập các tập đặc trưng
  - Đầu ra: Tập con đặc trưng tốt nhất
  - Phương pháp
    - Dò tìm "cải tiến" bộ đặc trưng: Thuật toán tối ưu hóa
    - Đánh giá chất lượng mô hình: Dùng chính thuật toán học để đánh giá

Thu gọn đặc trưng văn bản text



- Thu gọn đặc trưng
  - Đầu vào: Vector đặc trưng
  - Đầu ra: k đặc trưng tốt nhất
  - Phương pháp (lùi)
    - Sắp xếp các đặc trưng theo độ "tốt" (để loại bỏ bớt)
    - Tính lại độ "tốt" của các đặc trưng
    - Chọn ra k-đặc trưng tốt nhất
- Các kiểu phương pháp
  - Tiến / Tiến bậc thang (có xem xét thay thế khi tiến)
  - Lùi / Lùi bậc thang (có xem xét thay thế khi lùi)

Thu gọn đặc trưng phân lớp text nhị phân



```
SELECTFEATURES(𝔻, c, k)
1  V ← EXTRACTVOCABULARY(𝔻)
2  L ← []
3  for each t ∈ V
4  do A(t, c) ← COMPUTEFEATUREUTILITY(𝔻, t, c)
5     APPEND(L, ⟨A(t, c), t⟩)
6  return FEATURESWITHLARGESTVALUES(L, k)
```

- Một thuật toán lựa chọn đặc trưng text
  - V: Bảng từ vựng có được từ tập văn bản D
  - c: lớp đang được quan tâm
  - giá trị A(t,c): một trong ba thủ tục tính toán
- Ba kiểu thủ tục tính toán A(t,c)
  - Thông tin tương hỗ
  - Lựa chọn đặc trưng theo khi-bình phương (chi-square)
  - Lựa chọn đặc trưng theo tần suất

Thu gọn đặc trưng: thông tin tương hỗ

$$I(U;C) = \sum_{e_t \in \{1,0\}} \sum_{e_c \in \{1,0\}} P(U = e_t, C = e_c) \log_2 \frac{P(U = e_t, C = e_c)}{P(U = e_t)P(C = e_c)}$$

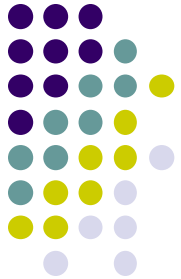
$$I(U;C) = \frac{N_{11}}{N} \log_2 \frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N} \log_2 \frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N} \log_2 \frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N} \log_2 \frac{NN_{00}}{N_{0.}N_{.0}}$$

- Công thức MI (Mutual Information)
  - Biến ngẫu nhiên U: từ khóa t *xuất hiện*/*không xuất hiện*
  - Biến ngẫu nhiên c: tài liệu *thuộc*/*không thuộc* lớp c
  - Ước lượng cho MI
- Ví dụ: Bộ dữ liệu Reuter-RCV1
  - Lớp poultry, từ khóa export

	$e_t = e_{\text{export}} = 1$	$e_t = e_{\text{export}} = 0$
$e_c = e_{\text{poultry}} = 1$	$N_{11} = 49$	$N_{01} = 141$
$e_c = e_{\text{poultry}} = 0$	$N_{10} = 27{,}652$	$N_{00} = 774{,}106$

$$I(U;C) = \frac{49}{801{,}948} \log_2 \frac{801{,}948 \cdot 49}{(49+27{,}652)(49+141)} + \frac{141}{801{,}948} \log_2 \frac{801{,}948 \cdot 141}{(141+774{,}106)(49+141)}$$
$$+ \frac{27{,}652}{801{,}948} \log_2 \frac{801{,}948 \cdot 27{,}652}{(49+27{,}652)(27{,}652+774{,}106)} + \frac{774{,}106}{801{,}948} \log_2 \frac{801{,}948 \cdot 774{,}106}{(141+774{,}106)(27{,}652+774{,}106)}$$
$$\approx 0.000105$$

10 đặc trưng tốt nhất cho 6 lớp



UK

london	0.1925
uk	0.0755
british	0.0596
stg	0.0555
britain	0.0469
plc	0.0357
england	0.0238
pence	0.0212
pounds	0.0149
english	0.0126

China

china	0.0997
chinese	0.0523
beijing	0.0444
yuan	0.0344
shanghai	0.0292
hong	0.0198
kong	0.0195
xinhua	0.0155
province	0.0117
taiwan	0.0108

poultry

poultry	0.0013
meat	0.0008
chicken	0.0006
agriculture	0.0005
avian	0.0004
broiler	0.0003
veterinary	0.0003
birds	0.0003
inspection	0.0003
pathogenic	0.0003

coffee

coffee	0.0111
bags	0.0042
growers	0.0025
kg	0.0019
colombia	0.0018
brazil	0.0016
export	0.0014
exporters	0.0013
exports	0.0013
crop	0.0012

elections

election	0.0519
elections	0.0342
polls	0.0339
voters	0.0315
party	0.0303
vote	0.0299
poll	0.0225
candidate	0.0202
campaign	0.0202
democratic	0.0198

sports

soccer	0.0681
cup	0.0515
match	0.0441
matches	0.0408
played	0.0388
league	0.0386
beat	0.0301
game	0.0299
games	0.0284
team	0.0264

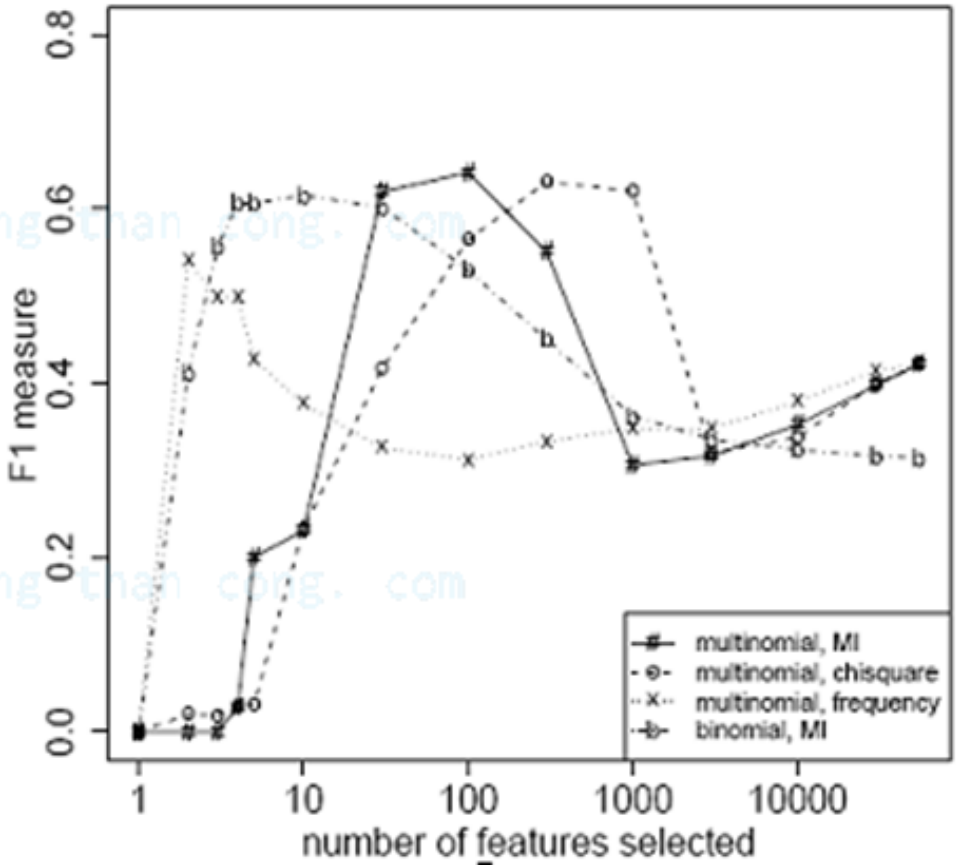
Bộ dữ liệu Reuter-RCV1

Thống kê khi-bình phương và tần số

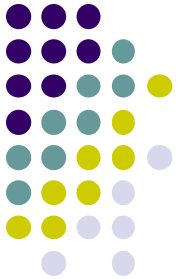
$$X^2(\mathbb{D},t,c) = \sum_{e_t\in\{0,1\}}\sum_{e_c\in\{0,1\}} \frac{(N_{e_te_c} - E_{e_te_c})^2}{E_{e_te_c}}$$

$$X^2(\mathbb{D},t,c) = \frac{(N_{11}+N_{10}+N_{01}+N_{00}) \times (N_{11}N_{00} - N_{10}N_{01})^2}{(N_{11}+N_{01}) \times (N_{11}+N_{10}) \times (N_{10}+N_{00}) \times (N_{01}+N_{00})}$$

- Thống kê khi-bình phương
  - Công thức xác suất: e_t, e_c : như MI, các biến E là kỳ vọng, N là tần số quan sát được từ tập tài liệu D
  - Ước lượng cho MI: các giá trị N như MI
- Tần số
  - Một ước lượng xác suất



Thu gọn đặc trưng phân lớp text đa lớp



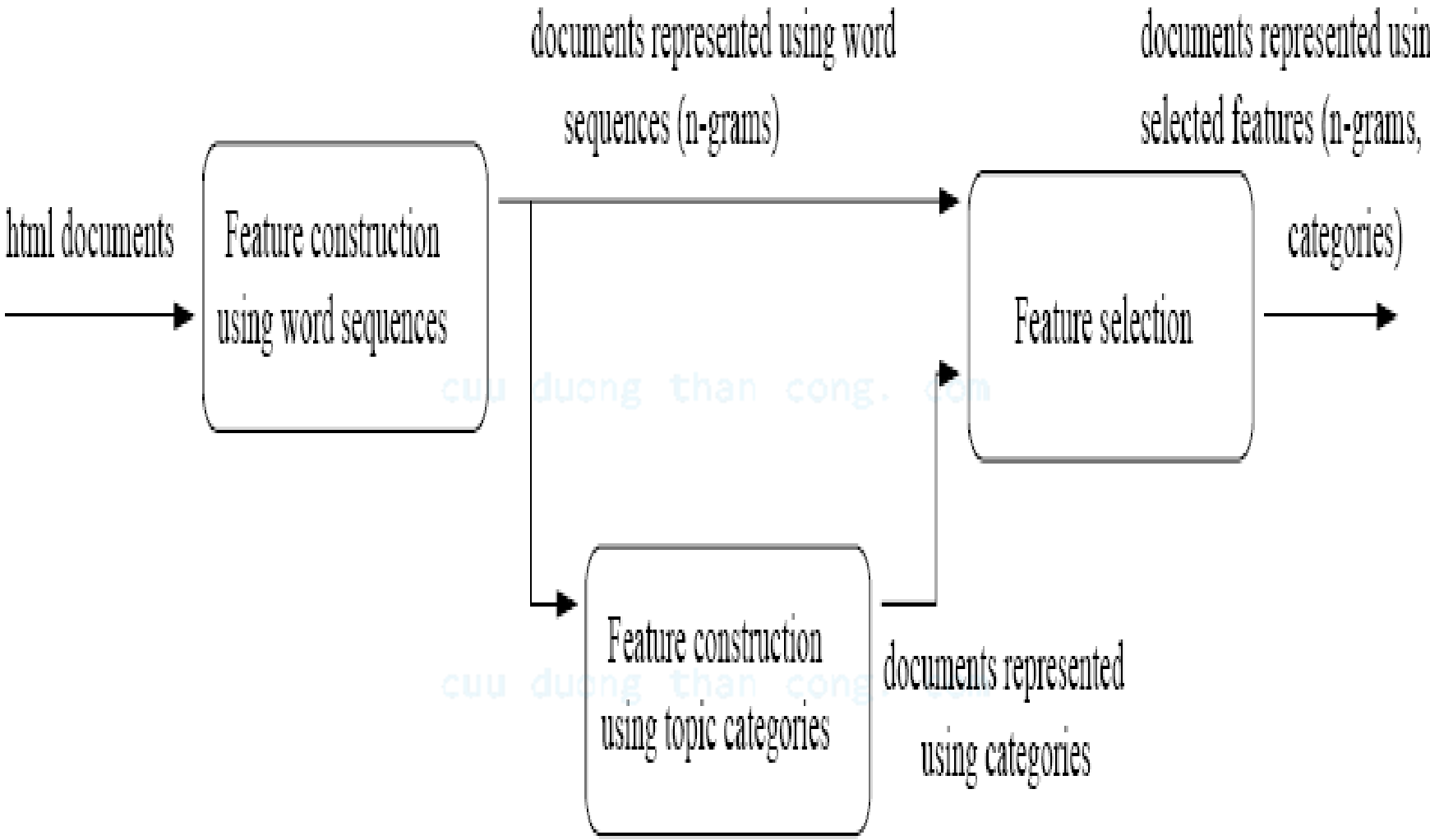
- Bài toán phân lớp đa lớp
  - Tập C = {c_1, c_2, …, c_n)
  - Cần chọ đặc trưng tốt nhất cho bộ phân lớp đa lớp
- Phương pháp thống kê khi-bình phương
  - Mỗi từ khóa
    - Lập bảng xuất hiện/không xuất hiện các đặc trưng trong lớp văn bản
    - Tính giá trị thống kê khi-bình phương
  - Chọn k đặc trưng (từ khóa)
- Phương pháp lựa chon từng lớp
  - Tính bộ đặc trưng tốt cho từng phân lớp thành phần
  - Kết hợp các bộ đặc trưng tốt
    - Tính toán giá trị kết hợp: trung bình (có trọng số xuất hiện) khi kết hợp
    - Chọn k-đặc trưng tốt nhất sau khi tính toán kết hợp

Biểu diễn Web

- Đồ thị Web
  - Web có cấu trúc đồ thị
    - Đồ thị Web: nút ⇔ trang Web, liên kết ngoài ⇔ cung (có hướng, vô hướng).
    - Bản thân trang Web cũng có tính cấu trúc cây (đồ thị)
  - Một vài bài toán đồ thị Web
    - Biểu diễn nội dung, cấu trúc
    - Tính hạng các đối tượng trong đồ thị Web: tính hạng trang, tính hạng cung..

  Nghiên cứu về đồ thị Web (xem trang sau)
- Đồ thị ngẫu nhiên
  - Tính ngẫu nhiên trong khai phá Web
    - WWW có tính ngẫu nhiên: mới, chỉnh sửa, loại bỏ
    - Hoạt động con người trên Web cũng có tính ngẫu nhiên
  - Là nội dung nghiên cứu thời sự

Một sơ đồ biểu diễn tài liệu Web



Dunja Mladenic' (1998). Machine Learning on Non-homogeneous, Distributed Text Data. *PhD. Thesis*, University of Ljubljana, Slovenia.

Một sơ đồ biểu diễn tài liệu Web

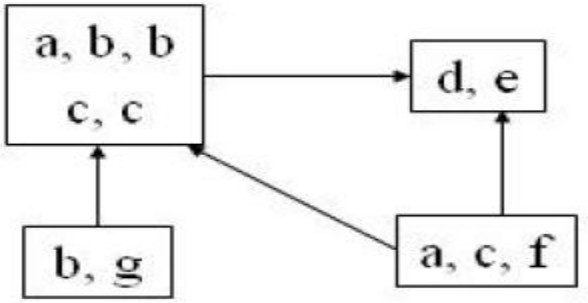
Các biểu diễn vector trang Web

Phương pháp 1:

a	b	c	d	e	f	g
1	2	2	0	0	0	0

Phương pháp 2:

a	b	c	d	e	f	g
2	3	3	1	1	1	1



Phương pháp 3:

Đoạn 1							Đoạn 2						
a	b	c	d	e	f	g	a	b	c	d	e	f	g
1	2	2	0	0	0	0	1	1	1	1	1	1	1

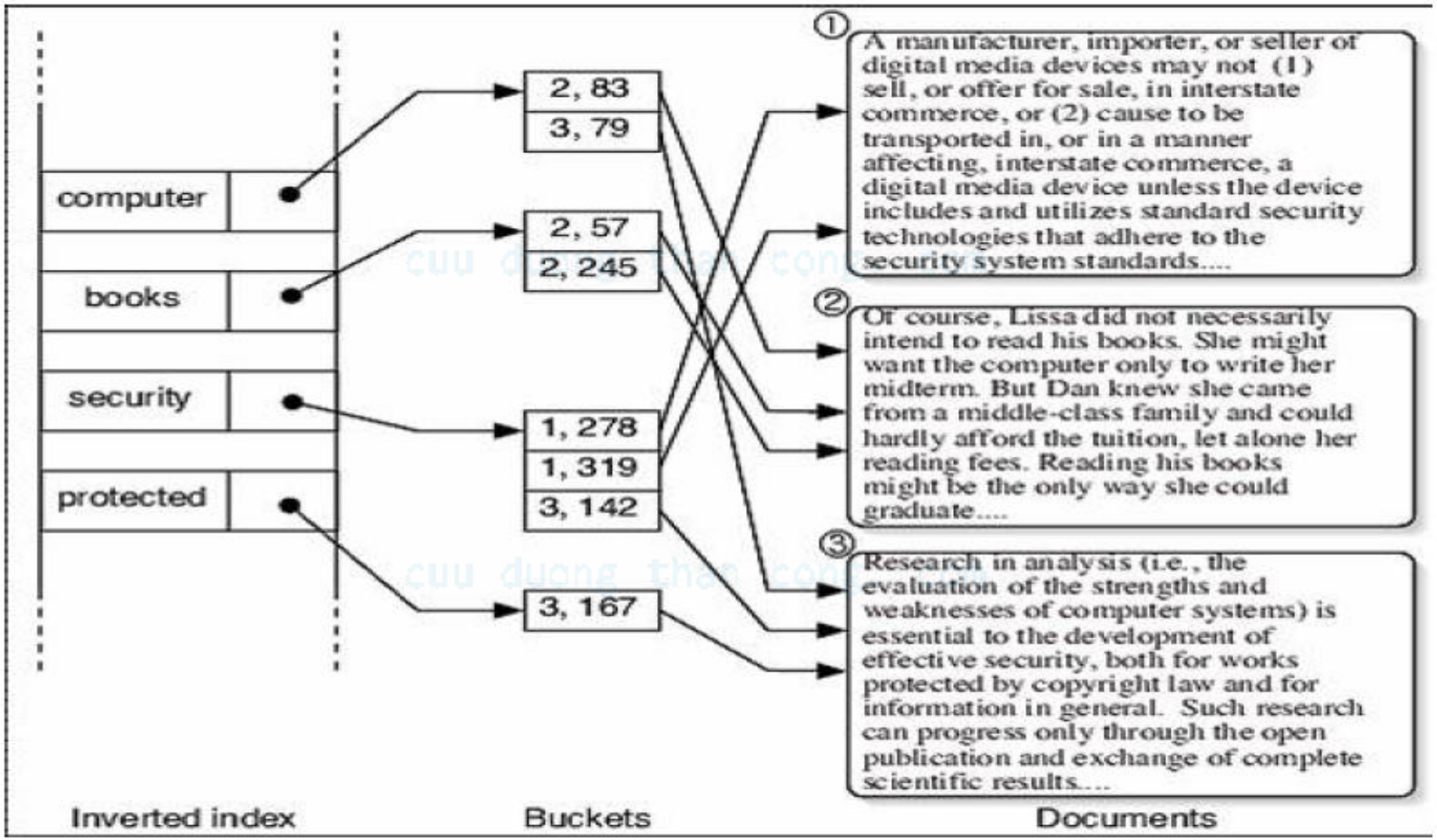
Phương pháp 4:

Đoạn1							Đoạn 2							Đoạn 3							Đoạn 4						
a	b	c	d	e	f	g	a	b	c	d	e	f	g	a	b	c	d	e	f	g	a	b	c	d	e	f	g
1	2	2	0	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	1	0	0	1	0	0	0	0	1
1	2	2	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	1	0	0

Dunja Mladenic' (1998). Machine Learning on Non-homogeneous, Distributed Text Data. *PhD. Thesis*, University of Ljubljana, Slovenia.

Một sơ đồ biểu diễn tài liệu Web

Máy tìm kiếm từ khóa nhanh: Hệ thống chỉ số ngược (Inverted Index)



KHAI PHÁ WEB
CHƯƠNG 6. TÌM KIẾM WEB

Giảng viên: Hà Quang Thụy
email: thuyhq@coltech.vnu.vn

Hà Nội, 11-2010

CHƯƠNG 6. TÌM KIẾM VĂN BẢN VÀ MÁY TÌM KIẾM

- Bài toán tìm kiếm văn bản
  - Khái niệm
  - Đánh giá
  - Tìm kiếm xấp xỉ
- Máy tìm kiếm
  - Công cụ tìm kiếm trên Internet
  - Một số máy tìm kiếm điển hình
  - Các thành phần cơ bản
  - Crawling
  - Đánh chỉ số và lưu trữ
  - Tính hạng và tìm kiếm

CHƯƠNG 6. TÌM KIẾM VĂN BẢN VÀ MÁY TÌM KIẾM

- Máy tìm kiếm thực thể
  - Khái niệm
  - Một số nội dung cơ bản
  - Một số nghiên cứu tìm kiếm thực thể
- Máy tìm kiếm ở Việt Nam

6.1. BÀI TOÁN TÌM KIẾM VĂN BẢN

- Nguồn tài nguyên
  - D = {d_i: các văn bản}
  - cho trước: trong CSDL
  - văn bản web trên Internet: cần thu thập về (máy tìm kiếm)
- Đầu vào
  - q: Câu hỏi người dùng (q D)
  - Từ khóa/ Cụm từ khóa/ "Biểu thức" hỏi
- Kết quả
  - Tập R (q) các văn bản thuộc D "liên quan" tới câu hỏi q
  - "liên quan": ngầm định một ánh xạ {q} 2^D
  - Hệ thống tìm kiếm "xấp xỉ" ánh xạ nói trên

6.1. BÀI TOÁN TÌM KIẾM VĂN BẢN

- Lời giải
  - q: hệ thống cho tập R'(q) xấp xỉ R(q)
  - Đánh giá hệ thống: đối sánh R'(q) với R(q)
  - R chưa biết Đánh giá qua các ví dụ đã có
  - Học ánh xạ R': xấp xỉ R cho hệ thống
- Phân loại tìm kiếm
  - Tìm kiếm theo lựa chọn (Document Selection)
  - Tìm kiếm theo tính hạng liên quan (Document Ranking)
  - Kết hợp cả lựa chọn lẫn ranking

TÌM KIẾM THEO LỰA CHỌN

- Học hàm f (d, q): D D {0,1}
  - Chọn/Không chọn
  - Thực tiễn: Module tìm kiếm của hệ thống.
  - Ngôn ngữ hỏi và "ngữ nghĩa" cho từng câu hỏi
  - câu hỏi q: Câu trả lời là R'(q)={d| f(d,q)=1}
- Ví dụ
  - hệ thống thư viện điện tử Greenstone
  - hệ thống tài liệu điện tử CiteSeer: http://citeseer.ist.psu.edu/
- Nhận xét
  - Đơn giản, dễ thực hiện
  - Hạn chế
    - Câu hỏi q "quá phổ dụng": kết quả có rất nhiều văn bản
    - Câu hỏi q "quá chuyên biệt": rất ít hoặc không có văn bản

TÌM KIẾM THEO TÍNH HẠNG

- Học hàm (mô hình) f (d, q): D D [0,1]
  - "Liên quan": Độ gần nhau giữa các tài liệu, hạng
  - Hạng tính trước, hạng với câu hỏi
- câu hỏi q: Câu trả lời là R'(q)={d| f(d,q) }
  - Hệ thống có ngưỡng >0
- Yêu cầu học
  - f (d, q) cần thỏa tính đơn điệu: d_1 "liên quan" tới q nhiều hơn d_2 thì $f(d,q_1)$ $f(d,q_2)$
  - Kiểm nghiệm: công nhận tương đối
- Ví dụ
  - Máy tìm kiếm
- Nhận xét
  - Mềm dẻo, khắc phục hạn chế của lựa chọn

BÀI TOÁN HỌC (NHẮC LẠI)

- Có sẵn tập ví dụ học D_E D
  - d D_E đã biết R(d) D
- Thuật toán học
  1. Chia ngẫu nhiên tập D_E thành hai tập D_{learn} và D_{test} , $|D_{test}|$ $|D_{learn}|/2$.
  2. Dùng D_{learn} học mô hình (xác định tham số)
  3. Dùng D_{test} đánh giá mô hình
  4. Kiểm tra điều kiện kết thúc: chưa kết thúc về 1
     - Thông thường kết thúc ngay
- Sử dụng đánh giá chéo (cross validation)
  - thông qua k lần thực hiện quá trình trên: Kết hợp đánh giá k lần.

ĐÁNH GIÁ MÔ HÌNH TÌM KIẾM

- Giải thích ký hiệu
  - R, R' liên quan đến các văn bản trong D_{test}
    - R: tập đúng hoàn toàn, R' là tập hệ thống cho là đúng
- Độ hồi phục (recall)
- Độ chính xác (precision)
- Độ đo F và độ đo F_1 . Độ đo F là tổng quát còn F_1 là thông dụng.

$$\frac{|R \cap R'|}{|R|} \qquad \pi = \frac{|R \cap R'|}{|R'|} \qquad f_\beta = \frac{(\beta^2+1)\pi\rho}{\beta^2\pi+\rho} \qquad f_1 = \frac{2\pi\rho}{\pi+\rho}$$

TÌM KIẾM XẤP XỈ

- Đặt vấn đề
  - Tính xấp xỉ trong ngôn ngữ tự nhiên: từ đồng nghĩa, từ gần nghĩa, phù hợp ngữ cảnh
  - Tính xấp xỉ trong biểu diễn văn bản
    - Biểu diễn vectơ: cô đọng, tiện lợi xử lý song tính ngữ nghĩa kém bỏ đi nhiều thứ (chẳng hạn, vị trí xuất hiện của các từ khóa)
    - Biểu diễn "xâu các từ": có ngữ nghĩa cao hơn song lưu trữ và xử lý phức tạp, bỏ đi một số yêu tố ngữ nghĩa (từ dừng...)
  - Vấn đề tìm kiếm xấp xỉ là vấn đề tự nhiên
- Độ hồi phục (recall)
- Độ chính xác (precision)
- Độ đo F và độ đo F_1 . Độ đo F là tổng quát còn F_1 là thông dụng.

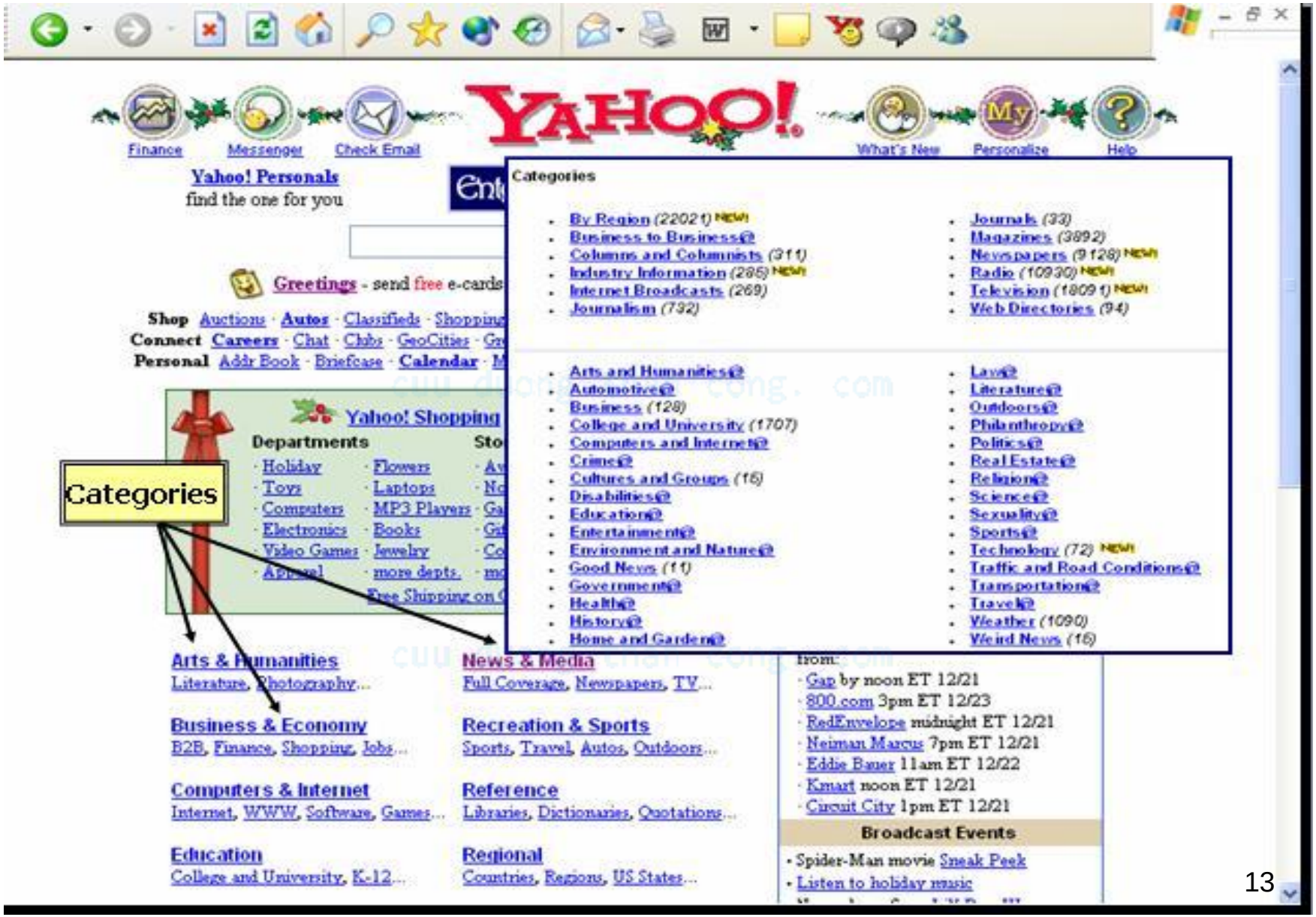
6.2. MÁY TÌM KIẾM

- Công cụ tìm kiếm trên Internet
- Một số máy tìm kiếm điển hình
- Một số đặc trưng và xu thế phát triển
- Các thành phần cơ bản
- Crawling
- Đánh chỉ số và lưu trữ
- Tính hạng và tìm kiếm

CÔNG CỤ TÌM KIẾM TRÊN INTERNET

- Hai kiểu công cụ tìm kiếm điển hình
  - Máy tìm kiếm (search engine)
  - Thư mục phân lớp (classified directory)
- Thư mục phân lớp
  - số lượng ít tài liệu Web
  - tổ chức dạng thư mục
  - tìm kiếm theo thư mục
  - kết quả danh sách theo thư mục.
  - Lycos, Yahoo, CiteSeer ... thư mục phân lớp điển hình.
- Kết hợp thư mục phân lớp vào máy tìm kiếm
  - AltaVista: có các dịch vụ catalog; Lycos: trộn dịch vụ vào chức năng.
  - Northern Light: có dịch vụ tìm kiếm tổ chức động kết quả của tìm theo từ khóa thành nhóm theo chủ đề tương tự hoặc nguồn/kiểu.

THƯ MỤC PHÂN LỚP: YAHOO



THƯ MỤC PHÂN LỚP: CiteSeer

http://citeseer.ist.psu.edu/directory.html

CiteSeer.IST
Scientific Literature Digital Library

Computer Science Directory

Agents Architecture Assistant Agent.. BDI Mobile Agents ...
Applications Face Recognitio.. Financial Predi.. Speech Recognit..
Architecture Clusters Distributed Arc.. Parallel
Artificial Intelligence Expert Systems Knowledge Repre.. Natural Languag.. Optimization ...
Compression Audio Text Video
Databases Concurrency Data Warehousin.. Deductive Object-oriented ...
Hardware CISC High Performanc.. Logic Design Memory Structur.. ...
Human Computer Interaction Collaboration Graphics Interface Desig.. Multimedia ...
Information Retrieval Classification Digital Librari.. Extraction Filtering ...
Machine Learning Case-based Lear.. Fuzzy Systems Genetic Algorit.. Neural Networks ...
Networking ATM Internet Local Area Multicast ...
Operating Systems Clusters Distributed Fault Tolerance Linux ...
Programming Compiler Design Compiler Optimi.. Functional Java ...
Security Access Control Encryption Information War.. Intellectual Pr.. ...
Software Engineering Data Structures Parallelism Randomized Algo..
Theory Computational C.. Formal Language.. Logic Quantum Computi.. ...
World Wide Web Agents Electronic Comm.. Metasearch Search Engines ...

