Automated Classification

Lars Marius Garshol
<larsga@bouvet.no>
Topic Maps 2007
2007-03-21

Automated classification

What is it?

Why do it?

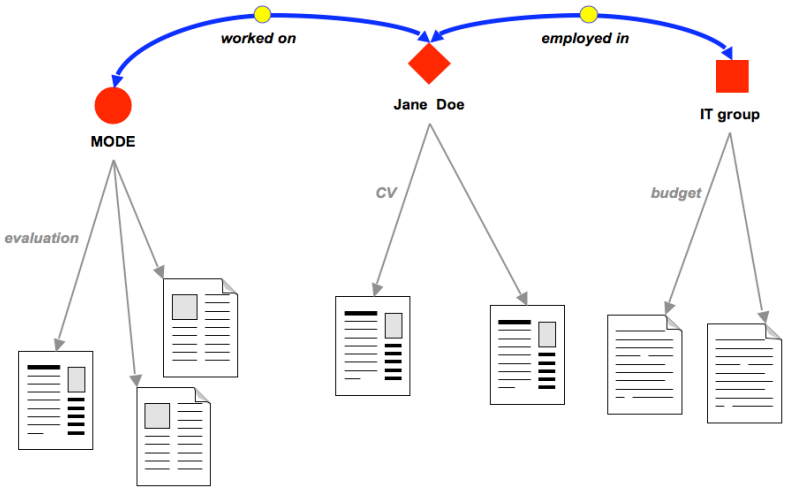
What is automated classification?

- **Create parts of a topic map automatically**
  - using the text in existing content as the source
  - not necessarily 100% automatic; user may help out
- **A hard task**
  - natural language processing is *very* complex
  - result is never perfect
- **However, it's possible to achieve some results**

Why automate classification?

- **Creating a topic map requires intellectual effort**
  - that is, it requires work by humans
- **Human effort = cost**
  - added value must be sufficient to justify the cost
  - in some cases either
    - the cost is too high, or
    - the value added is too limited
- **The purpose of automation is to lower the cost**
  - this increases the number of cases where the use of Topic Maps is justified

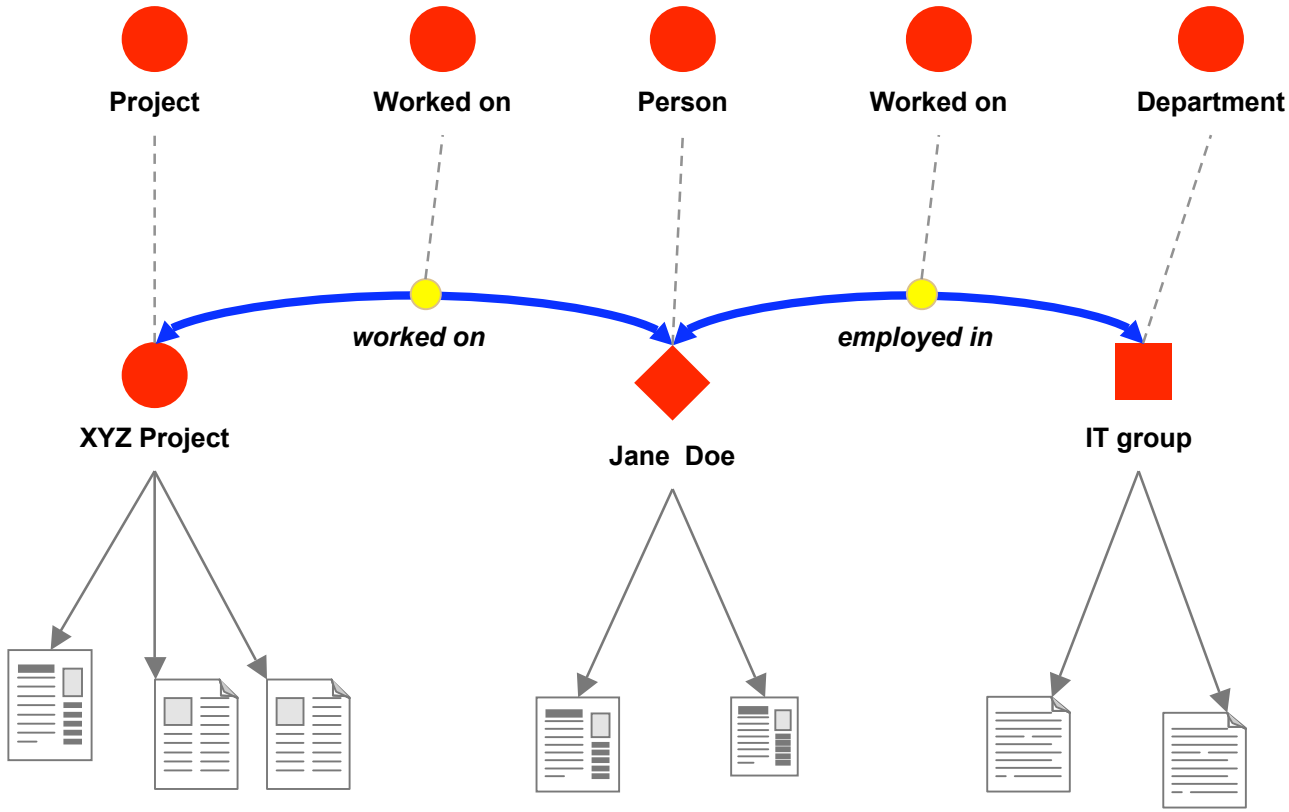
Classification competes with search



- **Requires intellectual effort**
- **Once created**
  - serves as site structure
  - allows navigation
  - allows improved search

