SIKS course
Adapting language modeling for applications

Wessel Kraaij

TNO ICT

Kraaijw@acm.org

LM, what is it good for?

- ASR system understands:
  - *That's peach wreck in kitchen*
- *What was said:*
  - *That's speech recognition*

- *Language modeling could help here!*
- *Applications:*
  - *Speech recognition, Machine Translation*
  - *Language & authorship identification*
  - *Topical relevance ranking, IR*
  - *Text compression*

Summary of the lecture

- Statistical language modeling offers a *clean*, *competitive* and *extensible* framework for a range of (IR) tasks

- *Parameter estimation* techniques accommodating the *sparse data problem* are key to its success

Outline

- Introducing generative probabilistic models ("language models")
- A basic retrieval model
  - The role of parameter estimation
  - The importance of priors
  - Relation with vector and probabilistic model
  - reformulation as cross-entropy
- Case studies
  - Entry page search (Web documents)
  - Cross Language Information Retrieval
  - Topic Tracking
- Assignments

Introduction to Generative Language Models

What is a language model?

- Simplified statistical model of text
  - Data driven, as opposed to rule based, symbolic models of text.
  - Assigns a probability of a string given a language (fragment) vs. syntactical well-formedness of that string.

  - P1 = P("For he is a jolly good"|"English")
  - P2 = P("For he jolly good"|"English")
  - P3 = P("For lui is a jolly good"|"English")

- Intuition: $P_1>P_2>P_3$

How can we compute P?

- Starting point: generative model
  - String is a series of ordered terms <$t_0 t_1$-t_n>
  - Probability of term t_i depends on previous terms
  - P("for he is a") = P("for").P("he"|"for").P("is"|"for he").P("a"|"for he is") (chain rule)

$$P(S) = \prod_{i=0}^{n} P(t_n | t_0 .. t_{n-1})$$

Memory

- "Memory" of generative model is usually restricted. Why?
  - E.g. Memory=1: First order Markov model

'Traditional' use of statistical language models

Noisy Channel

Claude Shannon

- Automatic Speech Recognition

Likelihood of observation given interpretation

$$\hat{s} = \arg\max_s (P(s \mid a)) = \arg\max_s (\frac{P(a \mid s)P(s)}{P(a)})$$

$$\approx \arg\max_s (P(a \mid s)P(s))$$

- Statistical Machine Translation

$$\hat{e} = \arg\max_e (P(e \mid f)) = \arg\max_e (\frac{P(f \mid e)P(e)}{P(f)})$$

$$\approx \arg\max_e (P(f \mid e)P(e))$$

Word order model, bigrams/trigrams

- <u>Simplifying models is an important technique!</u>

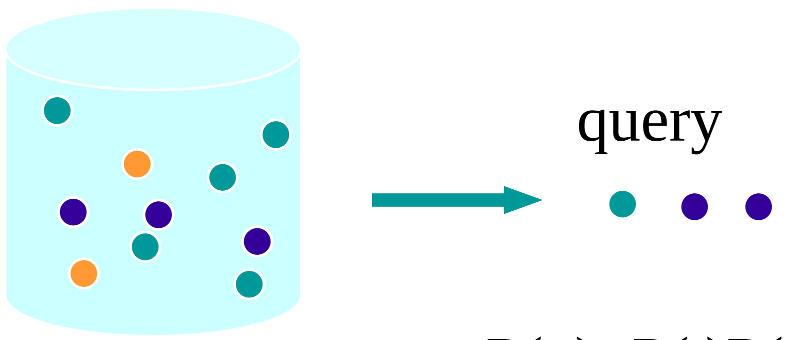
Decoder (e.g. Viterbi)

Application in IR

- Intuition: each document is represented by a language model *D.*
- A user constructs a query *Q* by choosing some terms of which he *assumes* that they occur in relevant documents.
- Rank documents according to $P(Q \mid D)$
  - How probable is Q, when taking a random text sample from D.
- Simple model (memory=0) works surprisingly well!
  - This means that we assume that all terms are chosen independently, which is clearly wrong.

Unigram language models

- Words are generated independent of the “history”.
  - Urn model: sampling with replacement.



P(q)=P(●)P(●)P(●) = 0.5x0.3x0.3

$$P(t_0 \ldots t_n \mid M) = \prod_{t=0}^{n} P(t_i \mid M)$$

Basic retrieval model

Language Models: implementation

- A unigram language model contains parameters, which all have to be estimated
  - Common method: Maximum likelihood estimation = relative frequencies

$$P_{ml}(w \mid D) = \frac{c(w,D)}{\sum_{w \in D} c(w,D)}$$

- Example:
  - D1= “Iran's Parliament has overwhelmingly approved a bill ordering a resumption of work on the government's nuclear program, including uranium enrichment”
  - D2=“US resumes Africa HIV medication program”.
  - Q= “Iran resumes nuclear program”

  - P(Q|D1) = ? P(Q]D2) = ?

