Query Modelling

Autocompletions, Temporal profiling
Query Modelling

- Query modelling is used to better capture the users information need

- Bridges the vocabulary gap between the query and the documents to be retrieved
  - bicycle vs bike, LOTUS vs POTUS vs president of US
  - EOS 1000D vs canon rebel series

- Used in Query expansions, suggestions and auto-completions

- Temporal Profiling for improving result quality
Query Modelling Applications - Auto-completions

- **Query Auto-completions**: Given a prefix text suggest the most probable queries

- **Better auto-completions are based on better modelling user intents**
• Query Suggestions: Given the complete query, try to guess related queries or what the user might be interested in

• spelling corrections are a subset of it
Query Modelling Applications - Expansions

- Autocompletions and suggestions are explicit

- To improve the quality of results the search engines implicitly enrich or expand queries
  
  - Input query: bike prices
  
  - Expanded query: bike prices OR bicycle price OR bicycle cost OR two-wheeler cost OR …

How can queries be modelled?

How to use temporal information to better model queries?
Query Modelling - Ingredients

- **Query log mining:** Usage of query logs and behavioral statistics while interacting with the search engines

- Query logs are not always available especially query logs for a long duration of time

- Information about new and emerging topics are unavailable even in query logs

- **Pseudo-relevance feedback:** Assuming top documents retrieved by the search engine to be relevant
Query Log mining: Usage of query logs and behavioral statistics while interacting with the search engines

Example of query logs and usage logs:

[10/09 06:39:59] Click: [webresult][q=holiday decorations][21]
http://www.stretcher.com/stories/99/991129b.cfm
[10/09 06:41:31] Click: [webresult][q=home made halloween decorations][6]
http://www.rats2u.com/halloween/halloween_crafts.htm
[10/09 06:52:18] Click: [webresult][q=home made halloween decorations][8]
http://www.rpmwebworx.com/halloweenhouse/index.html
[10/09 06:53:30] Click: [webresult][q=home made halloween decorations][20]
http://www.halloween-magazine.com/
Query Logs

- **Query log mining**: Usage of query logs and behavioral statistics while interacting with the search engines

- Example of query logs and usage logs:

```
1326  coats tire equipment  2006-04-28 15:53:18
1326  coats tire equipment  2006-05-03 19:15:01
1326  verizon wireless  2006-05-09 00:09:22
1326  www.crazycrudeals.com  2006-05-23 18:00:30
1337  uslandrecords.com  2006-03-01 11:50:34  1  http://www.seda-cog.org
1337  titlesourceinc  2006-03-14 15:45:55  1  http://www.titlesourceinc.com
1337  select business services  2006-03-14 15:51:41
1337  select business services title  2006-03-14 15:52:10
1337  cbc companies  2006-03-14 15:52:44  2  http://www.cbc-companies.com
1337  cbc companies  2006-03-14 15:52:44  3  http://www.cbc-companies.com
1337  cbc companies  2006-03-14 15:52:44  4  http://www.mktgservices.com
1337  national real estate settlement services  2006-03-14 15:59:13  1  http://www.realtms.com
1337  national real estate settlement services  2006-03-14 15:59:13  7  http://dmoz.org
1337  pennsylvania real estate settlement services  2006-03-14 16:04:40
1337  pennsylvania real estate settlement services  2006-03-14 16:05:11
1337  sunbury pennsylvania real estate settlement services  2006-03-14 16:05:47
1337  sunbury pennsylvania real estate settlement services  2006-03-14 16:06:28  14  http://pa.optimuslaw.com
```

[10/09 06:53:53] Click: [webresult][q=home made halloween decorations][20]
http://www.halloween-magazine.com/
Query Auto-completions

- Candidate set generation for a given prefix $p$

- Candidates are ranked according to the *most popular completion* to the given prefix and top-$k$ are presented as most promising

- A weight $w(q)$ for each candidate $q$ is estimated from the document collection of query log

- How are weights computed?

