A Language Modeling Approach to Information Retrieval

Melih Baydar
Murat Demirbüken

Outline

- Description of the Problem
- Motivation/Importance
- Previous Work
- Model Description
- Empirical Results

Description of the Problem

A model for IR

For effectively retrieving relevant documents by IR strategies, the documents are transformed into a suitable representation.

IR strategies contains a specific model for its document representation purposes.

Motivation/Importance

- **(Without) The parametric assumption**

  "It is unnecessary to construct a parametric model of the data when we have the actual data."

- **Documents are (not) members of pre-defined classes**

  "Each query needs to be looked at individually and documents will not necessarily fall cleanly into predetermined sets"

Previous Work

2-Poisson model

- make distributional assumption
- assume a pre-existing classification of documents

Fury model

- the integration of indexing and retrieval models
- the collection statistics are used in a heuristic fashion in order to estimate the probabilities of assigning concepts to documents

Kalt model

- *tf* and document length are both integral parts of the model rather than being heuristics as they are in many other models
- assume that documents were necessarily drawn from k language models representing the k classes of interest.

Model Description

- Infer a language model M_d for each document
  - A model for each class in previous works.
- Rank documents according to the estimate of producing the query
- MLE of probability of t under the term distribution for doc d

$$\hat{p}_{ml}(t|M_d) = \frac{tf_{(t,d)}}{dl_d}$$

Model Description (cont'd)

- Assumption : Given a particular language model, query terms occur independently
  - Allows ranking formula

$$\prod_{t \in Q} \hat{p}_{ml}(t, d)$$

  - Problem 1 -> If a document d is missing one or more query terms,then $p(t|M_d) = 0$

•Solution ? Non-existing term is possible!

$\frac{cf_t}{cs}$, where cf_t is the total # of term t in collection, cs is the total # of all terms

Model Description (cont'd)

- Problem 2 -> To make algorithm work on document sized samples, take mean probability of t in documents containing it

$$\hat{p}_{avg}(t) = \frac{\left(\sum_{d_{(t \in d)}} p_{ml}(t|M_d)\right)}{df_t}$$

  - But not every d that contains t is from the same language model M_d,
  - Calculate risk of using the mean to estimate $p(t|M_d)$

Model Description (cont'd)

- To minimize the risk, model risk for term t in doc d

$$\hat{R}_{t,d} = \left(\frac{1.0}{(1.0 + \overline{f}_t)}\right) \times \left(\frac{\overline{f}_t}{(1.0 + \overline{f}_t)}\right)^{tf_{t,d}}$$

where $\overline{f}_t$ is the mean of tf of t in docs it occurs i.e. $p_{avg}(t) \times dl_d$

Model Description (cont'd)

- Estimate of probability of producing the query for a given document model

$$\hat{p}(t|M_d) = \begin{cases} p_{ml}(t,d)^{(1.0-\hat{R}_{t,d})} \times p_{avg}(t)^{\hat{R}_{t,d}} & \text{if } tf_{(t,d)} > 0 \\ \frac{cf_t}{cs} & \text{otherwise} \end{cases}$$

Then,

$$\hat{p}(Q|M_d) = \prod_{t \in Q} \hat{p}(t|M_d) \times \prod_{t \notin Q} 1.0 - \hat{p}(t|M_d)$$

Empirical Results

	tf.idf	LM	%chg	I/D	Sign	Wilc.
Rel:	6501	6501				
Rret.:	3201	3364	+5.09	36/43	0.0000⋆	0.0002⋆
Prec.						
0.00	0.7439	0.7590	+2.0	10/22	0.7383	0.5709
0.10	0.4521	0.4910	+8.6	24/42	0.2204	0.0761
0.20	0.3514	0.4045	+15.1	27/44	0.0871	0.0081⋆
0.30	0.2761	0.3342	+21.0	28/43	0.0330⋆	0.0054⋆
0.40	0.2093	0.2572	+22.9	25/39	0.0541	0.0158⋆
0.50	0.1558	0.2061	+32.3	24/35	0.0205⋆	0.0018⋆
0.60	0.1024	0.1405	+37.1	22/27	0.0008⋆	0.0027⋆
0.70	0.0451	0.0760	+68.7	13/15	0.0037⋆	0.0062⋆
0.80	0.0160	0.0432	+169.6	9/10	0.0107⋆	0.0035⋆
0.90	0.0033	0.0063	+89.3	2/3	0.5000	undef
1.00	0.0028	0.0050	+76.9	2/3	0.5000	undef
Avg:	0.1868	0.2233	+19.55	32/49	0.0222⋆	0.0003⋆
Prec.						
5	0.4939	0.5020	+1.7	10/21	0.6682	0.4106
10	0.4449	0.4898	+10.1	22/30	0.0081⋆	0.0154⋆
15	0.3932	0.4435	+12.8	19/26	0.0145⋆	0.0038⋆
20	0.3643	0.4051	+11.2	22/34	0.0607	0.0218⋆
30	0.3313	0.3707	+11.9	28/41	0.0138⋆	0.0070⋆
100	0.2157	0.2500	+15.9	32/42	0.0005⋆	0.0003⋆
200	0.1655	0.1903	+15.0	35/44	0.0001⋆	0.0000⋆
500	0.1004	0.1119	+11.4	36/44	0.0000⋆	0.0000⋆
1000	0.0653	0.0687	+5.1	36/43	0.0000⋆	0.0002⋆
RPr	0.2473	0.2876	+16.32	34/43	0.0001⋆	0.0000⋆

I -> count of queries for which performance improved using new method

D -> count of queries for which perf. was different

Sign -> significance val acc. to sign test

Wilc -> significance val acc. to Wilcoxon test

Question&Comments

Thanks for listening…