- **Promises a plug-and-play solution**
- **Provides quite good findability**
- **Supporting navigation is harder**

Automatable tasks



- **Ontology**
  - hard
  - depends on requirements
  - one time only
- **Instance data**
  - hard
  - usually exists in other sources
- **Document keywords**
  - easier
  - frequent operation
  - usually no other sources

Classification

Requirements

Two different tasks

Two different approaches

Formats

- **Handle formats**
  - must automatically detect the formats of documents
  - must extract the text
  - preserving document structure is a plus

Many classification tools don't do this for you...

Relevance (1)

- It's not enough just to extract the text
  - must find the *relevant* text
- Can be difficult
- Data may need to be cleaned in advance

Relevance (2)

To take a simple example, if we were to do a full-text search for "XSLT" in the conference proceedings for the IDEAlliance conferences, there is of course a huge number of hits, but at the very top comes the topic "XSLT", which represents the XSLT standard. From there one can find the specification, papers about XSLT, which standards organization produced XSLT, tools implementing XSLT, tools using XSLT, etc

- This text says “XSLT” a lot
  - *every single occurrence* is an example
  - no useful information about XSLT
- XSLT is *not* a suitable keyword for this text
  - how is the computer to know?

Languages

- **Handle languages**
  - must automatically recognize the language of the text
  - must support the language
  - note that quality of results from one tool often varies from language to language

What's needed in language support?

- **Know what are common words (stop words)**
  - like "the", "of", "and", "in", ...
- **Recognizing different forms of the same word**
  - Topic Maps — emnekart
  - topic map — emnekartet
  - Topic Maps' — emnekartene
  - emnekartenes
  - emnekartets
- **Understanding of grammar**
  - in tools which analyze sentences support for new languages is quite hard to add
- **Word boundary detection**
  - some languages are written without spaces

Basic classification requirements

- **Correctly attach keywords to documents**
- **Discover new keywords**
  - handle compound keywords like “Topic Maps” and “New York”
- **Make use of existing list of keywords**
  - use to separate interesting keywords from non-interesting ones
  - use to recognize compound keywords
- **Infer document type**
  - is this a specification or an interview?

Many classification tools don’t do this for you...

Two kinds of categorization

- **Broad categorization**
  - categories are broadly defined
  - include many different subjects
- **Narrow categorization**
  - uses very specific keywords
  - each keyword is a single subject

Broad:

Environment, Crisis management

Narrow:

Water, Norway, drought, Drought Act, Cloud seeding, Morecambe Bay

December 29, 2006

Water from Norway to ease drought

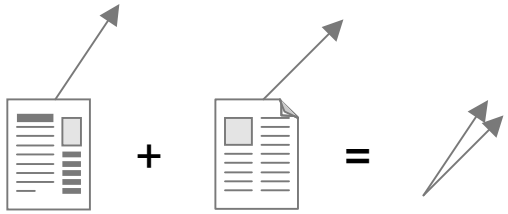
During the long, hot summer of 1976, when Britain faced its worst drought in 250 years, the Government considered a number of unusual solutions.

An emergency Drought Act was passed on August 6 and, by August 20, the Government had gathered information on the sinking of bore holes, the use of oil tankers to bring water from Norway, and the seeding of rain clouds — a method of forcing clouds to rain by spraying chemicals into the air.

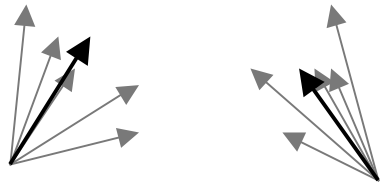
But cloud-seeding was ruled out and ministers were told that building a barrage at Morecambe Bay would be a cheaper way access water than importing it from Norway.

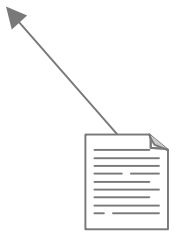
Latent semantic analysis (LSA)

- **Statistical approach to classification**
  - based on vector spaces and linear algebra
  - computes a vector for each document
  - angle between vectors = similarity of documents
  - assumes similar documents are about the same subject



- **Categorization**
  - needs a collection of documents for each category
  - computes a “footprint” for each category based on this
  - footprint is basically average vector for the documents
  - new document compared to footprints, and placed in categories where documents are “the most similar”





Keyword extraction

- **Also a statistical approach**
  - basically counts the number of times words appear in the text
  - uses various “tricks” to figure out what the best terms are, based on the counts
  - some implementations do full sentence analysis (parsing) of the text
  - produces a “relevance score” for each term in the document

Comparison

	LSA	Extraction
Training	yes	no
Best for	broad	narrow
New keywords	no	yes
Document type	yes	no
Scalability	challenging	no worries

Application

Classification results

Applications

Further work

Example of keyword extraction

Metadata? Thesauri? Taxonomies? Topic Maps!