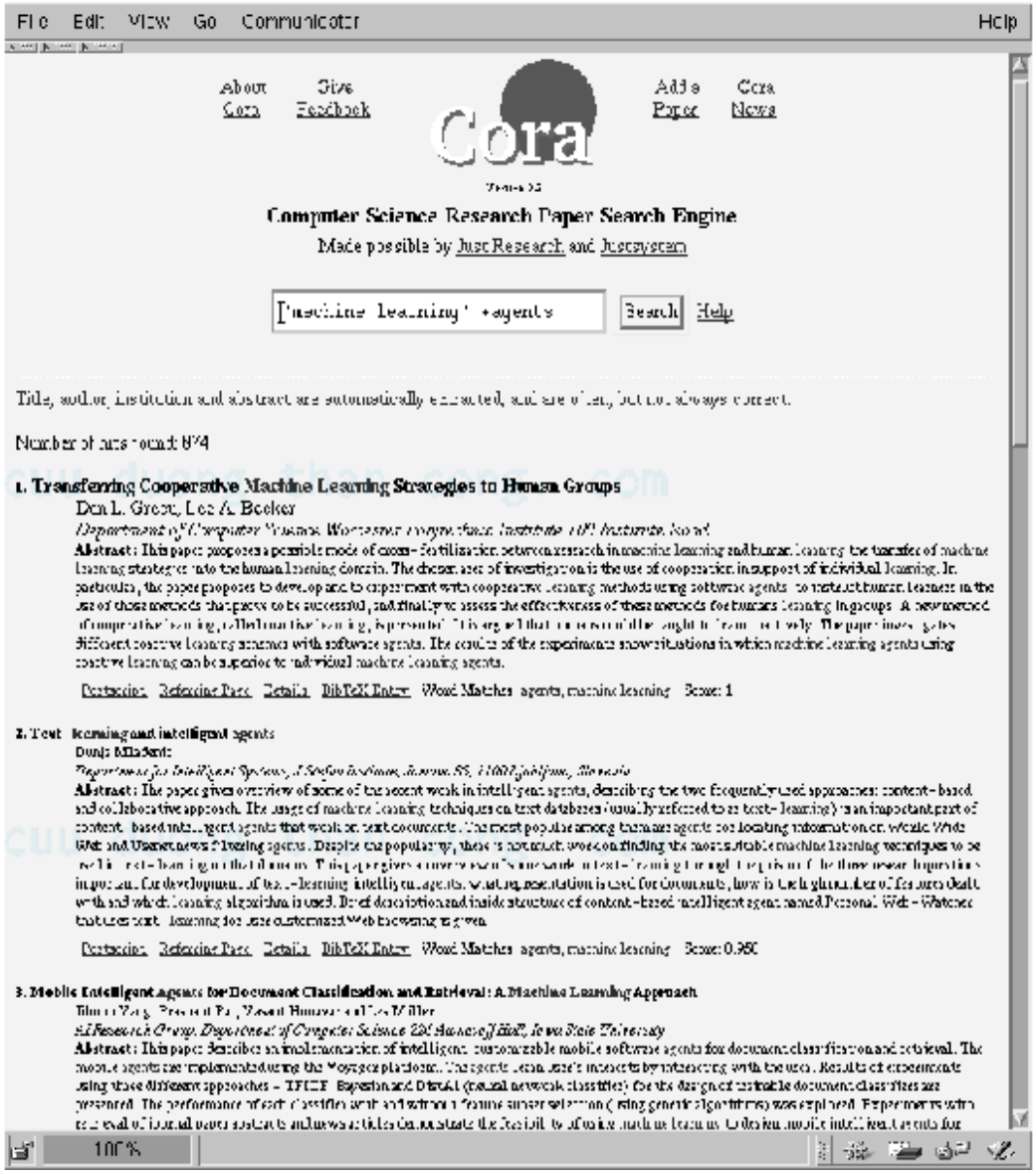
CiteSeer: Thư mục phân lớp & hệ tìm kiếm

http://citeseer.ist.psu.edu/

CÔNG CỤ TÌM KIẾM TRÊN INTERNET

- Máy tìm kiếm
  - Có trước tập lớn các tài liệu Web
  - Tìm kiếm dựa theo từ khóa
  - Kết quả: danh sách tài liệu theo tập xếp hạng
- Hạn chế
  - số lượng từ khóa ít, danh sách kết quả dài, ngữ nghĩa kém.
- Phân loại
  - Máy tìm kiếm chung
    - độ chính xác thấp
    - AltaVista, Hotbot, Infoseek
  - Dịch vụ tìm kiếm
    - Miền thu hẹp
    - Chính xác cao
    - Inktomi, Excite, www.netpart.com, Cora

MÁY TÌM KIẾM CORA



SƠ BỘ QUÁ TRÌNH PHÁT TRIỂN MÁY TÌM KIẾM

- 1994
  - Máy tìm kiếm đầu tiên WWWW (WWW Worm)
  - McBryan
  - Index chừng 110.000 trang web
  - 3/1994-4/1994: nhận 1500 câu hỏi hàng ngày
- 1997 (khi xuất hiện Google)
  - WebCrawler: 2 triệu ->Watch 100 triệu trang web
  - Alta Vista nhận 20 triệu câu hỏi / ngày
- 2000-nay
  - Tăng nhanh về số lượng
  - hàng tỷ trang web
  - hàng trăm triệu câu hỏi / ngày

MÁY TÌM KIẾM ALTA VISTA

- Hệ thống
  - Một module tìm kiếm
  - Log câu hỏi
- Module tìm kiếm
  - Mô hình viector có trọng số
  - Ngôn ngữ hỏi: hai mode hỏi
    - Đơn giản: từ khóa/dãy từ khóa (hoặc phép toán OR)/-word (tài liệu không chứa word -phép toán NOT)/+word : tài liệu chứa cả word/"dãy từ": tài liệu chứa dãy từ có thứ tự chặt như câu hỏi.
    - mở rộng : phép toán lôgic ***and***, ***or***, ***not*** thực hiện theo tài liệu; phép toán ***near*** các từ lân cận không chặt như “”. Cho chức năng đặt câu hỏi theo "vết".
    - Kết quả: Hiện 10 URL / 1 trang, theo thứ tự "hạng". Mỗi URL có tiêu đề và một số thông tin khác.

MÁY TÌM KIẾM ALTA VISTA

- Log câu hỏi
  - Mục tiêu: Hướng người dùng (Khai phá yêu cầu sử dụng)
  - Log câu hỏi gồm file text và một số thành phần khác
- File text
  - Câu hỏi mới
  - Màn hình kết quả từ yêu cầu đã gửi
  - Câu hỏi
    - tem thời gian được gửi (đơn vị mili giây từ 01/01/1970)
    - cookie: có không hai câu hỏi từ cùng một người dùng
    - tem các số hạng được gửi đi
    - màn hình kết quả
    - các biến dạng từ người dùng: ngày/khoảng ngày
    - dạng câu hỏi: đơn giản/mở rộng
    - trình duyệt, địa chỉ IP
- Các khái niệm phiên, tập dữ liệu log

SƠ BỘ MÁY TÌM KIẾM GOOGLE

- Tên gọi và tác giả
  - tên gọi: chơi chữ 10^{100}: máy tìm kiếm lớn
  - từ năm 1997
  - Sergey Brin và Lawren Page: hai nghiên cứu sinh Stanfort

- Một số thông số
  - Định hướng người dùng: có log câu hỏi
  - Yêu cầu
    - crawling nhanh: thu thập tài liệu web và cập nhật vào kho
    - Hệ thống lưu trữ hiệu quả: chỉ số và chính tài liệu
    - Hệ thống index: hàng trăm gigabyte dữ liệu hiệu quả
    - Hỏi/đáp nhanh: trăm nghìn câu hỏi / giây.

SƠ BỘ MÁY TÌM KIẾM GOOGLE

Bài viết đầu tiên về Google (năm 1998)

File: brin98anatomy.pdf

Sergey Brin received his B.S. degree in mathematics and computer science from the University of Maryland at College Park in 1993. Currently, he is a Ph.D. candidate in computer science at Stanford University where he received his M.S. in 1995. He is a recipient of a National Science Foundation Graduate Fellowship. His research interests include search engines, information extraction from unstructured sources, and data mining of large text collections and scientific data.

Lawrence Page was born in East Lansing, Michigan, and received a B.S.E. in Computer Engineering at the University of Michigan Ann Arbor in 1995. He is currently a Ph.D. candidate in Computer Science at Stanford University. Some of his research interests include the link structure of the web, human computer interaction, search engines, scalability of information access interfaces, and personal data mining.

The Anatomy of a Large-Scale Hypertextual Web Search Engine

Sergey Brin and Lawrence Page

Computer Science Department,

Stanford University, Stanford, CA 94305, USA

SƠ BỘ MÁY TÌM KIẾM GOOGLE

- Một số phân tích
  - hiệu quả: tối ưu truy nhập nhanh và hiệu quả
    - chỉ số: thuận/ngược
    - cấu trúc dữ liệu tốt: hệ thống file riêng
    - yêu tố "tập trung hóa".
  - Đánh chỉ số
    - 1994: mọi thứ tốt nếu bộ chỉ số đầy đủ
    - 1997: luận điểm trên không đúng. Chỉ số đầy đủ không tăng chất lượng tìm kiếm (chỉ có 1/4 top máy TK thương mại tìm được chính mình trong top ten).
    - Định hướng: cần tăng độ chính xác của các trang, đặc biệt 10 trang đầu.
    - Tập trung hóa chỉ số: tăng tốc độ tìm kiếm
- Môi trường hoạt động
  - Hệ điều hành Linux

MỘT SỐ ĐẶC TRƯNG VÀ THỊ TRƯỜNG

Search Engine Showdown
The Users' Guide to Web Searching

Search Engine Features Chart

* See also Search Engines by Search Features.
* Search engines grouped by size; all words link to more detailed reviews.

Last updated Oct. 31, 2003.
by Greg R. Notess.

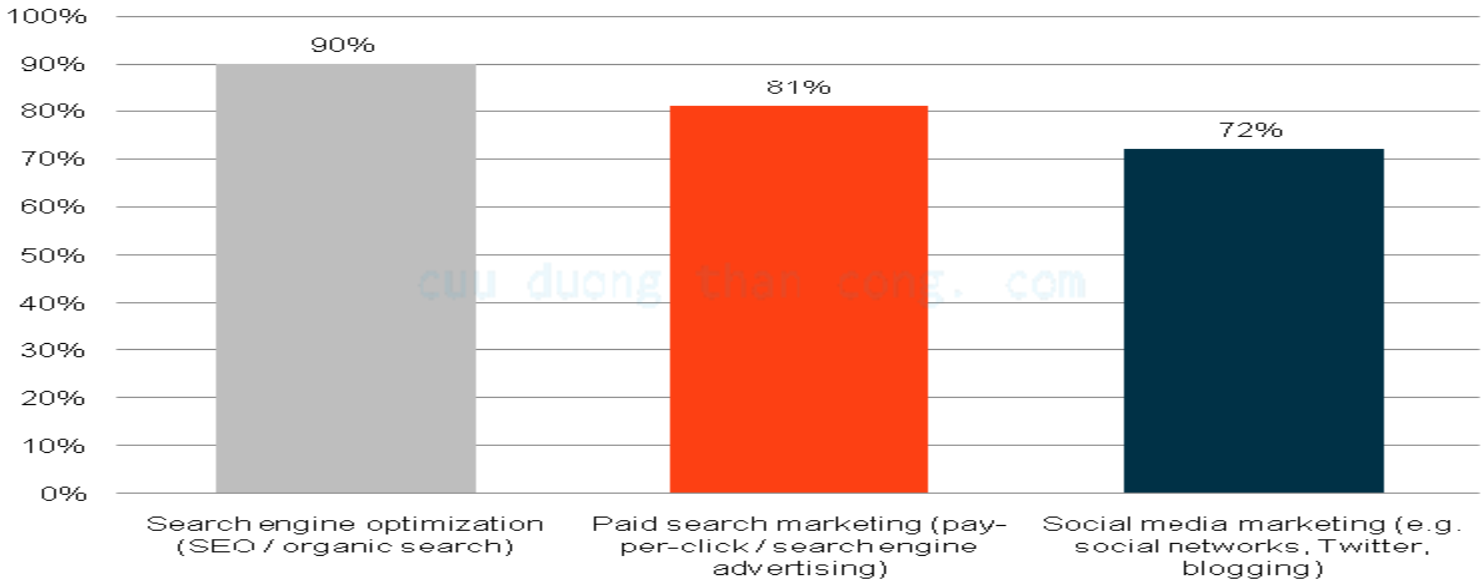
Search Engines	Boolean	Default	Proximity	Truncation	Case	Fields	Limits	Stop	Sorting
Google Review	-, OR	and	Phrase	No	No	intitle, inurl, more	Language, filetype, date, domain	Varies, + searches	Relevance, site
AlltheWeb Review	and, or, andnot, ( ), +, -, or with ()	and	Phrase	No	No	title, URL, link, more	Language, filetype, date, domain	No if not rewritten	Relevance, site
Lycos Review	+, -	and	Phrase	No	No	title, URL, link, more	Language, domain	No	Relevance
AltaVista Simple Review	+, -, AND, OR, AND NOT, ( )	and, phrase	Phrase, NEAR	Yes *	No	title, URL, link, more	Language, filetype	Yes	Relevance, site
AltaVista Adv. Review	and, or, and not, ( )	phrase	Phrase, near, within, <, <=	Yes *	Yes	title, URL, link, more	Language, filetype, date	No	Relevance, if used
HotBot (Inktomi) Review	AND, OR, NOT, ( ), -	and	Phrase	No	Yes	title, more	Language, date	Some	Relevance, site
MSN Search Review	AND, OR, NOT, ( ), -	and	Phrase	No	Yes	title, link	Language, filetype, date	Some	Relevance
Teoma Review	-, OR	and	Phrase	No	No	intitle, inurl	Language, site	Yes, + searches	Relevance, metasites
WiseNut Review	- only	and	Phrase	No	No	No	Language	Yes, + searches	Relevance, site
Gigablast Review	AND, OR, AND NOT, ( ),+, -	or	Phrase	No	No	title, site, ip, more	Domain, type	No	Relevance

MÁY TÌM KIẾM: THỊ TRƯỜNG

Năm 2010:

- Larry Page và Sergey Brin cùng xếp thứ 11 với tài sản 15 tỷ US$. http://www.forbes.com/wealth/forbes-400?boxes=listschannellatest#p_2_s_arank_-1_
- Báo cáo *Hiện trạng thị trường máy tìm kiếm thường niên lần thứ sáu* của SEMPO (Search Engine Marketing Professional Organization) thị trường công nghiệp tiếp thị máy tìm kiếm khu vực Bắc Mỹ năm 2010 *tăng trưởng 14% từ 14,6 tỷ đô la Mỹ năm 2009 lên 16,6 tỷ đô la Mỹ năm 2010 (*Chris Sherman (2010). The State Of Search Engine Marketing 2010, http://searchengineland.com/the-state-of-search-engine-marketing-2010-38826. Mar 25, 2010 at 5:00pm ET).

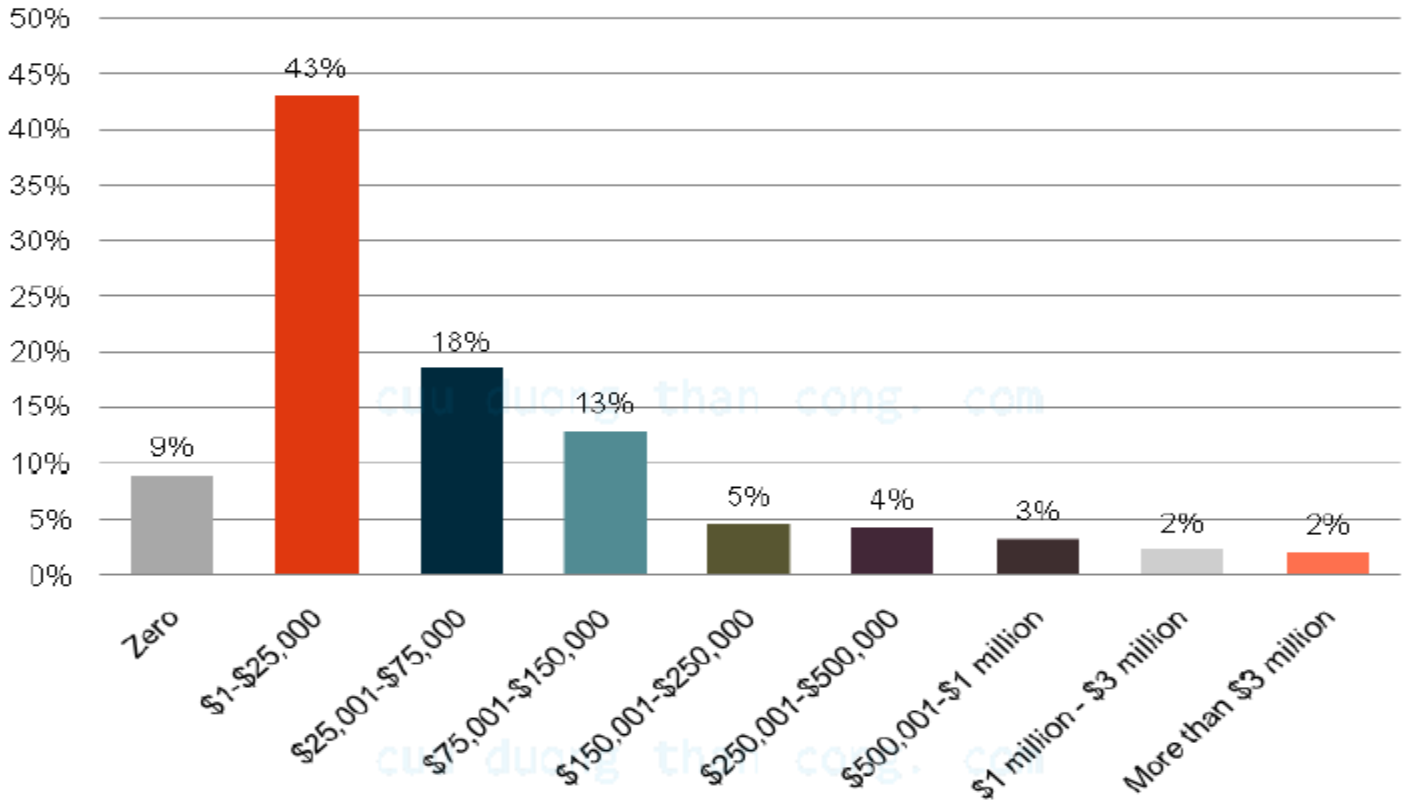
Figure 7: Which of the following types of activity does your organization carry out?



Respondents: 527

MÁY TÌM KIẾM: THỊ TRƯỜNG

Figure 8: What was your company's budget for search engine optimization (organic search) in 2009? (*Including agency, staff and technology costs*)



Respondents: 416

Năm 2010: Kinh phí tiếp thị trên máy tìm kiếm

- http://searchengineland.com/the-state-of-search-engine-marketing-2010-38826. Mar 25,
- Search engine optimization (SEO): nâng cao khả năng hiện thị trên máy tìm kiếm theo kết quả tìm kiếm, mở rộng giải pháp tiếp thị
- Search engine marketing (SEM): được đưa vào danh sách ưu tiên do có trả phí

NGHIÊN CỨU THU HỒI THÔNG TIN

- Theo Google Scholar, số bài chứa “Search Engine”: mọi nơi: 424.000 bài; tiêu đề: 6350 (2730 bài từ 2006-nay)
- Theo thư viện bài báo khoa học của ACM (ACM Digital Library): có trên 40.400 bài báo khoa học trong thư viên có liên quan tới “search engine”.

Top organizations in Data Mining — Filter: Data Mining — All Years

Author
Publication
Conference
Journal
Organization

Organization (View By Location)	Publications	Citations
Stanford University (H-Index: 363)	3180	362848
University of California Berkeley (H-Index: 345)	2555	319482
Massachusetts Institute of Technology (H-Index: 3	2496	314022
Microsoft (H-Index: 280)	3909	281107
Carnegie Mellon University (H-Index: 278)	2933	233987
IBM (H-Index: 241)	4159	226917
Cornell University (H-Index: 240)	1450	154205
Princeton University (H-Index: 272)	917	135970
University of Illinois Urbana Champaign (H-Index:	1959	125676
AT&T Labs Research (H-Index: 195)	1263	119500
Harvard University (H-Index: 319)	861	116990
University of California San Diego (H-Index: 232)	1226	110461
University of Washington (H-Index: 232)	1233	105580
University of Southern California (H-Index: 205)	1231	102271
University of Michigan (H-Index: 212)	1268	99059
Columbia University (H-Index: 223)	1233	94526
University of Maryland (H-Index: 207)	1686	90996
University of Pennsylvania (H-Index: 203)	1000	89839
University of Toronto (H-Index: 201)	1141	88896
Institute of Space and Astronautical Science (H-Index: 26)	9	54
Vietnam National University (H-Index: 6)	6	54
Shizuoka University (H-Index: 29)	30	54
China Medical University Taiwan (H-Index: 15)	18	54
Universidade do Vale do Rio Dos Sinos (H-Index: 13)	4	53

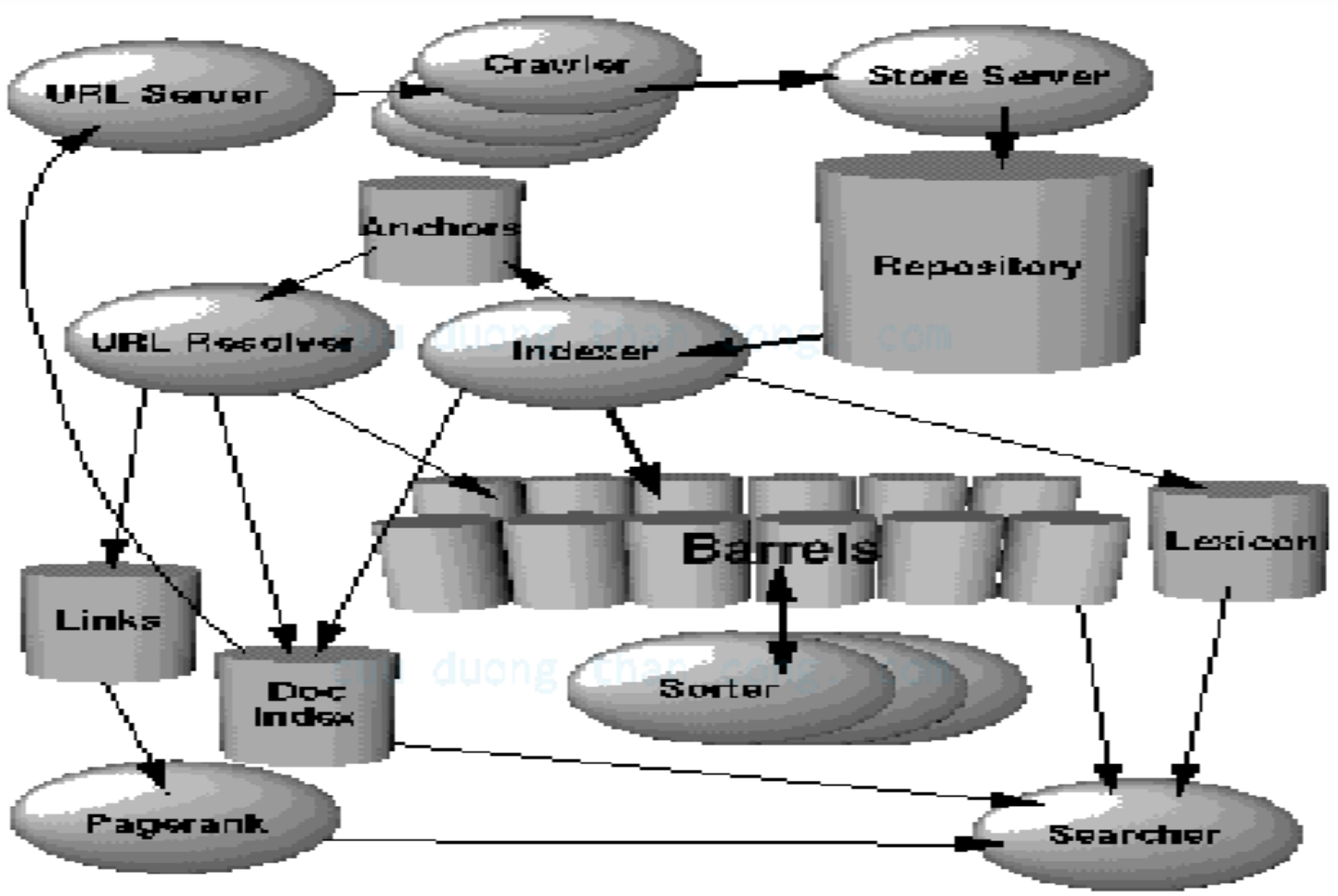
Nguồn: http://academic.research.microsoft.com/CSDirectory/Org_category_8.htm

CÁC THÀNH PHẦN CƠ BẢN CỦA MÁY TÌM KIẾM

- Một số thành phần cơ bản
  - Module phần mềm cơ bản
    - module crawling (crawler)
      - dò theo liên kết trên Web
        - thu thập nội dung trang Web
        - lưu vào các kho chứa
    - module indexing (indexer - đánh chỉ mục)
      - duyệt nội dung trang web đã tải
        - phân tích, tính hạng cho các trang này
      - lưu trữ trong các cấu trúc
    - module ranking
      - tính hạng các trang: cố đinh, theo câu hỏi
    - module searching (Tìm kiếm)
      - truy xuất cơ sở dữ liệu
        - trả về danh sách tài liệu thỏa mãn yêu cầu người dùng
      - sắp xếp các tài liệu này theo mức độ hợp lệ so với câu hỏi
    - module interface (giao diện)
      - nhận câu truy vấn của người dùng
      - gửi cho module tìm kiếm
      - nhận kết quả trả về và hiển thị
  - Tổ chức dữ liệu
    - Hệ thống file
    - Các cấu trúc dữ liệu

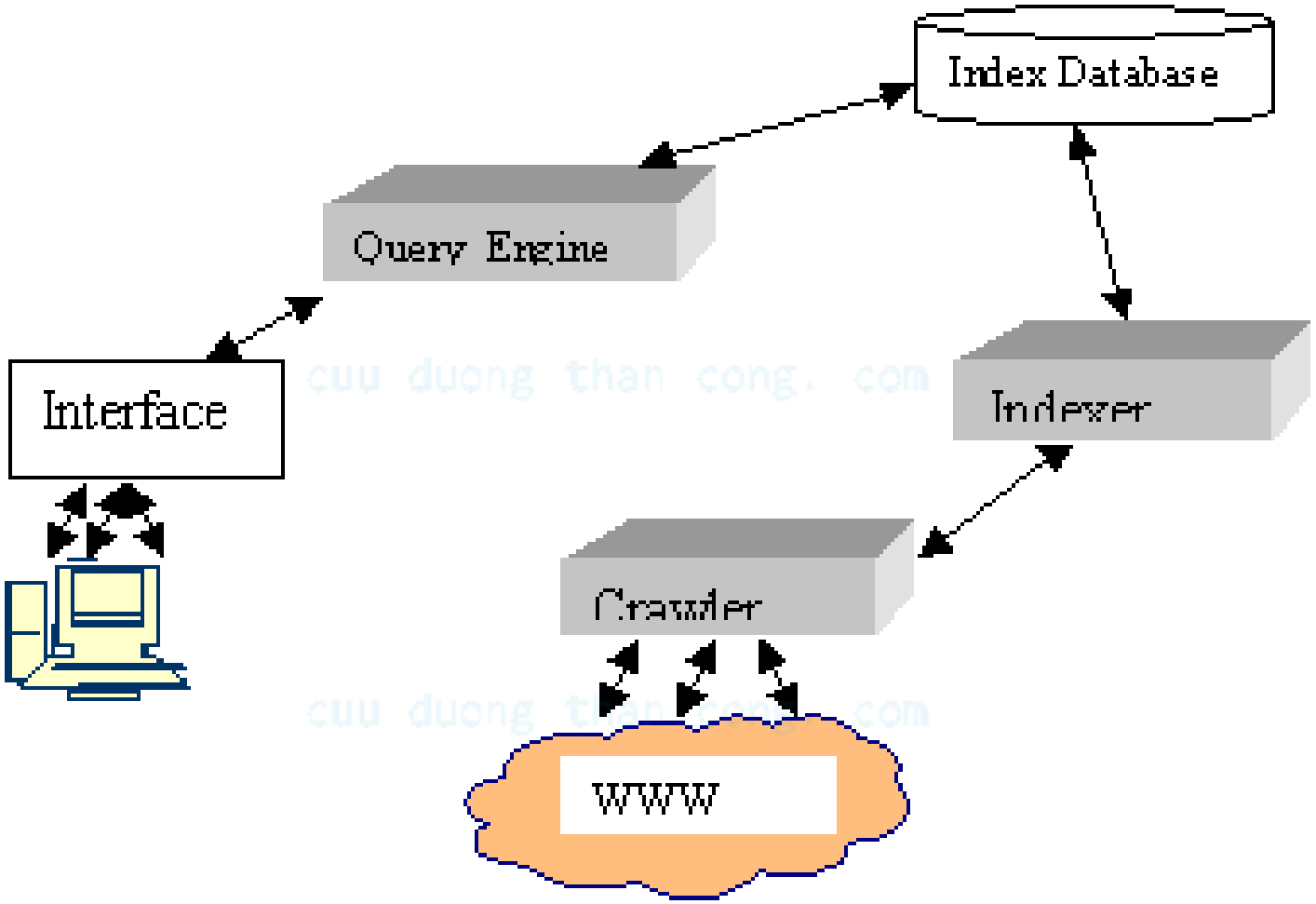
CÁC THÀNH PHẦN CƠ BẢN CỦA MÁY TÌM KIẾM

Máy tìm kiếm Google

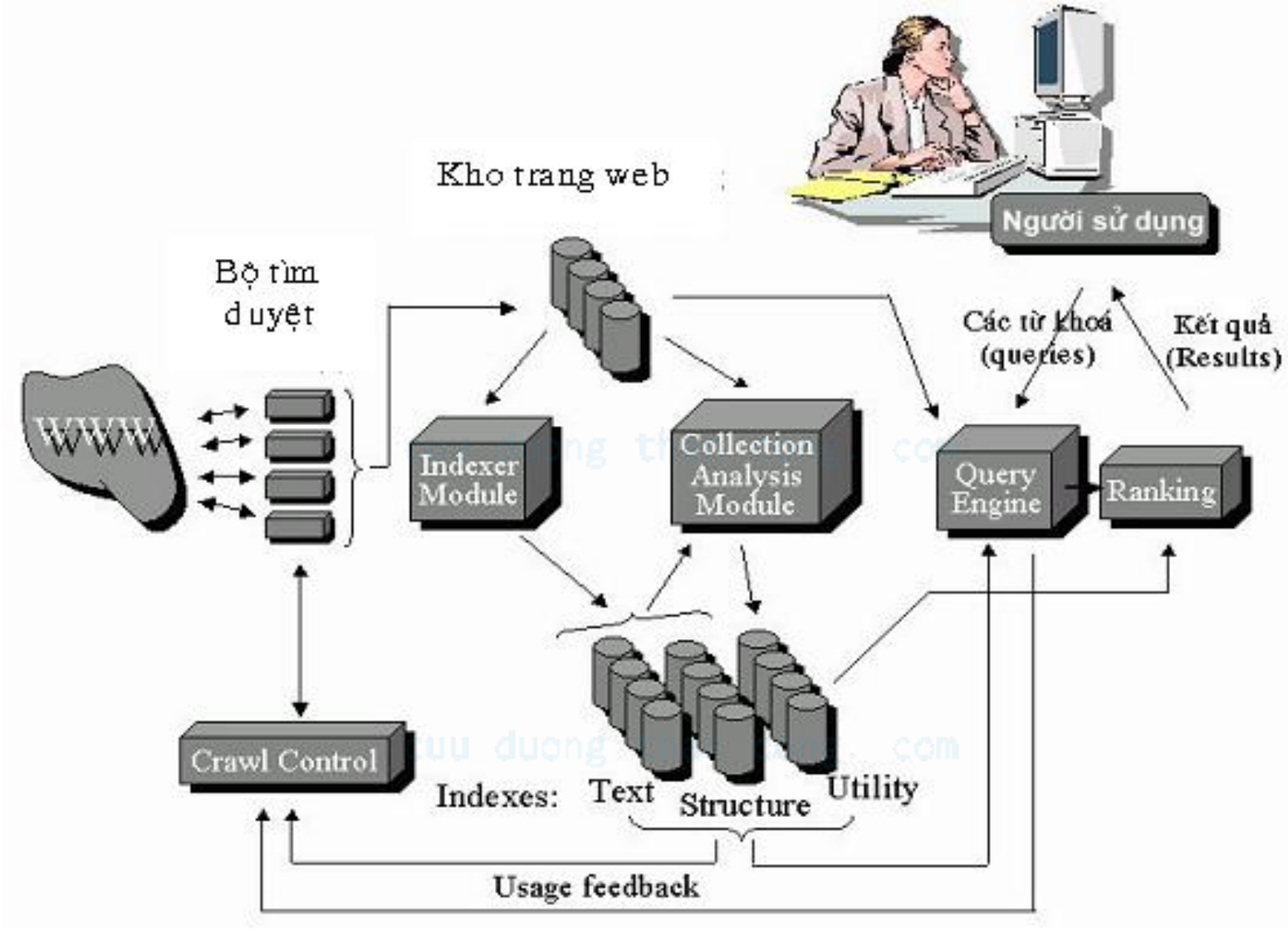


CÁC THÀNH PHẦN CƠ BẢN CỦA MÁY TÌM KIẾM

Máy tìm kiếm AltaVista



MÁY TÌM KIẾM ASPSEEK



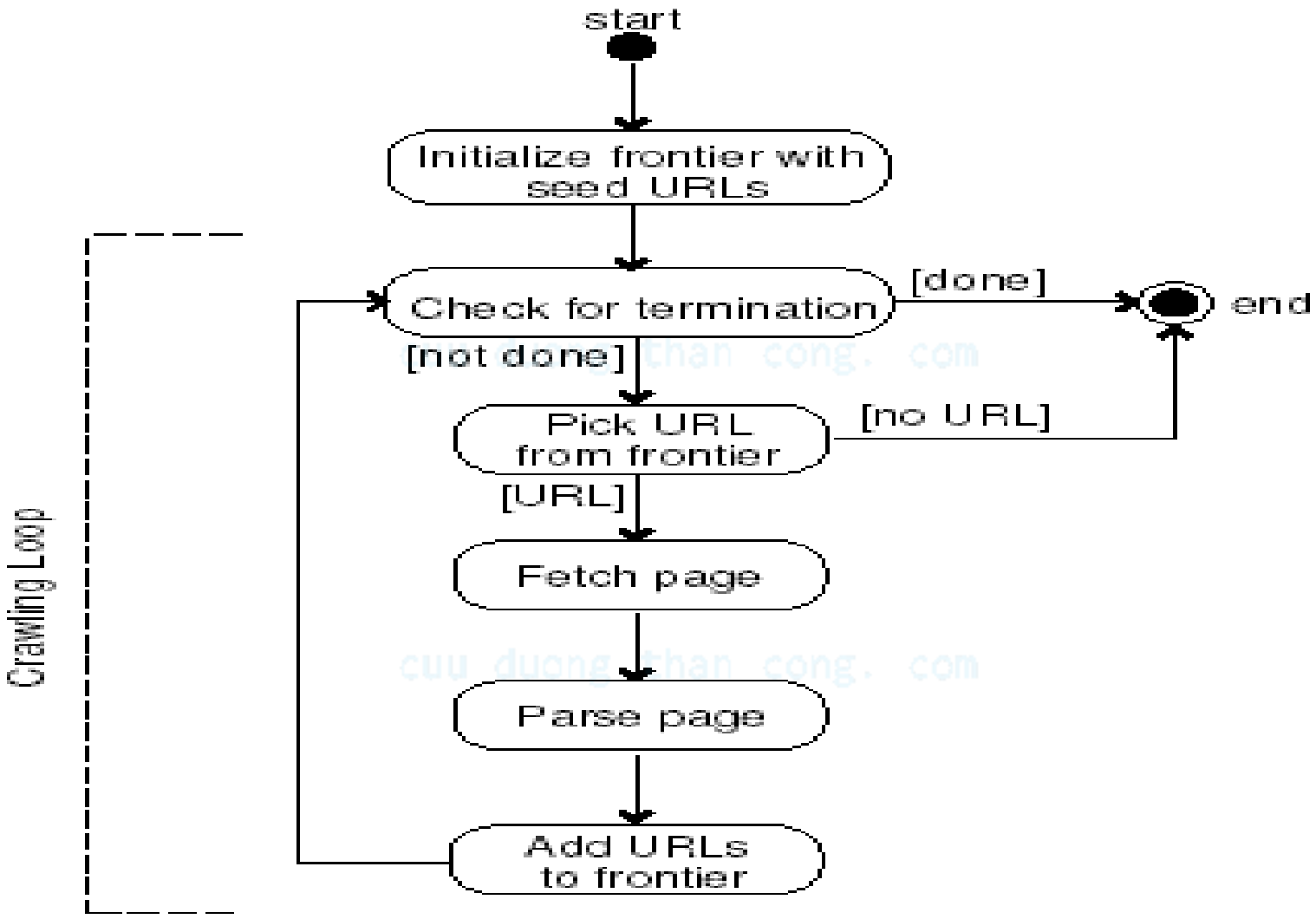
Máy tìm kiếm Vietseek (trên nền ASPseek)