Sparse data problem

- Feature space is large
  - ➔ nr of parameters is extremely high (all words in a language).
- Relatively little data for estimation (just 1 document)
  - This explains why higher order models (bigrams and up) are hardly feasible for IR.
- Solution: “smoothing”

Smoothing is a key element of parameter estimation

- Aim: avoid zero probabilities

- Methods:
  - Discounting: subtract constant ε, renormalize
    - E.g. Laplace, Good-Turing
    - Problem: all unseen terms are assigned an equal probability
  - Interpolation with a more general model
    - E.g. smooth a trigram model with a bigram model, which in turn is smoothed by a unigram model (ASR)
    - Or: smooth a document unigram model with a collection unigram model (*background model)*

Basic ranking formula

$$P(T_1, T_2, \cdots T_n \mid D) = \prod_{j=1}^{n} P(T_j \mid D)$$

Generative model, term independence

- Add smoothing to P(Q|D)

$$P(T_1, T_2, \cdots T_n \mid D) = \prod_{j=1}^{n} \lambda P(T_j \mid D) + (1-\lambda) P(T_j \mid C)$$

$$\Downarrow$$

$$\log P(T_1, T_2, \cdots T_n \mid D) = \sum_{j=1}^{n} \log\left[\lambda P(T_j \mid D) + (1-\lambda) P(T_j \mid C)\right]$$

- λ is usually a constant (e.g. 0.15), is this light or heavy smoothing?
- How does the model behave with λ = 0 and λ =1?

What about term weighting?

- Just like the vector space model, an LM model can be rewritten as an additive model, with one addend per query term
- Are there more relationships? Yes:

$$\log\left[\lambda P(T_j \mid D) + (1-\lambda)P(T_j \mid C)\right] =$$

$$\log\left[\frac{\lambda P(T_j \mid D) + (1-\lambda)P(T_j \mid C)}{(1-\lambda)P(T_j \mid C)}\right] + (1-\lambda)P(T_j \mid C) =$$

$$\log\left[1 + \frac{\lambda P(T_j \mid D)}{(1-\lambda)P(T_j \mid C)}\right] + (1-\lambda)P(T_j \mid C)$$

Constant

tf.idf

Classic trade-offs; where does NLP fit in?

- IR: Precision vs. Recall
  - Increased recall leads to reduced precision
- Machine Learning: Bias vs. Variance
  - Choosing the appropriate number of model parameters, which minimizes the error
  - Model too simple: High bias/low variance
  - Model too complex: Low bias/high variance (overfitting)

- Documents are (usually) short:
  - Danger for high variance error (sample too small), leading to low recall, the model is not robust
  - Standard operations: I) case normalization ii) stemming; map to a reduced feature space
  - More rigourous step: Latent Semantic Indexing, over-generalization?

Relations between prob. models

- Starting point: Odds of relevance (cf. Robertson/Sparck Jones)
  - Two ways to apply Bayes rule:

	discriminative	generative
$\log \frac{P(L \mid D,Q)}{P(\bar{L} \mid D,Q)} = \log \frac{P(D \mid L,Q)}{P(D \mid \bar{L},Q)} + \cancel{\log \frac{P(L \mid Q)}{P(\bar{L} \mid Q)}}$	BIR	Document likelihood ratio
$\log \frac{P(L \mid D,Q)}{P(\bar{L} \mid D,Q)} = \log \frac{P(Q \mid L,D)}{P(Q \mid \bar{L},D)} + \log \frac{P(L \mid D)}{P(\bar{L} \mid D)}$	BII	Query likelihood ratio

Cf. slide 46/47 lecture 3, slide 60/61 lecture 1

The importance of priors

Query likelihood (ratio)

$$\log\frac{P(L\mid D,Q)}{P(\bar{L}\mid D,Q)}=\log\frac{P(Q\mid L,D)}{P(Q\mid \bar{L},D)}+\log\frac{P(L\mid D)}{P(\bar{L}\mid D)}\approx$$

$$\sum_{w\in Q}c(w,Q)\log\frac{P(w\mid D)}{P(w\mid C)}+\log\frac{P(L\mid D)}{P(\bar{L}\mid D)}\approx$$

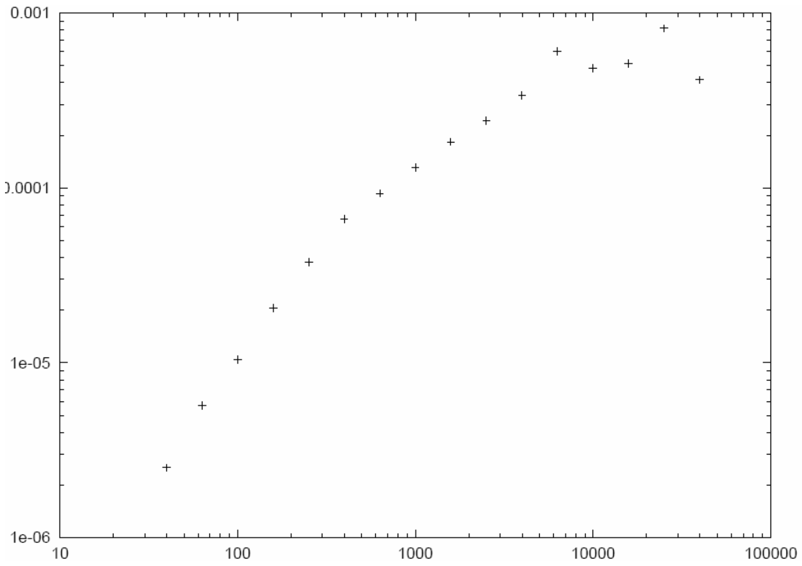
$$\sum_{w\in Q}c(w,Q)\log(1+\frac{\lambda P(w\mid D)}{(1-\lambda)P(w\mid C)})+\log P(L\mid D)$$

