Finding and visualising contextual information in hierarchically structured documents

Fabio Crestani

Dept. Computer and Information Sciences
University of Strathclyde, Glasgow, UK
f.crestani@cis.strath.ac.uk
Outline

● Objectives
● The importance of “context”
● Hierarchically structured documents
● Identifying context: the SRIDE\textsuperscript{RB} model
● Visualising context: the DocBall metaphor
● Application areas: PENG and REVEAL-THIS projects
● Future work
Rationale

- Several new standards of representation of structured documents
  - SGML, XML, MPEG-7, MPEG-21, …
- Document structure provides information
- Users are often interested only in some small relevant part of long and structurally complex documents
  - Textbooks, manuals, videos, news bulletins, …
- Some parts of documents, though not relevant, provide “contextual information” that increases the usefulness of relevant information
Example

- A user needs to write a short report on the “liberty style in Scotland” and needs to find information about “Charles Rennie Macintosh” in a large heterogeneous collection of text and video material.
Objectives

- An information retrieval system for structured documents should:
  1. Enable the user to find potentially relevant parts of documents
  2. Identify parts that, though no relevant, are useful to better assess the relevance of potentially relevant parts of documents (the “context”)
  3. Visualise/present the potentially relevant parts “in context”
  4. Enable the user to browse the document and select the parts that he/she thinks are relevant and useful

- We want to design models that enable to build such system
The importance of context

- Context always been recognised as important
  - KWIC, TileBars, …
- Context more important for structured documents, Web documents, and for streaming media
- Context recently recognised as crucial to improve information access satisfaction and usefulness
  - EU 6th Framework RTD Programme, IRiX NoE, CoLIS5, Context Conference, SIGIR workshops, …
- Context can be approached from different research directions
  - Document segmentation, retrieval models, visualisation, interaction, …
Context: what is it?

- Many different definitions on context
  - See for example the CFP of the Context-05 Conference
- Formal definitions of context can be found in logical approaches to IR, AI, etc.
- Simple and intuitive definition for our use:
  
  Additional information that enhances the understanding of some information being looked at. This includes data, e.g. surrounding information, and metadata, e.g. the position of the information in the document, the relation with other sections of the document, the author and title of the section or the document.
Context: where is it?

- In long and structurally complex documents we need to identify where the context is
  - Use document structure and/or segment the document in information coherent segments
  - There might not be a single best segment proving context
  - Metadata, like document structure, document author, title, etc. needs to be presented
- This is a complex issue
  - A study is currently being carried out with journalists to identify what constitute context
  - Many variables to consider, e.g. user background knowledge, domain, application, etc.
Context: when is it useful?

- Context can be used to help the user to better identify potentially relevant documents.

- The search process [Shneiderman]:
  1. formulation
  2. search
  3. reviewing
  4. refinement

Present context to user here
A study of how to capture and use

- At the University of Strathclyde, for the past 5 years, we carried out a number of projects aimed at studying how to identify, capture and use context in:
  - Multimedia Information Retrieval (IR) applications
  - Mobile information access applications
  - Electronic publishing and digital libraries applications
- We use a combination of formal approaches and user interaction studies
- In the following I present some work on a specific type of document: hierarchically structured documents
Hierarchically structured documents (HSD)

- Many types of documents are naturally hierarchically structured
- Textbooks, manuals, scientific articles, Web pages, …
Creating HSD

- Some documents are not naturally HSD, but they can be made so
- Educational videos, news bulletins, etc.
Estimating relevance in HSD

- A number of models have been proposed for indexing and retrieving HSD
- Most approaches let the user indicate the specific structural elements of interest in the query
- We want the user just to specify the topic of interest and have the system deal with finding the best structural element that contain the information sought and the context
  - The user might not care about the specific structural element
  - The user might not know the document structure
Identifying context in HSD: the SRIDE\textsuperscript{RB} model

- We developed a model, SRIDE\textsuperscript{RB}, to estimate relevance in HSD that enables a system to identify the most useful structural element containing the information sought.

- The model combines:
  - A Multilayered Bayesian Network model -> estimates relevance at each HSD element
  - A Utility Theory model -> use Decision theory to decide the best HSD element to present to the user, based on some utility function

- Collaboration with the University of Granada (Spain), funded by the Spanish Research Council
The SRIDE$\text{RB}$ model

- In short, relevance is estimated as follows:
  1. A HSD is modelled as a Multilayered Bayesian Network
  2. Each unit is a binary random variable, taking values in the set \( \{u^+_{i,j}, u^-_{i,j}\} \)
  3. Relevance is estimated at the smallest structural level (\( \mathcal{L}_l \)) using term frequency statistics
  4. Relevance is propagated to all other structural levels
  5. Each element is ranked according to its estimated relevance
- But this does not take into consideration the context and does not identify the most useful structural element to present to the user
The SRIDE$^{RB}$ model (cont.)