CRAWLING

- Giới thiệu
  - một thành phần quan trọng
  - hầu hết các máy tìm kiếm
- Chức năng
  - thu thập các trang web từ các site khác nhau trên Internet
  - lưu giữ vào kho lưu trữ (phục vụ bộ tạo chỉ mục)
  - làm tương nội dung các trang web được lưu trữ
- Hoạt động
  - khai thác cấu trúc liên kết web
  - lần theo các trang web
  - thu thập và làm tươi

CRAWLING

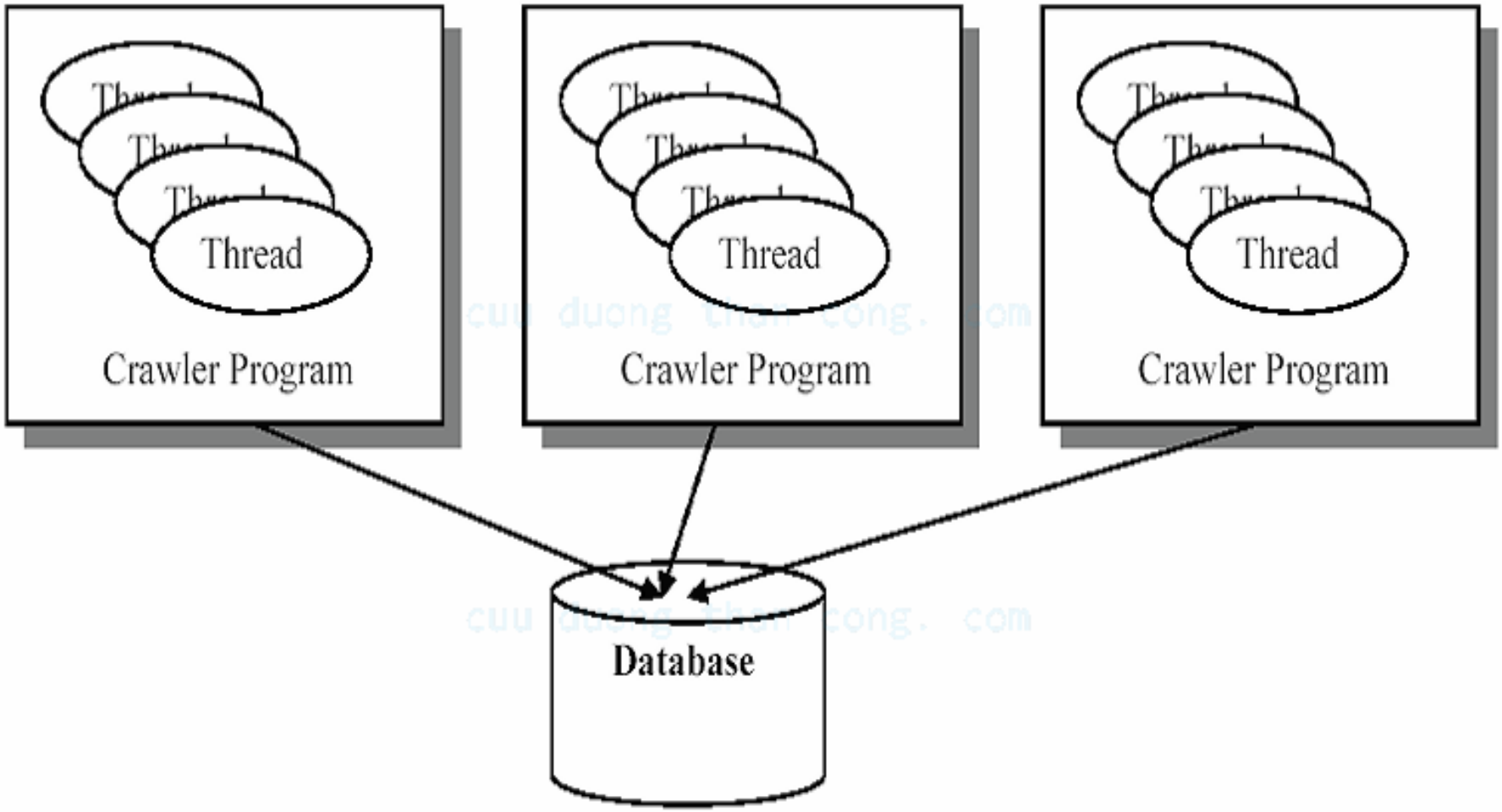
Thuật toán Crawler tuần tự tổng quát



CRAWLING TRONG Virginia

- Tập trung thuật toán nhằm tăng tốc độ thời gian
- Môđun được viết trên Java và kết nối CSDL MySQL
  - *Tại sao Java mà không phải C++ hay ngôn ngữ khác* ?
    - Chương trình trên ngôn ngữ khác chạy nhanh hơn
      - đặc biệt khi được tối ưu hóa mã
    - Crawler: nhiều vào-ra mà không quá nhiều xử lý của CPU
      - thời gian đáng kể *mạng và đọc/ghi đĩa*.
      - độ nhanh - chậm CPU Java và C++ không khác
    - Java độc lập nền hạ tầng (dịch sang mã byte)
      - di chú crawler sang máy khác để thực hiện.

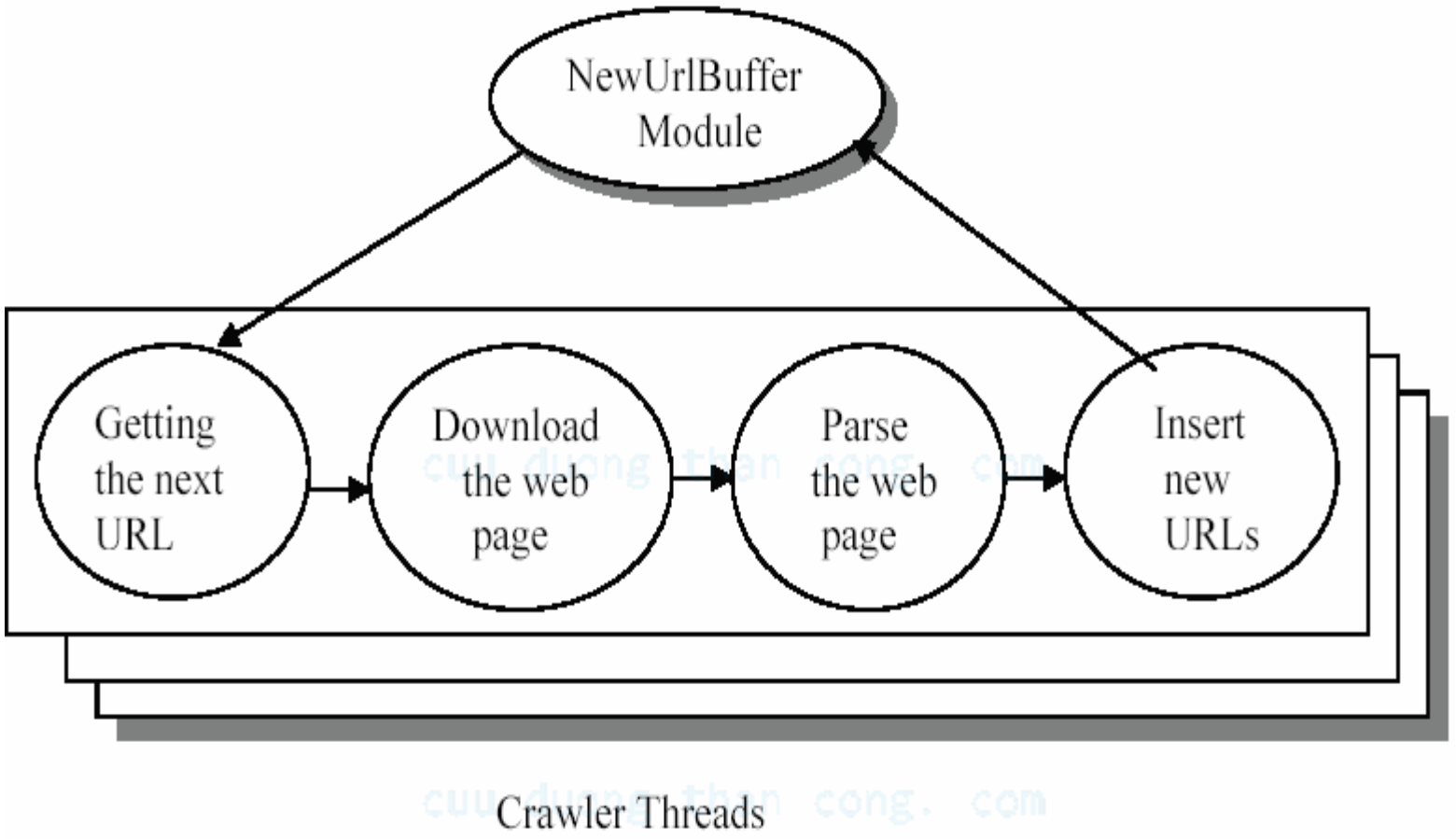
CRAWLING TRONG Virginia



CRAWLING TRONG Virginia

- Tăng tốc
  - cấu trúc hai mức đồng thời
    - mức crawler
    - mức luồng trong crawler
- Đồng thời mức luồng
  - luồng chạy đồng thời trong chương trình
  - đồng nhất
  - được điều khiển bởi môđun NewUrlBuffers
- Đồng thời crawler
  - file cấu hình: phân hoạch miền Internet để tải
  - tải các trang web theo phân hoạch

CRAWLING TRONG Virginia



Các thao tác chức năng một luồng trong crawler
(chồng luồng trong một crawler)

CRAWLING TRONG Virginia

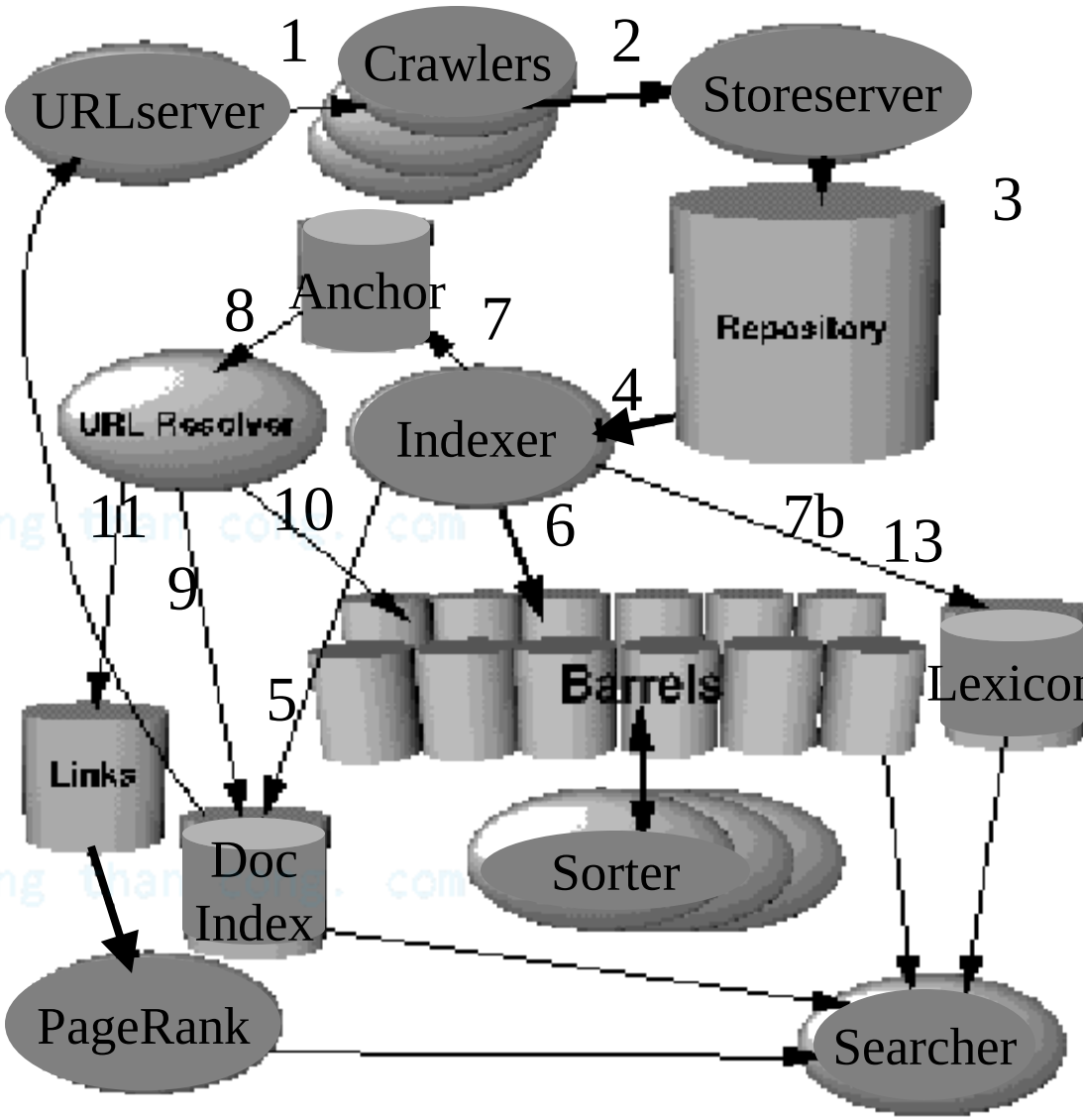
- Thư viện chạy luồng
  - Mã thao tác luồng
    - file CrawlerThread.java
  - Điều khiển chạy luồng
    - NewUrlBuffer
    - file NewUrlBuffer.java
- Quá trình Parser
  - file Parser.java
  - Tìm kiếm thông tin hữu dụng
    - URL
    - thông tin liên kết,
    - meta-thông tin trong HTML,
    - text
    - ...

CRAWLING TRONG Virginia

- Hoạt động của một luồng
  - Nhận một URL mới từ buffer
    - Nếu có, chuẩn bị tải file HTML tại server từ xa
    - ngược lại, chờ một số giây: nhận tiếp URL (nếu có)
    - Đồng bộ giữa tải và Parser: Buffer cỡ 10 URL
  - Nhận trường đầu HTML cần tải từ server từ xa
    - thu thông tin trạng thái của file (kiểu file, dung lượng, ...)
    - quyết định có tải hay không ?
  - Trường hợp quyết định
    - Tải thực sự
    - Đưa nội dung file vào bộ nhớ
    - Parser phân tích và đưa nội dung của file lên đĩa
  - Tiếp tục chu trình
- viết tới 7 phiên bản cho Crawler

CRAWLING TRONG GOOGLE

1. URLserver
   - gửi danh sách URL webpage sẽ đưa về cho các crawler phân tán.
2. Các crawler
   - crawling webpage về gửi cho StoreServer.
3. StoreServer
   - nén và lưu webpage lên đĩa (vào kho chứa).
4. Indexer có các chức năng:
   - đọc tài liệu từ kho chứa
   - giải nén
   - Parser



CRAWLING TRONG GOOGLE

5. Index cùng Sorter

- gán DocID cho Web page (DocID được gán mỗi khi Parser phát hiện một URL mới).

6. Mỗi tài liệu

- Được biến đổi thành tập các xuất hiện của các từ khóa (gọi là hit)
- Hit: từ khóa, vị trí trong tài liệu, font (cỡ, ...), hoa/thường.

Indexer

- phân bố các hit thành tập các "barrel" lưu trữ các chỉ số đã được sắp xếp.

7. Indexer

- phân tích các siêu liên kết
- lưu các thông tin quan trọng trong file "anchor" cho phép xác định
  - nguồn, đích của siêu liên kết
  - nội dung văn bản trong siêu liên kết.

- (7b) sinh từ điển tra cứu từ khóa.

CRAWLING TRONG GOOGLE

Văn bản trong siêu liên kết:

- nhiều hệ chỉ gắn vào trang nguồn
- Google gắn vào cả trang đích lợi ích
  - cho thông tin chính xác hơn, thậm chí chính trang web
    - "tóm tắt"
    - "qua chuyên gia xử lý"
- index cho trang web
  - "không văn bản"(ảnh, chương trình, CSDL ...)
  - xử trí trường hợp trang web chưa tồn tại
  - lấy văn bản anchor làm "nội dung"!
- Tư tưởng này có trong WWW Worm (1994) và có trong Google
  - o kết quả chất lượng hơn
  - o chú ý: crawling 24 triệu trang có tới 259 triệu anchor.

8. URLsolver

- đọc file anchor
- biến đổi URL tương đối URL tuyệt đối

CRAWLING TRONG GOOGLE

9. URLsolver

- cập nhật lại DocID

10. URLsolver

- đưa text anchor vào *index thuận* (hướng trỏ anchor).

11.URLsolver

sinh CSDL liên kết gồm các cặp liên kết (được dùng để tính PageRank).

12. Sorter

- đọc các Barrel (xếp theo DocID) để sắp lại theo WordID tạo ra các *index ngược*.
- Sinh ra danh sách các wordID và gia số trong index ngược.

13. DumpLexicon

- lấy từ lexicon+danh sách wordID
- sinh ra lexicon mới.

14. Searcher

- chạy do webserver trả lời câu hỏi
- dựa trên lexicon mới PageRank, index ngược.

CRAWLING TRONG GOOGLE

- Chạy craw Google:
- Chạy craw là bài toán thách thức: cần thi hành tinh xảo - tin cậy (quan trọng). Dễ đổ vỡ do tương tác hàng trăm nghìn phục vụ web và vô số phục vụ tên.
- Hàng trăm triệu webpage, hệ crawler Google *phân tán*, nhanh. 1 phục_vụ_URL đơn đưa danh sách các URL cho các crawler (Google dùng 3 trình Crawler).
- Phục vụ URL và các crawler viết trên *Python*.
- Mỗi crawler giữ 300 kết nối tại một thời điểm. Cần tìm kiếm page đủ nhanh. Máy chậm làm 4 crawler, mỗi crawler giữ 100 kết nối. Hiệu năng chính xem DNS

  crawler duy trì cache SNS. - "Trạng thái": nhìn DNS, kết nối host, gửi yêu cầu, nhận trả lời.

Phân một crawler kết nối với hơn nửa triệu phục vụ, sinh hàng chục triệu thực thể log, vô số cuộc thoại và thư,

CRAWLING: BÀI TOÁN LÀM TƯƠI TRANG WEB

Web search Engine dùng crawler đa thành phần:

- Duy trì bản sao địa phương của trang web,
- Tạo các cấu trúc dữ liệu (như index ngược)

Các trang web được thay đổi thường xuyên:

- 23% trang web thay đổi hàng ngày
- 40% trang web thương mại thay đổi hàng ngày
- Chu kỳ phân rã một trang web 20 ngày (half-life 10 ngày)

Crawler thường xuyên thăm trang web để bảo đảm tính "tươi"
"Thăm" thế nào cho tối ưu?

CRAWLING: CHIẾN LƯỢC TỐI ƯU

Sơ đồ chiến lược tối ưu gồm hai thành phần:

- Giải bài toán "Tính thường xuyên"
- Giải bài toán "Lập lịch crawling"

Thành phần 1: Giải bài toán "tính thường xuyên":

+ Mục tiêu:

(1) Tối ưu số lượng lần crawling mỗi trang,

(2) Tối ưu hóa thời điểm crawling mỗi trang.

+ Nội dung:

(1) Xác định *metric tối ưu* thích hợp hơn: dựa trên mức độ "khó xử" ? mà không theo "tình trạng cũ"

CRAWLING: CHIẾN LƯỢC TỐI ƯU

(2) Khung cảnh hợp nhất (xử lý điểm bất động) trên cơ sở kiểu phân bố phổ biến cập nhật trang web:

- Possion, Pareto, Weibull,
- Quasi-deterministic

(3) Thuật toán hiện đại nhất (state-of-the-art) để tìm số tối ưu dò tìm:

- chung và riêng: các ràng buộc đời sống thực,
- hiệu quả tính toán đặc biệt: lượng tính đồ sộ

(4) Thuật toán tìm ra các thời điểm dò tìm lý tưởng

CRAWLING: CHIẾN LƯỢC TỐI ƯU

Thành phần 2: Giải bài toán "Lập lịch crawling":

+ *Mục tiêu*: Tạo lịch crawling thực hiện được tối ưu dựa trên các thời điểm crawling lý tưởng hóa,

+ *Nội dung*: - Giải pháp vấn đề chuyển tải chính xác,

- Các ràng buộc cuộc sống thực

+ *Thử nghiệm*: Phân tích mẫu cập nhất từ một số website mà IBM có:

-Grant Slam Tennis: Úc+Pháp+Mỹ mở rộng, Wimbledon

- Golf: Các cup Master, Ryder,

- Olympic: Đông Nagano-1998, Hè Sydney-2000

- Awards: Tonys, Grammys

+ *Kết quả*: phân bố thời gian liên cập nhật *phủ miền rộng* theo ứng xử

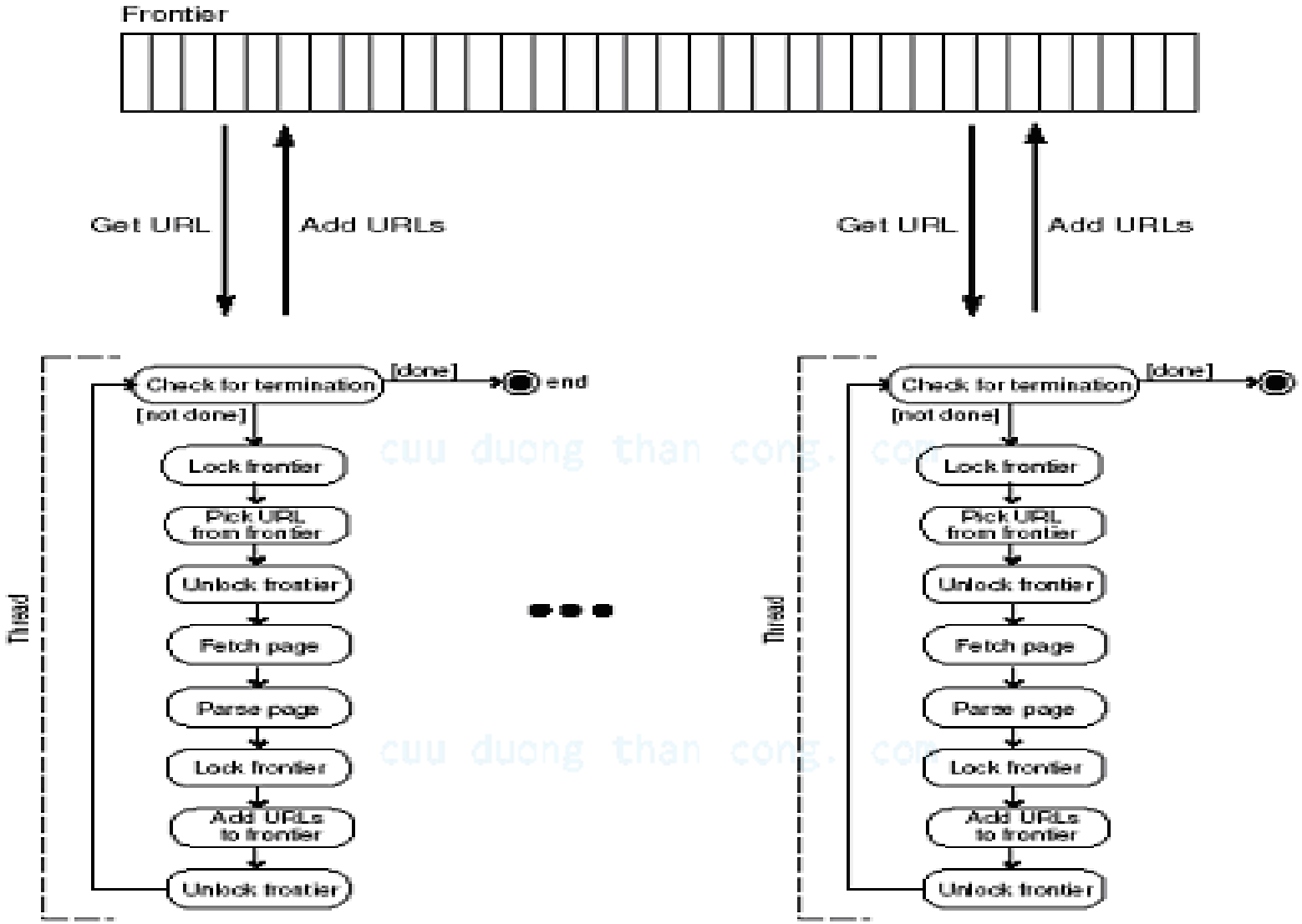
MỘT SỐ VẤN ĐỀ LIÊN QUAN VỚI CRAWLER

- Cách chọn tải trang web
  - Không thể tải mọi trang web
    - Không gian web quá lớn
    - Tập tải về có "giá trị" nhất
  - Tập khởi động
    - Nguồn để tải tập trang web: "nhân" crawler
    - Được công bố
  - Thứ tự trong frontier
    - Chọn trang "quan trọng" ?
    - Hạng ?
- Cách làm tươi trang web
  - Thứ tự làm tươi
    - Làm tươi theo định kỳ
    - Trang biến đổi nhiều làm tươi nhanh hơn
  - Tính tươi của trang web
    - Chữ ký nội dung trang (Vinahoo: Thuật toán MD5)
    - So sánh hai chữ ký

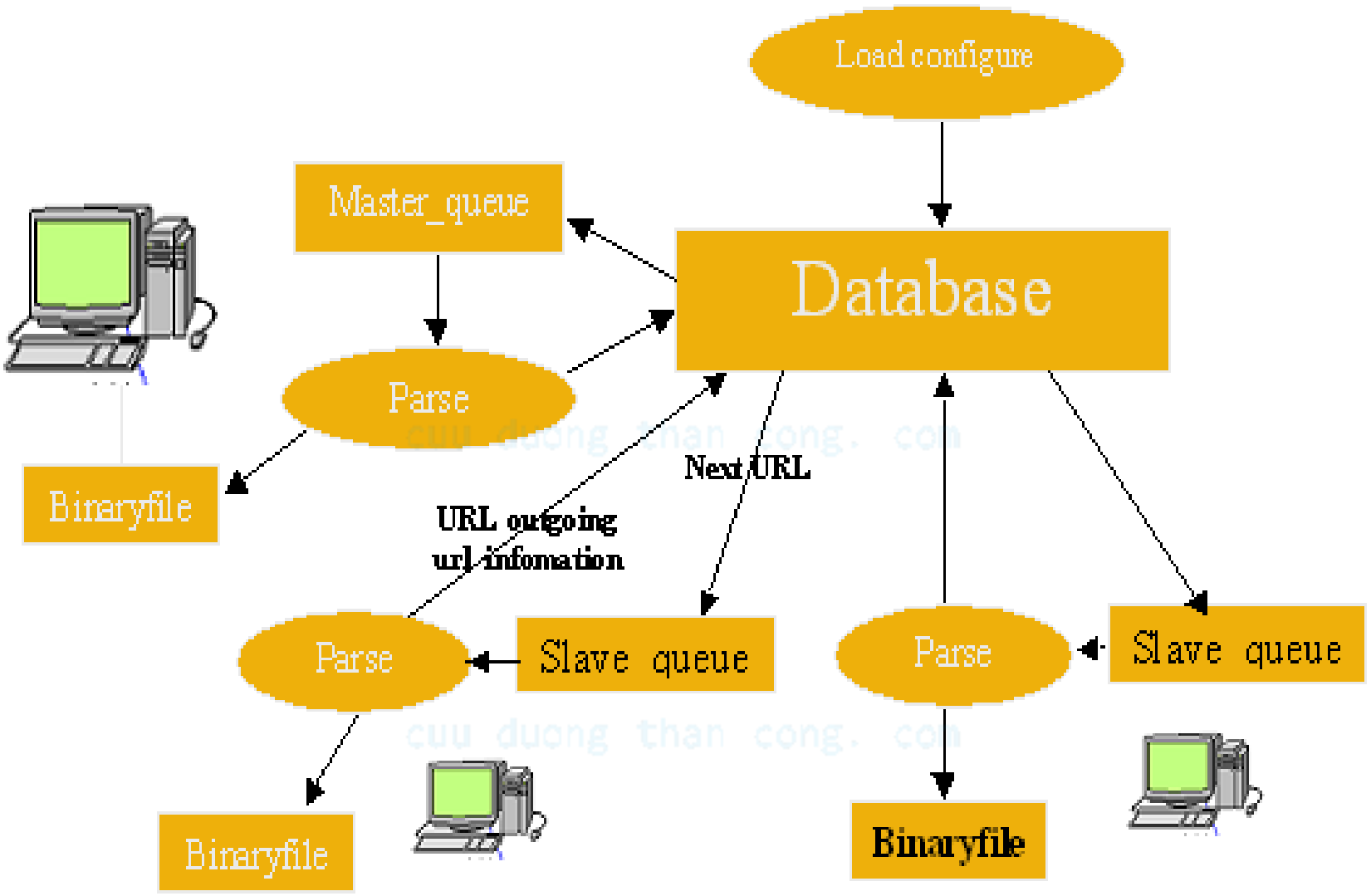
MỘT SỐ VẤN ĐỀ LIÊN QUAN VỚI CRAWLER

- Tối thiểu hoá việc tải nạp các site đã thăm
  - Ghi nhận các site đã thăm
  - So sánh: tương tự như URL, sử dụng thuật toán MD5
- Song song hóa quá trình dò tìm
  - chạy trên nhiều máy
  - song song thực hiện
  - không tải bội trang web

CRAWLER ĐA LUỒNG



CRAWLER SONG SONG



ĐÁNH CHỈ SỐ VÀ LƯU TRỮ TRONG GOOGLE: BIGFILES

Nguyên tắc

- "tối ưu"
  - tập dữ liệu lớn được dò tìm
  - index và tìm kiếm không tốn kém
- Một số căn cứ
  - phù hợp di chuyển đĩa
  - cấu trúc dữ liệu thích hợp

• Bigfiles

- Hệ thống file
  - đa thành phần
  - Hệ thống file ảo
- Gói Bigfiles
  - định vị trong h/thống file đa t/phần được tự động
  - đảm nhận định vị/giải định vị đặc tả file
  - hỗ trợ nén nội dung file

KHO LƯU TRỮ TÀI LIỆU TRONG GOOGLE

- Kho lưu trữ

- Lưu *mọi trang web*
- kế tiếp nhau
- Nén chuẩn zlib
  - dung hòa tốc độ nén tỷ lệ nén
  - nén 3:1 (so sánh với bzip 4:1 song tốc độ nén chậm)
- Với chỉ 1 cấu trúc
  - đảm bảo nhất quán
  - dễ phát triển

    xây dựng CTDL khác chỉ từ 1 kho và 1 file hiện lỗi clawer
- *Sync: tổng kiểm tra ?*

Repository: 50,5 GB-147,8 GB uncompressed

Sync	length	compressed packet
Sync	length	compressed packet

Từng gói nén (compressed packet)

DocID	ecode	urllen	pagelen	url	page

ĐÁNH CHỈ SỐ TÀI LIỆU

- Document index
  - giữ thông tin về từng document (TL)
  - cố định mode ISAM theo DocID
  - bản ghi
    - trạng thái hiện thời (clawled|chưa clawled)
    - con trỏ 1 (tới kho)
    - checksum,
    - các thống kê
    - con trỏ 2*:
      - tới file DocInfo độ dài biến thiên chứa URL+title (khi TL đã clawled)
      - tới URLlist chứa URL (chưa clawled).
  - file chuyển URL DocID
    - danh sách (checksum URL, DocID tương ứng) xếp theo checksum URL.
  - Kỹ thuật URLserver tìm DocID theo URL
    - Tính URLchecksum,
    - Tìm kiếm nhị phân file chuyển theo URLchecksum
    - Cho phép tìm kiếm theo mode lô. Mode lô cần cho dữ liệu lớn (ví dụ, 332 triệu links) ? câu hỏi hàng ngày nhiều ?

TỪ ĐIỂN VÀ DANH SÁCH HIT

(4) Từ điển

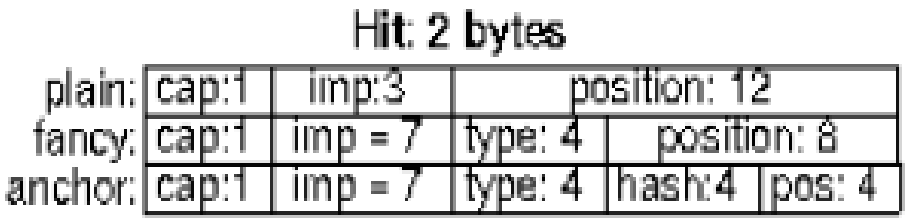
- Một số dạng biểu diễn
- đặt BNT khi thực hiện
- Gồm 2 phần
  - dãy từ cách nhau 1 dấu cách (ngoài ra có các thông số khác)
  - bảng băm các con trỏ,
- (1998): máy có BNT 256MB, 14 triệu từ,

(5) Các danh sách hit

- *một danh sách* dãy xuất hiện *một từ* ở *một tài liệu*
  - vị trí, font, hoa-thường,
- biểu diễn cả index - index ngược
  - trình bày hiệu quả nhất có thể được
- chọn lựa mã hóa
  - *đơn giản*|*cô đọng*
    - Đơn giản: bộ nhớ *ít hơn*
    - Cô đọng: chế biến bit *ít hơn*
    - Huffman |*Huffman* ? mã hóa cô đọng tối ưu (compact).