Djoerd Hiemstra and Wessel Kraaij, ``Twenty-One at TREC-7: ad-hoc and cross-language track'', *Proceedings of the seventh Text Retrieval Conference TREC-7*, NIST Special Publication 500-242, pages 227-238, 1999

Using the prior for document length normalization

- Almost linear relation between P(L) and document length

- Document priors improve average precision, especially for short queries
- TREC7 ad hoc:
  - Okapi: 0.232
  - LM: 0.241
  - LM+prior: 0.251



Web track: EP Task description

- TREC-10 (2001)
  - Collection of web documents (10Gbyte)
  - Find the entry page(s) of an organization
  - Just one or a few relevant documents

- Goal: explore different feature to estimate prior
  - Document length
  - Nr. of inlinks (remember Google's PageRank)
  - URL depth

Effectiveness of different features

- Document length: not effective at all for EP search
- Number of inlinks
  - Almost linear relationship with P(EP)
  - 1000 inlinks P=8E-3
  - 10 inlinks P=2E-4
- URL depth
  - Root P=6.4E-3
  - Subroot P=3.9E-5
  - Path P=9.6E-5
  - File P=3.8E-6

Results

- Mean reciprocal rank was computed over 100 EP queries:
  - No prior 0.3375
  - Inlink prior 0.5064
  - URL depth 0.7705
  - Inlink+URL(1) 0.7504
  - Inlink+URL(2) 0.7832
- Inlink+URL(1): assuming conditional independence➔ hurts MRR.
- Inlink+URL(2): estimation of joint probabilities

W. Kraaij, T. Westerveld, and D. Hiemstra. **The Importance of Prior Probabilities for Entry Page Search**. In M. Beaulieu, R. Baeza-Yates, S. H. Myaeng, and K. Järvelin, editors, *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002)*, pages 27-34, 2002. ACM Press.

Reformulation as cross entropy

Cross-entropy for monolingual IR

$$P(T_1, T_2, \cdots T_n \mid D) = \prod_{j=1}^{n} P(T_j \mid D)$$

Generative model, term independence

$$\log P(Q \mid D) = \sum_{w_i \in Q} c(w_i, Q) \log P(w_i \mid D)$$

From tokens to types

$$-H(w; D) = \sum_{i=1}^{n} P(w_i \mid Q) \log P_{sm}(w_i \mid D)$$

Formulation as a cross-entropy

Query expansion for query language model (Lavrenko&Croft)

- The idea is to compute the joint probability of w_i with the query:

$$P(w_i \mid R) \approx P(w_i \mid q_1,\ldots q_n) = \frac{P(w_i, q_1, q_2,\ldots q_n)}{P(q_1, q_2,\ldots q_n)}$$

- Introduce "hidden language models"

$$P(w_i, q_1..q_n) = \sum_M P(M, w_i, q_1..q_n) =$$

$$\sum_M P(M)P(w_i, q_1..q_n \mid M) =$$

$$\sum_M P(M)P(w_i \mid M)\prod_{k=1} P(q_k \mid M)$$

- Result: sparseness of query LM is reduced through massive query expansion (Relevance Models)

Case study
Cross Language Information Retrieval

Task description

- Cross Language Information Retrieval (CLIR):
  - Query and document are written in different languages
  - ➔ language models are instances of different feature spaces
  - Solution:
    - Translate documents or query
    - Map language models
  - Problems:
    - Availability of translation resources
    - Sense ambiguity

CLIR(1): generating the query

$$-H(w_s; D_t) = \sum_{i=1}^{n} P(s_i \mid Q) \log P_{sm}(s_i \mid D_t)$$

Matching in the query *(source)* language…

Statistical dictionary

$$P(s_i \mid D_t) = \sum_{j=1}^{T} P(s_i, t_j \mid D_t) = \sum_{j=1}^{T} P(s_i \mid t_j, D_t) P(t_j \mid D_t) \approx \sum_{j=1}^{T} P(s_i \mid t_j) P(t_j \mid D_t)$$

+ estimating P(w|D) in the query language…

$$-H(w_s; D_t) = \sum_{i=1}^{n} P(s_i \mid Q_s) \log \sum_{j=1}^{T} P(s_i \mid t_j) P_{sm}(t_j \mid D_t)$$

.. is a form of document translation!

