Statistical Language Modeling for Information Access

Theory, day 4: Between IR and IE

Maarten de Rijke   Edgar Meij

ISLA
University of Amsterdam

August 11–15, 2008 / ESSLLI 2008
August 14

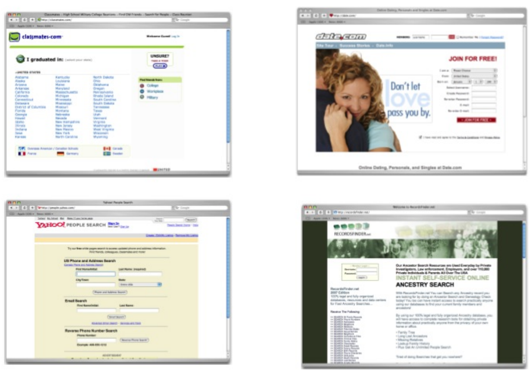
Outline

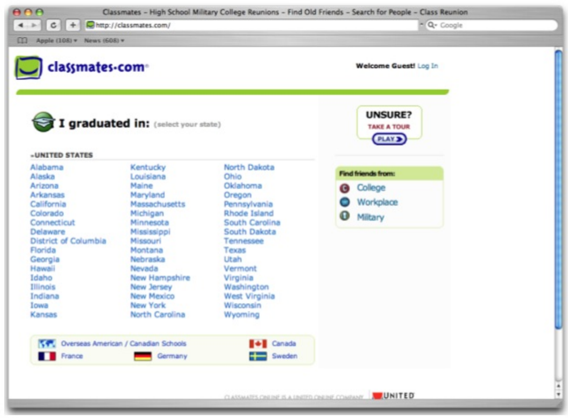
What Is Expertise Retrieval About?

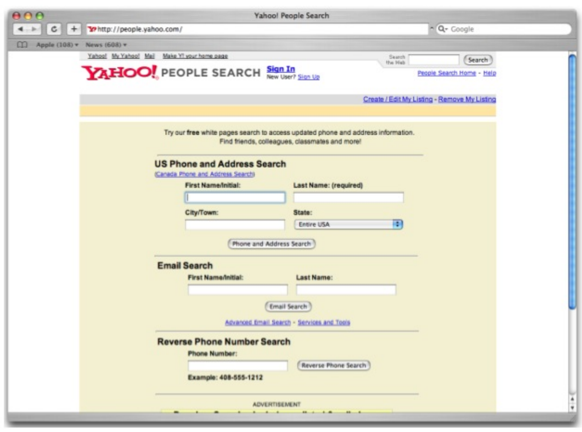
- One line summary: **finding and profiling people** within an organizational setting
- Background, models for expertise retrieval, experimental setup and evaluation, recent developments
- Presentation mostly based on Krisztian Balog, *People Search in the Enterprise*, PhD thesis, U. Amsterdam, July 2008
  - `http://www.science.uva.nl/~kbalog/phd-thesis/`

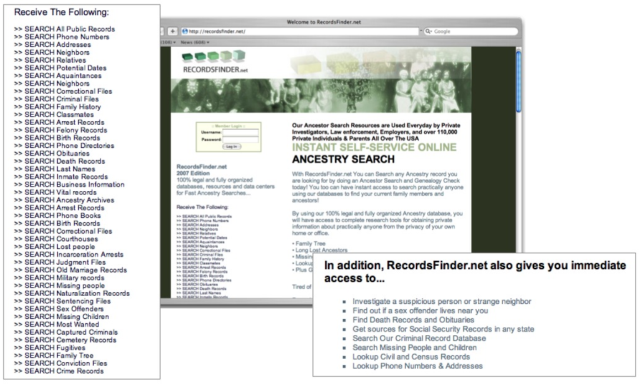
From Documents to Things to People

- Increasingly, search engines become aware of entities and entity like classes: CDs, books, people, locations, answers, . . .
- This lecture: people and answers
- Why interesting
  - From a modeling point of view: entities are directly represented (yet)—you need to get to them by collecting evidence and associating it to them, somehow
  - Mixes information retrieval and information extraction, providing a level of focus not offered by document retrieval
  - People love to search for people