CẤU TRÚC HIT TRONG GOOGLE

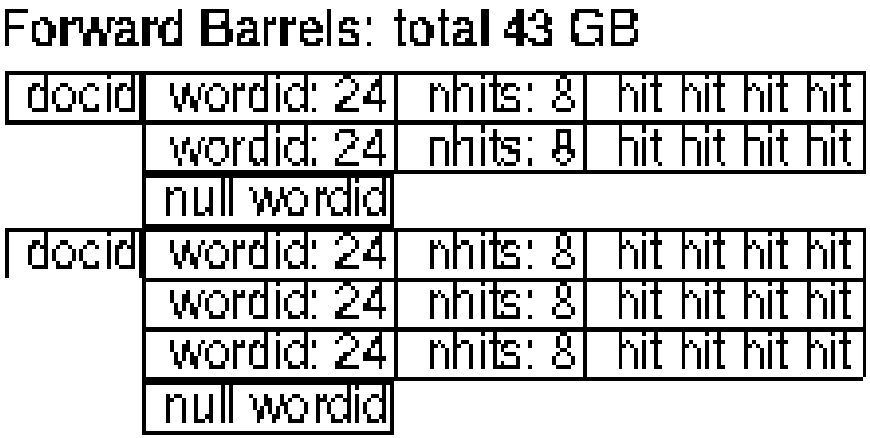
- hit (như hình vẽ)



  - 2 byte (16 bit)
  - hai kiểu: plain và fancy
    - *plain*: word trong nội dung; hoa/thường 1 bit; font 111 plain, =111 fancy; vị trí > 4096 đặt =4096
    - *fancy*: hai loại thường/anchor
      - *Anchor*: 4 bit vị trí trong anchor+4 bit hash cho DocID chứa anchor
      - Nghiên cứu giải pháp anchor+hash dài hơn
  - Tiết kiệm bộ nhớ
    - hit kết hợp WordID ở index thuận| DocID ở index ngược thành 4 byte
  - Độ dài thực lớn hơn
    - mã escape được dùng (00000-00000000 ?) và hai byte tiếp chứa độ dài thực
  - Thoả giới hạn thiết kế Google năm 1998
    - 2^{24} (14 triệu) Word
    - và 2^{27} (100 triệu) TL

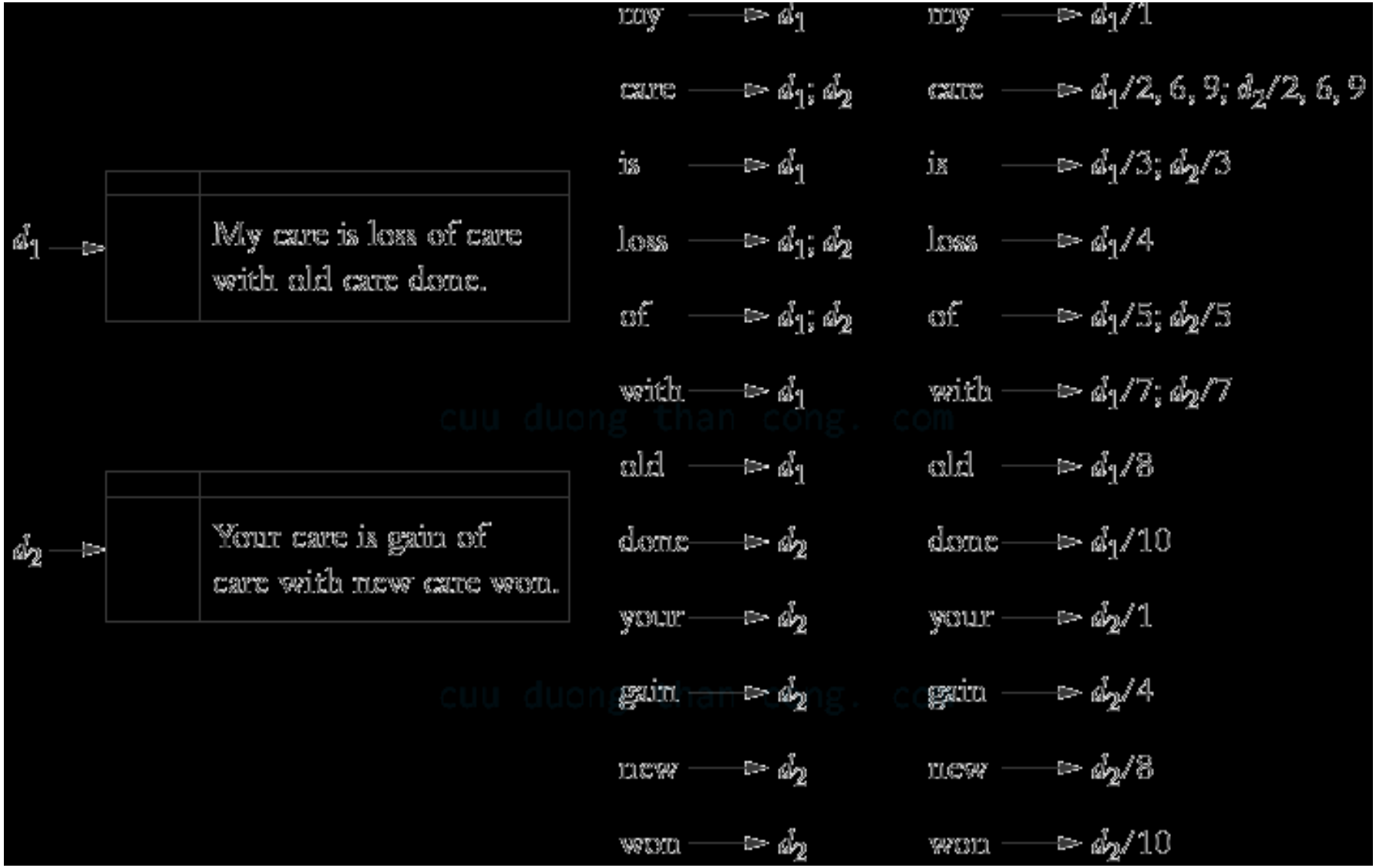
CHỈ SỐ THUẬN: TÌM THEO TÀI LIỆU

- Index thuận



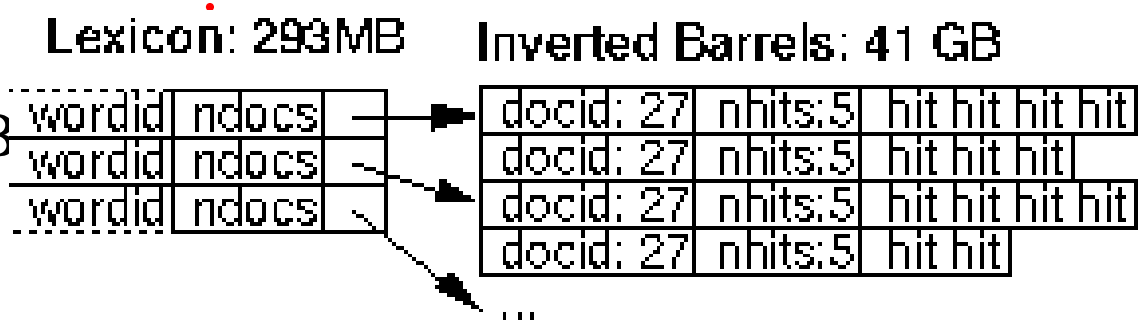
  - Phân hoạch 64 barrel thuận '''
  - Một barrel một vùng chỉ số WordID
    - Nếu Doc có Word ở barrel (vùng chỉ số)
      - ghi DocID vào barrel,
      - tiếp là dãy WordID kèm danh sách hit tương ứng words
    - thêm chút ít bộ nhớ (một docID ghi trên nhiều barrel) song ích lợi
      - về độ phức tạp thời gian
      - mã hóa khi index cuối cùng.
    - * ghi gia số WordID (+WordID đầu) tiết kiệm kh/gian
      - 24 bit cho WordID trong barrel chưa sắp
      - dành 8 bit ghi độ dài danh sách hit.

CHỈ SỐ NGƯỢC: TÌM THEO TỪ



CHỈ SỐ NGƯỢC: TÌM THEO TỪ

- Index ngược



- Từ vựng: 293 MB
- Barrels ngược 41 GB
- chứa barrel như thuận
- sắp theo wordID.
- wordID: từ điển trỏ barrel
  - chứa word tới doclist các DocID + dãy hit tương ứng.

* Quan trọng: Thứ tự DocID xuất hiện tại doclist ra sao ?

(1) Đơn giản: Sắp theo DocID trộn nhanh các doclist theo câu hỏi word đa thành phần ,

(2) Xếp theo "hạng" xuất hiện các word trong Doc: trả lời 1 word tầm thường, word đa thành phần nhanh;

- Khó khăn: (i) trộn, (ii) tính lại hạng khi dựng lại index

(3) giải pháp dung hòa (1)+(2) có hai tập index ngược:

(a) hit tiêu đề + hit anchor,

(b) mọi hit list kiểm tra tập hit tiêu đề + anchor.

(c) Nếu không đủ phù hợp kiểm tra tập thứ hai.

TRUNG TÂM DỮ LIỆU GOOGLE NGÀY NAY

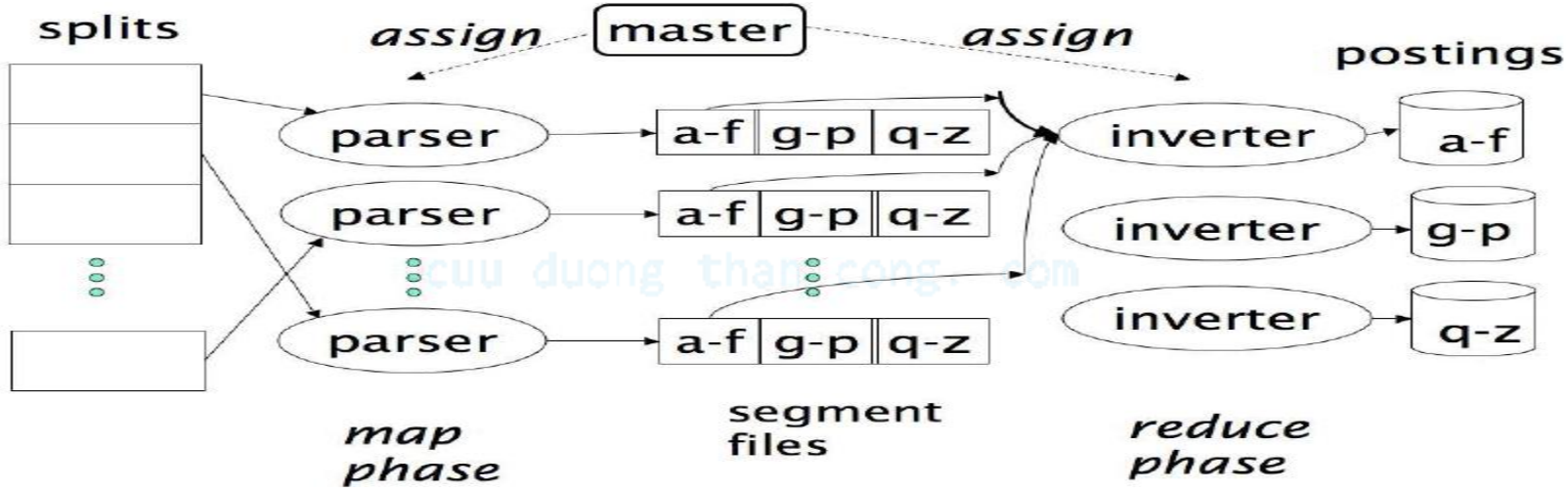
- Chỉ số quy mô WWW: phải sử dụng một cụm máy tính phân tán
  - Máy tính đơn dễ bị lỗi, dễ thất thường: chậm/thất bại
- Trung tâm dữ liệu Google:
  - Chủ yếu chứa các máy tính dịch vụ/hàng hóa
  - Được phân tán trên toàn thế giới
  - Ước lượng
    - Tổng cộng khoảng 1 triệu máy phục vụ, khoảng 3 triệu bộ xử lý lõi (Gartner, 2007)
    - Khởi động 100.000 máy phục vụ mỗi quý
    - Sẽ tiến tới khoảng 10% công suất tính toán trên thế giới
  - Chi phí trung tâm dữ liệu khoảng 200-250 triệu US$ mỗi năm
  - Vấn đề khai thác xâu máy tính đồ sộ

KHAI THÁC XÂU MÁY TÍNH

- Điều khiển chỉ mục phân tán
  - Duy trì một máy quản lý ("tập trung hóa") chỉ đạo công việc chỉ mục – giải pháp "tin cậy"
  - Phân chia công việc chỉ mục thành các tập việc song song
  - Máy quản lý gán mỗi việc cho một máy nhàn rỗi từ cụm.
- Việc "song song"
  - Hai kiểu việc song song và triển khai hai kiểu máy thi hành tương ứng kiểu việc song song
    - Phân tích cú pháp
    - Chỉ mục ngược
  - Tách bộ tài liệu đầu vào thành các đoạn theo hai loại trên
  - Mỗi đoạn là một tập con tài liệu

XỬ LÝ CÁC TẬP CON TÀI LIỆU

- Phân tích cú pháp
  - Máy quản lý gán mỗi đoạn tới một máy tính phân tích cú pháp rỗi
  - Bộ PTCP đọc từng tài liệu và cho ra các cặp (từ, tài liệu)
  - Bộ PTCP viết các cặp vào j phân hoạch theo miền chữ cái đầu tiên của từ, chẳng hạn với j=3: a-f, g-p, q-z.
- Chỉ mục ngược
  - Mỗi bộ chỉ mục ngược thu thập mọi thiết đặt = cặp (từ, tài liệu) cho một phân vùng từ khóa
  - Sắp xếp và ghi vào danh sách các thiết đặt

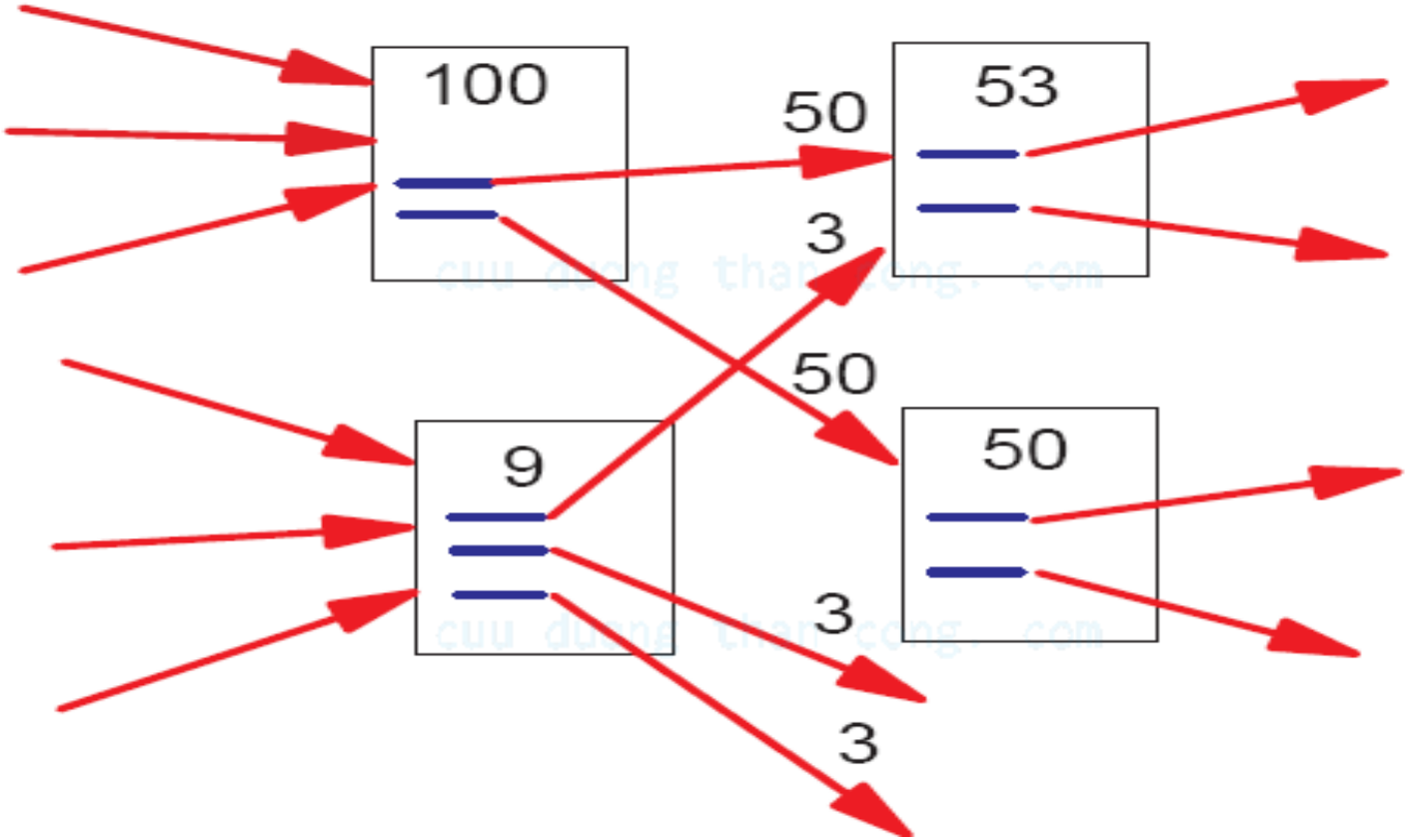


TÍNH HẠNG

- Hạng của trang web
  - Thuộc tính
    - "quan hệ" giữa các trang web
  - Theo nghĩa độ quan trọng
    - So sánh lẫn nhau
  - Sử dụng
    - Hiển thị khi trả lời người dùng
    - Khai thác, phát hiện các thuộc tính khác
- Phương pháp
  - Tính theo mô hình đồ thị web
  - Câu hỏi người dùng
- Là bài toán phổ dụng
  - Mạng phức hợp, mạng xã hội, mạng gene, đồ thị web

ĐỒ THỊ WEB

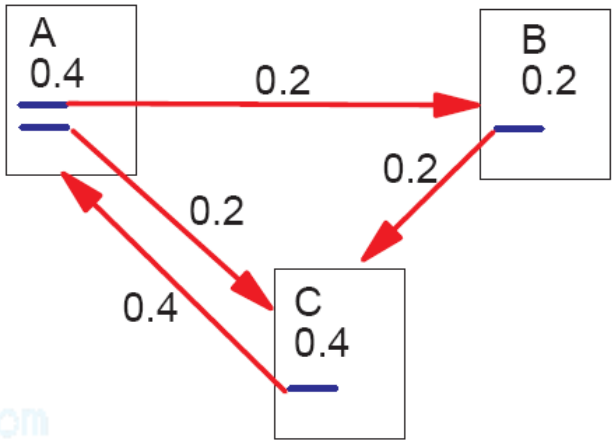
Chuyển giao hạng giữa các trang qua liên kết



TÍNH HẠNG ĐƠN GIẢN

- Công thức PageRank

$$r(i) \quad \sum_{j \ B(i)} r(j) / |N(j)|$$



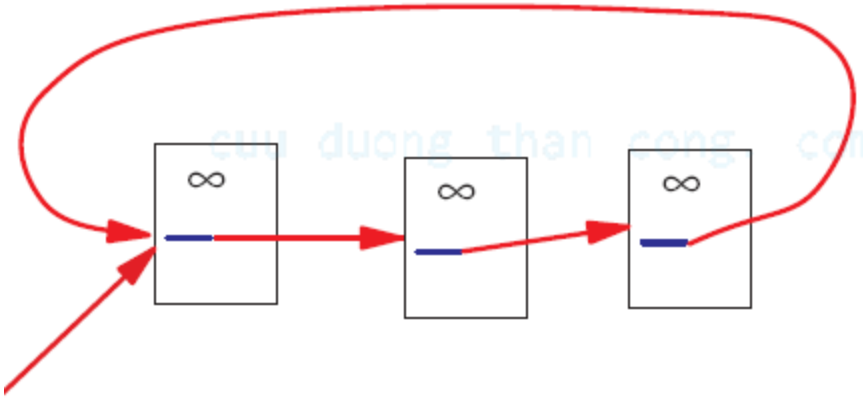
$$\leftrightarrow$$

$$r \quad A^T r$$

- N(i) : Số liên kết ra của trang i
- B(i) : Tập các trang có liên kết tới trang i
- $r_i = r(i)$: Hạng của trang i
- r là giá trị riêng của A^T
- Lặp để tính r.
- Vấn đề hội tụ ?

HẠNG TRANG ĐƠN GIẢN: TỒN TẠI

- Một số lưu ý
  - Chu trình: có thể lặp một vòng mãi mãi
  - Trang không có liên kết nào: hạng =0.
    - Trang nào cũng có ý nghĩa vì vậy hạng cần lớn hơn 0
- Cần công thức phức tạp hơn



HẠNG TRANG ĐƠN GIẢN: CẢI TIẾN

- Ma trận cải tiến được xây dựng từ ma trận đơn giản theo các bước:
  - Thêm 1/N vào hàng gồm toàn 0
  - Nhân ma trận với *d*
  - Cộng thêm giá trị *(1-d)/N*

$$\overline{A} \quad dA \quad (1 \quad d)(1/N)E$$

tồn tại vector PR riêng với $\overline{A}$

Lặp với vòng lặp 20

CÔNG THỨC CẢI TIẾN

- Công thức cải tiến

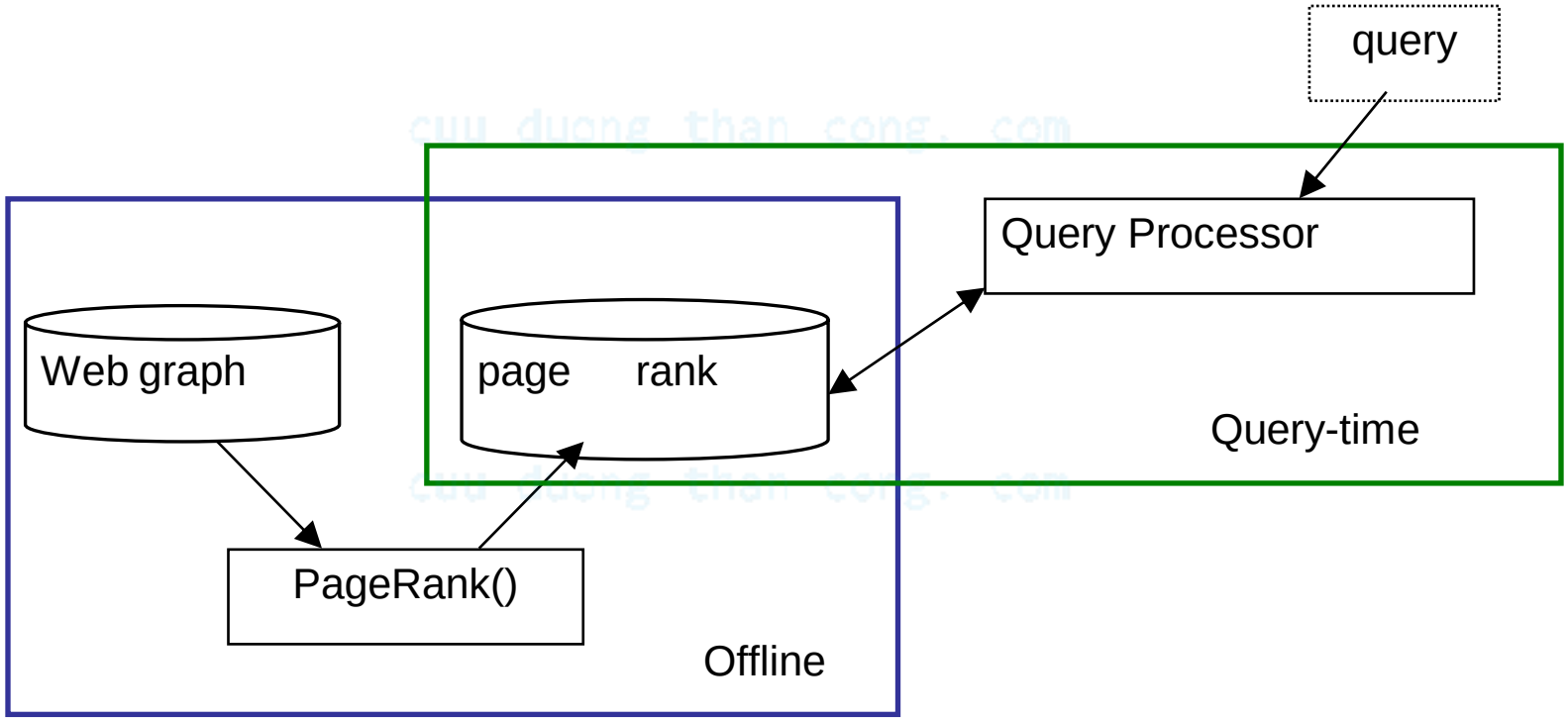
$$r(i) \quad d. \sum_{j \ B(i)} r(j) / |N(j)| \quad (1 \quad d)/n$$

- Độ hãm d (0.80-0.85)
  - Ma trận không suy biến Tồn tại vector riêng ổn định
  - Cơ sở toán học
    - Paolo Boldi, Massimo Santini and Sebastiano Vigna (2005). PageRank as a Function of the Damping Factor. *Proceedings of the 14th international conference on World Wide Web*, 557– 566, Chiba, Japan, 2005, ACM Press

HẠNG TRANG ĐƠN GIẢN VỚI CÂU TRUY VẤN

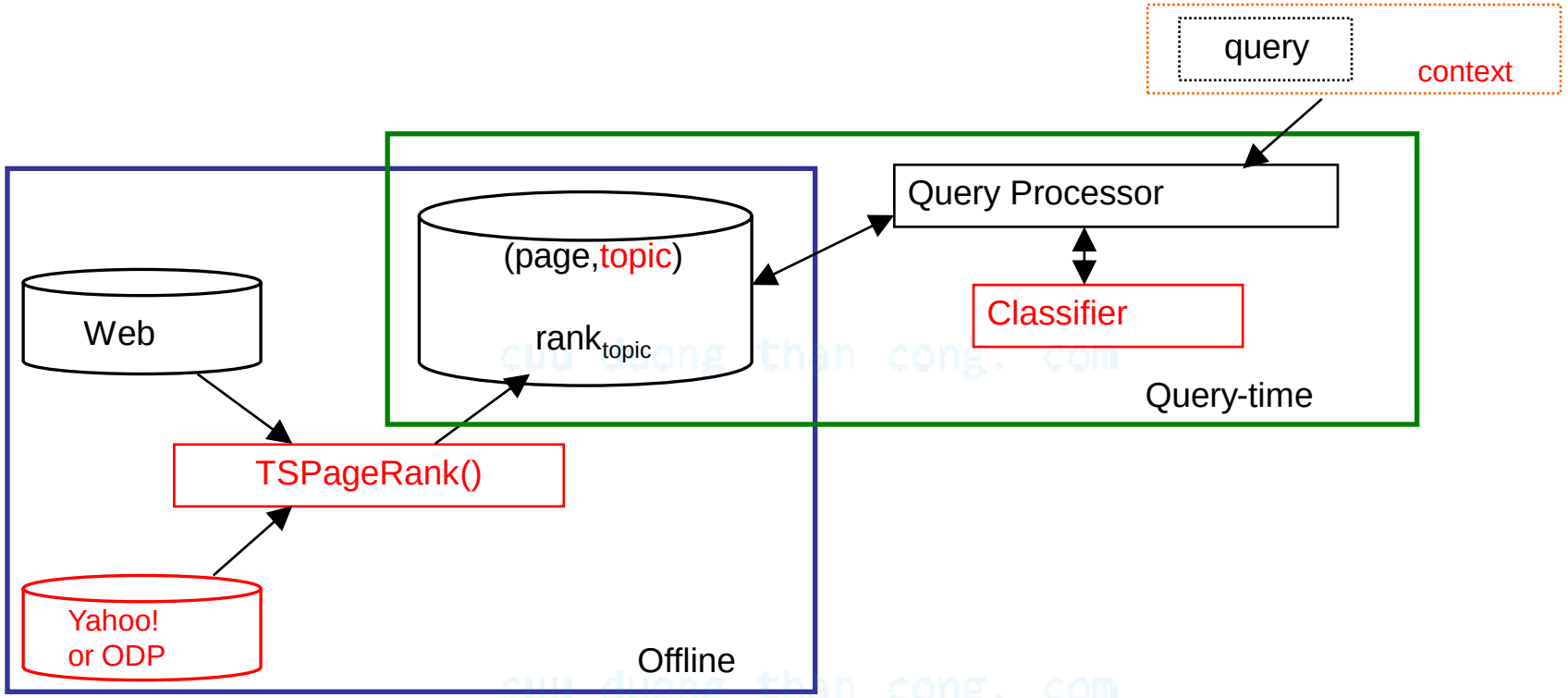
Nội dung

- Giá trị hạng PageRank
  - Được tính một lần
  - Ngoại tuyến cho toàn không gian Web, độc lập với câu truy vấn



PAGERANK HƯỚNG CHỦ ĐỀ

(Topic-Sensitive PageRank)



- Nội dung
  - Tính hạng dựa trên liên kết (Link-base score)
  - Quan tâm đến truy vấn (Topic-sensitive) lớp tài liệu lớp câu truy vấn
  - Tối thiểu truy vấn thời gian (Minimum query time processing)

HẠNG TRANG: MỘT SỐ NGHIÊN CỨU GẦN ĐÂY

- Bài toán thời sự trong máy tìm kiếm
  - Các nhóm nghiên cứu
    - Amy N. Langville, Carl D. Meyer
    - Paolo Boldi, Sebastiano Vigna
  - Các hội thảo quốc tế
    - Hội thảo WWW05
    - Fourteenth International World Wide Web Conference, Chiba, Japan, 2005
  - Khử spam
    - Tự nâng hạng trang
    - Một số phương pháp: nội dung (nội dung: (1) che phần nâng hạng, (2) giả dạng; liên kết: tăng liên kết tới)
- Áp dụng lĩnh vực khác
  - Mạng phức hợp: mạng xã hội, mạng gene
  - Vai trò các thực thể trong mạng

Tham khảo Nguyễn Thu Trang, Nguyễn Hoài Nam, Đặng Thanh Hải

Thuật toán PageRank

Một số bài báo Fan Chung Graham (Internet Mathematics)

http://math.ucsd.edu/~fan/

- PageRank and random walks on graphs , Proceedings of the "Fete of Combinatorics" conference in honor of Lovasz, to appear,
  (with Wenbo Zhao)
- Diameter of random spanning trees in a given graph , Journal of Graph Theory, to appeear,
  (with P. Horn and L. Lu)
- Small spectral gap in the combinatorial Laplacian implies Hamiltonian,
  Annals of Combinatorics, 13, (2010), 403--412.
  (with S. Butler)
- A local graph partitioning algorithm using heat kernel pagerank, WAW 2009, LNCS 5427, (2009), 62-75.
- A network color game , WINE 2008, Lecture Notes in Computer Science, Volume 5385 (2008), 522--530.
  (with K. Chaudhuri and M. S. Jamall)
- The giant component in a random subgraph of a given graph , Proceedings of WAW2009, Lecture Notes in Computer Science 5427, 38--49.
  (with P. Horn and L. Lu)
- PageRank as a discrete Green's function, *Geometry and Analysis*, I, ALM 17, (2010), 285--302.

Data-Aware Search on the Web

Act. 2: Entity Search

Are you searching for what you want?

Kevin C. Chang

WISDM
Web Indexing and Search for Data Mining

DAIS
The Database and Information Systems Laboratory
at The University of Illinois at Urbana-Champaign
Large Scale Information Management

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

What have you been searching lately?

- The **university** and **areas** of Kevin Chang?
- The **email** of Marc Snir?
- Customer service **phone** number of Amazon?
- What **profs** are doing databases at UIUC?
- The **papers** and **presentations** of SIGMOD 2007?
- Due **date** of SIGMOD 2008?
- Sale **price** of "Canon PowerShot A400"?
- "Hamlet" **books** available at bookstores?