CLIR(2): "translation" of the query

$$-H(w_t;D_t) = \sum_{i=1}^{n} P(t_i \mid Q_s) \log P_{sm}(t_i \mid D_t)$$

Matching in the document *(target)* language..

$$P(t_j \mid Q_s) = \sum_{j=1}^{S} P(s_j, t_i \mid Q_s) = \sum_{j=1}^{S} P(t_i \mid s_j, Q_s) P(s_j \mid Q_s) \approx \sum_{j=1}^{S} P(t_i \mid s_j) P(s_j \mid Q_s)$$

requires estimating P(w|Q) in the document language…

$$-H(w_t;D_t) = \sum_{i=1}^{n} \sum_{j=1}^{S} P(t_i \mid s_j) P(s_j \mid Q_s) \log P_{sm}(t_i \mid D_t)$$

…via mapping the query LM onto the document language

Research roadmap for CLIR

J. Allan (ed.): Challenges in Information Retrieval and Language Modeling, SIGIR Forum 2003:

- Observation: <u>CLIR effectiveness has reached the level of monolingual effectiveness</u>

- New challenges for CLIR research (a.o.):
  - More tightly integrated models for CLIR
  - Languages with sparse resources (low cost)
  - Scalability to multiple query and document languages
  - Exploiting parallel corpora to improve monolingual IR

Proposed approach (1)

- Transitive translation using English as a pivot language
  - 2(N-1) vs. N(N-1) directional language pairs

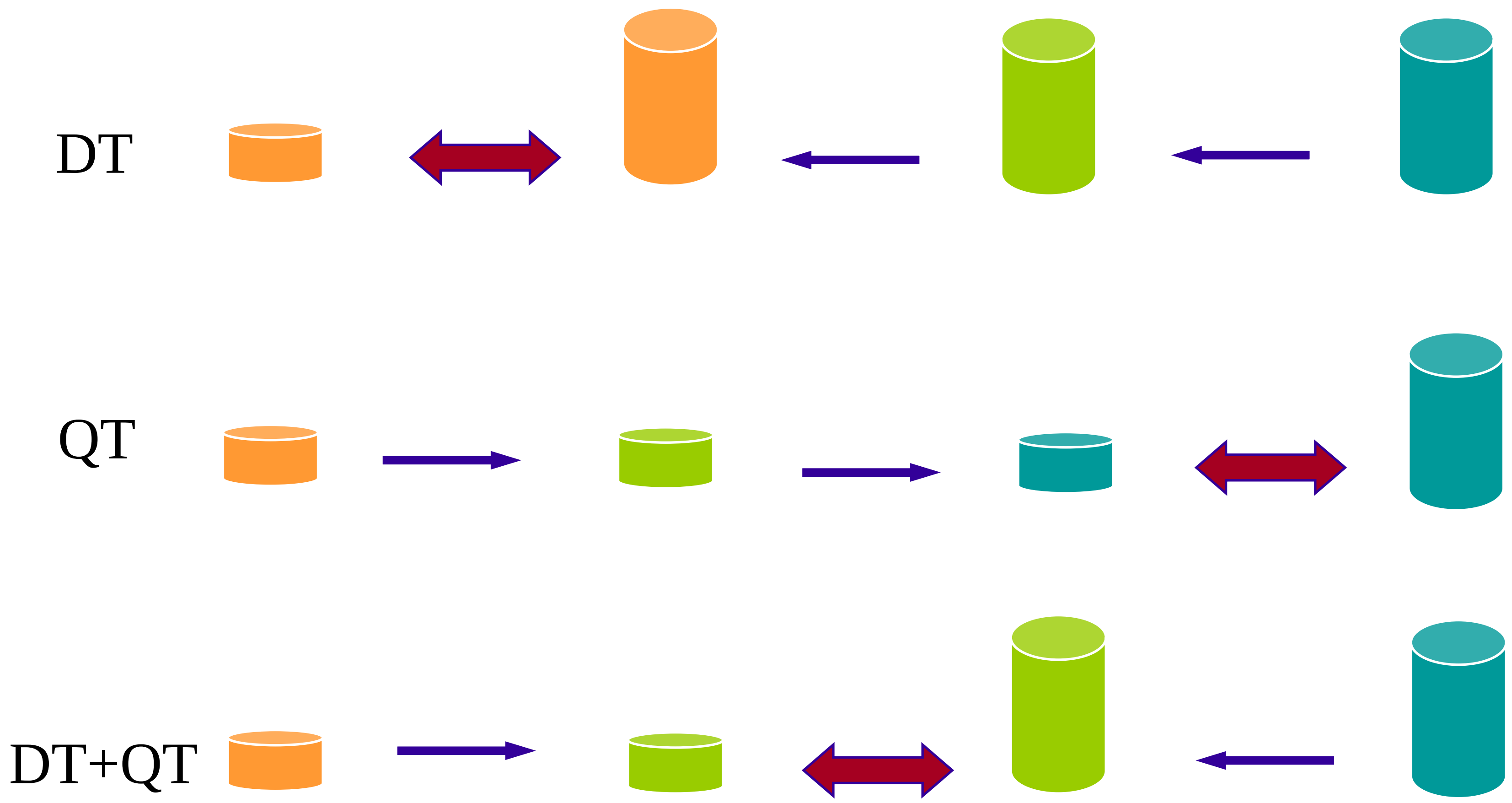
  - Which effect on effectiveness?
- Mining parallel pages from the Web
  - Inexpensive way to build statistical dictionaries
  - Quality varies from medium to high
  - Easy to combine with bilingual dictionaries

Proposed approach (2)

- Use statistical language models as underlying IR framework

- Research question:
  1. Comparison of various alternative ways to model transitive CLIR using word-by-word translation and language models

