Retrieval Model: Language Models

• Introduction to language models
• Unigram language model
• Document language model estimation
  – Maximum Likelihood estimation
  – Maximum a posterior estimation
  – Jelinek Mercer Smoothing
• Model-based feedback
Language Models: Motivation

• Vector space model for information retrieval
  – Documents and queries are vectors in the term space
  – Relevance is measured by the similarity between document vectors and query vector

• Problems for vector space model
  – Ad-hoc term weighting schemes
  – Ad-hoc similarity measurement
  – No justification of the relationship between relevance and similarity

• We need more principled retrieval models…

Introduction to Language Models:

● A Language model can be created for any language sample
  ➢ A document
  ➢ A collection of documents
  ➢ Sentence, paragraph, chapter, query…

● The size of the language sample affects the quality of the language model
  ➢ Long documents have a more accurate model
  ➢ Short documents have a less accurate model
  ➢ Model for sentence, paragraph or query may not be reliable
Introduction to Language Models:

- A document **language model** defines a probability distribution over indexed terms
  - E.g., the probability of generating a term
  - Sum of the probabilities is 1
- A query can be seen as observed data from unknown models
  - Query also defines a language model (more on this later)
- How might the models be used for IR?
  - Rank documents by \( \text{Pr}(\tilde{q} | \tilde{d}_i) \)
  - Rank documents by language models of \( \tilde{q} \) and \( \tilde{d}_i \) based on kullback-Leibler (KL) divergence between the models (come later)

Language Model for IR: Example

1. \( \tilde{q} \): sport, basketball
2. \( \tilde{d}_1 \): sport, basketball, ticket, sport
3. \( \tilde{d}_2 \): basketball, ticket, finance, ticket, sport
4. \( \tilde{d}_3 \): stock, finance, finance, stock

Estimating language model for each document

Generate retrieval results
Language Models

Three basic problems for language models

• What type of probabilistic distribution can be used to construct language models?
• How to estimate the parameters of the distribution of the language models?
• How to compute the likelihood of generating queries given the language modes of documents?

Multinomial/Unigram Language Models

Language model built by multinomial distribution on single terms (i.e., unigram) in the vocabulary

Examples:
Five words in vocabulary (sport, basketball, ticket, finance, stock)

For a document $d_i$, its language mode is:

$\{P_i(\text{sport}), P_i(\text{basketball}), P_i(\text{ticket}), P_i(\text{finance}), P_i(\text{stock})\}$

Formally:

The language model is: $\{P_i(w) \text{ for any word } w \text{ in vocabulary } V\}$

$$\sum_k P_i(w_k) = 1 \quad 0 \leq P_i(w_k) \leq 1$$
Multinomial/Unigram Language Models

- Multinomial Model for \( d_1 \): sport, basketball, ticket, sport
- Multinomial Model for \( d_2 \): basketball, ticket, finance, ticket, sport
- Multinomial Model for \( d_3 \): stock, finance, finance, stock

Estimating language model for each document

Maximum Likelihood Estimation (MLE)

**Maximum Likelihood Estimation:**
- Find model parameters that make generation likelihood reach maximum:
  \[ M^* = \text{argmax}_M \Pr(D|M) \]

  There are K words in vocabulary, \( w_1, ..., w_K \) (e.g., 5)

  Data: one document \( d_i \) with counts \( tf_i(w_1), ..., tf_i(w_K) \), and length \( |d_i| \)

  Model: multinomial M with parameters \( \{p_i(w_k)\} \)

  Likelihood: \( \Pr(d_i|M) \)
  \[ M^* = \text{argmax}_M \Pr(d_i|M) \]
Maximum Likelihood Estimation (MLE)

\[
p(\tilde{d}_i | M) = \left( \frac{1}{|\tilde{d}_i|} \right) \prod_{k=1}^{K} p_i(w_k)^{t_f(w_k)} \propto \prod_{k=1}^{K} p_i(w_k)^{t_f(w_k)}
\]

\[
l(\tilde{d}_i | M) = \log p(\tilde{d}_i | M) = \sum_k t_f(w_k) \log p_i(w_k)
\]

\[
\frac{\partial l}{\partial p_i(w_k)} = \frac{t_f(w_k)}{p_i(w_k)} + \lambda = 0 \quad \Rightarrow \quad p_i(w_k) = \frac{-t_f(w_k)}{\lambda}
\]

Since \( \sum_k p_i(w_k) = 1 \), \( \lambda = -\sum_k t_f(w_k) = |\tilde{d}_i| \) So, \( p_i(w_k) = \frac{c_i(w_k)}{|\tilde{d}_i|} \)

Use Lagrange multiplier approach
Set partial derivatives to zero
Get maximum likelihood estimate