WISDM
Web Indexing and Search

ILLINOIS

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

WISDM
Web Indexing and Search

ILLINOIS

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

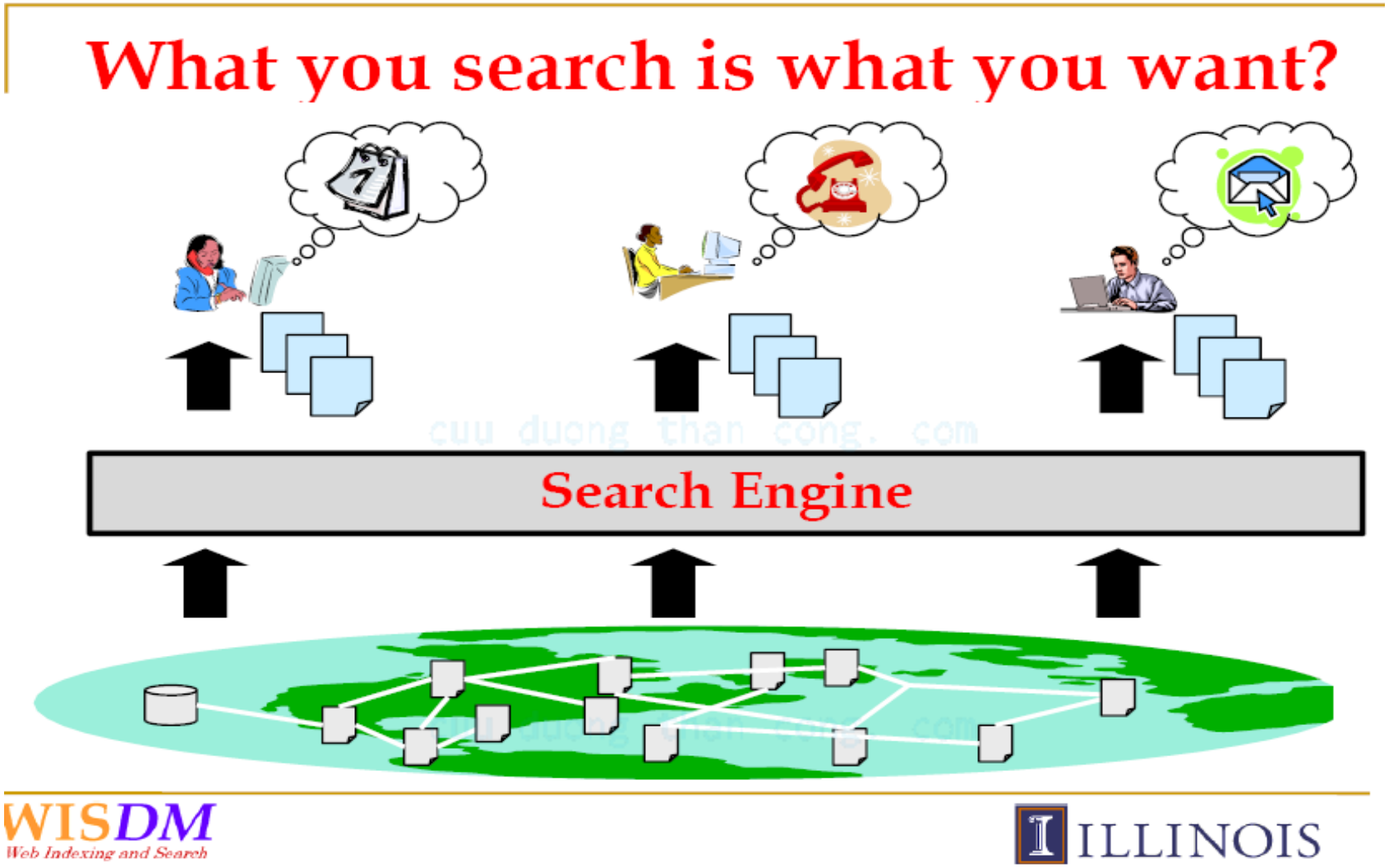
The Web is a Huge Supermarket!

WISDM
Web Indexing and Search

ILLINOIS

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE



PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

Our View–

Web is "interlinked pages" of "data objects".

WISDM
Web Indexing and Search

ILLINOIS

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

Challenges on the Web come in "dual":

Getting access to the structured information!

☞ Kevin's 4-quardants:

	Deep Web	*Surface Web*
Access		
Structure		

WISDM
Web Indexing and Search

ILLINOIS

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

Act 1. Challenge of the Deep Web:

MetaQuerier:

Data Hidden *behind* Query Forms.

MetaQuerier: Exploring and Integrating the Deep Web [*SIGMOD'03*, *SIGMOD'04*, *KDD'04*, *VLDB'05*, *CIDR'05*, *KDD'05*, *TODS'06*, *CACM'07*, *VLDB'07a*] (Demos: *SIGMOD'04*, *SIGMOD'05*, *ICDE'05*, *ICDE'07*)

WISDM
Web Indexing and Search

ILLINOIS

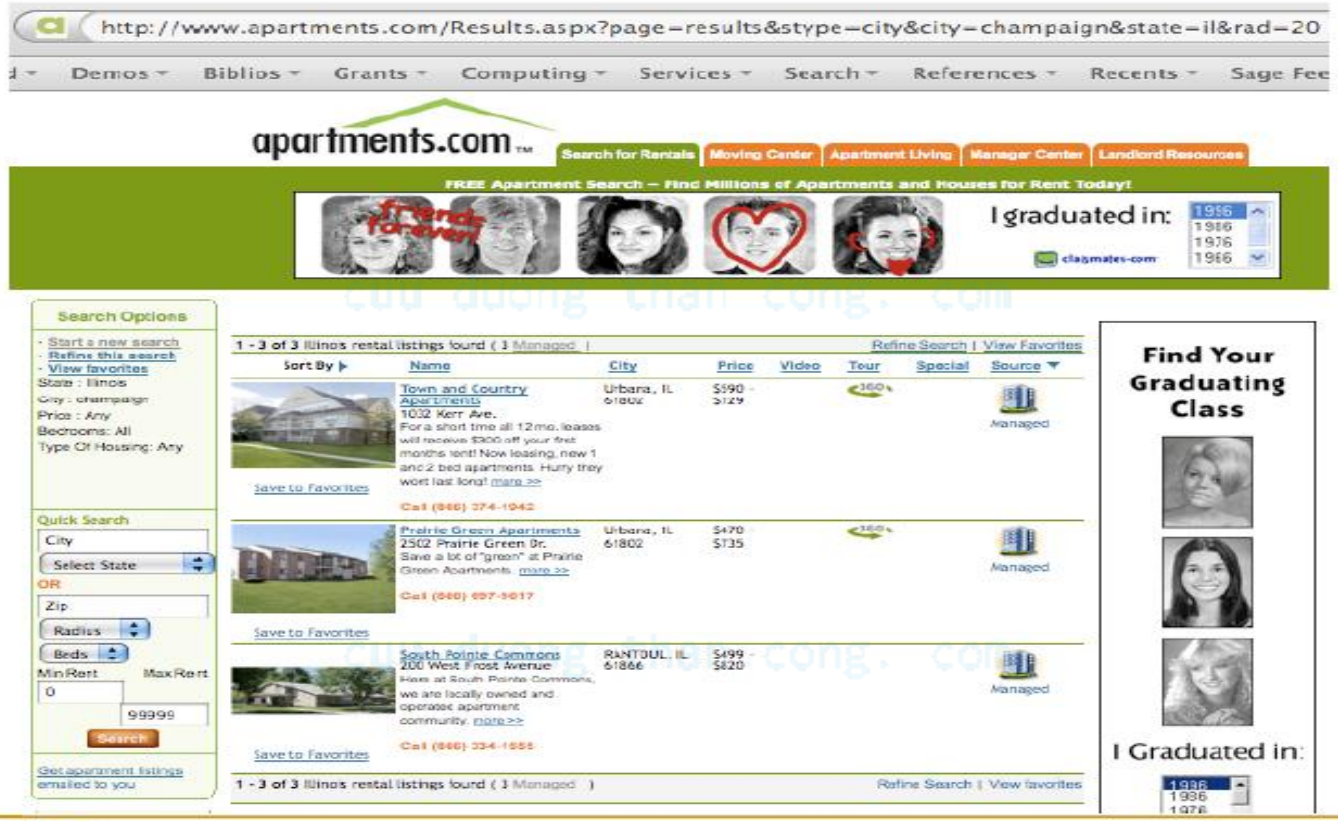
PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

Example Applications

CAZOODLE Apartment Search BETA

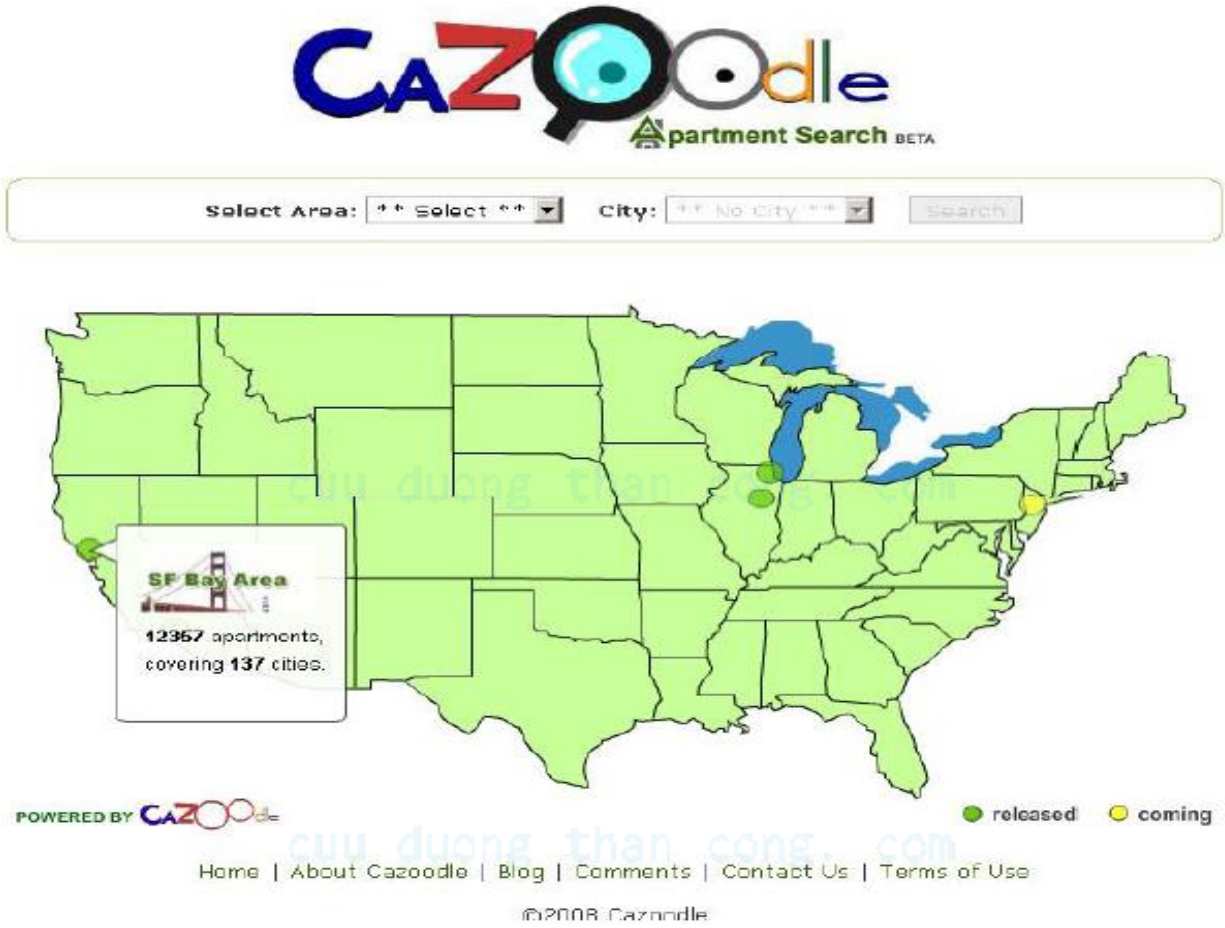
Current State of the Art



WISDM Web Indexing and Search

ILLINOIS

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE



WISDM
Web Indexing and Search

ILLINOIS

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

Act 2. Challenge of the Surface Web:

WISDM
Web Indexing and Search for Data Mining

WISDM:

Data Hidden *within* HTML Web Pages.

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

We take an **entity view** of the Web:

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

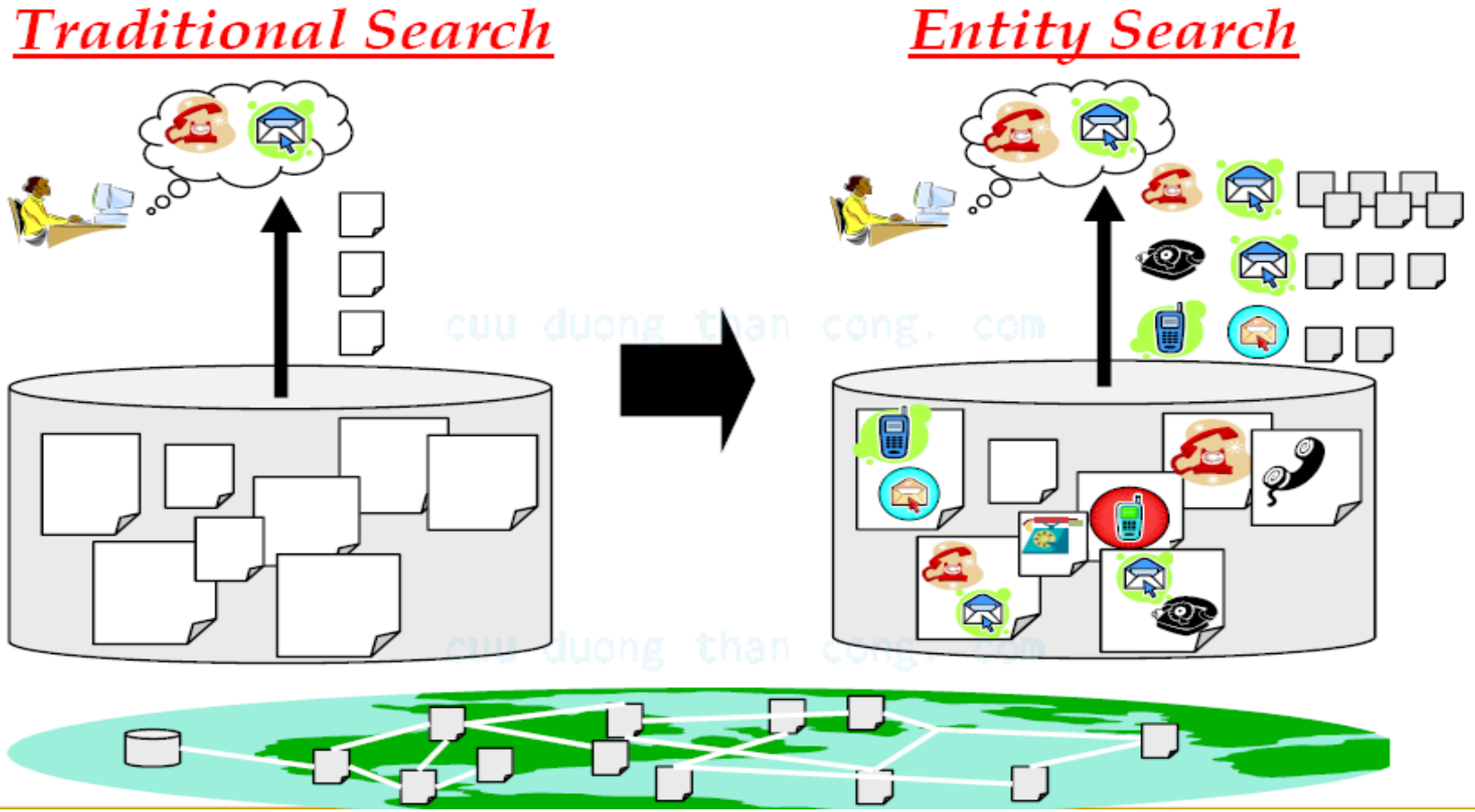
What is an "entity"?
Your target of information– or, anything.

- Phone number
- Email address
- PDF
- Image
- Person name
- Book title, author, …
- Price (of something)

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

From pages to entities

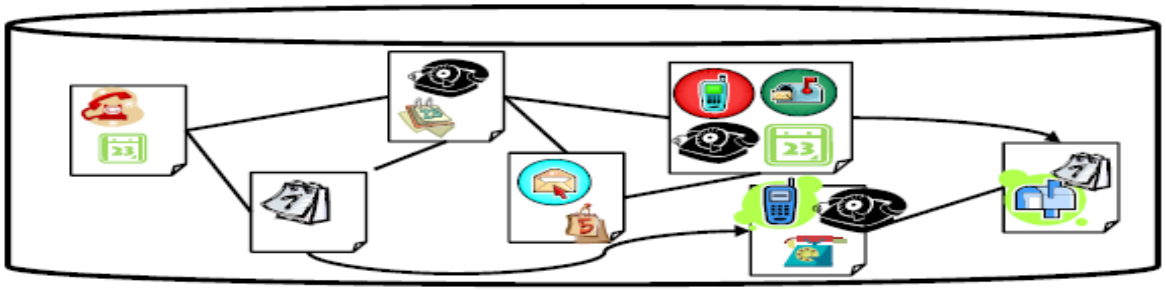


PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

TÌM KIẾM THỰC THỂ - ENTITY/OBJECT SEARCH ENGINE

Entity Search Problem:

Given:



Input: Keywords & Entities (optionally with a pattern)

E.g. Amazon Customer Service #phone

Output: Ranked Entity Tuples

	0.90
	0.80
	0.60
...	...

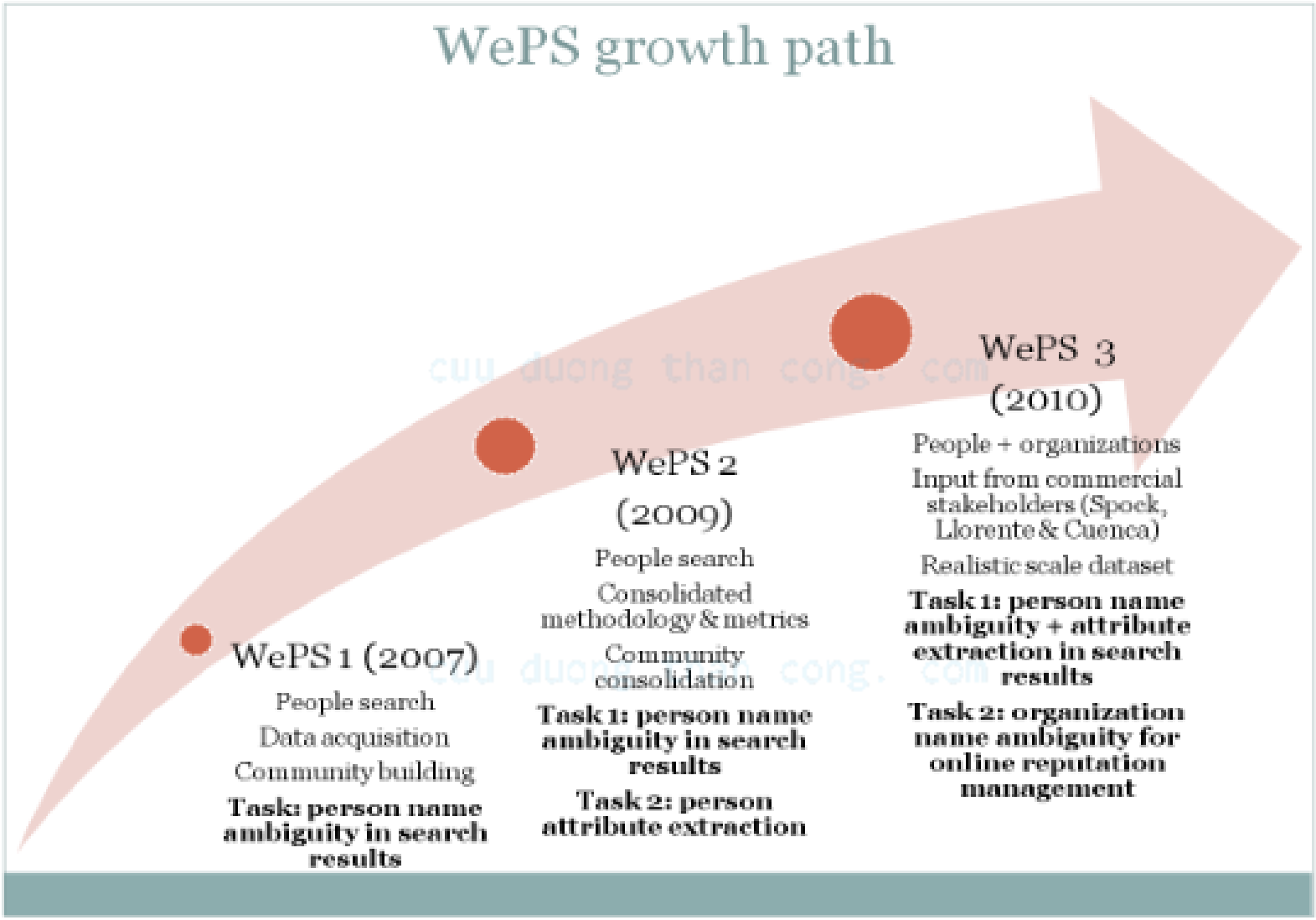
Kết quả trích chọn thông tin: chương 9.

PGS. Kevin C. Chang: Bài trình bày tại ĐHCN, ĐHQGHN ngày 08/7/2008

Hệ thống tìm kiếm thực thể người

- Dãy hội thảo WePS Web People Search
  - http://nlp.uned.es/weps/
  - WePS-1
    - June 23-24, 2007
    - Prague, Czech Republic
    - in association with Semeval/ACL 2007
  - WePS-2
    - April 21st
    - Madrid, Spain
    - Co-located with the WWW2009 conference
  - WePS-3
    - 23 September 2010
    - CLEF 2010 Lab in Padova (Italy)

WePS 123 - Web People Search



WePS 123 - Web People Search

- WePS-1
  - Bài toán: Person names disambiguation in a Web searching
  - 29 nhóm tham gia và 15 bài báo công bố
- WePS-2
  - Bài toán
    - Person names disambiguation in a Web searching
    - Person Attribute Extraction
  - 21 bài báo công bố
- WePS-3
  - Bài toán:
    - Kết hợp hai bài toán từ WePS-2: *? Tiểu sử người*
    - Mơ hồ tên "tổ chức" và quản lý danh tiếng "tổ chức"
  - 2 báo cáo mời và 13 báo cáo khác

WePS-1: Bài báo mô tả bài toán

- Task description
  - Bộ dữ liệu dùng thử: phiên bản chuyển thể WePS-06
  - Bộ dữ liệu học, kiểm tra: cùng cách lấy từ 3 nguồn (điều tra dân số Mỹ, Wikipedia tiếng Anh, và Ban chương trình hội nghị ECDL06)
  - Tiến hành chú giải
- Số nhóm tham gia và báo cáo
  - 29 nhóm
  - 16 báo cáo đúng hạn

[AGS07] Javier Artiles, Julio Gonzalo and Satoshi Sekine (2007). The SemEval-2007 WePS Evaluation: Establishing a benchmark for the Web People Search Task., *WePS-1*

WePS-1: Bài báo mô tả bài toán [AGS07]

Name	entities	documents	discarded
Wikipedia names			
John Kennedy	27	99	6
George Clinton	27	99	6
Michael Howard	32	99	8
Paul Collins	37	98	6
Tony Abbott	7	98	9
Alexander Macomb	21	100	14
David Lodge	11	100	9
Average	23,14	99,00	8,29
ECDL-06 Names			
Edward Fox	16	100	36
Allan Hanbury	2	100	32
Donna Harman	7	98	6
Andrew Powell	19	98	48
Gregory Crane	4	99	17
Jane Hunter	15	99	59
Paul Clough	14	100	35
Thomas Baker	60	100	31
Christine Borgman	7	99	11
Anita Coleman	9	99	28
Average	15,30	99,20	30,30
WEB03 Corpus			
Tim Whisler	10	33	8
Roy Tamashiro	5	23	6
Cynthia Voigt	1	405	314
Miranda Bollinger	2	2	0
Guy Dunbar	4	51	34
Todd Platts	2	239	144
Stacey Doughty	1	2	0
Young Dawkins	4	61	35
Luke Choi	13	20	6
Gregory Brennan	32	96	38
Ione Westover	1	4	0
Patrick Karlsson	10	24	8
Celeste Paquette	2	17	2
Elmo Hardy	3	55	15
Louis Sidoti	2	6	3
Alexander Markham	9	32	16
Helen Cawthorne	3	46	13
Dan Rhone	2	4	2
Maile Doyle	1	13	1
Alice Gilbreath	8	74	30
Sidney Shorter	3	4	0
Alfred Schroeder	35	112	58
Cathie Ely	1	2	0
Martin Nagel	14	55	31
Abby Watkins	13	124	35
Mary Lemanski	2	152	78
Gillian Symons	3	30	6
Pam Tetu	1	4	2
Guy Crider	2	2	0
Armando Valencia	16	79	20
Hannah Bassham	2	3	0
Charlotte Bergeron	5	21	8
Average	5,90	47,20	18,00
Global average	10,76	71,02	26,00

- Task description
  - Bộ dữ liệu dùng thử: phiên bản chuyển thể

WePS-1: Bài báo mô tả bài toán [AGS07]

Name	entities	documents	discarded
Wikipedia names			
John Kennedy	27	99	6
George Clinton	27	99	6
Michael Howard	32	99	8
Paul Collins	37	98	6
Tony Abbott	7	98	9
Alexander Macomb	21	100	14
David Lodge	11	100	9
Average	23,14	99,00	8,29
ECDL-06 Names			
Edward Fox	16	100	36
Allan Hanbury	2	100	32
Donna Harman	7	98	6
Andrew Powell	19	98	48
Gregory Crane	4	99	17
Jane Hunter	15	99	59
Paul Clough	14	100	35
Thomas Baker	60	100	31
Christine Borgman	7	99	11
Anita Coleman	9	99	28
Average	15,30	99,20	30,30
WEB03 Corpus			
Tim Whisler	10	33	8
Roy Tamashiro	5	23	6
Cynthia Voigt	1	405	314
Miranda Bollinger	2	2	0
Guy Dunbar	4	51	34
Todd Platts	2	239	144
Todd Platts	2	239	144
Stacey Doughty	1	2	0
Young Dawkins	4	61	35
Luke Choi	13	20	6
Gregory Brennan	32	96	38
Ione Westover	1	4	0
Patrick Karlsson	10	24	8
Celeste Paquette	2	17	2
Elmo Hardy	3	55	15
Louis Sidoti	2	6	3
Alexander Markham	9	32	16
Helen Cawthorne	3	46	13
Dan Rhone	2	4	2
Maile Doyle	1	13	1
Alice Gilbreath	8	74	30
Sidney Shorter	3	4	0
Alfred Schroeder	35	112	58
Cathie Ely	1	2	0
Martin Nagel	14	55	31
Abby Watkins	13	124	35
Mary Lemanski	2	152	78
Gillian Symons	3	30	6
Pam Tetu	1	4	2
Guy Crider	2	2	0
Armando Valencia	16	79	20
Hannah Bassham	2	3	0
Charlotte Bergeron	5	21	8
Average	5,90	47,20	18,00
Global average	10,76	71,02	26,00

Bộ dữ liệu học: số thực thể, số tài liệu, độ thải hồi(?)

WePS-1: Bài báo mô tả bài toán [AGS07]

Name	entities	documents	discarded
Wikipedia names			
Arthur Morgan	19	100	52
James Morehead	48	100	11
James Davidson	59	98	16
Patrick Killen	25	96	4
William Dickson	91	100	8
George Foster	42	99	11
James Hamilton	81	100	15
John Nelson	55	100	25
Thomas Fraser	73	100	13
Thomas Kirk	72	100	20
Average	56,50	99,30	17,50
ACL06 Names			
Dekang Lin	1	99	0
Chris Brockett	19	98	5
James Curran	63	99	9
Mark Johnson	70	99	7
Jerry Hobbs	15	99	7
Frank Keller	28	100	20
Leon Barrett	33	98	9
Leon Barrett	33	98	9
Robert Moore	38	98	28
Sharon Goldwater	2	97	4
Stephen Clark	41	97	39
Average	31,00	98,40	12,80
US Census Names			
Alvin Cooper	43	99	9
Harry Hughes	39	98	9
Jonathan Brooks	83	97	8
Jude Brown	32	100	39
Karen Peterson	64	100	16
Marcy Jackson	51	100	5
Martha Edwards	82	100	9
Neil Clark	21	99	7
Stephan Johnson	36	100	20
Violet Howard	52	98	27
Average	50,30	99,10	14,90
Global average	45,93	98,93	15,07

Bộ dữ liệu kiểm tra

WePS-1: Bài báo mô tả bài toán [AGS07]

$$\text{Precision}(C_i, L_j) = \frac{|C_i \cap L_j|}{|C_i|}$$

$$\text{Purity} = \sum_i \frac{|C_i|}{n} \max \text{Precision}(C_i, L_j)$$

$$\text{Inverse Purity} = \sum_j \frac{|L_i|}{n} \max \text{Precision}(L_i, C_j)$$

$$\text{F} = \frac{1}{\alpha \frac{1}{\text{Purity}} + (1-\alpha) \frac{1}{\text{Inverse Purity}}}$$

Các độ đo đánh giá

- Độ chính xác: có từ tìm kiếm thông tin
- C: tập cụm được đánh giá; L: tập các lớp (chú giải bằng tay), n: số phần tử được phân cụm
- = 0.2 : IP trọng số cao hơn P (độ thuần khiết)

WePS-1: Bài báo mô tả bài toán [AGS07]

		Macro-averaged Scores			
		F-measures			
rank	team-id	$\alpha = .5$	$\alpha = .2$	Pur	Inv_Pur
1	CU_COMSEM	.78	.83	.72	.88
2	IRST-BP	.75	.77	.75	.80
3	PSNUS	.75	.78	.73	.82
4	UVA	.67	.62	.81	.60
5	SHEF	.66	.73	.60	.82
6	FICO	.64	.76	.53	.90
7	UNN	.62	.67	.60	.73
8	ONE-IN-ONE	.61	.52	1.00	.47
9	AUG	.60	.73	.50	.88
10	SWAT-IV	.58	.64	.53	.71
11	UA-ZSA	.58	.60	.58	.64
12	TITPI	.37	.71	.43	.80
13	JHU1-13	.53	.65	.43	.82
14	DFKI2	.50	.63	.39	.83
15	WIT	.49	.66	.36	.93
16	UC3M_13	.48	.66	.35	.95
17	UBC-AS	.40	.53	.30	.91
18	ALL-IN-ONE	.40	.52	.29	1.00

Table 3: Team ranking

Kết quả đánh giá các đội

Lưu ý ONE-IN-ONE: Các thực thể mà chỉ một tài liệu đại diện

WePS-1: 13 bài báo mô tả hệ thống

1. **PSNUS: Web People Name Disambiguation by Simple Clustering with Rich Features.** *Ergin Elmacioglu, Yee Fan Tan, Su Yan, Min-Yen Kan and Dongwon Lee;* The Pennsylvania State University, (Mỹ) + NUS
2. **AUG: A combined classification and clustering approach for web people disambiguation.***Els Lefever, Véronique Hoste and Timur Fayruzov;* Ghent University Association; Mỹ ?
3. **CU-COMSEM: Exploring Rich Features for Unsupervised Web Personal Name Disambiguation.***Ying Chen and James H. Martin;* University of Colorado at Boulder, Bỉ
4. **DFKI2: An Information Extraction Based Approach to People Disambiguation.***Andrea Heyl and Günter Neumann;* Artificial Intelligence – DFKI, Đức
5. **FICO: Web Person Disambiguation Via Weighted Similarity of Entity Contexts.***Paul Kalmar and Matthias Blume,* Fair Isaac Corporation, Mỹ
6. **IRST-BP: Web People Search Using Name Entities.***Octavian Popescu and Bernardo Magnini;* FBK-irst, Trento (Italy)
7. **JHU1 : An Unsupervised Approach to Person Name Disambiguation using Web Snippets.***Delip Rao, Nikesh Garera and David Yarowsky;* Johns Hopkins University (Mỹ?)

WePS-1: 13 bài báo mô tả hệ thống

8. **SHEF: Semantic Tagging and Summarization Techniques Applied to Cross-document Coreference.***Horacio Saggion;* University of Sheffield, Anh
9. **TITPI: Web People Search Task Using Semi-Supervised Clustering Approach.***Kazunari Sugiyama and Manabu Okumura;* Tokyo Institute of Technology
10. **UA-ZSA: Web Page Clustering on the basis of Name Disambiguation.** *Zornitsa Kozareva, Sonia Vazquez and Andres Montoyo,* University of Alicante, Tây Ban Nha
11. **UC3M_13: Disambiguation of Person Names Based on the Composition of Simple Bags of Typed Terms.***David del Valle-Agudo, César de Pablo-Sánchez and María Teresa Vicente-Díez;* Universidad Carlos III de Madrid, Tấy Ban Nha
12. **UNN-WePS: Web Person Search using co-Present Names and Lexical Chains.***Jeremy Ellman and Gary Emery;* Northumbria University, Anh
13. **UVA: Language Modeling Techniques for Web People Search.** *Krisztian Balog, Leif Azzopardi and Maarten de Rijke;* University of Amsterdam, Hà Lan
14. **WIT: Web People Search Disambiguation using Random Walks.***José Iria, Lei Xia and Ziqi Zhang;* The University of Sheffield, Anh

WePS-2: mô tả bài toán và độ đánh giá

- Mô tả bài toán (2 bài báo)
  - WePS 2 Evaluation Campaign: Overview of the Web People Search Clustering Task, *Javier Artiles, Julio Gonzalo and Satoshi Sekine*
  - WePS 2 Evaluation Campaign: Overview of the Web People Search Attribute Extraction Task, *Satoshi Sekine and Javier Artiles.*
- Độ đo đánh giá (1 bài báo)
  - Combining Evaluation Metrics with a Unanimous Improvement Ratio and its Application to the Web People Search Clustering Task. *Enrique Ámigo, Javier Artiles and Julio Gonzalo.*

WePS-2:

[UVA] The University of Amsterdam at WePS2. *Krisztian Balog, Jiyin He, Katja Hofmann, Valentin Jijkoun, Christof Monz, Manos Tsagkias, Wouter Weerkamp and Maarten de Rijke.*

[UPM] Learning by doing: A baseline approach to the clustering of web people search results. *José Carlos González, Pablo Maté, Laura Vadillo, Rocío Sotomayor and Álvaro Carrera.*

- **[PolyUHK] PolyUHK: A Robust Information Extraction System for Web Personal Names.** *Ying Chen, Sophia Yat Mei Lee and Chu-Ren Huang,* The Hong Kong Polytechnic University

[UMD] Determine the Entity Number in Hierarchical Clustering for Web Personal Name Disambiguation. *Jun Gong and Douglas Oard.*

[CASIANED] CASIANED: Web Personal Name Disambiguation Based on Professional Categorization. *Xianpei Han and Jun Zhao.*

[CASIANED] CASIANED: People Attribute Extraction based on Information Extraction. *Xianpei Han and Jun Zhao.*

[ITC_UT] Person Name Disambiguation on the Web by TwoStage Clustering. *Masaki Ikeda, Shingo Ono, Issei Sato, Minoru Yoshida and Hiroshi Nakagawa.*

[FICO] Features for Web Person Disambiguation. *Paul Kalmar and Dayne Freitag.*

[ECNU] Which Who are They? People Attribute Extraction and Disambiguation in Web Search Results., *Man Lan, Yu Zhe Zhang, Yue Lu, Jian Su and Chew Lim Tan,* East China Normal University

WePS-2:

[AUG] Fuzzy Ants Clustering for Web People Search. *Els Lefever, Timur Fayruzov, Véronique Hoste and Martine De Cock.*

[UNED] Web People Search Disambiguation using Language Model Techniques. *Juan Martinez-Romo and Lourdes Araujo.*

[UCI] Exploiting Web querying for Web People Search in WePS2. *Rabia Nuray-Turan, Zhaoqi Chen, Dmitri Kalashnikov and Sharad Mehrotra.*

[UC3M] UC3M at WePS2-AE: Acquiring Patterns for People Attribute Extraction from Webpages. *César de Pablo Sánchez and Paloma Martínez Fernández.*

[BUAP] An Unsupervised Approach based on Fingerprinting to the Web People Search task. *David Pinto, Mireya Tovar, Darnes Vilariño, Héctor Díaz and Héctor Jiménez-Salazar.*

[XMedia] XMedia: Web People Search by Clustering with Machinely Learned Similarity Measures. *Lorenza Romano, Krisztian Buza, Claudio Giuliano and Lars Schmidt-Thieme.*

[GUELP] Web People Search Based on Locality and Relative Similarity Measures. *Fei Song, Robin Cohen and Song Lin.*

[PRIYAVEN] Clustering Web People Search Results Using Fuzzy Ant-Based Clustering. *Priya Venkateshan.*

[MIVTU] A Two-Step Approach to Extracting Attributes for People on the Web. *Keigo Watanabe, Danushka Bollegala, Yutaka Matsuo and Mitsuru Ishizuka.*

WePS-2: Bài về độ đo

F function [10]:

$$F_\alpha(A,B) = A * B/(\alpha * R + (1-\alpha) * P)$$

Thường kết hợp các chỉ số C. Van Rijsbergen (1974) như công thức trên (với R: Độ hồi tưởng và P: độ chính xác).

