UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Ranking Models

Language Models

Rodrygo L. T. Santos
rodrygo@dcc.ufmg.br

The ranking problem

$$f(q,d)$$

Ranking models recap

Boolean model
- Boolean query
- Set-based retrieval (no actual ranking)

Vector space models
- Query and documents as vectors
- Similarity-based ranking

Probabilistic ranking models

Probabilistic relevance models
- Query and documents as random variables
- Ranking based on probability of relevance

$$f(q,d) = P(R|d,q)$$

Language modeling approach

Key intuition
- Users who try to think of a good query, think of words that are likely to appear in relevant documents
- A document is a good match to a query if it uses the same underlying ***language*** as the query

Statistical language model

A probability distribution over word sequences
- P(“Today is Wednesday”) ≈ 0.001
- P(“Today Wednesday is”) ≈ 0.0000000000001
- P(“The eigenvalue is positive”) ≈ 0.00001

Can also be regarded as a probabilistic mechanism for “generating” text, thus also called a “generative” model

Types of language models

Full dependence model

- $P(w_1 \dots w_k) = P(w_1)P(w_2|w_1) \dots P(w_k|w_1 \dots w_{k-1})$

Infeasible in practice

- Expensive computation
- Weak estimates (data sparsity)

Types of language models

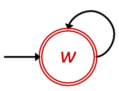
Tunable dependence via n-grams

- 3-gram ("trigram")
  $P(w_1 \dots w_k) = P(w_1)P(w_2|w_1) \dots P(w_k|w_{k-2}, w_{k-1})$
- 2-gram ("bigram")
  $P(w_1 \dots w_k) = P(w_1)P(w_2|w_1) \dots P(w_k|w_{k-1})$
- 1-gram ("unigram")
  $P(w_1 \dots w_k) = P(w_1)P(w_2) \dots P(w_k)$

Unigram language model

The simplest language model

- A one-state probabilistic finite automaton

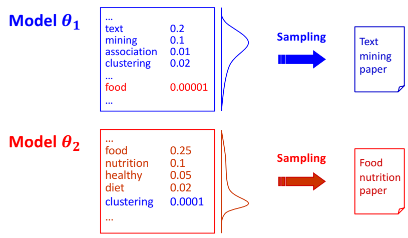


state emission probabilities

the	0.20
a	0.10
frog	0.01
toad	0.01
said	0.03
likes	0.02
that	0.04
…	
STOP	0.20

P("frog said that toad likes frog STOP")
= 0.01 · 0.03 · 0.04 · 0.01 · 0.02 · 0.01 · 0.02
= 0.0000000000048

Example text generation



Evaluation of language models

Direct evaluation criterion

- Goodness of fit: data likelihood, perplexity, cross entropy, Kullback-Leibler divergence

Indirect evaluation criterion

- Task-dependent: we hope more "reasonable" LMs would achieve better task performance

Applications of language models

Language modeling offers a principled way to quantify the uncertainties associated with natural language

Speech recognition

- Given that we see "John" and "feels", how likely will we see "happy" as opposed to "habit" next?

Applications of language models

Language modeling offers a principled way to quantify the uncertainties associated with natural language

Text categorization

- Given that we see “baseball” three times and “game” once in an article, how likely is it about “sports”?

Applications of language models

Language modeling offers a principled way to quantify the uncertainties associated with natural language

Document ranking

- Given that a user is interested in sports news, how likely would he or she use “baseball” in a query?

Language models for ranking

$$\begin{aligned} f(q,d) &= P(R|d,q) && \text{PRP} \\ &\propto P(q,d|R) && \text{Bayes' rule} \\ &\approx P(q,d) \end{aligned}$$

Relevance assumed implicitly!

Query likelihood model

$$\begin{aligned} f(q,d) &\approx P(q,d) \\ &= P(q|d)P(d) && \text{Bayes' rule} \end{aligned}$$

Two core components

- $P(q|d)$: query likelihood
- $P(d)$: document prior

Computing $P(d)$

Uninformative (uniform) prior

- $P(d) = 1/n$, for n documents in the corpus

Informative prior

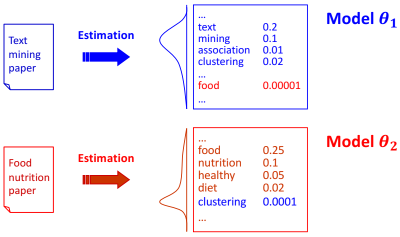
- Authoritativeness (e.g., PageRank)
- Accessibility (e.g., URL length)
- Readability (e.g., avg. sentence length)

Computing $P(q|d)$

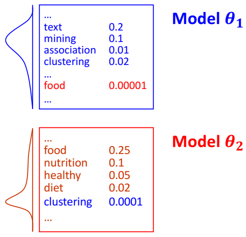
Generative process

- Intuitively, the probability that a user who likes document d (modeled by θ_d) will pose query q

Ranking via query likelihood



Ranking via query likelihood



q = [data mining algorithms]

$$P(q|\theta_1)$$
vs.
$$P(q|\theta_2)$$

Computing $P(q|d)$

Under a unigram (term independence) assumption

$$P(q|\theta_d) = \prod_{t \in q} P(t|\theta_d)^{\mathrm{tf}_{t,q}}$$
$$\propto \sum_{t \in q} \mathrm{tf}_{t,q} \log P(t|\theta_d)$$

How to estimate $P(t|\theta_d)$?