Transitive configurations



Alternative transitive CLIR models

$$-H(w_s;D_t)=\sum_{i=1}^{n}P(s_i\mid Q_s)\log\sum_{j}^{I}\sum_{k}^{T}P(s_i\mid v_j)P(v_j\mid t_k)P_{sm}(t_k\mid D_t)$$

DT: transitive document model translation

$$-H(w_t;D_t)=\sum_{i=1}^{n}\sum_{k}^{I}\sum_{j}^{S}P(t_i\mid v_k)P(v_k\mid s_j)P(s_j\mid Q_s)\log P_{sm}(t_i\mid D_t)$$

QT: transitive query model translation

$$-H(w_t;D_t)=\sum_{k}^{V}\sum_{j}^{S}P(v_k\mid s_j)P(s_j\mid Q_s)\log\sum_{i}^{T}P(v_k\mid t_i)P_{sm}(t_i\mid D_t)$$

QT+DT: match in pivot language

CLIR
Resources: estimating the translation "models"

Mining parallel Web pages (1)

- Observation: many web pages have an English version.
- Several tools available: e.g. PTMINER (Université de Montréal)
- PTMINER can be used to construct parallel corpora by automatic mining:
    - Select candidate sites:
        - SE query: anchor: “french version” etc.
    - Find files on the candidate sites:
        - SE query: host: <hostname>
    - Host crawling, use files as seeds for a within site recursive crawl

Mining parallel Web pages (2)

- Pair Scan: exploit conventions: www.asite.ca/en/afile.html vs. www.asite.ca/fr/afile.html
- Postfiltering:
  - Text length ratio
  - HTML structure
  - Language identification
- Corpora for EN-FR/DE/NL/CH/IT
  - EN-IT: 8504 pairs, 1.2 M words
  - EN-FR: 18,807 pairs , 6.7 M words

Building translation models

- Sentence alignment taking advantage of paragraph and HTML structure
- Tokenization, lemmatization, stop-word removal
- Train simple statistical translation model (IBM model 1)
  - 1-1 alignment
  - Assume translation model P(S|T) is independent of word order.
- Prune models:
  - Best N parameters (entropy criterion)
- Coverage: EN – IT 35K, EN – FR 50K

Example: translation of ‘drugs’

- Systran:
  - drogues
- Dictionary:
  - 1 drogue, stupéfiant, narcotique; 2 drogue, médicament
- Parallel corpus:
  - P(f|e) drogue(0.55), médicament(0.45)
  - P(e|f) médicament(0.79), drogue(1.0), toxicomane(0.23), drug(1.0), alcoolisme(0.24), stupéfiant(0.34), antidrogue(1.0), pharmacothérapie(0.25), immodium (0.12), ….