**Estimating language model for each document**

- \((p_{sp}, p_b, p_t, p_s) = (0.5, 0.25, 0.25, 0, 0)\)
- \((p_{sp}, p_b, p_t, p_s) = (0.2, 0.2, 0.4, 0, 0)\)
- \((p_{sp}, p_b, p_t, p_s) = (0, 0, 0, 0.5, 0.5)\)
Maximum Likelihood Estimation (MLE)

Maximum Likelihood Estimation:
- Assign zero probabilities to unseen words in small sample

A specific example:
Only two words in vocabulary ($w_1$=sport, $w_2$=business) like (head, tail) for a coin; A document $\bar{d}$ generates sequence of two words or draw a coin for many times

$$Pr(\bar{d} | M) = \frac{\bar{d}_i}{tf_i(w_1) \cdot tf_i(w_2)} p_i(w_1)^{\delta_i(w_1)}(1 - p_i(w_1))^\delta_i(w_2)$$

Only observe two words (flip the coin twice) and MLE estimators are:
- “business sport” $P_i(w_1)$*=0.5
- “sport sport” $P_i(w_1)$*=1  ?
- “business business” $P_i(w_1)$*=0  ?

Data sparseness problem
Solution to Sparse Data Problems

- Maximum a posterior (MAP) estimation
- Shrinkage
- Bayesian ensemble approach

Maximum A Posterior (MAP) Estimation

Maximum A Posterior Estimation:
- Select a model that maximizes the probability of model given observed data
  \[ M^* = \arg \max_M \Pr(M|D) = \arg \max_M \Pr(D|M) \Pr(M) \]
  - \( \Pr(M) \): Prior belief/knowledge
  - Use prior \( \Pr(M) \) to avoid zero probabilities

A specific example:
Only two words in vocabulary (sport, business)

For a document \( d_i \):

\[
\Pr(M | d_i) = \left( \frac{d_i}{tf_i(w_1) tf_i(w_2)} \right) p_i(w_1)^{y_i(w_1)} p_i(w_2)^{y_i(w_2)} \Pr(M)
\]

Prior Distribution
Maximum A Posterior (MAP) Estimation

**Maximum A Posterior Estimation:**
- Introduce prior on the multinomial distribution
  - Use prior $Pr(M)$ to avoid zero probabilities, most of coins are more or less unbiased
  - Use Dirichlet prior on $p(w)$
- $\text{Dir}(\vec{\alpha} | \alpha_1, \ldots, \alpha_K) = \frac{\Gamma(\alpha_1 + \cdots + \alpha_K)}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_K)} \prod_k p_i(w_k)^{\alpha_i - 1}, \quad \sum_k p_i(w_k) = 1, \quad 0 \leq p_i(w_k) \leq 1$

Hyper-parameters
- Constant for $p_K$
- $\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} \, dt$,
  - $\Gamma(n+1) = n!$ if $n \in \mathbb{Z}$

---

Maximum A Posterior (MAP) Estimation

For the two word example:
- a Dirichlet prior $Pr(M) \propto p(w_1)^2 (1 - p(w_1))^2$
Maximum A Posterior (MAP) Estimation

- **Maximum A Posterior:**
  
  \[ M^* = \arg\max_M \Pr(M|D) = \arg\max_M \Pr(D|M) \Pr(M) \]

  \[
  \Pr(d_i | M) \Pr(M) \propto p_i(w_{i1}) \frac{1}{2} (1 - p_i(w_{i1})) p_i(w_{i2}) \frac{1}{2} (1 - p_i(w_{i2}))^{-1}
  \]

  \[
  = p_i(w_{i1}) \frac{1}{2} (1 - p_i(w_{i2}))^{-1}
  \]

  Pseudo Counts

  \[
  M^* = \arg\max_{p_i(w_i)} p_i(w_{i1}) \frac{1}{2} (1 - p_i(w_{i2}))^{-1}
  \]

A specific example:

Only observe two words (flip a coin twice):

“sport sport” \( p_i(w_{i1})^*=1 \) ?

![Graphs showing the probability of observing specific words](image-url)
Maximum A Posterior (MAP) Estimation

A specific example:

Only observe two words (flip a coin twice):
“sport sport” \( P(w_1)^* = 1 \) ?

\[
p(w_1)^* = \frac{f(w_1) + \alpha_1 - 1}{f(w_1) + \alpha_1 - 1 + f(w_2) + \alpha_2 - 1}
\]

\[
= \frac{2 + 3 - 1}{2 + 3 - 1 + 0 + 3 - 1} = \frac{4}{6} = \frac{2}{3}
\]

MAP Estimation
Unigram Language Model

Maximum A Posterior Estimation:

- Use Dirichlet prior for multinomial distribution
- How to set the parameters for Dirichlet prior?
MAP Estimation
Unigram Language Model

Maximum A Posterior Estimation:
- Use Dirichlet prior for multinomial distribution

There are $K$ terms in the vocabulary:

Multinomial: $\vec{p}_i = \{ p_i(w_1), \ldots, p_K(w_i) \}; \sum_k p_i(w_k) = 1, \ 0 \leq p_i(w_k) \leq 1$

$\text{Dir}(p_i | \alpha_1, \ldots, \alpha_K) = \frac{\Gamma(\alpha_i + \cdots + \alpha_K)}{\Gamma(\alpha_i) \cdots \Gamma(\alpha_K)} \prod_k p_i(w_k)^{\alpha_i - 1}, \ \sum_k p_i(w_k) = 1, \ 0 \leq p_i(w_k) \leq 1$

Hyper-parameters

Constant for $p_K$

MAP Estimation for unigram language model:

$\hat{p}_i = \arg \max \frac{\Gamma(\alpha_i + \cdots + \alpha_K)}{\Gamma(\alpha_i) \cdots \Gamma(\alpha_K)} \prod_k p_i(w_k)^{\gamma_i(w_i) + \alpha_i - 1} \prod_k p_i(w_k)^{\alpha_i - 1}$

st. $\sum_k p_i(w_k) = 1, \ 0 \leq p_i(w_k) \leq 1$

Use Lagrange Multiplier; Set derivative to 0

$\hat{p}_i(w_k) = \frac{tf_i(w_k) + \alpha_k - 1}{\sum_k (tf_i(w_k) + \alpha_k - 1)}$
MAP Estimation
Unigram Language Model

MAP Estimation for unigram language model:
Use Lagrange Multiplier; Set derivative to 0

\[ \tilde{p}_i(w_k) = \frac{tf_i(w_k) + \alpha_i - 1}{\sum_k (tf_k(w_k) + \alpha_k - 1)} \]

How to determine the appropriate value for hyper-parameters?

- When nothing observed from a document
  \[ \tilde{p}_i(w_k) = \frac{\alpha_k - 1}{\sum_i (\alpha_i - 1)} \]

- What is most likely \( p_i(w_k) \) without looking at the content of the document?

MAP Estimation
Unigram Language Model

MAP Estimation for unigram language model:

- What is most likely \( p_i(w_k) \) without looking at the content of the document?

- The most likely \( p_i(w_k) \) without looking into the content of the document \( d \) is the unigram probability of the collection:

  \[ \{p(w_1|c), p(w_2|c), \ldots, p(w_K|c)\} \]

Without any information, guess the behavior of one member on the behavior of whole population

\[ \tilde{p}_i(w_k) = \frac{\alpha_i - 1}{\sum_i (\alpha_i - 1)} = p_i(w_k) \Rightarrow \alpha_i - 1 = \mu p_i(w_k) \]
**MAP Estimation**

**Unigram Language Model**

**MAP Estimation for unigram language model:**

\[
\hat{p} = \operatorname*{arg\,max}_{\hat{p}} \frac{\Gamma(\alpha_i + \cdots + \alpha_K)}{\Gamma(\alpha_i) \cdots \Gamma(\alpha_K)} \prod_k p_i(w_k)^{f_i(w_k)} \prod_k p_i(w_k)^{\mu \varphi_i(w_k)} \\
\text{st. } \sum_k p_i(w_k) = 1, \quad 0 \leq p_i(w_k) \leq 1 \\
= \operatorname*{arg\,max}_{\hat{p}} \prod_k p_i(w_k)^{f_i(w_k) + \mu \varphi_i(w_k)} \\
\text{st. } \sum_k p_i(w_k) = 1, \quad 0 \leq p_i(w_k) \leq 1
\]

Use Lagrange Multiplier; Set derivative to 0

\[
\hat{p}_i(w_k) = \frac{tf_i(w_k) + \mu \varphi_i(w_k)}{\sum_k tf_i(w_k) + \mu}
\]

Pseudo counts
Pseudo document length

**Maximum A Posterior (MAP) Estimation**

**Dirichlet MAP Estimation for unigram language model:**

Step 0: compute the probability on whole collection based collection unigram language model

\[
p_c(w_k) = \frac{\sum_i tf_i(w_k)}{\sum_i |d_i|}
\]

Step 1: for each document \(d_i\), compute its smoothed unigram language model (Dirichlet smoothing) as

\[
p_l(w_k) = \frac{tf_i(w_k) + \mu \varphi_l(w_k)}{|d_i| + \mu}
\]
Step 2: For a given query $q = \{ tf_q(w_1), \ldots, tf_q(w_k) \}$

- For each document $d_i$, compute likelihood

$$p(q \mid d_i) = \prod_{k=1}^{K} \left[ p(w_i \mid d_i) \right]^{tf_q(w_k)} = \prod_{k=1}^{K} \left[ \frac{tf_i(w_k) + \mu p_c(w_k)}{|d_i| + \mu} \right]^{tf_q(w_k)}$$

- The larger the likelihood, the more relevant the document is to the query