Lecture 12: Language Models for IR

William Webber (william@williamwebber.com)

COMP90042, 2014, Semester 1, Lecture 12

What we'll learn in this lecture

- Language models:
  - Alternative probabilistic approach to IR
  - Nice theory, good effectiveness

Trad probab IR vs. language modelling

Probabilistic IR

$$P(R = 1|q, d) \tag{1}$$

Language modelling for IR

$$P(q|d) \tag{2}$$

Language model

$$P(w_1, \ldots, w_m) \tag{3}$$

- Language model assigns a *probability* to a sequence of *terms* (utterance)
- Used in speech recognition, machine translation, POS tagging ...

n-gram language model

$$P(w_i|\cdot) = P(w_i|w_{i-1}, \ldots, w_{i-(n-1)}) \tag{4}$$

$$\begin{aligned} P(w_1, \ldots, w_m) = P(w_1) \times P(w_2|w_1) \times \ldots \\ \times P(w_i|w_{i-1}, \ldots, w_{i-(n-1)}) \times \ldots \\ \times P(w_m|w_{m-1}, \ldots, w_{m-(n-1)}) \end{aligned} \tag{5}$$

- In n-gram, prob of word depends only on previous $n-1$ words
- Prob of utterance product of prob of each word
- No backwards, long-range dependencies

Unigram language model

$$P(w_i|\cdot) = P(w_i) \tag{6}$$

$$P(w_1, \ldots, w_m) = \prod_{i=1}^{m} P(w_i) \tag{7}$$

- In unigram model, word probability is independent
- Probability of utterance is product of probability of individual words

Building a unigram language model (MLE)

Nay, but this dotage of our general's . . . We were dissever'd. Hastily lead away.

the	0.031208
and	0.029623
. . .	. . .
agamemnon	0.000110
troy	0.000109
. . .	. . .
abaissiez	0.000001
aarons	0.000001

- A ULM can be built from relative frequency of terms in example text, or *discourse*, D
- Refer to this as maximum likelihood estimator (MLE)

$$\hat{P}_{\text{mle}}(t|D) = \frac{f_{D,t}}{|D|} \tag{8}$$

Testing utterances

$$
\begin{aligned}
&P(\text{"In delivering my son from me I bury a second husband"} \,|\, D) \\
&\quad = P(\text{"In"} \,|\, D) \times P(\text{"delivering"} \,|\, D) \times \ldots \times P(\text{"husband"} \,|\, D) \\
&\quad\quad = 0.012255 \times 0.000004 \times \ldots \times 0.000328 \\
&\quad\quad\quad = 5.00e-32 \qquad (9)
\end{aligned}
$$

- Probability of given utterance testable against model
- Any utterance will have a very low probability
- But what we usually care about is relative probability
  - Computation note: best to sum log probabilities; product of raw probabilities underflows

Unseen terms

$P(s\|D)$	s
4.45e-32	in delivering my son from me i bury a second husband
...	...
2.23e-25	what hope is there of his majestys amendment
6.09e-28	he hath abandond his physicians madam under whose
0.00e+00	practices he hath persecuted time with hope and finds no other
3.71e-35	advantage in the process but only the losing of hope by time
1.03e-27	this young gentlewoman had a father o that had how

- With MLE language model, if term never occurs in discourse D
- then utterance containing that term has probability 0
- which is generally undesirable

Smoothed language models

To avoid this, *smooth* the language model.

- Add (e.g.) half to count of every term (uniform prior):

$$P_U(t|D) = \frac{f_{D,t} + 0.5}{|D| + |D_C| \cdot 0.5} \tag{10}$$

  (note: no longer a proper probability distribution, when taking unseen documents into account)

- Linearly interpolate with some "background" universe, C:

$$P_\lambda(t|D) = \lambda \frac{f_{D,t}}{|D|} + (1 - \lambda) P(t|C) \tag{11}$$

Matching utterance to models

- Utterance to test is Shakespeare's play "All's Well that Ends Well" (1605)
- Compare against unigram models from:
  - Other plays by William Shakespeare (1564–1616)
  - King James Version of the Bible (1611)
  - Librettos by W.S. Gilbert (1836–1911) for operas of Arthur Sullivan (1842–1900)
  - Plays by John Galsworthy (1867–1933)
  - Lyrics to first five albums by Miley Cyrus (1992–)
- Assign unseen terms a probability of 1 in 10 million.
- Calculate log probability of AWtEW given each corpus

Find model for utterance

$\log P(\text{AWtEW})$	C
−170,531	Shakespeare's other plays
−183,387	Gilbert and Sullivan's operas
−192,887	Plays for John Galsworthy (1867-1933)
−196,460	King James Version of the Bible
−225,877	Lyrics to first five Miley Cyrus albums

- Probability of "All's Well that Ends Well" being randomly generated from any unigram model very low
  - If $\log P()$ is $-170,000$, then $P()$ is around 1 in $1.0e+73658$ (1 divided by 1 followed by 73,658 zeros).
- But more likely to be generated by model of Shakespeare's plays than any other model

Language models for IR

In LM in IR, estimate probability of query given (model of) document:

$$\hat{P}(q|d) = P(q|M_d) \tag{12}$$

where M_d is a unigram language model of document:

$$P_{\text{mle}}(q|d) = \prod_{t \in q} P(q_t|d) \tag{13}$$

which (naively) could be MLE model

$$P_{\text{mle}}(q_t|d) = \frac{f_{d,t}}{|d|} \tag{14}$$

Then rank queries by decreasing $P(q|M_d)$. (We don't care about absolute probabilities, only relative ones.)