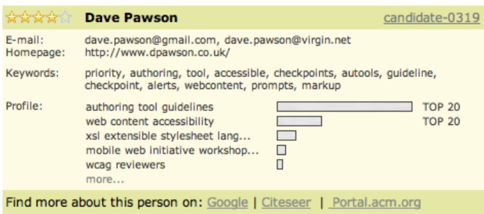


Flavors of People Search

- Locating classmates and old friends
- Finding dates, partners
- White/yellow pages (name, addres, phone, ...)
- Background check (recordsfinder.com: "investigate a susicious person or strange neighbor")
- Interest in this lecture: professional or work-related people search applications
  - A personnel officer wants to find information about a person who applied for a specific position
  - A company requires the state-of-the-art in some field, therefore they want to contact with someone from a knowledge institute
  - An enterprise needs to set up a task force to accomplish some objective

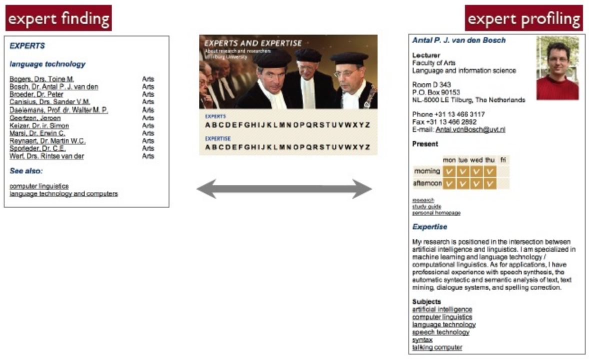
Two Main Tasks

- Expert finding
  - Identifying a list of people who are knowledgeable about a given topic Who are the experts on topic X?
- Expert profiling
  - Returning a list of topics that a person is knowledgeable about
  - What topics does person Y know about?
- Concretely:



Two Main Tasks

- Concretely: `http://www.uvt.nl/webwijs/`



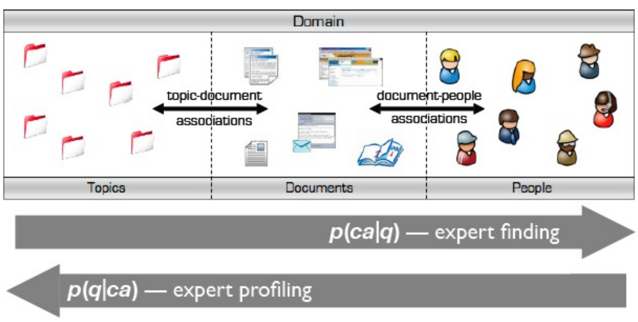
Additional Tasks

- Mining contact details
  - Essential for an operational system
- Finding similar experts
  - Counterpart of "find similar pages" feature of Web search engines
- Enterprise document search
  - Not just names, but documents relevant to the topic

Language Modeling Framework

- Expert finding: $p(ca|q)$ — the probability of a candidate being an expert given the query topic q?
- Expert profiling: $p(q|ca)$ — the probability of a knowledge area (topic) being part of the candidate's profile?
- Use Bayes to reduce to $p(q|ca)$

Main Building Blocks

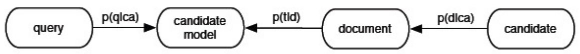


Quickly: Two Models for Expertise Retrieval

- Estimating $p(q|ca)$... how do we find experts?
  how do **you** find experts?
- An association finding problem
  - **candidate-based**: create a textual model candidates' knowledge according to the document with which they associated
  - **document-based**: identify the docs that best describe the topic, then find out who is most strongly associated with them

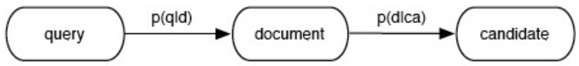
Model 1: Candidate Model

- Collect all term information from all documents associated with given candidate
- Smooth it with a background model
- Use this to represent candidate
- In a few steps
  - $p(t|M_{ca}) = (1-\lambda) \cdot p(t|ca) + \lambda \cdot p(t)$
  - $p(t|ca) = \sum_d p(t|d) \cdot p(d|ca)$
  - $p(q|M_{ca}) = \prod_{t \in q} p(t|M_{ca})^{n(t,q)}$
- Putting it altogether:
  $p(q|M_{ca}) =$
  $\prod_{t \in q} \left\{(1-\lambda) \cdot \left(\sum_d p(t|d) \cdot p(d|ca)\right) + \lambda \cdot p(t)\right\}^{n(t,q)}$



Model 2: Document Model

- Find docs relevant to query and determine who's most strongly associated with the relevant docs
- Step by step:
  - $p(q|ca) = \sum_d p(q|d)p(d|ca)$
  - $p(q|M_d) = \prod_{t \in q} p(t|M_d)^{n(t,q)}$
  - $p(t|M_d) = (1-\lambda) \cdot p(t|d) + \lambda \cdot p(t)$
- All in one:
  $p(q|ca) = \sum_d \left\{((1-\lambda) \cdot p(t|d) + \lambda \cdot p(t))^{n(t,q)}\right\} \cdot p(d|ca)$



Document-Candidate Associations

- *Need*: estimate the probability that a doc is associated with a candidate — $p(d|ca)$
- *Assume*: extraction component produces $n(d, ca)$, the number of times person ca appears in doc d

$$p(d|ca) = \frac{p(ca|d) \cdot p(d)}{p(ca)}$$

- Multiple choices
  - Boolean: associations are binary; $p(ca|d) = 1$ if $n(ca, d) > 0$, 0 otherwise
  - TF.IDF like features
  - KL divergence (see below)
  - ...