Making sense of it all

By: Lars Marius Garshol
Affiliation: Development Manager Ontopia
Date: $Date: 2004/10/26 12:15:27 $

ontopia

Abstract

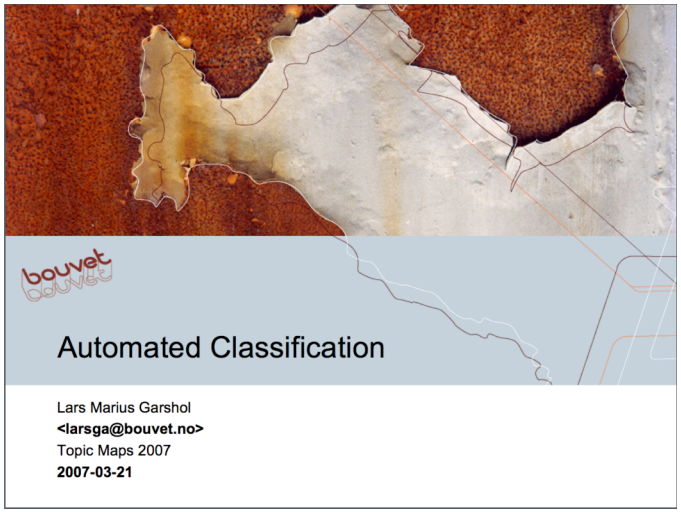
To be faced with a document collection and not to be able to find the information you know exists somewhere within it is a problem as old as the existence of document collections. Information Architecture is the discipline dealing with the modern version of this problem: how to organize web sites so that users actually can find what they are looking for.

Information architects have so far applied known and well-tried tools from library science to solve this problem, and now topic maps are sailing up as another potential tool for information architects. This raises the question of how topic maps compare with the traditional solutions, and that is the question this paper attempts to address.

The paper argues that topic maps go beyond the traditional solutions in the sense that it provides a framework within which they can be represented as they are, but also extended in ways which significantly improve information retrieval.

- **topic maps** **1.0**
- **metadata** **0.57**
- **subject-based class.** **0.42**
- *Core metadata* *0.42*
- **faceted classification** **0.34**
- **taxonomy** **0.22**
- **monolingual thesauri** **0.19**
- **controlled vocabulary** **0.19**
- **Dublin Core** **0.16**
- **thesauri** **0.16**
- *Dublin* *0.15*
- *keywords* *0.15*

Example #2



Automated classification	1.0	5
Topic Maps	0.51	14
XSLT	0.38	11
compound keywords	0.29	2
keywords	0.26	20
Lars	0.23	1
Marius	0.23	1
Garshol	0.22	1
...		

So how could this be used?

- **To help users classify new documents in a CMS interface**
  - suggest appropriate keywords, screened by user before approval
- **Automate classification of incoming documents**
  - this means lower quality, but also lower cost
- **Get an overview of interesting terms in a document corpus**
  - classify all documents, extract the most interesting terms
  - this can be used as the starting point for building an ontology
  - (keyword extraction only)

Example user interface

- **The user creates an article**
  - this screen then used to add keywords
  - user adjusts the proposals from the classifier

Example of corpus classification

- Processed 1334 papers from various XML conferences
- The terms shown are the most suggested keywords by the tool
- This is with no keyword list as input

- XML
- SGML
- DTD
- XSLT
- metadata
- *Markup*
- RDF
- topic maps
- *Internet*
- HTML
- SVG
- ebXML
- XSL

R&D stuff

- **It's possible to take this even further**
- **What is said about the keywords in the text can be used to group them by topic type**
  - it's very, very difficult to suggest specific topic types
  - some tools can do this for geographic entities, people, and companies
  - however, it *is* possible to produce nameless groups
- **This, of course, provides an even better starting point for an ontology**
  - in fact, it reduces the work of providing a rough starting topic map dramatically
  - this is R&D at the moment

Term grouping example

- **W3C, Oracle, DTD, metadata, authoring, ISO, SQL, DOM, SAX, SOAP, OASIS, IDEAlliance, Sun** ← a mess
- **PNG, GIF, JPEG** ← graphics formats
- **XML, RDF** ← well...
- **RTF, PDF** ← document formats
- **SVG, VML, WebCGM** ← XML graphics formats
- **Java, Python, Perl, XSLT, XQuery, ECMAScript, Prolog** ← programming languages
- **Microsoft, Excel, Adobe** ← not right
- **workflow, OMG** ← not right

Conclusion

Summing up

Conclusion

- **Automated classification is possible**
  - results are never perfect, however
  - the better the result, the less manual work required
- **Classifying documents**
  - this is the most important operation
  - this is where automation works best
- **Ontology generation**
  - a less important operation
  - automation can help, but still a mostly manual process