Mục đích bài báo: Nghiên cứu tác động ảnh hưởng của các trọng số

Đề xuất độ đo nâng cao chất lượng đánh giá không phụ thuộc vào trọng số.

Cải thiện theo nhất trí: Unanimous Improvements

$$Q_X(a) \geq_\forall Q_X(b) \text{ if and only if } x(a) \geq x(b) \forall x \in X$$

Hệ số cải thiện theo nhất trí: Unanimous Improvements Ratio

$$\mathrm{UIR}_{X,T}(a,b) = \frac{|T_{a \geq_\forall b}| - |T_{b \geq_\forall a}|}{|T|} = \frac{|t \in T / Q_X(a) \geq_\forall Q_X(b)| - |t \in T / Q_X(b) \geq_\forall Q_X(a)|}{|T|}$$

Mô hình Meta-Search [Glo01]

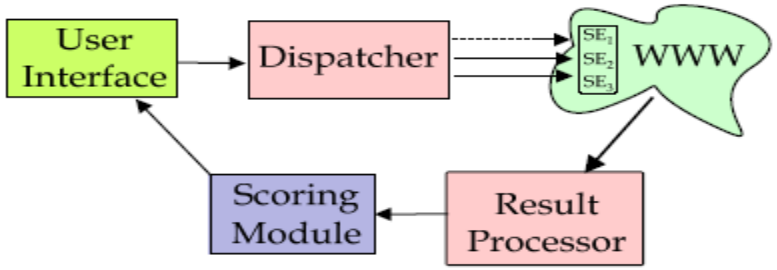


Figure 3.1: Architecture of a web metasearch engine.

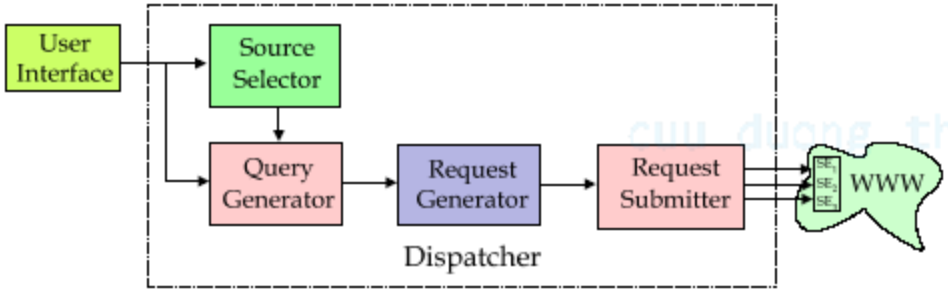


Figure 3.2: A possible design of a dispatcher.

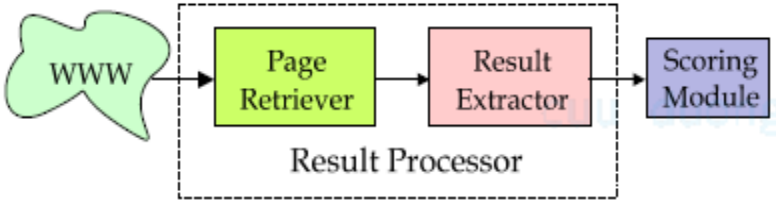


Figure 3.3: Two subcomponents of a generic result processor.

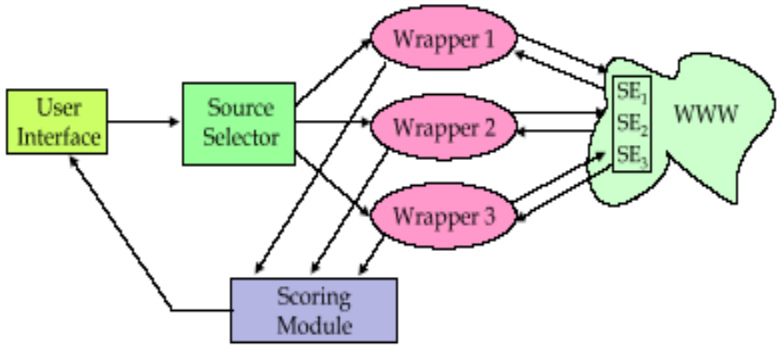


Figure 3.4: A metasearch engine based on wrappers.

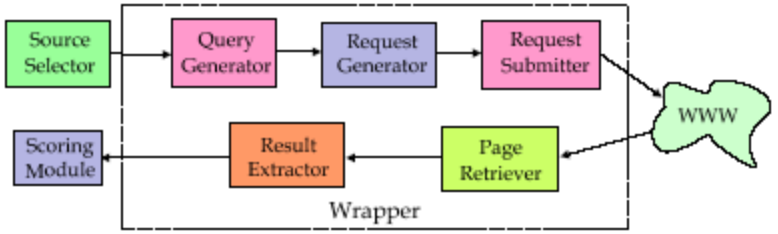


Figure 3.5: The internal subcomponents of a wrapper.

Eric J. Glover (2001). Using Extra-Topical User Preferences To Improve Web-Based Metasearch. *PhD Thesis,* The University of Michigan.

Mô hình Meta-Search [ME08]

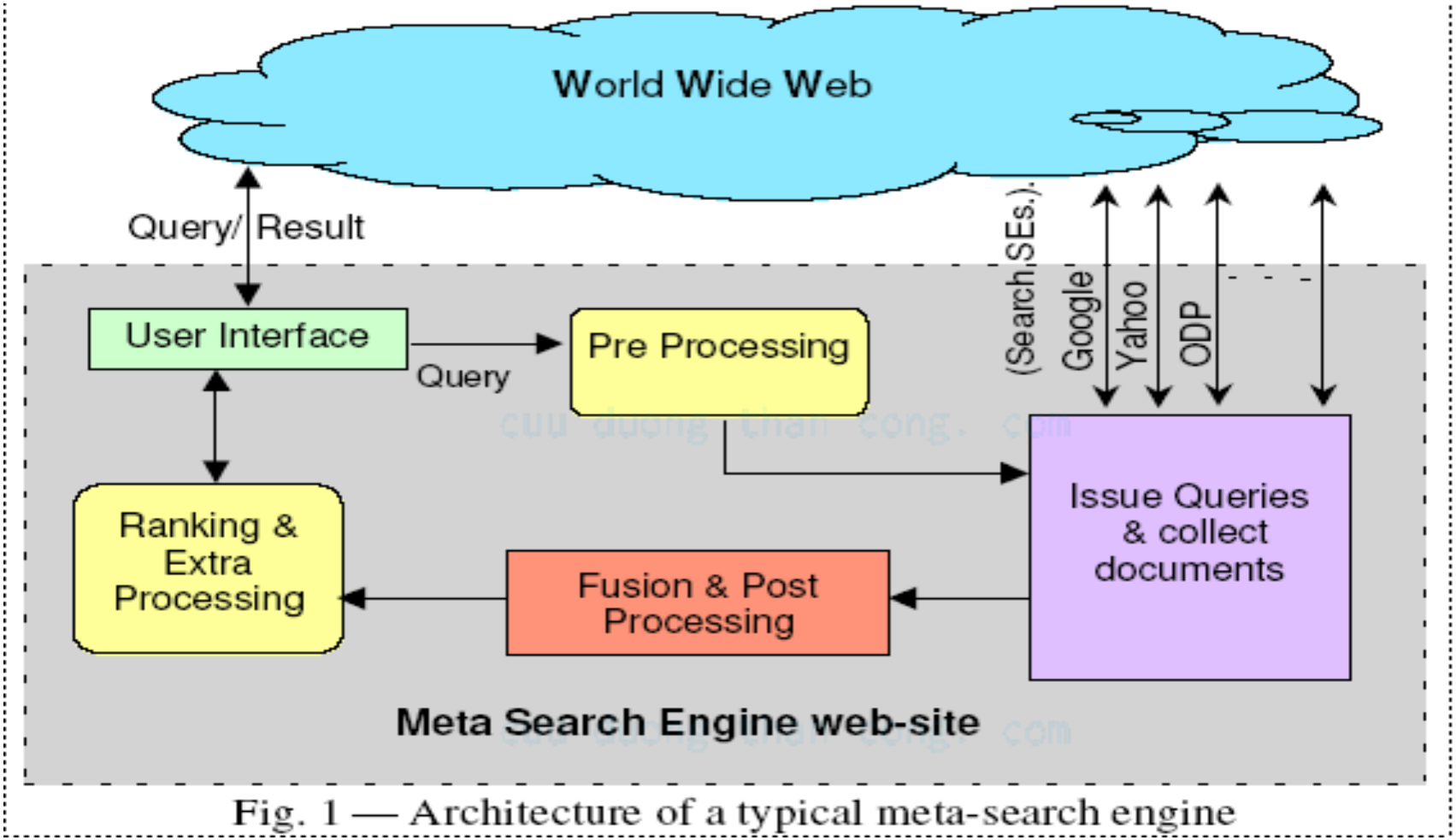


Fig. 1 — Architecture of a typical meta-search engine

[ME08] Manoj M, Elizabeth Jacob (2008). A Personalized Search Engine Based on MS engine: an overview, *J. of Scientific & Industries Research*, **67**: 379-386.

MÁY TÌM KIẾM Ở VIỆT NAM

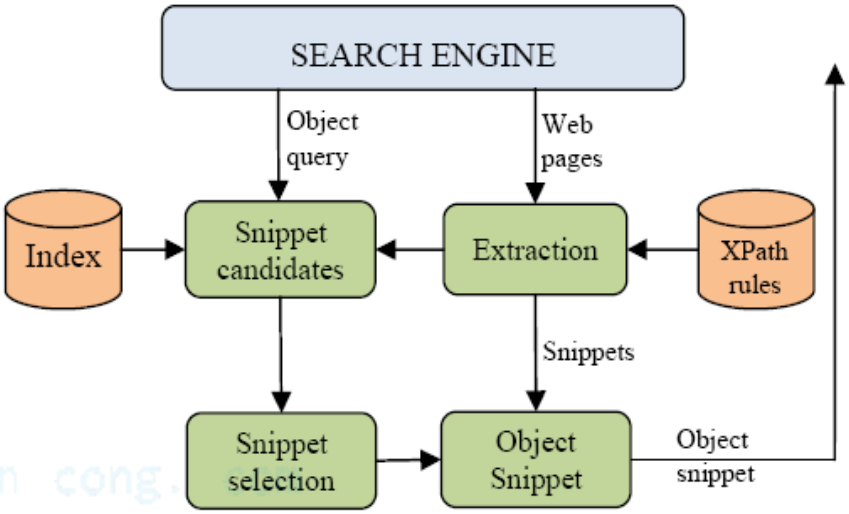
- Một số máy tìm kiếm TV trước đây:
  - **PanVietnam** của Netnam: giải 2002, cầm chừng
  - **VinaSEEK** của Tinh Vân: máy (200 tr), chạy như hiện nay
  - **Hoa tiêu** của FPT (Vương Quang Khải): nổi, cáo lui 8/2002
- Một số máy tìm kiếm hiện nay
  - Xalo.vn (http://www.xalo.com): C/ty Tinh Vân
  - Socbay (http://www.socbay.com): C/ty NAISCORP
  - Baamboo (http://www.baamboo.com): C/ty CP Truyền thông Việt Nam
- Một số nhận định trên mạng
  - Search của Việt Nam: Cảnh chợ chiều!
  - Bùi Dũng: *Google đã "nuốt chửng" công cụ tìm kiếm Việt Nam?* (http://vietnamnet.vn/cntt/2005/11/517349/)
  - Vẫn duy trì công cụ tìm kiếm của Việt Nam!
  - Lê Ngọc Quang (IDG Ventures Vietnam)
    - *gần như bỏ không ; không tạo doanh thu ;rất ít người dùng -> lãng phí*
  - "Công cụ tìm kiếm Việt Nam: Một cổ hai tròng" http://sggp.org.vn/dientutinhoc/2010/9/236722/

COLTECH-DM: CÁC MÁY TÌM KIẾM

- Xuất xứ
  - Vietseek do Bùi Quang Minh, 2002
  - Trên cơ sở ASPseek
  - Chạy thử trên cổng VDC với 2 triệu trang web
  - Module tiếng Việt
- Dự án Vinahoo (VNSEN)
  - Nhóm nghiên cứu "Khai phá dữ liệu"
  - Hoạt động và dự kiến
- Máy tìm kiếm thực thể
  - Liên kết với nhóm UIUC (PGS. Chang C. Kevin)
  - Tìm kiếm người, đa phương tiện.

COLTECH-DM:MÁY TÌM KIẾM VIETSEEK-2002

COLTECH-DM: MÁY TÌM KIẾM THỰC THỂ



Real estate Search

Location	hoang mai			
Property Type				
Price	min	10000000	max	1000000000
Bedrooms	min	2	max	9
Bathrooms	min		max	
Area	min		max	

Search Clear

Apartments for Rent: 229 C5 Dai Kim - 9a13e5ed

...Location: 229 C5 New Urban Dai Kim, **Hoang Mai** District, Ha Noi. Property type: Villas Price: **15,000,000 VND** Bedrooms: **8** Bathrooms: 5 Area: 320 m2...

www.metvuong.com - Original - Cached

House for Sale, House for Rent, Villas, Apartments, ...

...Location: 25 Tan Mai, Tan Mai 25 Tan Mai, **Hoang Mai** District, Ha Noi. Property type: Apartments, Building Price: **1,000,000,000 VND** . Built year: 1996 **2** bedrooms 2, bathrooms. Area: 88.00 m2, Garage...

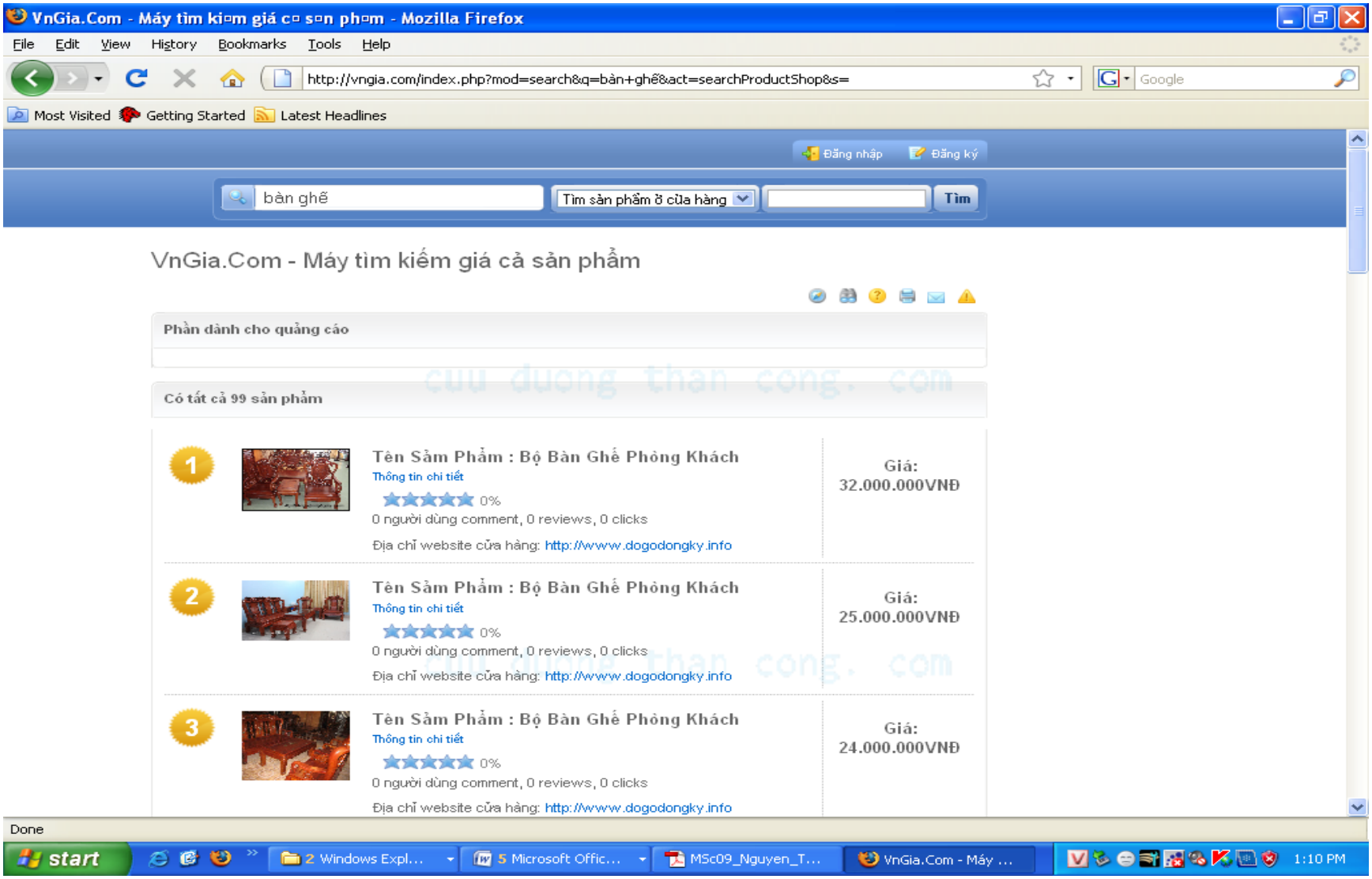
www.nhadat.com - Original - Cached

Cccca Nha Dana Van Phong Khac

...$/m2 rent luxury apartments **Hoang Mai** Ha Noi. Category: ...:Thanh Xuan, Ha Noi. Price: **16,200,000 vnd** / m2. Virtual Office in district...Details: Apartment 89 m2 with 2 bedrooms clear designed system

www.vanphongtrongoi.com - Original - Cached

COLTECH-DM: TÌM KIẾM GIÁ CẢ SẢN PHẨM



BÀI GIẢNG KHAI PHÁ DỮ LIỆU WEB

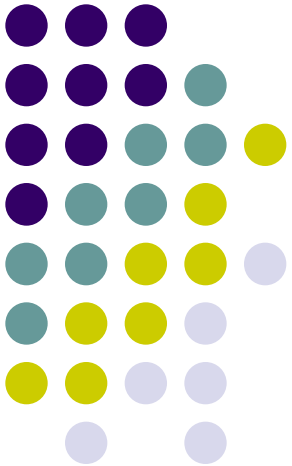
CHƯƠNG 7. PHÂN LỚP WEB

PGS. TS. HÀ QUANG THỤY

HÀ NỘI 10-2010

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

ĐẠI HỌC QUỐC GIA HÀ NỘI

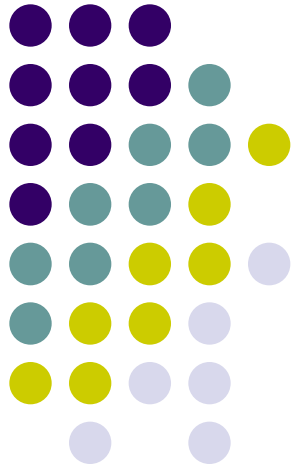


Nội dung

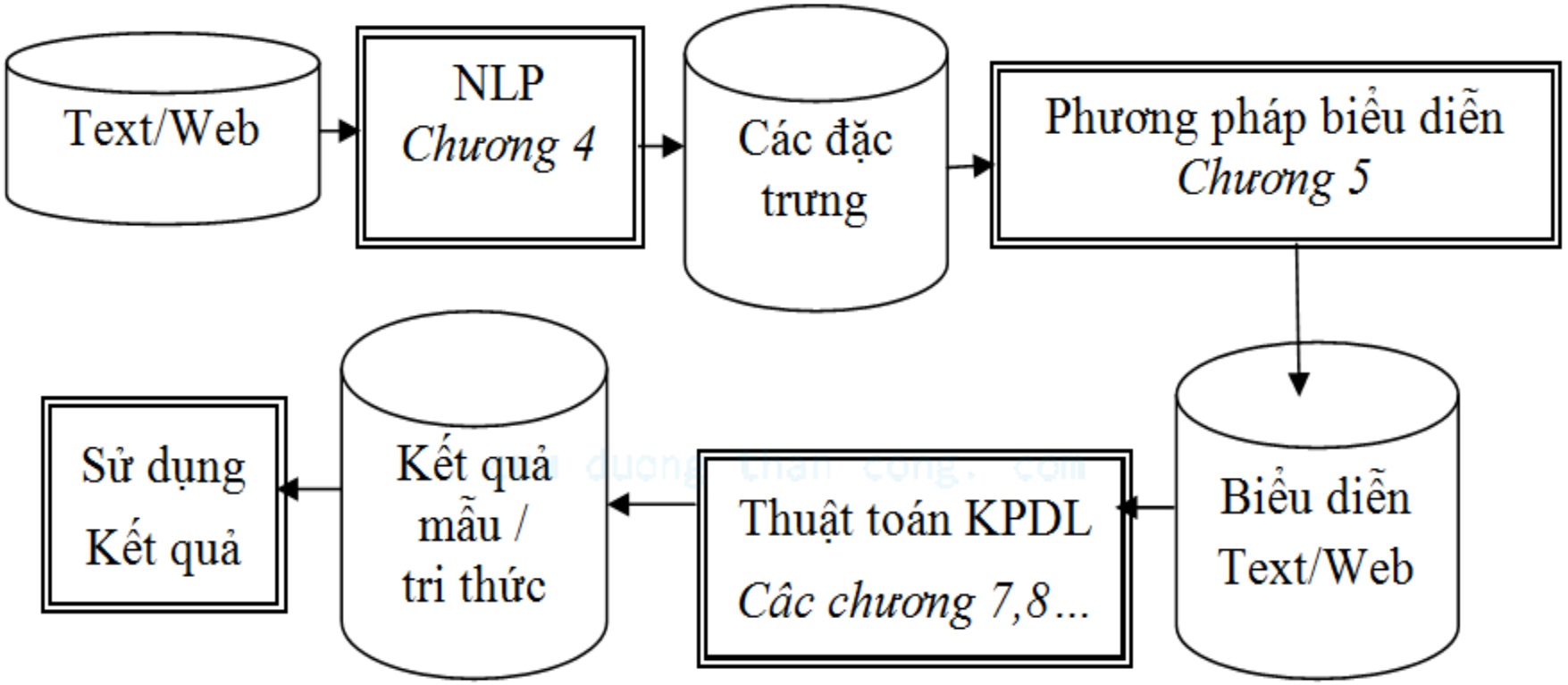
Giới thiệu phân lớp Web

Phân lớp học giám sát

Phân lớp học bán giám sát

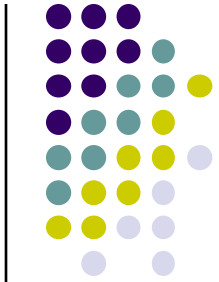


Giới thiệu: Sơ đồ khai phá Web



- Thuật toán KPDL: phân lớp, phân cụm, tóm tắt… Sử dụng các thuật toán KPDL chung (phân lớp, phân cụm…)
- Chọn các đặc trưng, chọn cách biểu diễn Web đóng vai trò quan trọng trong KPDL Web: Chương 4 và Chương 5.
- Các chương: phát biểu bài toán và một số thuật toán KPDL điển hình

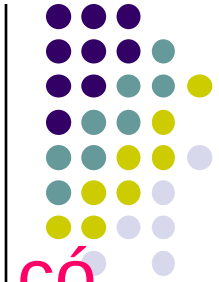
Bài toán phân lớp Web



- Đầu vào
  - Tập tài liệu web D = $\{d_i\}$
  - Tập các lớp $C_1, C_2, \ldots, C_k$ mỗi tài liệu d thuộc một lớp C_i
  - Tập ví dụ $D_{exam} = D_1 + D_2 + \ldots + D_k$ với $D_i = \{d \quad D_{exam}$: d thuộc $C_i\}$
  - Tập ví dụ D_{exam} đại diện cho tập D
- Đầu ra
  - Mô hình phân lớp: ánh xạ từ D sang C
- Sử dụng mô hình
  - d D \ D_{exam} : xác định lớp của tài liệu d
- Ví dụ
  - Crawler hướng chủ đề: Chủ đề Lớp
  - Phân lớp/phân cụm tập trang Web trả về "chủ đề/lớp

cuu duong than cong. com

Phân lớp: Quá trình hai pha



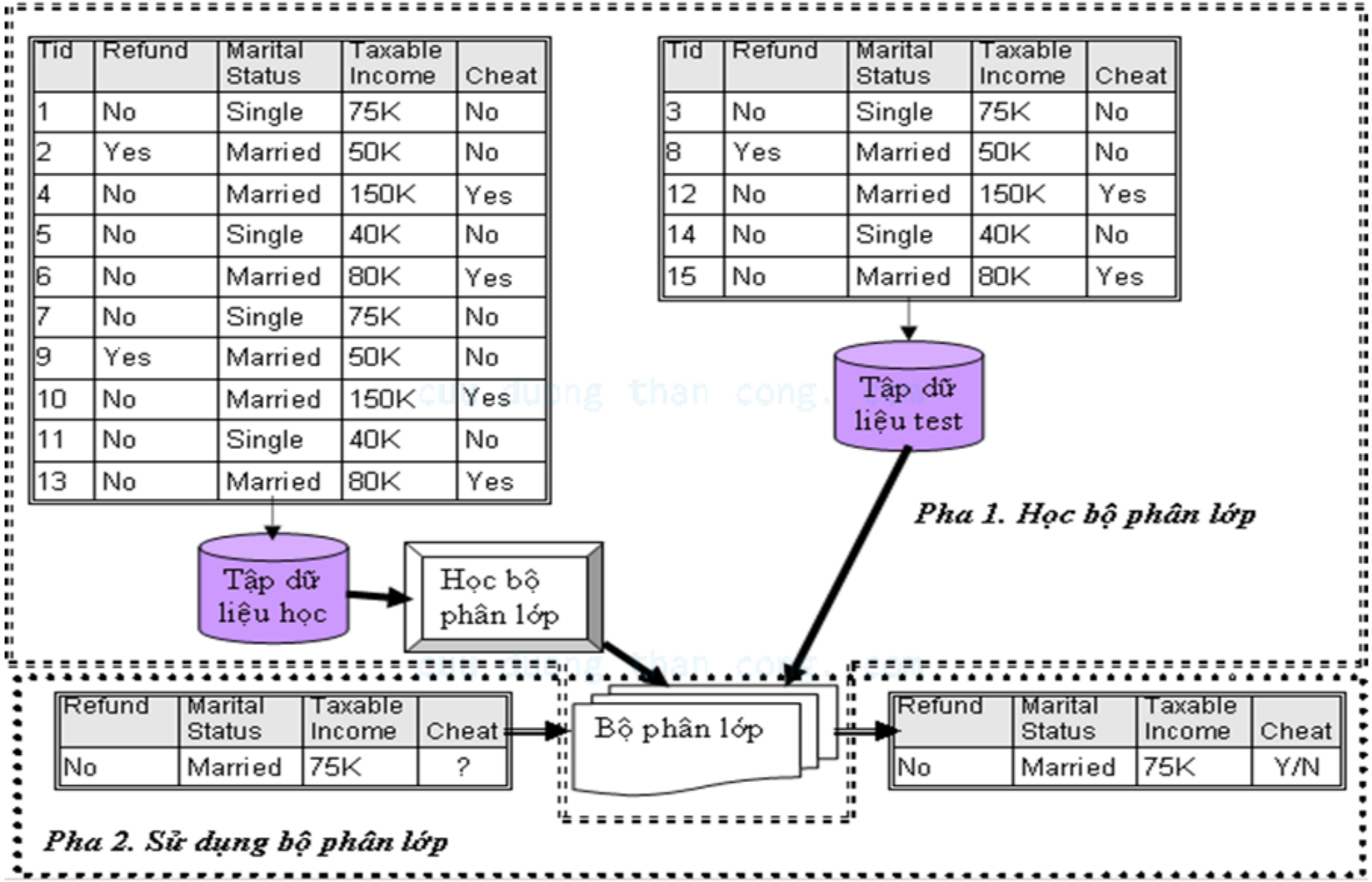
- Xây dựng mô hình: Tìm mô tả cho tập lớp đã có
  - Cho trước tập lớp C = {C_1, C_2, …, C_k}
  - Cho ánh xạ (chưa biết) từ miền D sang tập lớp C
  - Có tập ví dụ $D_{exam}=D_1+D_2+ \ldots + D_k$ với $D_i=\{d \ \ D_{exam}: d \ \ C_i\}$ D_{exam} được gọi là tập ví dụ mẫu.
  - Xây dựng ánh xạ (mô hình) phân lớp trên: Dạy bộ phân lớp.
  - Mô hình: Luật phân lớp, cây quyết định, công thức toán học…
- Pha 1: Dạy bộ phân lớp
  - Tách D_{exam} thành D_{train} (2/3) + D_{test} (1/3). D_{train} và D_{test} "tính đại diện" cho miền ứng dụng
  - D_{train} : xây dựng mô hình phân lớp (xác định tham số mô hình)
  - D_{test} : đánh giá mô hình phân lớp (các độ đo hiệu quả)
  - Chọn mô hình có chất lượng nhất
- Pha 2: Sử dụng bộ phân lớp
  - d D \ D_{exam} : xác định lớp của d.

Ví dụ phân lớp: Bài toán cho vay

Tid	*Refund*	*Marital Status*	*Taxable Income*	*Cheat*
1	No	Single	75K	No
2	Yes	Married	50K	No
3	No	Single	75K	No
4	No	Married	150K	Yes
5	No	Single	40K	No
6	No	Married	80K	Yes
7	No	Single	75K	No
8	Yes	Married	50K	No
9	Yes	Married	50K	No
10	No	Married	150K	Yes
11	No	Single	40K	No
12	No	Married	150K	Yes
13	No	Married	80K	Yes
14	No	Single	40K	No
15	No	Married	80K	Yes

B

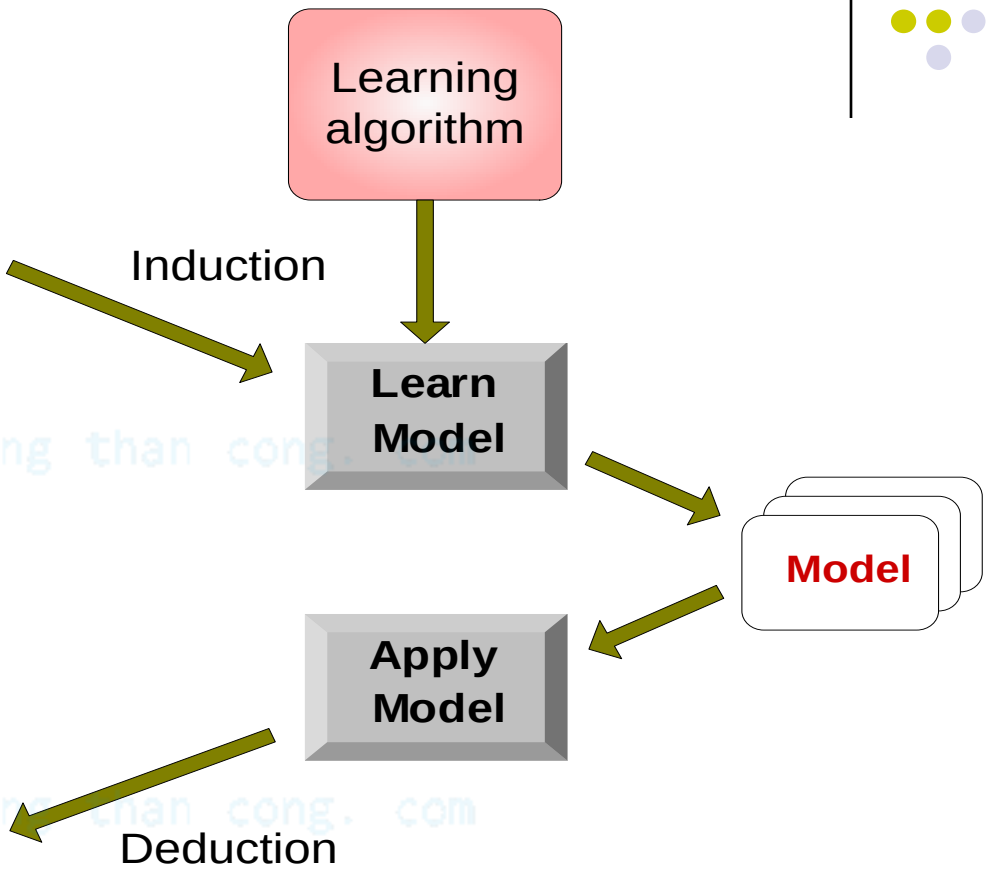
Phân lớp: Quá trình hai pha



Phân lớp: Quá trình hai pha

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

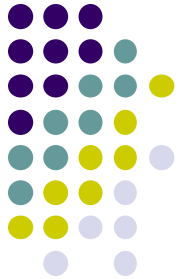
Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

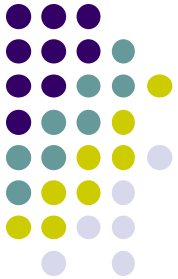
Test Set

Các loại phân lớp



- Phân lớp nhị phân/ đa lớp:
  - |C|=***2****: phân lớp nhị phân.*
  - |C|***>2:*** *phân lớp đa lớp.*
- Phân lớp đơn nhãn/ đa nhãn:
  - Đơn nhãn: mỗi tài liệu được gán vào chính xác một lớp.
  - Đa nhãn: một tài liệu có thể được gán nhiều hơn một lớp.
  - Phân cấp: lớp này là cha/con của lớp kia

Các vấn đề đánh giá mô hình



- Các phương pháp đánh giá hiệu quả

  Câu hỏi: Làm thế nào để đánh giá được hiệu quả của một mô hình?