CLIR case study: Experiments

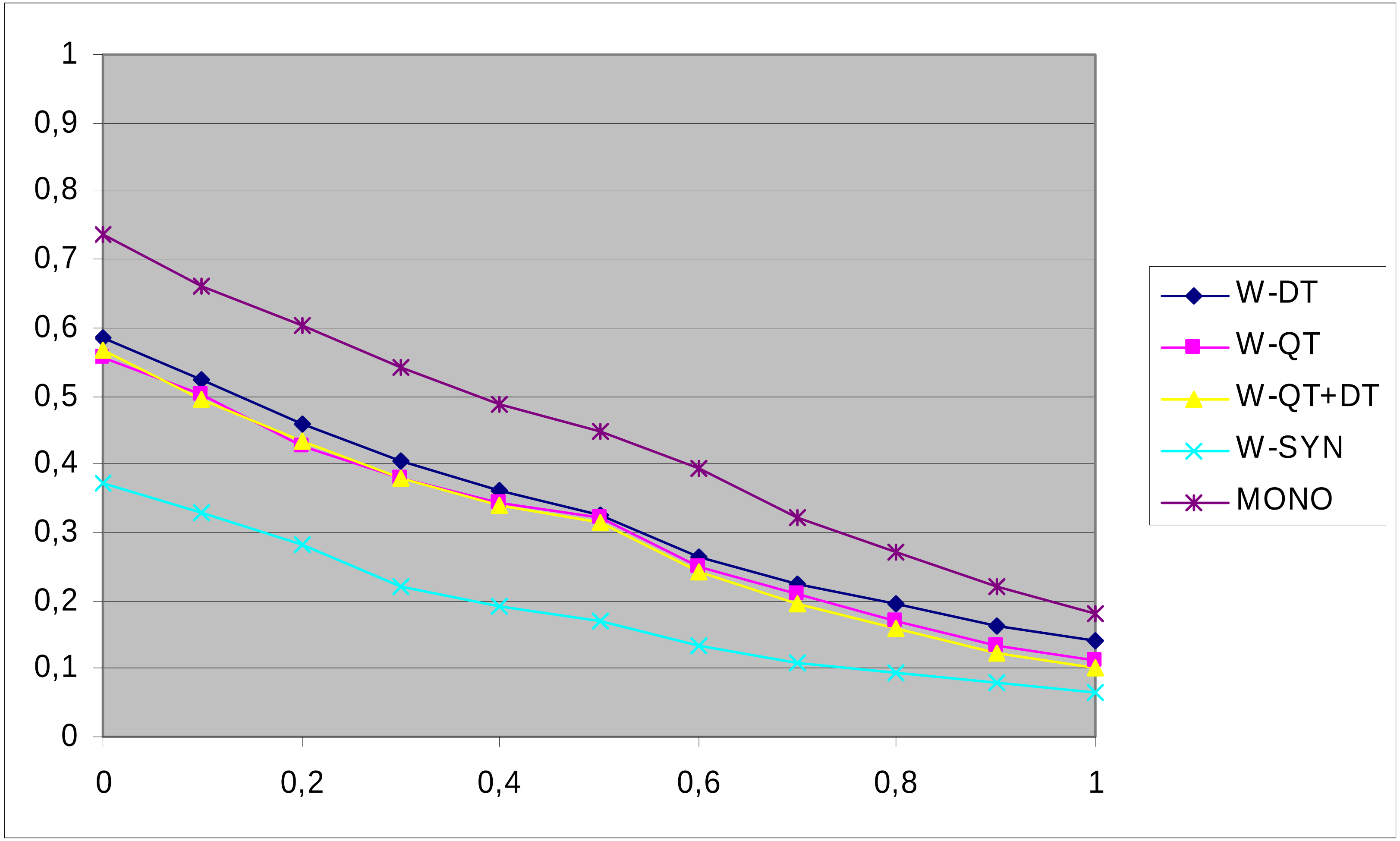
Experimental conditions

- Compare…
  - Models:
    - Three different transitive configurations
    - Baseline: using SYN operator (Pirkola,1998)

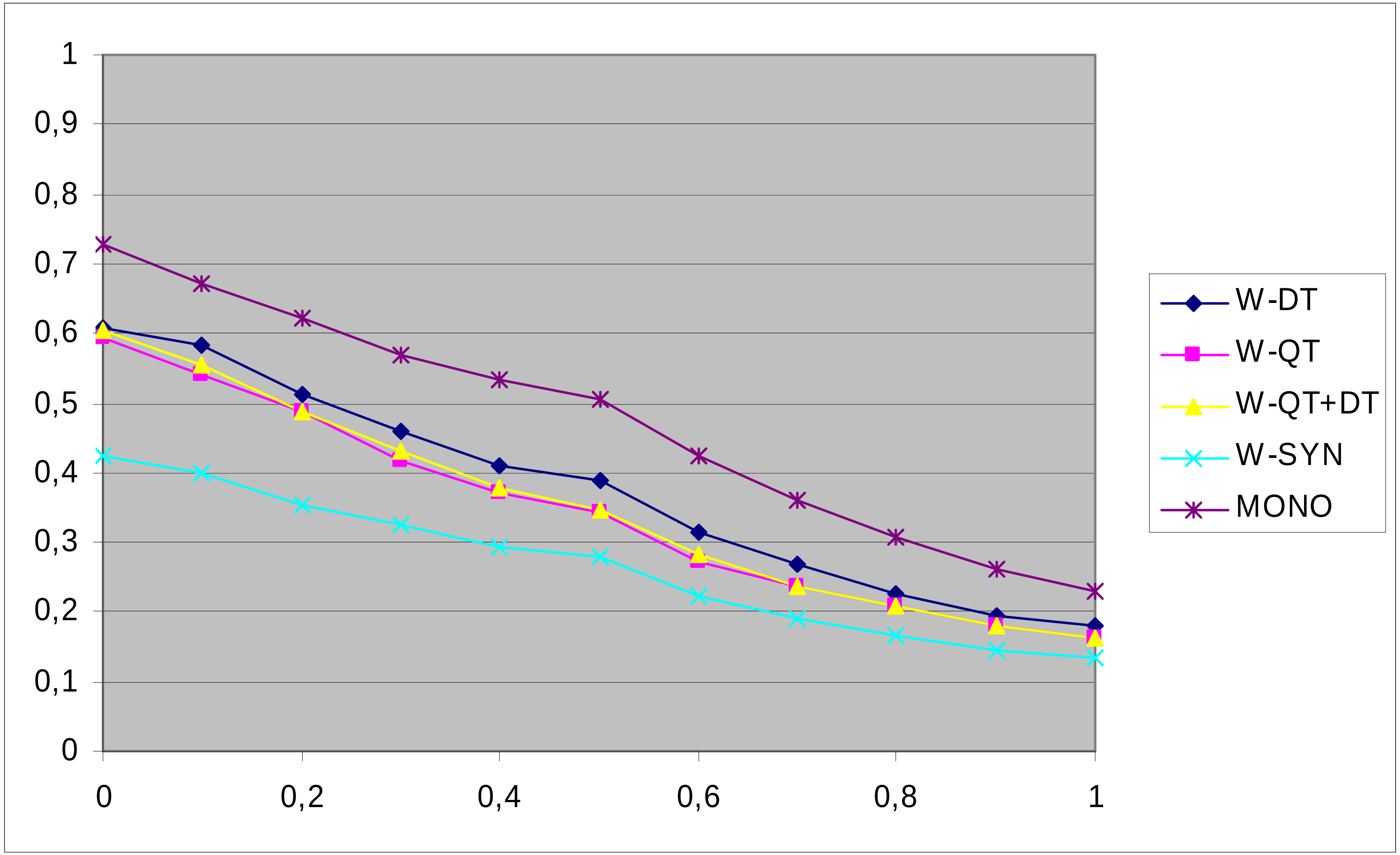
Experimental setup

- CLEF 2000/1/2 query set (T+D), CLEF 2000 document set: Le Monde, La Stampa.
- Data preprocessing
  - Tokenization
  - Remove stopwords
  - Morphological normalization
    - IT: OMSEEK stemmer
    - EN/FR: POS tagging + inflectional stemming (Xelda)
  - For queries, remove stop-structure: …*are relevant, documents that discuss…*