Smoothing

- JM
- Dirichlet $\lambda = \frac{\beta}{\beta + n(x)}$
  where $n(x)$ is
  - Model 1: sum of lengths of all docs assocated with a given candidate ($x = ca$)
  - Model 2: document lenght ($x = d$)

  and β is the avg representation length
  - Model 1: of a candidate representation
  - Model 2: of a doc

TREC enterprise track

- Tasks at the enterprise track

Task	TREC 2005	TREC 2006	TREC 2007
Expert search	x	x	x
E-mail known item search	x		
E-mail discussion search	x	x	
Document search			x

- Standard metrics: MAP, MRR, both for expert finding and for expert profiling
- Multiple collections, with their own characteristics...
  - W3C (TREC 2006, 2006): w3c.org
  - CSIRO (TREC 2007, 2008): csiro.au
  - UvT Epert Collection: uvt.nl/webwijs

Expert Finding: Model 1 vs Model 2

Model	TREC 2005 MAP	TREC 2005 MRR	TREC 2006 MAP	TREC 2006 MRR	TREC 2007 MAP	TREC 2007 MRR
1	.1883	.4692	.3206	.7264	.3700	.5303
2	**.2053**	**.6088**$^{(2)}$	**.4660**$^{(3)}$	**.9354**$^{(3)}$	**.4137**$^{(1)}$	**.5666**

Table 5.1: Model 1 vs. Model 2 on the expert finding task, using the TREC 2005–2007 test collections. Best scores for each year are in boldface.

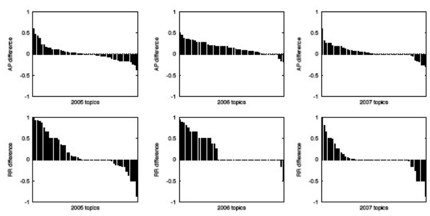


Figure 6.2: Topic-level differences in scores, Model 1 (baseline) vs Model 2. (Top): AP; (Bottom): RR. From left to right: TREC 2005, 2006, 2007.

Expert Profiling

Language	UvT ALL Model 1 MAP	UvT ALL Model 1 MRR	UvT ALL Model 2 MAP	UvT ALL Model 2 MRR	UvT MAIN Model 1 MAP	UvT MAIN Model 1 MRR	UvT MAIN Model 2 MAP	UvT MAIN Model 2 MRR
English	.2023	.3913	**.2682**$^{(3)}$	**.4968**$^{(3)}$	.3003	.4375	**.3549**$^{(3)}$	**.5198**$^{(3)}$
Dutch	.2081	.4130	**.2503**$^{(3)}$	**.4963**$^{(3)}$	.2782	.4155	**.3102**$^{(3)}$	**.4854**$^{(3)}$

Table 5.5: Model 1 vs. Model 2 on ALL vs. MAIN topics of the UvT collection. Best scores for each language are in boldface.

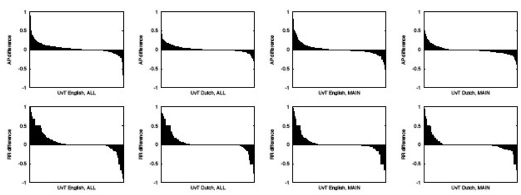


Figure 6.5: Topic-level differences in scores, Model 1 (baseline) vs Model 2. (Top): AP; (Bottom): RR. From left to right: English ALL, Dutch ALL, English MAIN, and Dutch MAIN.