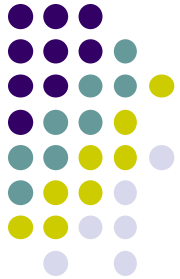
- Độ đo để đánh giá hiệu quả

  Câu hỏi: Làm thế nào để có được ước tính đáng tin cậy?

- Phương pháp so sánh mô hình

  Câu hỏi: Làm thế nào để so sánh hiệu quả tương đối giữa các mô hình có tính cạnh tranh?

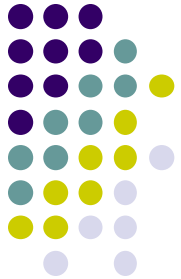
Đánh giá phân lớp nhị phân



- Theo dữ liệu test
- Giá trị thực: P dương / N âm; Giá trị qua phân lớp: T đúng/F sai. : còn gọi là *ma trận nhầm lẫn*
- Sử dụng các ký hiệu TP (true positives), TN (true negatives), FP (false positives), FN (false negatives)
  - TP: số ví dụ dương P mà thuật toán phân lớp cho giá trị đúng T
  - TN: số ví dụ âm N mà thuật toán phân lớp cho giá trị đúng T
  - FP: số ví dụ dương P mà thuật toán phân lớp cho giá trị sai F
  - FN: số ví dụ âm N mà thuật toán phân lớp cho giá trị sai F
- Độ hồi tưởng , độ chính xác , các độ đo F_1 và F

$$\frac{TP}{TP \quad FP} \qquad \frac{TP}{TP \quad TN} \qquad f_{\beta} = \frac{(\beta^2 + 1)\pi\rho}{\beta^2 \pi + \rho} \qquad f_1 = \frac{2\pi\rho}{\pi + \rho}$$

Đánh giá phân lớp nhị phân



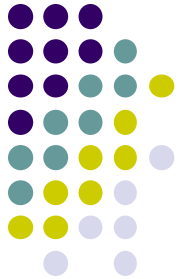
- Phương án khác đánh giá mô hình nhị phân theo độ chính xác (accuracy) và hệ số lỗi (Error rate)
- *Ma trận nhầm lẫn*

		Lớp dự báo	
		Lớp = 1	Lớp = 0
Lớp thực sự	Lớp = 1	f_{11}	f_{10}
	Lớp = 0	f_{01}	f_{00}

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{f_{11} + f_{00}}{f_{11} + f_{10} + f_{01} + f_{00}}$$

$$\text{Error rate} = \frac{\text{Number of wrong predictions}}{\text{Total number of predictions}} = \frac{f_{10} + f_{01}}{f_{11} + f_{10} + f_{01} + f_{00}}$$

So sánh hai phương án



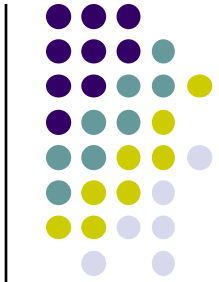
- Tập test có 9990 ví dụ lớp 0 và 10 ví dụ lớp 1. Kiểm thử: mô hình dự đoán cả 9999 ví dụ là lớp 0 và 1 ví dụ lớp 1 (chính xác: TP)
  - Theo phương án (precision, recall) có

    = 1/10=0.1; =1/1=1; f_1 = 2*0.1/(0.1+1.0)= 0.18
  - Theo phương án (accurary, error rate) có

    accurary=0.9991; error rate = 9/10000 = 0.0009

    Được coi là rất chính xác !
  - f_1 thể hiện việc đánh giá nhạy cảm với giá dữ liệu

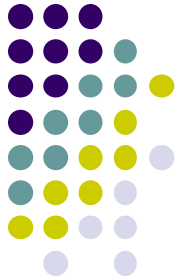
Đánh giá phân lớp đa lớp



- Bài toán ban đầu: C gồm có k lớp
- Đối với mỗi lớp C_i , cho thực hiện thuật toán với các dữ liệu thuộc D_{test} nhận được các đại lượng TP_i, TF_i, FP_i, FN_i (như bảng dưới đây)

Lớp C_i		Giá trị thực	
		Thuộc lớp C_i	Không thuộc lớp C_i
Giá trị qua bộ phân lớp đa lớp	Thuộc lớp C_i	TP_i	TN_i
	Không thuộc lớp C_i	FP_i	FN_i

Đánh giá phân lớp đa lớp



- Tương tự bộ phân lớp hai lớp (nhị phân)
  - Độ chính xác Pr_i của lớp C_i là tỷ lệ số ví dụ dương được thuật toán phân lớp cho giá trị đúng trên tổng số ví dụ được thuật toán phân lớp vào lớp C_i :

$$\mathrm{Pr}_i \quad \frac{TP_i}{TP_i \quad TN_i}$$

  - Độ hồi tưởng Re_i của lớp C_i là tỷ lệ số ví dụ dương được thuật toán phân lớp cho giá trị đúng trên tổng số ví dụ dương thực sự thuộc lớp C_i:

$$\mathrm{Re}_i \quad \frac{TP_i}{TP_i \quad FP_i}$$

Đánh giá phân lớp đa lớp

- Các giá trị ¡ và ¡ : độ hồi phục và độ chính xác đối với lớp $C_¡$.
- Đánh giá theo các độ đo
  - vi trung bình-microaveraging (được ưa chuộng) và
  - trung bình lớn-macroaveraging M và M

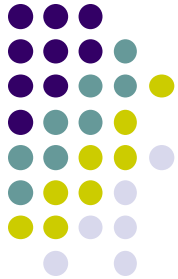
$$M \quad \frac{1}{K} \quad \frac{K}{c \ 1} \quad c$$

$$\frac{\frac{K}{c \ 1} TP_c}{\frac{K}{c \ 1} (TP_c \quad FP_c)}$$

$$M \quad \frac{1}{K} \quad \frac{K}{c \ 1} \quad c$$

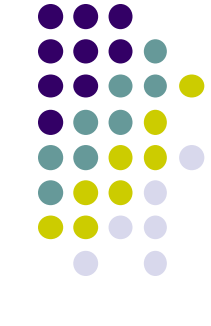
$$\frac{\frac{K}{c \ 1} TP_c}{\frac{K}{c \ 1} (TP_c \quad TN_c)}$$

Các kỹ thuật phân lớp



- Các phương pháp cây quyết định
  Decision Tree based Methods
- Các phương pháp dựa trên luật
  Rule-based Methods
- Các phương pháp Bayes «ngây thơ» và mạng tin cậy Bayes
  Naïve Bayes and Bayesian Belief Networks
- Các phương pháp máy vector hỗ trợ
  Support Vector Machines
- Lập luận dưa trên ghi nhớ
  Memory based reasoning
- Các phương pháp mạng nơron
  Neural Networks
- Một số phương pháp khác

Phân lớp cây quyết định



- Mô hình phân lớp là cây quyết định
- Cây quyết định
  - Gốc: tên thuộc tính; không có cung vào + không/một số cung ra
  - Nút trong: tên thuộc tính; có chính xác một cung vào và một số cung ra (gắn với điều kiện kiểm tra giá trị thuộc tính của nút)
  - Lá hoặc nút kết thúc: giá trị lớp; có chính xác một cung vào + không có cung ra.
  - Ví dụ: xem trang tiếp theo
- Xây dựng cây quyết định
  - Phương châm: "chia để trị", "chia nhỏ và chế ngự". Mỗi nút tương ứng với một tập các ví dụ học. Gốc: toàn bộ dữ liệu học
  - Một số thuật toán phổ biến: Hunt, họ ID3+C4.5+C5.x
- Sử dụng cây quyết định
  - Kiểm tra từ gốc theo các điều kiện

Ví dụ cây quyết định và sử dụng

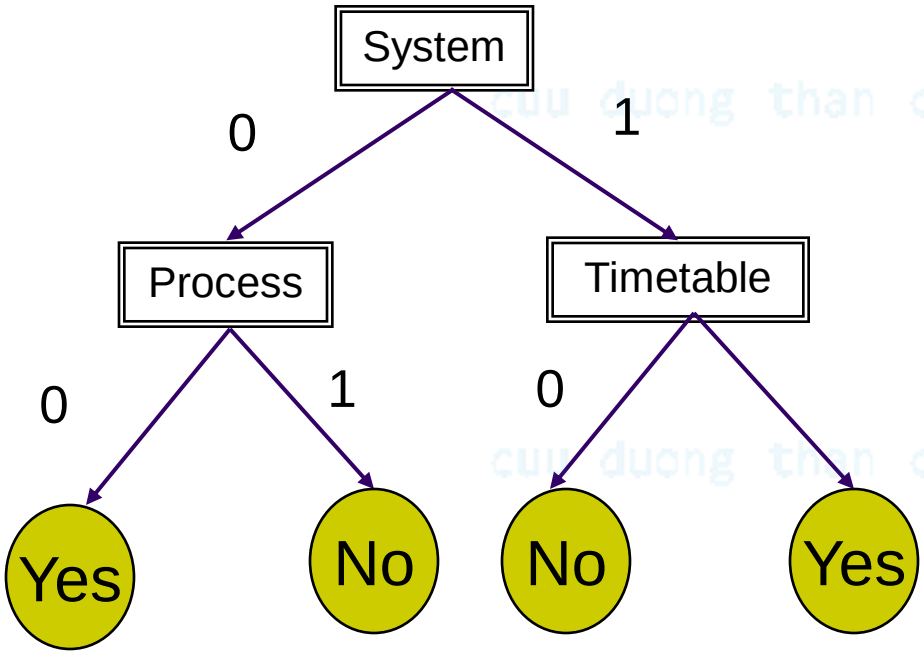
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Kết luận: Gán giá trị YES vào trường Cheat cho bản ghi

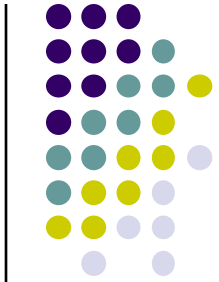
Ví dụ cây quyết định phân lớp văn bản

- Phân lớp văn bản vào lớp AI : trí tuệ nhân tạo
- Dựa vào các từ khóa có trong văn bản: System, Process, Timetable (Phân tích miền ứng dụng)



1. **If** System=0 **and** Process=0 **then** Class AI = Yes.
2. **If** System=0 **and** Process=1 **then** Class AI = No.
3. **If** System=1 **and** Timetable=1 **then** Class AI = Yes.
4. **If** System=1 **and** Timetable=0 **then** Class AI = No.

Dựng cây quyết định: thuật toán Hunt



- Thuật toán dựng cây quyết định sớm nhất, đệ quy theo nút của cây, bắt đầu từ gốc
- Input
  - Cho nút t trên cây quyết định đang được xem xét
  - Cho tập các ví dụ học D_t.
  - Cho tập nhãn lớp (giá trị lớp) y_1, y_1, … y_k. (k lớp)
- Output
  - Xác định nhãn nút t và các cung ra (nếu có) của t
- Nội dung

  1: Nếu mọi ví dụ trong D_t đều thuộc vào một lớp y thì nút t là một lá và được gán nhãn y.

  2: Nếu D_t chứa các ví dụ thuộc nhiều lớp thì

  2.1. Chọn 1 thuộc tính A để phân hoạch D_t và gán nhãn nút t là A

  2.2. Tạo phân hoạch D_t theo tập giá trị của A thành các tập con

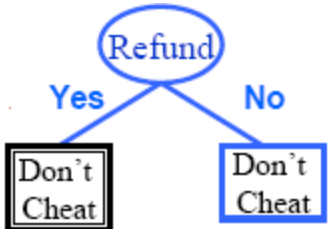
  2.3. Mỗi tập con theo phân hoạch của D_t tương ứng với một nút con u của t: cung nối t tới u là miền giá trị A theo phân hoạch, tập con nói trên được xem xét vơi u tiếp theo. Thực hiện thuật toán với từng nút con u của t.

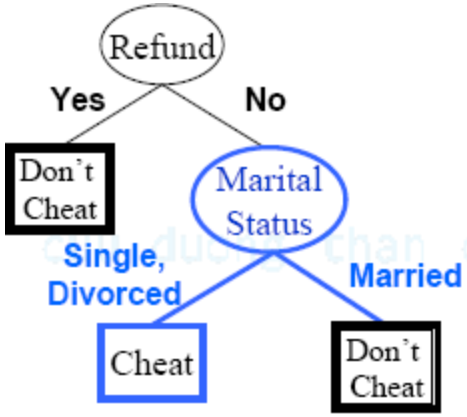
Ví dụ: thuật toán Hunt

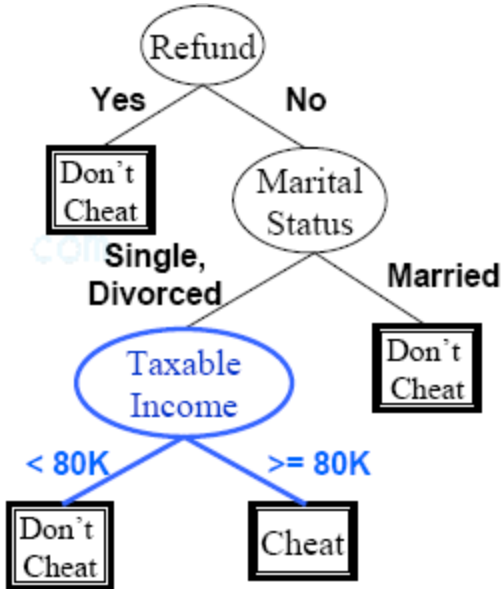
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Giải thích

- Xuất phát từ gốc với 10 bản ghi
- Thực hiện bước 2: chọn thuộc tính Refund có hai giá trị Yes, No. Chia thành hai tập gồm 3 bản ghi có Refund = Yes và 7 bản ghi có Refund = No
- Xét hai nút con của gốc từ trái sang phải. Nút trái có 3 bản ghi cùng thuộc lớp Cheat=No (Bước 1) nên là lá gán No (Don't cheat). Nút phải có 7 bản ghi có cả No và Yes nên áp dụng bước 2. Chọn thuộc tính Marital Status với phân hoạch Married và hai giá trị kia…







Thuật toán cây quyết định ID3

ID3 (*Examples, Target_attribute, Attributes*)

Ở đây: *Examples* là tập ví dụ học; *Target_attribute* là các thuộc tính đầu ra (lớp) cho cây quyết định dự đoán; *Attributes* là danh sách các thuộc tính khác tham gia trong quá trình học của cây quyết định. Kết quả thủ tục trả về cây quyết định phân lớp đúng các mẫu ví dụ đưa ra.

1. Tạo một nút gốc *Root* cho cây quyết định.
2. Nếu toàn bộ *Examples* đều là các ví dụ thuộc cùng một lớp thì trả lại cây *Root* một nút đơn với nhãn + (nếu các ví dụ thuộc lớp +) hoặc với nhãn - (nếu các ví dụ thuộc lớp -).
3. Nếu *Attributes* là rỗng thì trả lại cây *Root* một nút đơn với nhãn gán bằng giá trị phổ biến nhất của *Target_attribute* trong *Examples*.
4. Còn lại

   Begin

   4.1. Gán $A \leftarrow$ thuộc tính từ tập *Attributes* mà phân lớp tốt nhất tập *Examples*.

   4.2. Thuộc tính quyết định cho *Root* $\leftarrow A$

   4.3. Lặp với các giá trị có thể v_i của A,

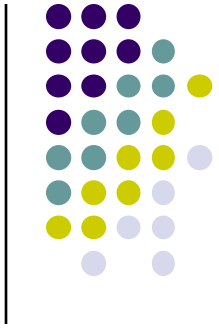
   - Cộng thêm một nhánh cây con ở dưới *Root*, phù hợp với biểu thức kiểm tra $A = v_i$.
   - Đặt $Examples_{v_i}$ là một tập con của tập các ví dụ có giá trị v_i cho A
   - Nếu $Examples_{v_i}$ rỗng
     + Thì dưới mỗi nhánh mới thêm một nút lá với nhãn = giá trị phổ biến nhất của *Target_attribute* trong tập *Examples*.
     + Ngược lại thì dưới nhánh mới này thêm một cây con

       ID3($Examples_{v_i}$, Target_attribute, Attribute - {A}).

   End

5. Return *Root*.

Thuộc tính tốt nhất: Độ đo Gini



- Bước 4.1. chọn thuộc tính A tốt nhất gán cho nút t.
- Tồn tại một số độ đo: Gini, Information gain…
- Độ đo Gini
  - Đo tính hỗn tạp của một tập ví dụ mẫu
  - Công thức tính độ đo Gini cho nút t:

$$Gini(t) = 1 - \sum_{j} [p(j|t)]^2$$

Trong đó p(j|t) là tần suất liên quan của lớp j tại nút t

  - Gini (t) lớn nhất = 1-1/n_c (với n_c là số các lớp tại nút t): khi các bản ghi tại t phân bố đều cho n_c lớp; tính hỗn tạp cao nhất, không có phân biệt giữa các lớp
  - Gini (t) nhỏ nhất = 0 khi tất cả các bản ghi thuộc một lớp duy nhất.
- Ví dụ: Bốn trường hợp

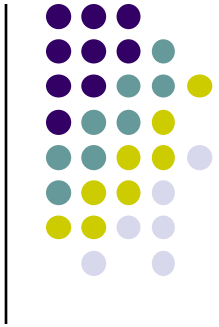
C1	**0**
C2	**6**
Gini=0.000	

C1	**1**
C2	**5**
Gini=0.278	

C1	**2**
C2	**4**
Gini=0.444	

C1	**3**
C2	**3**
Gini=0.500	

Chia tập theo độ đo Gini



- Dùng trong các thuật toán CART, SLIQ, SPRINT
- Khi một nút t được phân hoạch thành k phần (k nút con của t) thì chất lượng của việc chia tính bằng

$$GINI_{split} \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

trong đó

- n là số bản ghi của tập bản ghi tại nút t,
- $.n_i$ là số lượng bản ghi tại nút con I (của nút t).

Chia tập theo độ đo Gini: Ví dụ

- Tính toán GINI cho Refund (Yes, No), Marital Status (Single&Divorced, Married) và Taxable Income (<80K, 80K).
- Refund: 3/10 * (0) + 7/10 * (1-(3/7)^2 – (4/7)^2) = 7/10*(24/49) = 24/70
- Marital Status: 4/10 * 0 + 6/10 * (1- (3/6)^2 – (3/6)^2) = 6/10 * ½ = 3/10
- Taxable Income: thuộc tính liên tục cần chia khoảng (tồn tại một số phương pháp theo Gini, kết quả 2 thùng và 80K là mốc)

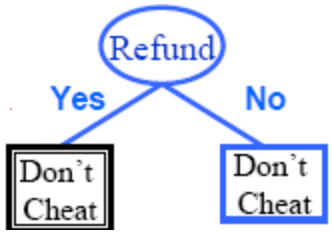
  3/10 * (0) + 7/10 * (1-(3/7)^2 – (4/7)^2) = 7/10*(24/49) = 24/70

Như vậy, Gini của Refund và Taxable Income bằng nhau (24/70) và lớn hơn Gini của Marital Status (3/10) nên chọn Refund cho gốc cây quyết định.

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

$$GINI_{split} \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

$$Gini(t) = 1 - \sum_{j=1} p(j \mid t)^2$$



Chọn thuộc tính: Information Gain

- Độ đo Information Gain
  - Thông tin thu được sau khi phân hoạch tập ví dụ
  - Dùng cho các thuật toán ID3, họ C4.5
- Entropy
  - Công thức tính entropy nút t:
    $$Entropy(t) \quad {}_{j} \; p(j \mid t) \log p(j \mid t)$$
    Trong đó p(j|t) là tần suất liên quan của lớp j tại nút t
    độ không đồng nhất tại nút t.
  - Entropy (t) lớn nhất = log (n_c) (với n_c là số các lớp tại nút t): khi các bản ghi tại t phân bố đều cho n_c lớp; tính hỗn tạp cao nhất, không có phân biệt giữa các lớp
  - Entropy (t) nhỏ nhất = 0 khi tất cả các bản ghi thuộc một lớp duy nhất.
  - Lấy loga cơ số 2 thay cho loga tự nhiên
- Tính toán entropy (t) cho một nút tương tự như Gini (t)

Chọn thuộc tính: Information Gain

- Độ đo Information Gain

$$Gain_{chia} \quad entropy(t) \quad \sum_{i=1}^{k} \frac{n_i}{n} entropy(i)$$

Trong đó, n là số lượng bản ghi tại nút t, k là số tập con trong phân hoạch, n_i là số lượng bản ghi trong tập con thứ i.

Độ đo giảm entropy sau khi phân hoạch: chọn thuộc tính làm cho Gain đạt lớn nhất.

C4.5 là một trong 10 thuật toán KPDL phổ biến nhất.

  - *Hạn chế*: Xu hướng chọn phân hoạch chia thành nhiều tập con

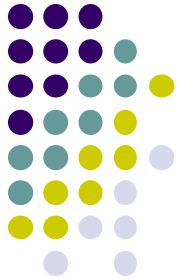
- Cải tiến

$$GainRATIO \quad \frac{Gain_{chia}}{SplitINFO} \qquad SplitINFO \quad \sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

  - Dùng GainRatio để khắc phục xu hướng chọn phân hoạch nhiều tập con

- Áp dụng: Tự tiến hành

Phân lớp dựa trên luật



- Giới thiệu
  - Phân lớp các bản ghi dựa vào tập các luật "kiểu" if … then
- Luật
  - Luật: <điều kiện> y

    Trong đó:

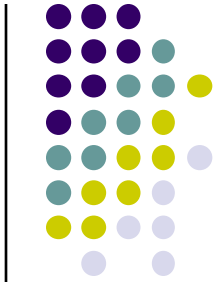
    <điều kiện> là sự kết nối các thuộc tính (còn gọi là tiên đề/điều kiện của luật: LHS bên trái)

    y là nhãn lớp (còn gọi là kết quả của luật: RHS bên phải).
  - Ví dụ

    Refund = 'Yes" Cheat = "No"

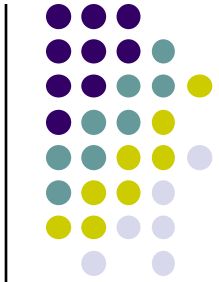
    (Refund = "No") (Marital Status = "Married") Cheat = "No"
- Sử dụng luật
  - Một luật được gọi là "bảo đảm" thể hiện r (bản ghi) nếu các thuộc tính của r đáp ứng điều kiện của luật.
  - Khi đó, vế phải của luật cũng được áp dụng cho thể hiện.

Xây dựng luật phân lớp



- Giới thiệu
  - Trực tiếp và gián tiếp
- Trực tiếp
  - Trích xuất luật trực tiếp từ dữ liệu
  - Ví dụ: RIPPER, CN2, Holte's 1R
  - Trích xuất luật trực tiếp từ dữ liệu
    1. Bắt đầu từ một tập rỗng
    2. Mở rộng luật bằng hàm Học_một_luật
    3. Xóa mọi bản ghi "bảo đảm" bởi luật vừa được học
    4. Lặp các bước 2-3 cho đến khi gặp điều kiện dừng
- Gián tiếp
  - Trích xuất luật từ mô hình phân lớp dữ liệu khác, chẳng hạn, mô hình cây quyết định, mô hình mạng nơ ron, …
  - Ví dụ:C4.5Rule

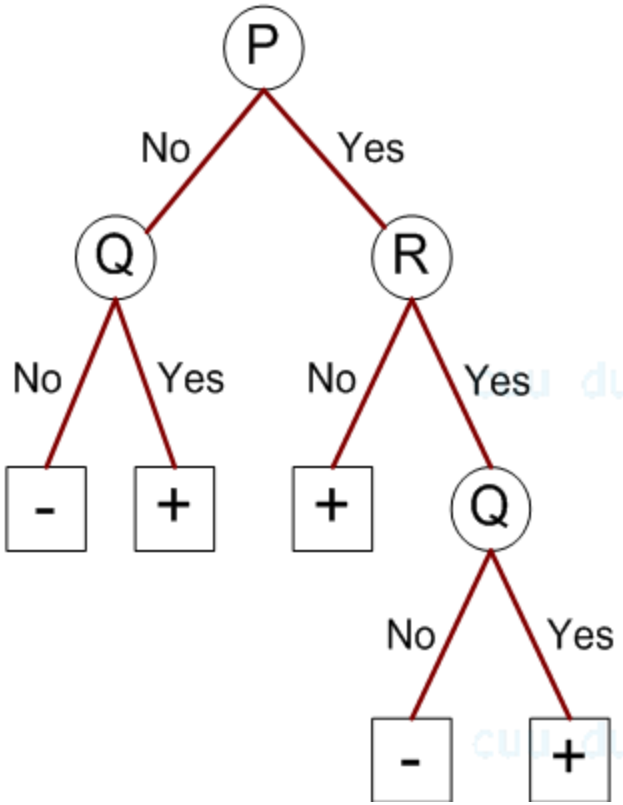
Mở rộng luật: một số phương án



- Sử dụng thống kê
  - Thống kê các đặc trưng cho ví dụ
  - Tìm đặc trưng điển hình cho từng lớp
- Thuật toán CN2
  - Khởi đầu bằng liên kết rỗng: {}
  - Bổ sung các liên kết làm cực tiểu entropy: {A}, {A, B}…
  - Xác định kết quả luật theo đa số của các bản ghi đảm bảo luật
- Thuật toán RIPPER
  - Bắt đầu từ một luật rỗng: {} lớp
  - Bổ sung các liên kết làm cực đại lợi ích thông tin FAIL
  - R0: {} => lớp (luật khởi động)
  - R1: {A} => lớp (quy tắc sau khi thêm liên kết)
  - Gain (R0, R1) = t [log (p1 / (p1 + n1)) - log (p0 / (p0 + n0))]

  với t: số thể hiện đúng đảm bảo cả hai R0 và R1
    - p0: số thể hiện đúng được bảo đảm bởi R0
    - n0: số thể hiện sai được đảm bảo bởi R0
    - P1: số thể hiện đúng được bảo đảm bởi R1
    - n 1: số trường hợp sai được đảm bảo bởi R1

Luật phân lớp: từ cây quyết định



Tập luật

Liệt kê các đường đi từ gốc

r1: (P=No,Q=No) ==> -

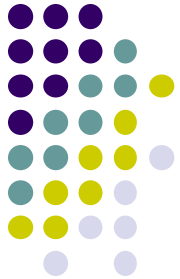
r2: (P=No,Q=Yes) ==> +

r3: (P=Yes,R=No) ==> +

r4: (P=Yes,R=Yes,Q=No) ==> -

r5: (P=Yes,R=Yes,Q=Yes) ==> +

Sinh luật gián tiếp: C4.5rules

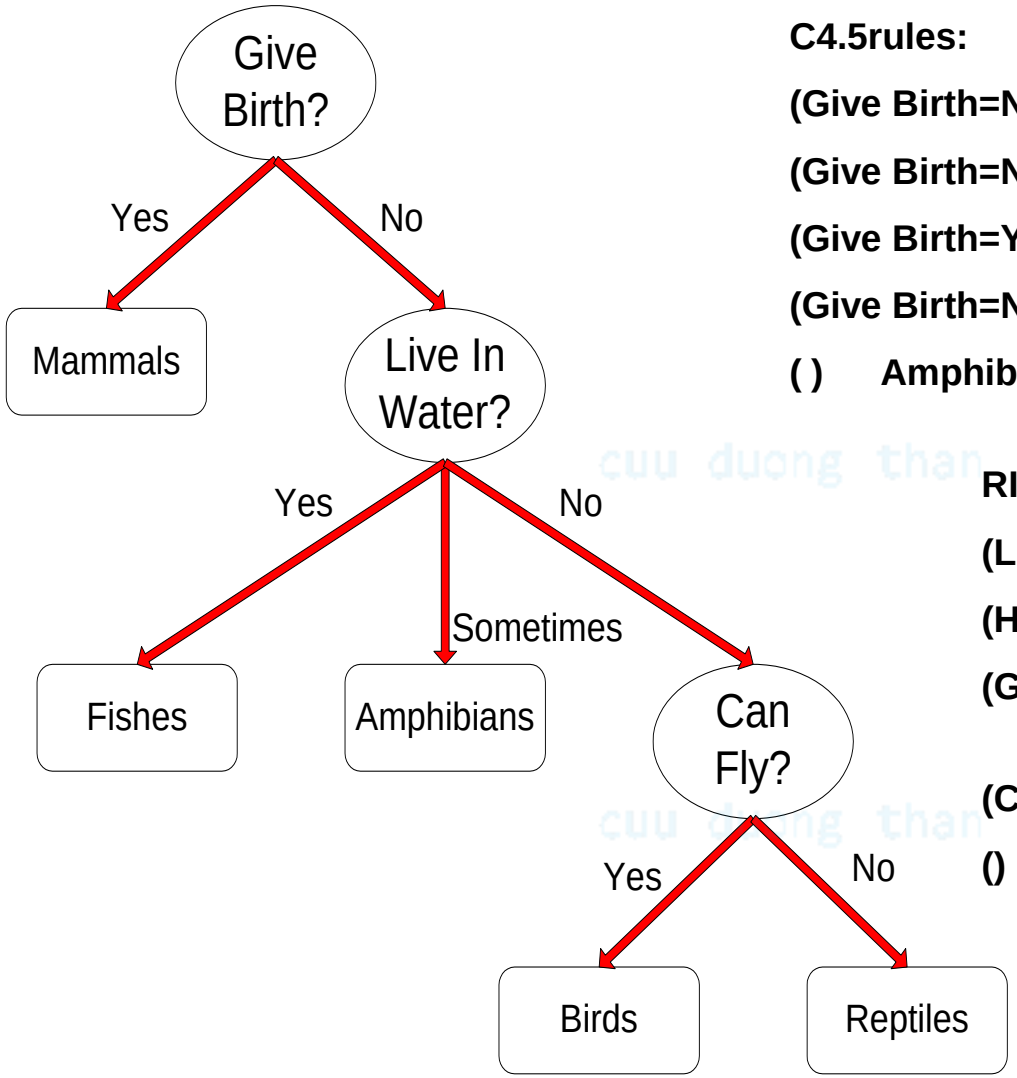


- Trích xuất luật từ cây quyết định chưa cắt tỉa
- Với mỗi luật, r: A → y
  - Xem xét luật thay thế r': A' → y, trong đó A' nhận được từ A bằng cách bỏ đi một liên kết
  - So sánh tỷ lệ lỗi r so với các r'
  - Loại bỏ các r' có lỗi thấp hơn r
  - Lặp lại cho đến khi không cải thiện được lỗi tổng thể
- Thay thế sắp xếp theo luật bằng sắp xếp theo tập con của luật (thứ tự lớp)
  - Mỗi tập con là một tập các luật với cùng một kết quả (lớp)
  - Tính toán độ dài mô tả của mỗi tập con
  - Độ dài mô tả = L(lỗi) + g* L(mô hình)
  - g : tham số đếm sự hiện diện của các thuộc tính dư thừa trong một tập luật (giá trị chuẩn, g=0.5)

C4.5rules: Ví dụ

Name	Give Birth	Lay Eggs	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	no	yes	mammals
python	no	yes	no	no	no	reptiles
salmon	no	yes	no	yes	no	fishes
whale	yes	no	no	yes	no	mammals
frog	no	yes	no	sometimes	yes	amphibians
komodo	no	yes	no	no	yes	reptiles
bat	yes	no	yes	no	yes	mammals
pigeon	no	yes	yes	no	yes	birds
cat	yes	no	no	no	yes	mammals
leopard shark	yes	no	no	yes	no	fishes
turtle	no	yes	no	sometimes	yes	reptiles
penguin	no	yes	no	sometimes	yes	birds
porcupine	yes	no	no	no	yes	mammals
eel	no	yes	no	yes	no	fishes
salamander	no	yes	no	sometimes	yes	amphibians
gila monster	no	yes	no	no	yes	reptiles
platypus	no	yes	no	no	yes	mammals
owl	no	yes	yes	no	yes	birds
dolphin	yes	no	no	yes	no	mammals
eagle	no	yes	yes	no	yes	birds

C4.5rules: Ví dụ



C4.5rules:

(Give Birth=No, Can Fly=Yes) Birds

(Give Birth=No, Live in Water=Yes) Fishes

(Give Birth=Yes) Mammals

(Give Birth=No, Can Fly=No, Live in Water=No) Reptiles

( ) Amphibians

RIPPER:

(Live in Water=Yes) Fishes

(Have Legs=No) Reptiles

(Give Birth=No, Can Fly=No, Live In Water=No) Reptiles

(Can Fly=Yes,Give Birth=No) Birds

() Mammals

Phân lớp Bayes

- Giới thiệu
  - Khung xác suất để xây dựng bộ phân lớp
  - Xác suất có điều kiện

Hai biến cố A và C

$$P(C \mid A) \quad \frac{P(A,C)}{P(A)}$$

$$P(A \mid C) \quad \frac{P(A,C)}{P(C)}$$

- Định lý Bayes:

$$P(c|x) = P(x|c).P(c)/P(x)$$

  - P(x) bằng nhau cho tất cả các lớp
  - Tìm c sao cho P(c|x) lớn nhất ⇔Tìm c sao cho P(x|c).P(c) lớn nhất
  - P(c): tần suất xuất hiện của các tài liệu thuộc lớp c
  - Vấn đề: làm thế nào để tính P(x|c)?