Intuition of LM for IR

$P(q|M_d)$ in words asks:

> *How likely is the model that generated the document to also generate the query?*

Understood as searcher behaviour:

- Searcher told (or learns) to build queries using words likely to occur in relevant documents
- Thus, their query attempts to approximate language of relevant documents
- Testing against document language models then reasonable

Bayesian development of LM for IR

$$P(d|q,R) = \frac{P(q|d,R)P(d|R)P(R)}{P(q|R)} \tag{15}$$

Drop document-independent values:

$$P(d|q,R) \propto P(q|d,R)P(d|R) \tag{16}$$

Assume uniform prior belief for $P(d|R)$

$$P(d|q,R) \propto P(q|d,R) \tag{17}$$

Smoothing of LM: Linear interpolation

Collection C is background model for linear interpolation (so-called Jelinek-Mercer smoothing):

$$P_{\mathrm{JM}}(w|d) = (1-\lambda)P_{\mathrm{mle}}(w|d) + \lambda P(w|C) \quad (18)$$

$$P_{\mathrm{mle}}(w|d) = \frac{f_{d,t}}{|d|} \quad (19)$$

$$P(w|C) = \frac{c_t}{C} \quad (20)$$

- λ needs to be tuned for corpus, query stream
- Larger λ for shorter queries, smaller for longer
- Larger λ for newswire, smaller for web

Why smoothing?

$$P_{\mathrm{JM}}(w|d) = (1-\lambda)P_{\mathrm{mle}}(w|d) + \lambda P(w|C) \tag{21}$$

- Originally, smoothing to avoid exact match requirement
- But in fact smoothing important for performance regardless
- To see why, perform analysis in terms of:
  - TF
  - IDF
  - Document length
  - QTF

Why smoothing?

$$\hat{P}_(w|d) = P_{\text{mle}}(w|d) = \frac{f_t}{|d|} \tag{22}$$

Without smoothing:

- TF comes in f_t
- Document length from $|d|$
- QTF from summing each occasion of query term
- But no IDF
- Frequent terms get high weight in document
- . . . even if non-discriminative (think "the")

Why smoothing?

$$P_{\mathrm{JM}}(w|d) = (1-\lambda)P_{\mathrm{mle}}(w|d) + \lambda P(w|C) \tag{23}$$

With smoothing:

- Note that $\lambda P(w|C)$ (collection frequency of term) added for every document
- Effect is to dampen importance of $P(w|d)$ for collection-frequent terms
- So, in effect, it is an IDF term

Alternative smoothing

Alternative smoothing model (known as Dirichlet smoothing):

$$P_{\text{Dir}}(w|d) = \frac{f_{dt} + \mu P(w|C)}{|d| + \mu} \tag{24}$$

- μ parameter needs setting
- But much less sensitive than λ in Jelinek-Mercer
- $\mu \approx 2,000$ good general choice
- Technically: language model is multinomial; Dirichlet is conjugate prior to multinomial
  - So also nicer theoretically

Comparative performance

		Method		
Collection	Query	BM25	LM (JM)	LM (Dir)
TREC8 Newswire	short	0.2292	0.2310	**0.2470**
	medium	0.2523	0.2582	**0.2621**
	long	0.2454	**0.2608**	0.2597
TREC9 Web	short	0.1602	0.1212	**0.1864**
	medium	0.1950	0.1799	**0.2302**
	long	0.2053	0.1788	**0.2164**

Table : Bennett, Scholer, and Uitdenbogerd, "A Comparative Study of Probabilistic and Language Models for IR", ADCS 2008

- Language models perform at least as well as BM25
- But with fewer parameters to tune (1 versus 3)
- Jelinek-Mercer better for long queries on newswire data
- . . . Dirichlet for short (e.g. web) ones
- . . . and for web data (any query length)

Looking back and forward

Back: Language Models

- Classical probabilistic IR tries to solve $P(R|d,q)$
- Language models instead solve $P(q|d)$
- (which is proportional to $P(d|q)$, assuming to document prior)
- Language model gives probability of sequence of words
- Unigram language model says probability of utterance $P(s|M)$ is product of probability of individual terms $\prod_{w \in s} P(w|M)$
- Simple model of $P(w|M)$ is frequency of word in source corpus

Looking back and forward

Back: Language Models in IR

- In LM4IR, language model calculates $P(q|M_d)$
- Model of document, M_d, unigram MLE of term frequency
- . . . smoothed with collection frequency
- LM4IR achieves at least as good effectiveness as BM25
- . . . with fewer parameters (1 versus 3)
- . . . and nicer theory

Looking back and forward

Forward

- Next lecture: extensions of language modelling
  - E.g. relevance feedback
- Later: topic models as extension of this “generative” model

Further reading

- Chapter 12, "Language models for information retrieval"[1], of Manning, Raghavan, and Schutze, *Introduction to Information Retrieval*, CUP, 2009.
- Ponte and Croft, "A Language Model Approach to IR", SIGIR, 1998 (one of earliest uses of LM in IR).
- Zhai and Lafferty, "A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval", SIGIR, 2001 (compares Jelinek-Mercer and Dirichlet smoothing)

[1] http://nlp.stanford.edu/IR-book/pdf/12lmodel.pdf