Ranking database results based on context.

Arthur van Bunningen
Overview

• Motivation
• A model of relevance
• Introduction to preference rules
• Connecting preference rules and relevance
• Implanting the model on top of a DBMS
• Summary and outlook
Motivation

• Many information available in digital form.
• Need for personalization
  – Based on person
  – Based on situation
  ➢ Based on context
A model of relevance

- Traditional:

\[ P(D = d|Q = q \land U = u) = \frac{P(Q = q|D = d \land U = u) \cdot P(D = d|U = u)}{P(Q = q|U = u)} \]

- Adding “situatedness” to user:

\[ P(D = d|Q = q \land U = u_{sit}) = \frac{P(Q = q|D = d \land U = u_{sit}) \cdot P(D = d|U = u_{sit})}{P(Q = q|U = u_{sit})} \]
Influence of context

- Context affects query generation.
- Context affects general relevance of tuple.
- If query has lower probability to follow from situation, resulting documents more likely to be relevant.

\[
P(D = d | Q = q \land U = u_{sit}) = \\
P(Q = q | D = d \land U = u_{sit}) P(D = d | U = u_{sit}) \\
P(Q = q | U = u_{sit})
\]
Influence of situatedness

• Context affects query generation.
• Context affects general relevance of a document.
• If query has lower probability to follow from situation, resulting documents more likely to be relevant.

\[
P(D = d | Q = q \land U = u_{sit}) = \frac{P(Q = q | D = d \land U = u_{sit}) P(D = d | U = u_{sit})}{P(Q = q | U = u_{sit})}
\]
Introduction to preference rules

• Relates contextual features to features of documents together with a weight \((C, P, \sigma)\)

• Examples:
  – (KoffieRoom, Jazz, 0.8)
  – (Cheerful, Eighties, 0.7)

• Semantics:
  – Preferences taken from choices in history.
  – *If we pick a random moment in history when a situation had feature C and a user made a choice, the chance that (s)he chose a document with feature P is \(\sigma\).*
Preference rules and relevance

- Introducing features in relevance model

\[ P(D = d | U = u_{sit}) = P(D = d | F(d) = f, G(u_{sit}) = g) \]

- The most relevant document in a situation:
  - The document for which the chance is highest that we chose before, in the situation with the same features, a document with the same features.
  - Relevance of one document:

\[
= \prod_{r_p | r_c \in g} \begin{cases} 
\sigma & \text{if } r_p \in f \\
1 - \sigma & \text{if } r_p \notin f 
\end{cases}
\]
Example & assumptions

Situation: KoffieRoom, Cheerful

\[ \neg \text{Jazz}, \neg \text{Eighties} : (1-0.8)(1-0.7) = 0.06 \]

- Document features of rules do not depend on document features of other rules.
- Rules define “everything”
  - Information not coded in rules (e.g. music is loud, reading a paper) does not play any role.
- What if situation did not occur before
  - System don’t know: Default rules
Uncertainty of features

- Conditioning

\[ P(D = d | G(u_{sit}) = g) = \sum_{F(d) = f} P(F(d) = f) \cdot P(D = d | F(d) = f, G(u_{sit}) = g) \]

\[ P(D = d) = \sum_{G(u_{sit}) = g} P(G(u_{sit}) = g) \cdot (\sum_{F(d) = f} P(F(d) = f) \cdot P(D = d | F(d) = f, G(u_{sit}) = g)) \]

- Leads to probability that document is the ideal document \( P(D=d) \) of:

\[ = \sum_{G(u_{sit}) = g} P(G(u_{sit}) = g) \cdot (\sum_{F(d) = f} P(F(d) = f) \cdot \prod_{r_p | r_c \in g} \begin{cases} \sigma & \text{if } r_p \in f \\ 1 - \sigma & \text{if } r_p \notin f \end{cases}) \]
A word about querying (1/2)

- Rules coded in Description Logics

\[
\text{Context} : \{PETER\} \sqcap (\exists \text{hasActivityType.FreeTimeActivity}) \\
\text{Preference} : \text{TvProgram} \sqcap (\exists \text{hasGenre.\{HUMAN-INTEREST\}})
\]

- DL model, maps to database
  - Concept tables
    - Movie, Person, Genre, ...
  - Role tables
    - hasGenre, hasRoom, hasMood, ...
Another word about querying (2/2)

• Uncertainty addressed with event expressions
  – Special DL operators that include calculation of event expression
  – Construct view

• Preference view is generated from rules
  – Contains view on all preferred documents with relevance score
  – Can be queried
  – Could be domain specific
Wrap-up and outlook

• Wrap-up
  – Relevance
  – Preference rules
  – Querying

• What’s next?
  – More implementation
  – Addressing efficiency
  – Explanation of results
Questions?

Ubiquitous computing will “make using a computer as refreshing as taking a walk in the woods” – Mark Weiser
Example

• Rules
  – (KoffieRoom, Jazz, 0.8)
  – (Cheerfull, Eighties, 0.7)

• Situation
  – KoffieRoom : 0.5
  – Cheerfull : 0.3
  – ...

• Document
  – Eighties: 0.9
  – ...

Example (continued)

- Sit: CoffeeRoom ∧ Cheerfull
  - Doc: Eighties ∧ Jazz
    • 0.8 * 0.7
  - Doc: Eighties ∧ ¬ Jazz
    • 0.8 * (1-0.7)
  - Doc: ¬ Eighties ∧ Jazz
    • (1-0.8) * 0.7
  - Doc: ¬ Eighties ∧ ¬ Jazz
    • (1-0.8) * (1-0.7)

- Sit: CoffeeRoom ∧ ¬ Cheerfull
- Sit: ¬ CoffeeRoom ∧ Cheerfull
- Sit ¬ CoffeeRoom ∧ ¬ Cheerfull