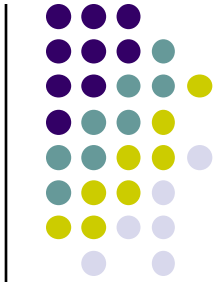
Định lý Bayes: Ví dụ

- Một bác sỹ biết
  - Bệnh nhân viêm màng não có triệu chứng cứng cổ S|M: 50%
  - Xác suất một bệnh nhân bị viêm màng não M là 1/50.000
  - Xác suất một bệnh nhân bị cứng cổ S là 1/20
- Một bệnh nhân bị cứng cổ hỏi xác suất anh/cô ta bị viêm màng não ?

$$P(M|S) \quad \frac{P(S|M)P(M)}{P(S)} \quad \frac{0.5 \quad 1/50000}{1/20} \quad 0.0002$$

Pang-Ning Tan, Michael Steinbach, Vipin Kumar. Introduction to Data Mining (Chapter 5: Classification: Alternative Techniques), *Addison Wesley*, 2005, http://www.cs.uu.nl/docs/vakken/dm/dmhc13.pdf

Phân lớp Bayes

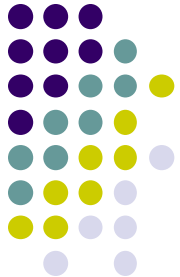


- Các thuộc tính (bao gồm nhãn lớp) là các biến ngẫu nhiên.
- Cho một bản ghi với các giá trị thuộc tính (A_1, A_2, …, A_n)
  - Cần dự báo nhãn c
  - Tìm lớp c để cực đại xác suất $P(C|A_1, A_2, \ldots, A_n)$
- Có thể tính xác suất $P(C|A_1, A_2, \ldots, A_n)$ từ dữ liệu học?

cuu duong than cong. com

Pang-Ning Tan, Michael Steinbach, Vipin Kumar. Introduction to Data Mining (Chapter 5: Classification: Alternative Techniques), *Addison Wesley*, 2005, http://www.cs.uu.nl/docs/vakken/dm/dmhc13.pdf

Phân lớp văn bản Naïve Bayes

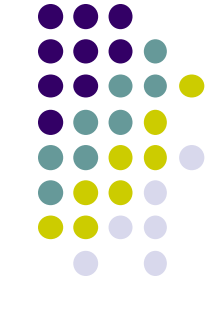


- Giả thiết Naïve Bayes:
  - giả thiết độc lập: xác suất xuất hiện của một từ khóa trong văn bản độc lập với ngữ cảnh và vị trí của nó trong văn bản:

$$p(c \mid x, \quad) \qquad \prod_{T \text{ in } \bar{x}} p(c \mid x, T)\, p(T \mid \bar{x})$$

$$\mathrm{P}(\mathrm{x}_1, \ldots, \mathrm{x}_\mathrm{k} \mid \mathrm{C}) = \mathrm{P}(\mathrm{x}_1 \mid \mathrm{C}) \cdot \ldots \cdot \mathrm{P}(\mathrm{x}_\mathrm{k} \mid \mathrm{C})$$

Phân lớp văn bản Naïve Bayes



- Cho
  - Tập ví dụ $D_{exam} = D_{learn} + D_{test}$
  - Tập từ vựng $V = \{f_1, f_2, \ldots, f_{||V||}\}$
  - Tập lớp C= {C1, C2, …, Cn} với mỗi C_i một ngưỡng $_i > 0$
- Tính xác suất tiên nghiệm
  - Trên tập ví dụ học D_{learn}
  - $p(C_i) = M_i/M$, M= $||D_{learn}||$, $M_i = ||Doc \quad D_{learn} / Doc \quad C_i||$
  - Xác suất một đặc trưng (từ) f_j thuộc lớp C:

$$P(f_j \mid C) \quad \frac{1 \quad TF(f_j, C)}{|V| \quad \sum_{i \ 1}^{n} TF(j_l, C_i)}$$

- Cho tài liệu Doc mới
  - Tính xác suất hậu nghiệm
  - Nếu P(C|Doc) > $_C$ thì Doc C!

$$P(C \mid Doc) \quad \frac{p(C) * \prod_{F_j \ V} (p(F_j \mid C)^{TF(F_j, Doc)}}{\sum_{i \ 1}^{n} p(C_i) * \prod_{F_j \ V} (p(F_j \mid C_i)^{TF(F_j, Doc)}}$$

Công thức phân lớp Bayes thứ hai

$$P(F_j|C) = \frac{1 + TF(F_j, C)}{|V| + \sum_{i=1}^{n} TF(F_i, C)}$$

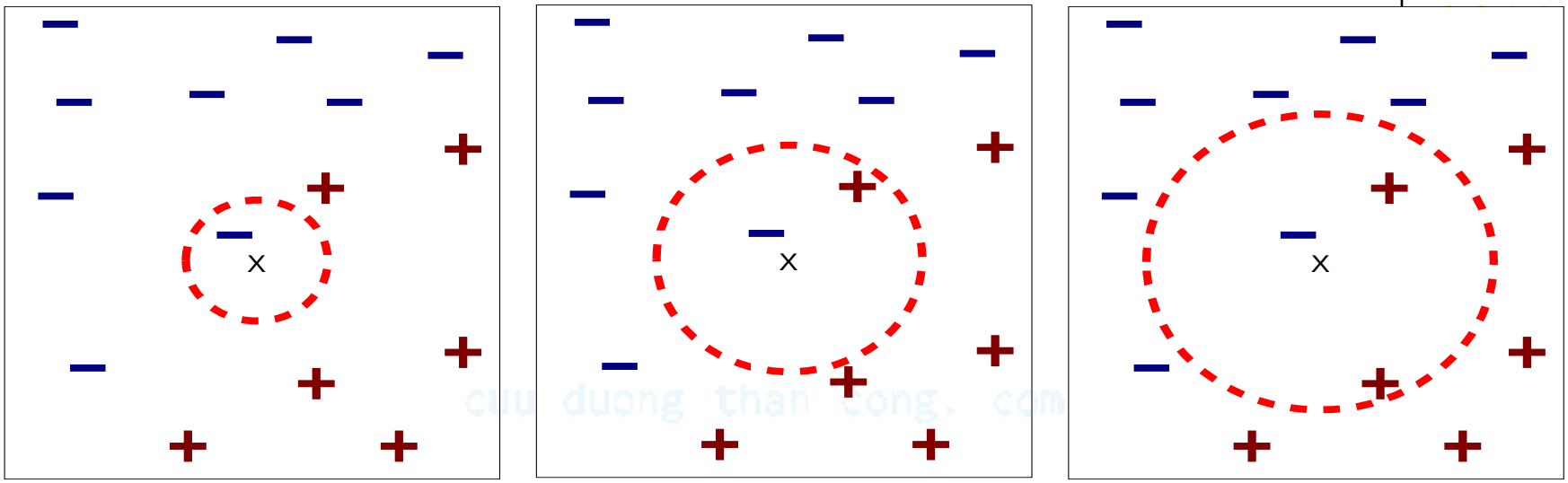
$$C|Doc) = \frac{P(C) \times \prod_{F_j \in Doc} P(F_j|C)^{TF(F_j, Doc)}}{\sum_{i=1}^{n} P(C_i) \times \prod_{F_i \in Doc} P(F_i|C_i)^{TF(F_i, Doc)}}$$

Phân lớp k-NN

$$Sm(Doc, D_i) \quad Cos(Doc, D_i) \quad \frac{\sum_{l} X_l * Y_l}{\sqrt{\sum_{l} X_l^2 \quad \sum_{l} Y_l^2}}$$

- Cho trước
  - Một tập D các tài liệu biểu diễn bản ghi các đặc trưng
  - Một đo đo khoảng cách (Ơcơlit) hoặc tương tự (như trên)
  - Một số k > 0 (láng giềng gần nhất
- Phân lớp tài liệu mới Doc được biểu diễn
  - Tính khoảng cách (độ tương tự) từ Doc tới tất cả tài liệu thuộc D
  - Tìm k tài liệu thuộc D gần Doc nhất
  - Dùng nhãn lớp của k-láng giềng gần nhất để xác định nhãn lớp của Doc: nhãn nhiều nhất trong k-láng giềng gần nhất

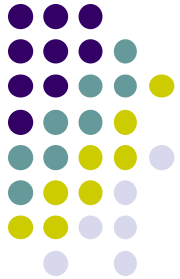
Phân lớp k-NN: Ví dụ



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

- Ba trường hợp như hình vẽ
  - 1-NN: Chọn lớp "-": láng giềng có nhãn "-" là nhiều nhất
  - 2-NN: Chọn lớp "-": hai nhãn có số lượng như nhau, chọn nhãn có tổng khoảng cách gần nhất
  - 3-NN: Chọn lớp "+": láng giềng có nhãn "+" là nhiều nhất

Thuật toán SVM

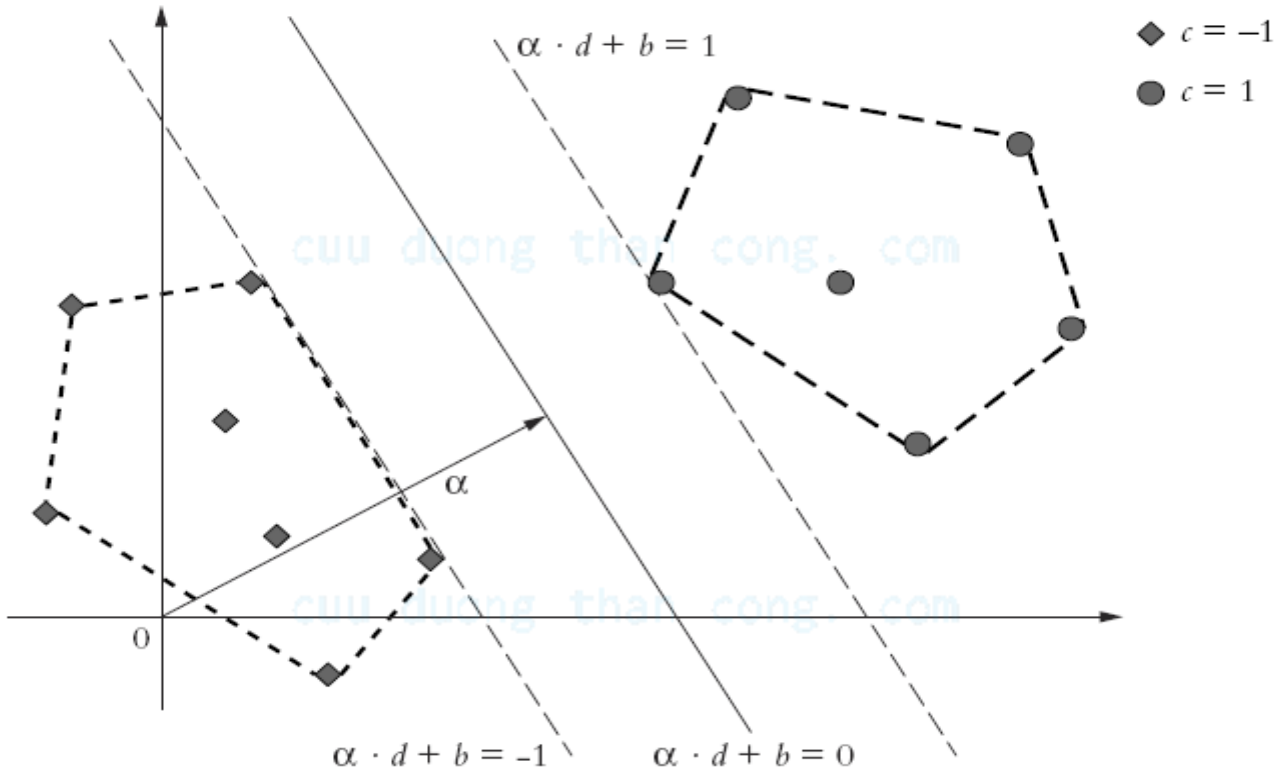


- Thuật toán máy vector hỗ trợ (Support Vector Machine – SVM): được Corters và Vapnik giới thiệu vào năm 1995.
- SVM rất hiệu quả để giải quyết các bài toán với dữ liệu có số chiều lớn (như các vector biểu diễn văn bản).

Thuật toán SVM

- Tập dữ liệu học: D= $\{(X_i, C_i), i=1,\ldots n\}$
  - $C_i \in \{-1,1\}$ xác định dữ liệu dương hay âm
- Tìm một siêu phẳng: $\alpha_{SVM} .\boldsymbol{d} + \boldsymbol{b}$ phân chia dữ liệu thành hai miền.
- Phân lớp một tài liệu mới: xác định dấu của
  - $f(d) = \alpha_{SVM} .\boldsymbol{d} + \boldsymbol{b}$
  - Thuộc lớp dương nếu $f(d) > 0$
  - Thuộc lớp âm nếu $f(d) < 0$

Thuật toán SVM



Thuật toán SVM

- Nếu dữ liệu học là tách rời tuyến tính:
  - Cực tiểu: $\frac{1}{2} \cdot \quad = \frac{1}{2} \| \ \|^2 \quad (1)$
  - Thỏa mãn:

$$c_i \quad .d_i \quad b \quad 1 \quad i \quad 1,\ldots,n \quad (2)$$

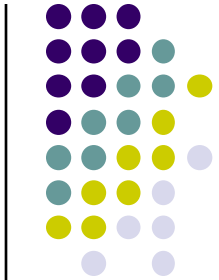
- Nếu dữ liệu học không tách rời tuyến tính: thêm biến {ξ1… ξn}:
  - Cực tiểu:
  - Thỏa mãn:

$$\frac{1}{2} \cdot \quad +C \sum_{i=1}^{n} \quad _i \quad (3)$$

$$c_i \quad .d_i \quad b \quad 1 \quad _i \quad i \quad 1,\ldots,n$$

$$_i \quad 0 \quad i \quad 1,\ldots,n \quad (4)$$

Phân lớp Web bán giám sát



- Giới thiệu phân lớp bán giám sát web
  - Khái niệm sơ bộ
  - Tại sao học bán giám sát
- Nội dung phân lớp bán giám sát web
  - Một số cách tiếp cận cơ bản
  - Các phương án học bán giám sát phân lớp web
- Phân lớp bán giám sát trong NLP

Học bán giám sát:Tài liệu tham khảo

1. Xiaojin Zhu (2006 ***). Semi-Supervised Learning Literature Survey, 1-2006. (Xiao Zhu [1])
   http://www.cs.wisc.edu/~jerryzhu/pub/ssl survey.pdf

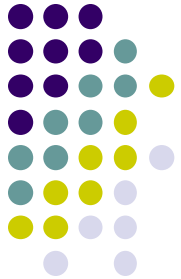
Zhou, D., Huang, J., & Scholkopf, B. (2005). Learning from labeled and unlabeled data on a directed graph. *ICML05, 22nd International Conference on Machine Learning*. Bonn, Germany.

Zhou, Z.-H., & Li, M. (2005). Semi-supervised regression with co-training. *InternationalJoint Conference on Artificial Intelligence (IJCAI)*.

Zhu, X. (2005). *Semi-supervised learning with graphs*. Doctoral dissertation, Carnegie Mellon University (mã số CMU-LTI-05-192).

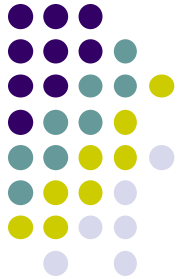
1. Olivier Chapelle, Mingmin Chi, Alexander Zien (2006) A Continuation Method for Semi-Supervised SVMs. Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, 2006.
   và các tài liệu khác

Sơ bộ về học bán giám sát



- Học bán giám sát là gì ? Xiao Zhu [1] FQA
  - Học giám sát: tập ví dụ học đã được gán nhãn (ví dụ gắn nhãn) là tập các cặp (tập thuộc tính, nhãn)
  - ví dụ gắn nhãn
    - Thủ công: khó khăn chuyên gia tốn thời gian, tiền
    - Tự động: như tự động sinh corpus song hiệu quả chưa cao
  - ví dụ chưa gắn nhãn
    - Dễ thu thập nhiều
      - xử lý tiếng nói: bài nói nhiều, xây dựng tài nguyên đòi hỏi công phu
      - xử lý văn bản: trang web vô cùng lớn, ngày càng được mở rộng
    - Có sẵn có điều kiện tiến hành tự động gắn nhãn
  - Học bán giám sát: dùng cả ví dụ có nhãn và ví dụ chưa gắn nhãn
    - Tạo ra bộ phân lớp tốt hơn so với chỉ dùng học giám sát: học bán giám sát đòi hỏi điều kiện về dung lượng khối lượng

Cơ sở của học bán giám sát

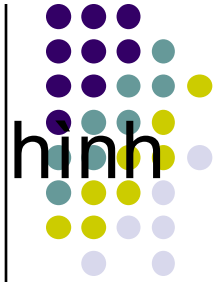


- Biểu diễn dữ liệu chưa mô tả hết ánh xạ gán nhãn trên dữ liệu
  - chẳng hạn, nghịch lý “hiệu quả như nhau” trong biểu diễn văn bản
- Ánh xạ gán nhãn có liên quan mô hình dữ liệu (mô hình / đặc trưng/ nhân / hàm tương tự) mô hình đã có theo tự nhiên hoặc giả thiết dữ liệu tuân theo.

Hiệu lực của học bán giám sát

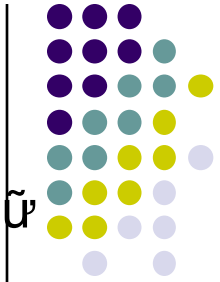
- Dữ liệu chưa nhãn không luôn luôn hiệu quả
  - Nếu giả thiết mô hình không phù hợp giảm hiệu quả
  - Một số phương pháp cần điều kiện về miền quyết định: tránh miền có mật độ cao:
    - Transductive SVM (máy hỗ trợ vector lan truyền)
    - Information Regularization (quy tắc hóa thông tin)
    - mô hình quá trinh Gauxơ với nhiễu phân lớp bằng không
    - phương pháp dựa theo đồ thị với trọng số cạnh là khoảng cách
  - "Tồi" khi dùng phương pháp này song lại "tốt" khi dùng phương pháp khác

Phương pháp học bán giám sát



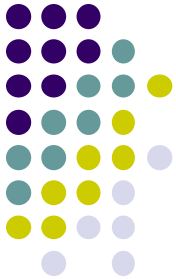
- Các phương pháp học bán giám sát điển hình
  - EM với mô hình trộn sinh
  - Self-training
  - Co-training
  - TSVM
  - Dựa trên đồ thị
  - ...
- So sánh các phương pháp
  - Đòi hỏi các giả thiết mô hình mạnh. Giả thiết mô hình phù hợp cấu trúc dữ liệu: khó kiểm nghiệm
  - Một số định hướng lựa chọn
    - Lớp phân cụm tốt: dùng EM với mô hình sinh trộn.
    - Đặc trưng phân thành hai phần riêng rẽ: co-training
    - Nếu hai điểm tương tự hướng tới một lớp: dựa trên đồ thị
    - Đã sử dụng SVM thì mở rộng TSVM
    - Khó nâng cấp học giám sát đã có: dùng self-traning

Phương pháp học bán giám sát



- Dùng dữ liệu chưa gán nhãn
  - Hoặc biến dạng hoặc thay đổi thứ tự giả thiết thu nhờ chỉ dữ liệu có nhãn
  - Mô tả chung
    - Giả thiết dưới dạng p(y|x) còn dữ liệu chưa có nhãn p(x)
    - Mô hình sinh có tham số chung phân bố kết nối p(x, y)
    - Mô hình trộn với EM mở rộng thêm self-training
    - Nhiều phương pháp là phân biệt: TSVM, quy tắc hóa thông tin, quá trình Gauxơ, dựa theo đồ thị
  - Có dữ liệu không nhãn: nhận được xác suất p(x)
- Phân biệt "học lan truyền" với "học bán giám sát"
  - Đa dạng về cách gọi. Hạn chế bài toán phân lớp.
  - "Bán giám sát"
    - dùng ví dụ có / không có nhãn,
    - "học dữ liệu nhãn/không nhãn,
    - "học dữ liệu phân lớp/có nhãn bộ phận".
    - Có cả lan truyền hoặc quy nạp.
  - Lan truyền để thu hẹp lại cho quy nạp: học chỉ dữ liệu sẵn. Quy nạp: có thể liên quan tới dữ liệu chưa có.

Mô hình sinh: Thuật toán EM

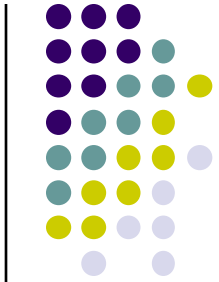


- Sơ bộ
  - Mô hình sớm nhất, phát triển lâu nhất
  - Mô hình có dạng p(x,y) = p(y)*p(x|y)
  - Với số lượng nhiều dữ liệu chưa nhãn cho P(x|y) mô hình trộn đồng nhất. Miền tài liệu được phân thành các thành phần,
  - Lý tưởng hóa tính "Đồng nhất": chỉ cần một đối tượng có nhãn cho mỗi thành phần
- Tính đồng nhất
  - Là tính chất cần có của mô hình
  - Cho họ phân bố {p } là đồng nhất nếu $_1$ $_2$ thì p $_1$ p $_2$ cho tới một hoán đổi vị trí các thành phần tính khả tách của phân bố tới các thành phần

Mô hình sinh: Thuật toán EM

- Tính xác thực của mô hình
  - Giả thiết mô hình trộn là chính xác dữ liệu không nhãn sẽ làm tăng độ chính xác phân lớp
  - Chú ý cấu trúc tốt mô hình trộn: nếu tiêu đề được chia thành các tiêu đề con thì nên mô hình hóa thành đa chiều thay cho đơn chiều
- Cực đại EM địa phương
  - Miền áp dụng
    - Khi mô hình trộn chính xác
  - Ký hiệu
    - D: tập ví dụ đã có (có nhẫn /chưa có nhãn)
    - D^K: tập ví dụ có nhãn trong D ($|D^K| << |D|$)

Mô hình sinh: Thuật toán EM



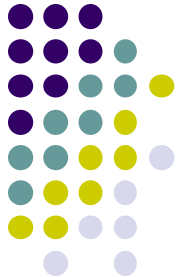
- Nội dung thuật toán

1: Cố định tập tài liệu không nhãn D^U $D \setminus D^K$ dùng trong E-bước và M-bước
2: dùng D^K xây dựng mô hình ban đầu $_0$
3: **for** i = 0, 1, 2, . . . cho đến khi kết quả đảm bảo **do**
4: **for** mỗi tài liệu d D^U **do**
5: E-bước: dùng phân lớp Bayes thứ nhất xác định P$(c|d,$ $_i)$
6: **end for**
7: **for** mỗi lớp c và từ khóa t **do**
8: M-bước: xác định $_{c,t}$ dùng công thức (*) để xây dựng mô hình i+1
9: **end for**
10: **end for**

$$\mathrm{P}(d|c) = \mathrm{P}(L = \ell_d|c)\binom{\ell_d}{\{n(d,t)\}}\prod_{t\in d}\theta_t^{n(d,t)}$$

$$\theta_{c,t} = \frac{1 + \sum_{d\in D}\mathrm{P}(c|d)\,n(d,t)}{|W| + \sum_{d\in D}\sum_{\tau}\mathrm{P}(c|d)\,n(d,\tau)} \qquad \mathrm{P}(c) = \frac{1}{|D|}\sum_{d\in D}\mathrm{P}(c|d)$$

Mô hình sinh: Thuật toán EM

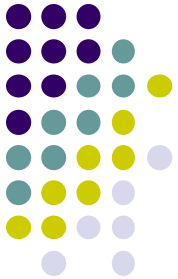


- Một số vấn đề với EM
  - Phạm vi áp dụng: mô hình trộn chính xác
  - Nếu cực trị địa phương khác xa cực trị toàn cục thì khai thác dữ liệu không nhãn không hiệu quả
  - "Kết quả đảm bảo yêu cầu": đánh giá theo các độ đo hồi tưởng, chính xác, F_1...
  - Một số vấn đề khác cần lưu ý:
    - Thuật toán nhân là Bayes naive: có thể chọn thuật toán cơ bản khác
    - Chọn điểm bắt đầu bằng học tích cực

Mô hình sinh: Thuật toán khác

- Phân cụm - và - Nhãn
  - Sử dụng phân cụm cho toàn bộ ví dụ
    - cả dữ liệu có nhãn và không có nhãn
    - dành tập D_{test} để đánh giá
  - Độ chính xác phân cụm cao
    - Mô hình phân cụm phù hợp dữ liệu
    - Nhãn cụm (nhãn dữ liệu có nhãn) làm nhãn dữ liệu khác
- Phương pháp nhân Fisher cho học phân biệt
  - Phương pháp nhân là một phương pháp điển hình
  - Nhân là gốc của mô hình sinh
  - Các ví dụ có nhãn được chuyển đổi thành vector Fisher để phân lớp

Self-Training



- Giới thiệu
  - Là kỹ thuật phổ biến trong SSL
    - EM địa phương là dạng đặc biệt của seft-training
- Nội dung

  Gọi

  L : Tập các dữ liệu gán nhãn.

  U : Tập các dữ liệu chưa gán nhãn

  Lặp (cho đến khi U =)

  Huấn luyện bộ phân lớp giám sát h trên tập L

  Sử dụng h để phân lớp dữ liệu trong tập U

  Tìm tập con U’ U có độ tin cậy cao nhất:

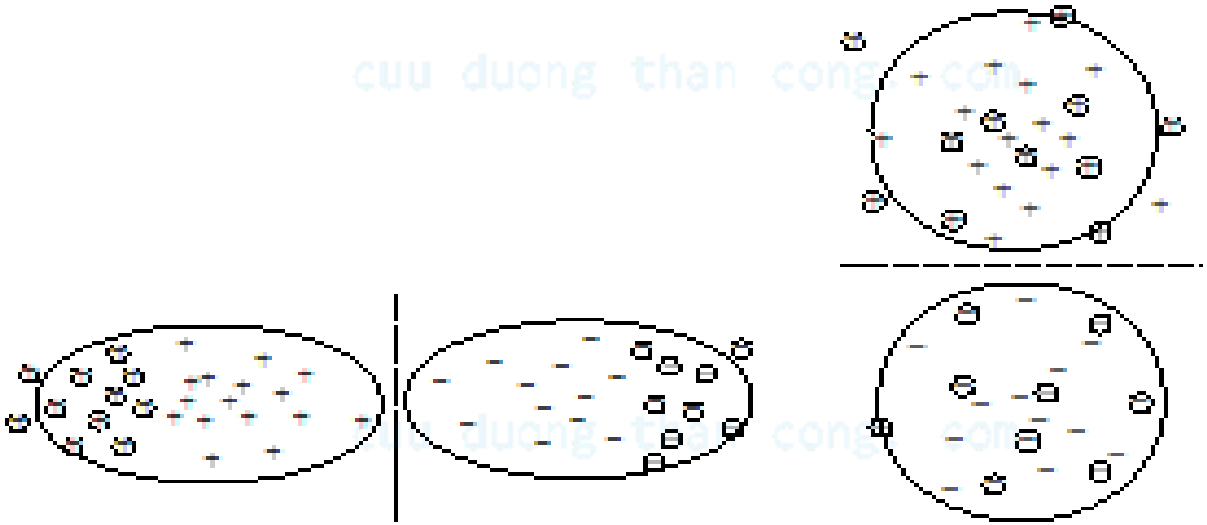
  L + U’ L

  U – U’ U

  Vấn đề tập U' có "độ tin cậy cao nhất"
- Thủ tục "bootstrapping"
- Thường được áp dụng cho các bài toán NLP

Co-Training

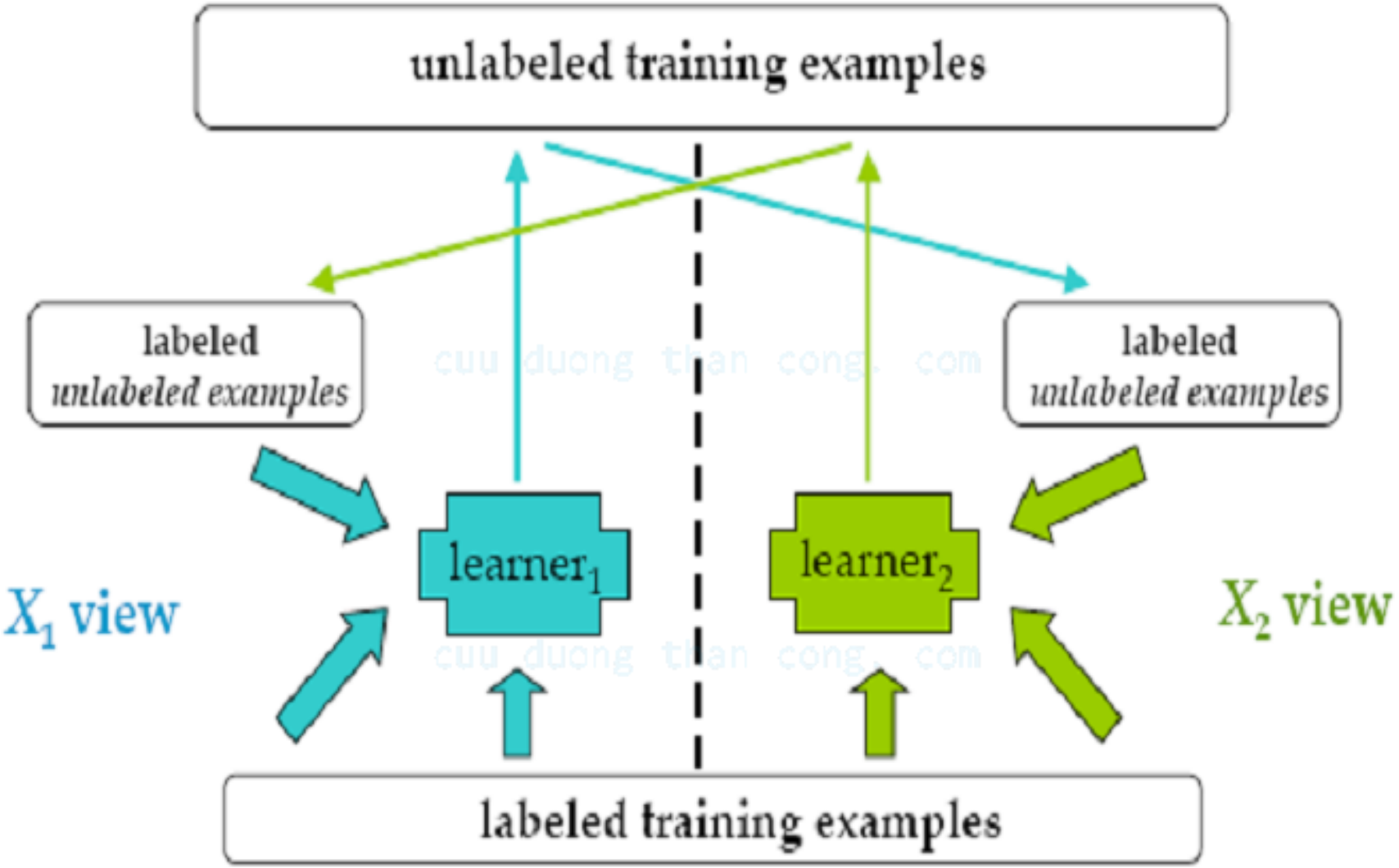
- Tư tưởng
  - Một dữ liệu có hai khung nhìn
  - Ví dụ, các trang web
    - Nội dung văn bản
    - Tiêu đề văn bản



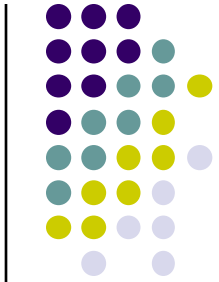
(a) x^1 view (b) x^2 view

Co-Training

- Mô hình thuật toán

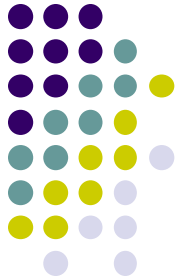


Co-Training



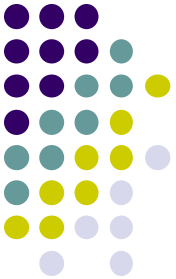
- Điều kiện dừng
  - hoặc tập dữ liệu chưa gán nhãn là rỗng
  - hoặc số vòng lặp đạt tới ngưỡng được xác định trước
- Một số lưu ý
  - Tập dữ liệu gán nhãn có ảnh hưởng lớn đến co-training
    - Quá ít: không hỗ trợ co-training
    - Quá nhiều: không thu lợi từ co-training
  - Cơ sở tăng hiệu quả co-training: thiết lập tham số
    - Kích cỡ tập dữ liệu gán nhãn
    - Kích cỡ tập dữ liệu chưa gán nhãn
    - Số các mẫu thêm vào sau mỗi vòng lặp
  - Bộ phân lớp thành phần rất quan trọng

Chặn thay đổi miền dày đặc



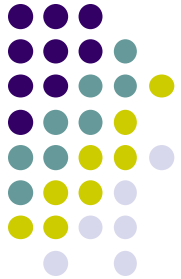
- Transductive SVMs (S3VMs)
  - Phương pháp phân biệt làm việc trên p(y|x) trực tiếp
  - Khi p(x) và p(y|x) không tương thích đưa p(x) ra khỏi miền dầy đặc
- Quá trình Gauxơ)

Mô hình đồ thị



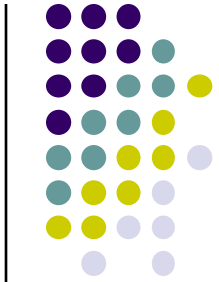
- Biểu diễn dữ liệu chưa mô tả hết ánh xạ gán nhãn trên dữ liệu (chẳng hạn, nghịch lý “hiệu quả như nhau” trong biểu diễn văn bản)
- Ánh xạ gán nhãn có liên quan mô hình dữ liệu (mô hình / đặc trưng/ nhân / hàm tương tự) mô hình đã có theo tự nhiên hoặc giả thiết dữ liệu tuân theo.

Học bán giám sát với dữ liệu Web



- Tài liệu tham khảo
  - Soumen Chakrabarti (2003). *Mining the Web: Discovering Knowledge from Hypertext Data.* Morgan Kaufmann Publishers. Chương 6. SEMISUPERVISED LEARNING)
  - Các tài liệu về học máy tài liệu chưa gán nhẫn.
  - Pierre Baldi, Paolo Frasconi, Padhraic Smyth (2003). *Modeling the Internet and the Web: Probabilistic Methods and Algorithms*. Wiley, 2003, ISBN: 0-470-84906-1(Tài liệu giảng day 2).

Học bán giám sát với dữ liệu Web



- Một số thuật toán điển hình (xem [chương 6])
  - Expectation Maximization
    - *Experimental Results*
    - *Reducing the Belief in Unlabeled Documents*
    - *Modeling Labels Using Many Mixture Components*
  - Labeling Hypertext Graphs
    - *Absorbing Features from Neighboring Pages*
    - *A Relaxation Labeling Algorithm*
    - *A Metric Graph- Labeling Problem*
  - Co- training