P/R graph Web IT-EN-FR



P/R graph Web FR-EN-IT



CLIR Case study:conclusions (1)

Integrated CLIR models:

- For WEB translation models, all probabilistic models outperform the SYN baseline;
- The SYN based baseline model breaks down under many translation alternatives per term;
- The alternative transitive CLIR models have roughly equivalent effectiveness;

➢ Proper probabilistic modeling yields best results

CLIR case study: conclusions (2)

<u>Transitive translation using Web-based lexicons:</u>

- Effectiveness ranges between 70-80% of bilingual, depending on query language and translation resource;
- Web-based translation resources are competitive with high quality MRD resources;

➢ Transitive translation is a viable approach to CLIR

- Lexical coverage of concatenated translation chain is critical for overall performance

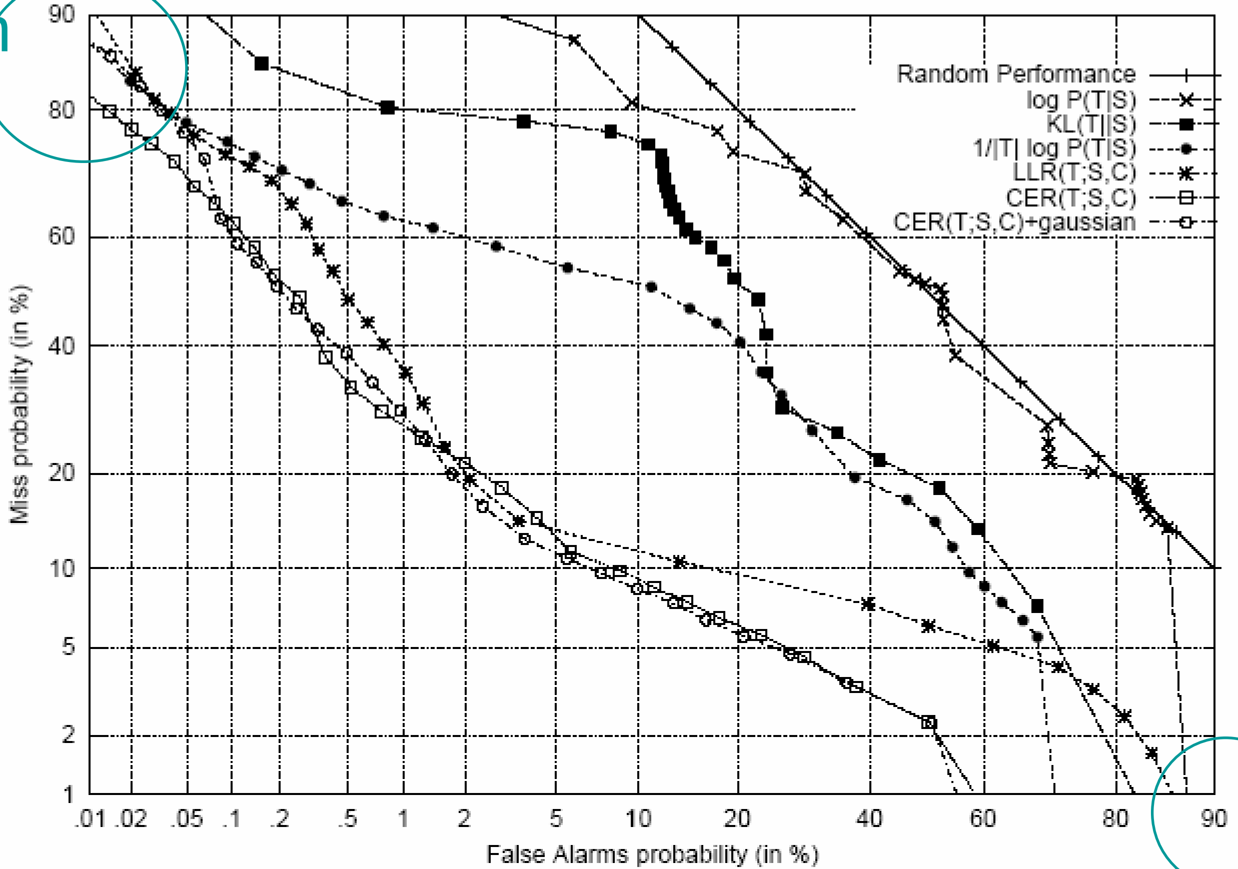
Case study:
Topic Tracking
-the importance of normalization-

Topic Tracking: Task description

- Given one or a few training documents, decide for an incoming stream of documents, for each document whether it is relevant or not (binary classification
- Challenge: the task is not only to rank documents, but the rank score must be "stable" on an absolute scale, in order apply a global decision rule based on thresholding

Normalization is important!