Variations and Improvements

- Better estimates of candidate-document associations
- Bring in organisational structure
  - Smooth with documents from colleagues in the same group
- Proximity-based models
  - Passage/window based (M1B, M2B)
- Weigh candidate's weight in doc using KL-divergence between candidate's LM and doc LM
- Boosting underlying doc retrieval (BFB, query expansion using expert profiles, doc priors, ...)
- Careful combination leads to MAP scores of **0.5267** on TREC 2007 data (M1B; SIGIR 2009)
- Up to **0.5405** with some "secret" ingredients (M1B; SIGIR 2009)
- Up to **0.5747** without secret sauce but with rich query model based on example documents (M1B; CIKM 2008)

Something Else

- Finding Similar Experts task
  - Balog and De Rijke, SIGIR 2007
- Complement topic-centric models with contextual factors
  - Media experience, "up-to-date-ness", organizational structure, reliability, proximity, position, ...
  - Model as priors
- Experiment with Tilburg University science communicators
  - If the expert you'd normally recommend is not available, whom would you recommend?
- Contextual factors significantly improve early precision (MRR): $0.54 \rightarrow 0.59$
  - Hofmann et all, Future Challenges in Expertise Retrieval Workshop, 2008

Expertise Retrieval Upshot

- Going beyond documents
  - After all, document search has become a commodity (on the web, at least)
- Language models offer a flexible setting for modeling ER, accommodating priors, mixtures, etc.
- Very competitive performance on a range of ER tasks
- Lots of modeling work left to be done, lots of work on the interface of IR/IE left to be done
  - Be creative

Hang on

If You Have A Hammer...

- Apply the underlying type-topic associations elsewhere
  - Stakeholders in the news
  - Influential authors on a given topic (digital library setting)
  - Intelligence
  - Blog distillation
  - Spotting moods associated with a given topic
  - Getting to know your politician
  - Automatic composition of committees, PCs, ...
  - ...
- What's next
  - Web-based ER
  - Result presentation
  - New evaluation/application settings

Question Answering vs Question Retrieval

- QA has been around since the early 1960s
- Initially as a front end to (structured database)
  - Early fame for systems provided access to baseball data, data on rocks collected by NASA during its moon missions, ...
- Since late 1990s lot of attention for *corpus-based* QA: given a text corpus and a question, a system has to identify and return "the answer" (in the corpus)
- Recent rise in interest in *community-based QA*: retrieving questions that are similar to a given input query
  - FAQs (Jijkoun and de Rijke, CIKM 2005)
  - Yahoo! Answers (Agichtein et al, WISDOM 2008)
  - `wondir.com` (Xue et al, SIGIR 2008)

Combining a Translation-Based LM with a QL Model

- Given a question, find a good answer in the repository
  - Unlike standard doc retrieval, can use both answer part and question part (of items in repository)
- Xue et al combine a translation-based language model for the question part with a query likelihood approach for the answer part
- Word mismatch problem ("the vocabulary gap") potentially worse than with doc retrieval
  - short bits of text, little redundancy

The Models

- Setting: query ("the user's question"): $\mathbf{q}$, archive consisting of (q, a) pairs

- $p(\mathbf{q}|(q,a)) = \prod_{w \in \mathbf{q}} p(w|(q,a))$
- $p(w|(q,a)) = \frac{|(q,a)|}{|(q,a)|+\lambda} p_{mx}(w|(q,a)) + \frac{\lambda}{|(q,a)|+\lambda} p_{ml}(w|GE)$
- $p_{mx}(w|(q,a)) = \alpha p_{ml}(w|q) + \beta \sum_{t \in q} p(w|t) p_{ml}(t|q) + \gamma p_{ml}(w|a)$

- Huh?
  - Generation probability of the question:

$$\alpha p_{ml}(w|q) + \beta \sum_{t \in q} p(w|t) p_{ml}(t|q)$$

  - Generation probility of the answer:

$$\gamma p_{ml}(w|a)$$

Evaluation

- Use IBM Model 1 to estimate translation probabilities $p(w_i|w_j)$, using (q,a) and (a,q) pairs as parallel corpus
  - Briefly: EM plus maximum likelihood estimates
- Compare: standard mixture LM ($\beta = 0$), translation model ($\gamma = 0$), everything together ($\alpha \cdot \beta \cdot \gamma > 0$)
- Evaluation: using 50 TREC QA questions, against a 1M (q,a) collection

Model	MAP	P@10
$\beta = 0$	0.3791	0.2368
$\gamma = 0$	0.4238	0.2868
full	0.4885	0.3053

What's Next Here?

- Parameter estimation
- Bringing in additional factors
  - Social features (number of stars)
  - Question class specific features
- ...

Wrap Up and Look Ahead

- The course wiki
  - http://www.science.uva.nl/~mdr/Teaching/ESSLLI2008
  - LostInHamburg (case sensitive!)
- Summary
  - Getting started with expertise retrieval
  - A bit on retrieving questions and answers
- Tomorrow
  - Learning to rank
  - Discriminative vs generative models
  - Issues you can work on
  - Issues that you requested