  - Most popular query — based on query frequency or how many times has the query been issues

  \[
  MPC(p) = \arg \max_{q \in C_p} w(q), \quad w(q) = \frac{f(q)}{\sum_{i \in Q} f(i)}
  \]
What is missed in such kind of a modelling approach?
• Temporal aspect of popularity not taken into account

• Historically popular candidates might overpower recent trends

• Periodically popular queries might not be represented
• Temporal aspect of popularity not taken into account

• Historically popular candidates might overpower recent trends

• Periodically popular queries might not be represented
• Weights assigned to candidates should not only take into account absolute historical frequencies but also
  • Trends
  • periodicities
  • bursts

• Time series analysis techniques can be used to determine the forecast the popularity weight
  • Trends - double exponential smoothing
  • periodicities - triple exponential smoothing
  • burst - burst detection techniques
Pseuso-Relevance feedback

- Assume top-k documents to be relevant
- Use this set for query modelling or retrieval effectiveness
Temporal query profiles

- Temporal profiles are constructed to determine how temporally relevant queries

- Queries can be classified into
  - Atemporal
  - Temporally Ambiguous
  - Temporally unambiguous

- Model the period of time relevant to the query
• For a given query rank the documents according to the standard retrieval models (say LM as discussed in the previous lectures)

• Each document has a score and a publication time

• Plot the time lines which will then be analysed to find the query classes
Estimating Time Series Values

• What would be the value at a given time point?

  • count of documents published at that time point (contribution of each doc same, i.e., 1.0)

  • sum of the scores of the documents (relevance score)

  • Language modelling approach to establish \( P(t|Q) \)

\[
P(t|Q) = \sum_{D \in R} P(t|D) \frac{P(Q|D)}{\sum_{D' \in R} P(Q|D')}
\]
What about the time points with no documents published?

- Distribution of documents containing the query term irregular (vocabulary gap)
- Neighbouring time points having high values increases the probability of having a non-zero value for a time point
Smoothing using Background Model

- Smoothing using language model

- Take the distribution of the entire collection $P(t \mid C)$

- What is the concentration of documents in the underlying distribution at $t$
Smoothing using Background Model

\[ P'(t|Q) = \lambda P(t|Q) + (1 - \lambda) P(t|c) \]

- \( P(t|Q) \) is the original score at \( t \)
- \( P(t|c) \) is the collection score at \( t \)
- \( \lambda \) is the smoothing parameter
Smoothing using Background Model

\[
P'(t|Q) = \lambda P(t|Q) + (1 - \lambda) P(t|c)
\]

How do we incorporate information from neighbouring time points?
Smoothing using Background Model

\[ P'(t|Q) = \lambda P(t|Q) + (1 - \lambda) P(t|c) \]

How do we incorporate information from neighbouring time points?

- Use time series prediction methods like exponential smoothing
• How do we compare time series?

• **Clarity** - Based on KL divergence between collection and query distribution
  
  • KL divergence is used to compare two distribution
  
  • The more the divergence the more clear the query is

• **Periodicity** - detect if the query time-series so obtained is periodic
  
  • Use auto-correlation or similar methods (discussed in lecture before)
• Statistics of Rank order
  
  • How much of the power of the distribution contained in the peaks?

  • To focus on peaks we use rank order of high peaks using the Kurtosis measure

• Burst Model
  
  • Identify the burstiness of a distribution using burst detection techniques

  • Finally using these features classify the queries into the query classes
Temporal query profiles

Temporal Unambiguous Queries

- **earthquake in Armenia**
- **matrix**

Temporal Ambiguous Queries

- **hostage taking**
- **nba playoffs**

6.1 Manual Classification of TREC ad-hoc Queries

Attempting to work with a standard set of queries, we used the TREC ad hoc query sets. Since none of the queries in these sets were annotated with temporal classes, they had to be hand-classified. We classified more than 50 queries using the TREC topic descriptions. Annotators were not provided temporal profiles.

Specifically, annotators were asked to judge, based on the topic, description, and narrative fields, whether a query was requesting multiple events, a single event, or had no preference. The annotators labeled 18 common...
References and Further Readings