High precision



High recall

Detection error trade-off curve

Score distribution properties

- Using P(T|S) has two problems:
  - Score distribution is dependent on the length of T (topic)
  - Score distribution is dependent on the probability of occurrence in the background collection (since it is used for smoothing)

  - Solution: use the odds of relevance as starting point

$$\log \frac{P(L \mid D,Q)}{P(\bar{L} \mid D,Q)} \approx \sum_{w \in Q} c(w,Q) \log \frac{P(w \mid D)}{P(w \mid C)}$$

  - Formulate as "reduction in cross entropy"

$$CER(T;S,C) = -H(T,S) + H(T,C) = \sum_{w \in T} P(w \mid T) \log \frac{P(w \mid S)}{P(w \mid C)}$$

The effect of normalization

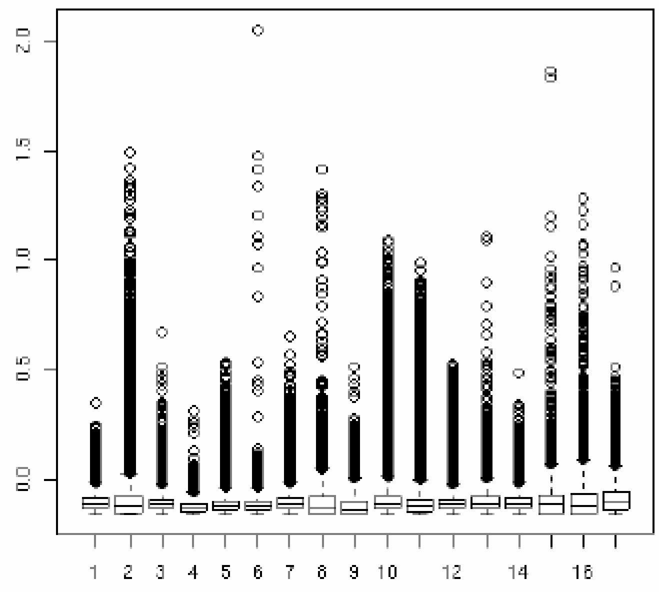


Figure 7.3. Score distributions of $CER(T;S,C)$

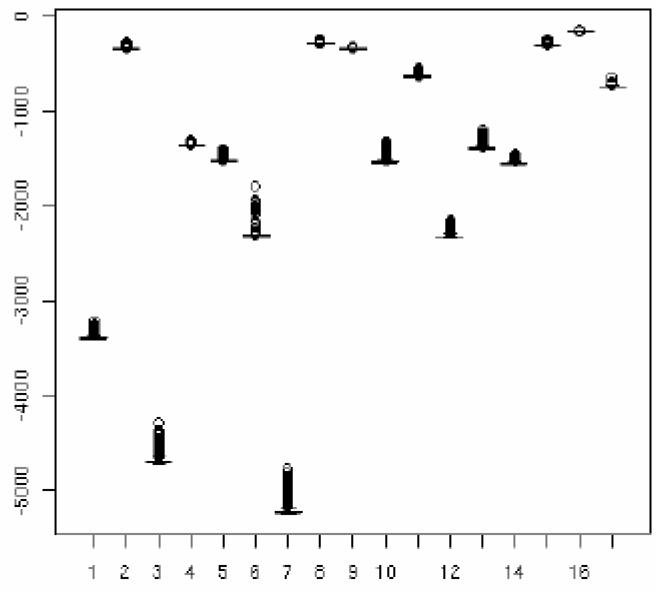


Figure 7.4. Score distributions of $P(T|S)$

Some references

- **TREC Experiment and Evaluation in Information Retrieval**
  Edited by Ellen M. Voorhees and Donna K. Harman
  MIT Press
- **Language Modeling for Information Retrieval**
  Series: The Kluwer International Series on Information Retrieval, Vol. 13
  Croft, W. Bruce; Lafferty, John (Eds.)
  2003, 264 p., Hardcover
  ISBN: 1-4020-1216-0
- **Relevance-Based Language Models**, by Lavrenko, V. and Croft, W.B., in Proceedings of the 24th annual international ACM SIGIR conference, New Orleans, LA, September 7 - 12, 2001.
- **Using Language Models for Information Retrieval**, by Djoerd Hiemstra, Ph.D. Thesis, Centre for Telematics and Information Technology, University of Twente, January 2001, ISSN 1381-3617 (no. 01-32), ISBN 90-75296-05-3
- **Transitive probabilistic CLIR models**, by Wessel Kraaij and Franciska de Jong. In Proceedings of RIAO 2004, 2004 .
- **Variations on Language Modeling for Information Retrieval**, by Wessel Kraaij. PhD thesis, University of Twente, June 2004
- **Foundations of Statistical Natural Language Processing**, Manning and Schuetze, MIT press. http://nlp.stanford.edu/fsnlp/

Student project openings @ TNO ICT // media mining

- Our department has challenging topics for internships or master projects
- Topics include:
  - Document clustering
  - Metadata extraction, semantic tagging
  - Genomics
  - Segmentation of audio / video
  - Ontologies
- Feel free to contact me at
  - Wessel.kraaij@tno.nl
  - 015 2857194