Maximum likelihood estimation (MLE)

Simply count observed occurrences

- $P_{\mathrm{MLE}}(t|\theta_d) = \frac{\mathrm{tf}_{t,d}}{|d|}$

Problems

- Observed terms will be scored too optimistically
- Unobserved terms will receive zero probability

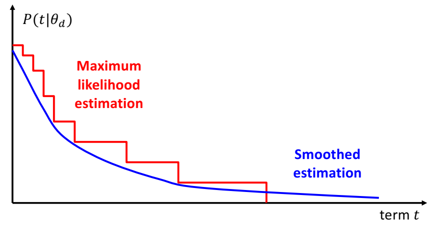
Smoothing probabilities

Smooth probabilities in language models

- Discount the probability of seen words
- Give some probability mass to unseen words

The probability of a non-occurring term should be close to its probability of occurrence in the corpus C

Smoothing probabilities



Smoothing probabilities

General form

- $P(t|\theta_d) = (1-\alpha)\, P_{\text{MLE}}(t|\theta_d) + \alpha\, P_{\text{MLE}}(t|\theta_C)$

Smoothing controlled through parameter α

- Jelinek-Mercer: $\alpha = \lambda, 0 < \lambda < 1$

Jelinek-Mercer smoothed model

$$f(q,d) \propto \prod_{t \in q} P(t|\theta_d)^{\text{tf}_{t,q}}$$

$$= \prod_{t \in q} \left((1-\lambda) \frac{\text{tf}_{t,d}}{|d|} + \lambda \frac{\text{tf}_{t,C}}{|C|} \right)^{\text{tf}_{t,q}}$$

Jelinek-Mercer example ($\lambda = 1/2$)

q: [michael jackson]

- d_1: Jackson was a gifted entertainer
- d_2: Michael Jackson anointed himself King of Pop

$$P(q|\theta_1) = \left[\frac{0/5 + 1/12}{2}\right] \times \left[\frac{1/5 + 2/12}{2}\right] \approx 0.008$$

$$P(q|\theta_2) = \left[\frac{1/7 + 1/12}{2}\right] \times \left[\frac{1/7 + 2/12}{2}\right] \approx 0.018$$

Where is tf-idf weighting?

$$f(q,d) \propto \sum_{t \in q} \text{tf}_{t,q} \log P(t|\theta_d)$$

$$= \sum_{t \in q} \text{tf}_{t,q} \log \left((1-\lambda) \frac{\text{tf}_{t,d}}{|d|} + \lambda \frac{\text{tf}_{t,C}}{|C|} \right)$$

Where is tf-idf weighting?

$$f(q,d) \propto \sum_{t \in q} \text{tf}_{t,q} \log \left(\frac{(1-\lambda) \frac{\text{tf}_{t,d}}{|d|}}{\lambda \frac{\text{tf}_{t,C}}{|C|}} + 1 \right)$$

Where is tf-idf weighting?

$$f(q,d) \propto \sum_{t \in q} \mathrm{tf}_{t,q} \log\left(\frac{(1-\lambda)\frac{\mathrm{tf}_{t,d}}{|d|}}{\lambda\frac{\mathrm{tf}_{t,C}}{|C|}} + 1\right)$$

- Proportional to term frequency: tf effect
- Inv. proportional to collection frequency: idf effect

How about document length?

General form

- $P(t|\theta_d) = (1-\alpha)\, P_{\mathrm{MLE}}(t|\theta_d) + \alpha\, P_{\mathrm{MLE}}(t|\theta_C)$

Smoothing controlled through parameter α

- Jelinek-Mercer: $\alpha = \lambda, 0 < \lambda < 1$
- Dirichlet: $\alpha = \frac{\mu}{|d|+\mu}$ $\quad |d| \to 0 \therefore \alpha \to 1$
 $|d| \to \infty \therefore \alpha \to 0$

How effective are these?

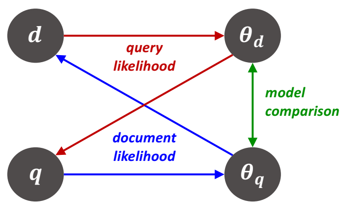
Dirichlet smoothing works the best

- Particularly effective for keyword queries

Length-adjusted smoothing

- Shorter documents get more smoothing
- Longer documents get less smoothing

Extended approaches



Document likelihood model

$$\begin{aligned} f(q,d) &\approx P(q,d) \\ &= P(d|q)P(q) \quad \text{Bayes' rule} \\ &\propto P(d|\theta_q) \end{aligned}$$

Problem: queries are short

- Poor estimation of θ_q

Solution: improve query model via feedback

Model comparison

Two steps

- Build both θ_q and θ_d
- Measure how much they ***diverge***

Divergence-based ranking

- $f(q,d) = -D(\theta_q || \theta_d)$

Model comparison

Kullback-Leibler (KL) divergence
- Asymmetric measure [0,∞]

$$f(q,d) = -D_{\mathrm{KL}}(\theta_q || \theta_d)$$
$$= -\sum_t P(t|\theta_q) \log \frac{P(t|\theta_q)}{P(t|\theta_d)}$$

Summary

Principled ranking approach
- Statistical foundations (better parameter setting)

Effectiveness through smoothing
- Add key ranking components (idf, document length)

Flexible and extendable
- Relevance feedback, priors, other tasks

References

Introduction to Information Retrieval, Ch. 12
Manning et al., 2008

Search Engines: Information Retrieval in Practice, Ch. 7
Croft et al., 2009

Statistical Language Models for Information Retrieval
Zhai, FnTIR 2008

References

A language modeling approach to information retrieval
Ponte and Croft, SIGIR 1998

A study of smoothing methods for language models applied to information retrieval
Zhai and Lafferty, SIGIR 2001

UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Coming next…

Information-theoretic Models

Rodrygo L. T. Santos
rodrygo@dcc.ufmg.br