- Graphical representation:
So, a re-ranking based on Utility Theory is carried out as follows:

1. Consider the following decision: \( r^+_{i,j} \), retrieve \( U_{i,j} \), or \( r^-_{i,j} \) do not retrieve \( U_{i,j} \), based the relevance of \( U_{i,j} \) and \( U_{i,j-1} \)

2. Use a Utility Function \( V(r_{i,j}) \), assigning utilities values to each decision, as from the following tables:

3. Evaluate the Expected Utility of retrieving a structural unit \( U_{i,j} \) with \( j \neq 1 \), given a query \( Q \) (combining utility and relevance, assuming conditional impendence)

4. Re-ranked elements considering the Expected Utility
The SRIDE$^\text{RB}$ model (cont.)

- Re-ranking can be carried out using different strategies:
  - Consider only $EU(r_{+i,j}/Q)$
  - Consider the ratio $EU(r_{+i,j}/Q) / EU(r_{-i,j}/Q)$
  - Consider the different $EU(r_{+i,j}/Q) - EU(r_{-i,j}/Q)$
- Utilities values can be learned and can be personalised:
  - E.g. is retrieval of a rel section of a non-rel chapter ($v^+_{+-}$) preferable to retrieval a non-rel section of a rel chapter ($v^+_{-+}$)?
  - Provide complete ordering of decisions, with associated values (in $[0,1]$):
    $$v^-_+ \leq v^+_+ \leq v^+_+ \leq v^-_+ \leq v^-_+ \leq v^+_+ \leq v^-_+ \leq v^+_+$$
- More details in the papers
Visualising context in HSD

- Sometime it is better to leave to the user the choice of the most relevant and useful structural element
  1. Specify desired element in the query
  2. Select the desired element when looking at the retrieval results
- In case 2, the system needs to visualise:
  - The document structure
  - The estimated relevance of each element
  - Context information (data and metadata)
- We need an integrated approach, presenting the above information with just one metaphor
Visualising context: past approaches

- Many different approaches proposed in the past in the document visualisation research area
- Some approaches are directed to specific types of documents and specific application
- Best general approaches for IR: Tilebars, Relevance Curves and Thumbnails
Tilebars and relevance curves

- Show the relevance of each passage or segment of text
- Show the document size
- Does not show the hierarchical relations of the document structural elements
- Does not show the context of the relevant elements
Thumbnails

- Show the appearance of the document
- Useful when the user works frequently with the same document set and can recognise a document by its thumbnail view.
- Could show context (e.g. KWIC)
- Does not show the size of the document
- Does not show the structure of the document
- Does not show the relevance of each element of the document
- Could be a good complement to Tilebars and Relevance Curves
The Docball metaphor

- We developed a visualisation metaphor that enables to show the structure, size, relevance of each element in the structure, and context
- Collaboration with the University of Valladolid (Spain), funded by the Royal Society
- Simple idea:
DocBalls applied to IR

- We built a prototype system that combines SRIDE$^{RB}$ with DocBall
- We introduced also other functionalities: like relevance feedback, query session history, zooming at a specific structural level, etc.
- User evaluation showed high levels of user satisfaction and effectiveness in search tasks
- Example of user session:
Presentation of retrieval results (example)
DocBall applied to Web IR

- An extension of the DocBall metaphor to the Web domain
  - Web pages have structure: HTML, XML
- A visual metaphor that can explain the user:
  - Where are relevant elements in relation to structure of a Web document
  - Where relevant Web documents are in relation to the entire Web site (work in progress)
The WebDocBall prototype system architecture

- We built a prototype Web search engine
  - Search carried out by Google Web APIs
  - Visualisation with SVG APIs
The WebDocBall system: results presentation

- Steps involved (after results are returned):
  1. Determine the structure of the Web document
  2. Estimate the relevance of each structural element with respect to the query
  3. Display structure and relevance using the DocBall metaphor

- Problems:
  - HTML labels are used with no discipline
  - HTML labels express structural information in an implicit way.

- Question:
  - Which are the most common HTML labels and what are their structural significance?
Determining the structure of a Web Document

- What are the most important tags to consider to determine the structure of a Web document?

- Small experiment:
  - Generate random URL using Google
    - 30,000 distinct HTML pages
    - 13,124,339 HTML labels
  - Find the HTML markup labels that are most often used:
    - content, p-implied, td, br, tr, img, p, comment, table, span, li, script, div, meta, input, title, center, hr, ul, body, html, head, form, link, select, frame, object, ol
  - We consider only those labels with evident structural meaning, used to express hierarchical relations
Web document structure (example)
Presentation of retrieval results (example)
Evaluation

- The SRIDE\textsuperscript{RB} model and the DocBall metaphor have been evaluated in small user and task oriented studies

- Very positive results:
  - The DocBall metaphor is intuitive
  - Compared with standard IR and Web search systems, our prototype improve user satisfaction and task effectiveness

- But:
  - More work is needed to improve efficiency (system was very slow)
  - More work is needed to model other HTML markup labels
Application and exploitation

- The SRIDE\textsuperscript{RB} model and the DocBall metaphor are currently being used in a number of projects we are involved:
  - EU FP6 STREP project “PENG” (PErsонаlised News content programmiNG)
  - EU FP6 STREP project “REVEAL-THIS” (Retrieval of Video and Language for The Home user in an Information Society)
  - INEX (INitiative for the Evaluation of XML Retrieval)
Conclusions and future work

- Identifying and visualise relevant information *in context* improves user effectiveness and satisfaction in the information access task

- But:
  - Context is domain and application specific
  - Context is user specific, i.e. personal
  - Context is time and location specific

- We need to be able to capture all these characteristics of context