MULTI-CRITERIA EVALUATION OF INFORMATION RETRIEVAL TOOLS

Nishant Kumar and Jan Vanthienen
Katholieke Universiteit Leuven, Leuven Institute for Research on Information Systems
Naamsestraat 69, 3000 Leuven, Belgium

Jan De Beer and Marie-Francine Moens
Katholieke Universiteit Leuven, Legal Informatics and Information Retrieval
ICRI, Tiensestraat 41, 3000 Leuven, Belgium

Keywords: information retrieval, text mining, decision support systems, knowledge representation.

Abstract: We propose a generic methodology for the evaluation of Text Mining/Search and Information Retrieval tools based on their functional conformity to a predefined set of functional requirements prioritized by distinguishable user profiles. The methodology is worked out and applied within the context of a research project concerning the assessment of intelligent exploitation tools for unstructured information sources in the police domain. We present the general setting of our work, give an overview of our evaluation approach, and discuss our methodology for testing in greater detail. These kinds of evaluations are particularly useful for both (potential)purchasers of exploitation tools, given the high investments in time and money required in becoming proficient in their use, and developers who aim at producing better quality software products.

1 INTRODUCTION

The invent of various text and data mining algorithms and their continuous improvements in terms of accuracy, performance, scalability,... paired with an ever expanding market of software producers turning the algorithms into general-purpose, versatile, fully fledged and easy-to-use software products, driven by an ever growing interest and desire for such tools in various domains, strengthens the need to develop solid and sound evaluation procedures to test tools' absolute competence and relative competitiveness for their application and integration in live environments. As software vendors tend to proclaim superiority and supreme adequacy of their products, it is yet to be studied and verified through objective and sound means whether these claims hold true in practice. In our project this is achieved through the definition of various evaluation criteria, which will be used in a subsequent benchmarking stage.

Defining objective and adequate criteria is surely not a trivial task. First, the concept of relevance as the perceived quality or usability of any generated results, is by itself very subjective in nature, depending partly on the context of the task, the user, the anticipated outcome, the objective, etc. As a consequence, evaluation usually entails and is founded on human interaction and judgment, severely constraining the amount of testing and effort that can be spend. Lastly, a great number of heterogeneous and seemingly incomparable factors and criteria take part in a cognitive human judgment process, which is hard to reveal and formalize.

With their vast amounts of interconnected structured and unstructured data files, police forces throughout the world are gaining interests in powerful and reliable automated tools that turn data into useful, concise, accurate, and timely information and knowledge, to improve or assist in information sharing and criminal intelligence analysis. For police forces, information and knowledge make vital elements for the efficient and effective practicing of their operations. This widely known and well understood fact is translated into the concept of *Intelligence Led Policing* (ILP), as opposed to the more traditional, labor-intensive and less efficient strategy of *crime fighting*.

Section 2 briefly describes the project INFO-NS as the context of and as a case study for the development and application of our generic evaluation methodology. One facet of the evaluation spectrum, which consists of assessing the functional support of tools, termed *conformity testing*, will be covered in depth in Sect. 3. After a discussion of the evaluation model, we conclude in Sect. 4 with related work and references for further reading.

Kumar N., Vanthienen J., De Beer J. and Moens M. (2006).
MULTI-CRITERIA EVALUATION OF INFORMATION RETRIEVAL TOOLS.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - AIDSS*, pages 150-155
DOI: 10.5220/0002463601500155

2 PROJECT DESCRIPTION

2.1 Overview

The INFO-NS research project is an initiative of the Belgian Science Policy Office in collaboration with the Belgian police, and is carried out by the research groups pertaining to the authors of this paper. The aim of the project is to provide an objective study to the applicability of exploitation tools for unstructured information sources of the Belgian police. More specifically, it is studied how information retrieval, information extraction and information processing tools might leverage intelligence and decision support by exploiting, linking, and contextualizing the unstructured information that is contained in vast amounts of available free text material.

The project will achieve its objective through a thorough evaluation of existing (off-the-shelf) retrieval and text mining products of some of the leading and most promising software producers in the field on a workbench of test cases and evaluation criteria worked out in collaboration with police departments.

2.2 Evaluation Approach

We identified three major groups of evaluation criteria, capturing the applicability, the competence, and the practicality of the tools under evaluation.

Applicability The extent to which each of the preselected tools (an initial market selection) answers the identified functional needs of the various user profiles.

Competence The extent to which each of the tools performs at quality measures like capability, accuracy, flexibility, scalability, etc. For this purpose, task-specific evaluation procedures and criteria are devised.

Practicality Includes performance, as the extent to which system resources (memory, disk space, network band width,...) are efficiently utilised, considering extensive document collections and a large potential number of concurrent users, next to various, more subjective criteria, such as user-friendliness, user-system interaction, the quality of documentation, etc.

For the remaining of this paper, we restrict ourselves to the application of the first of these groups, coined conformity evaluation.

3 CONFORMITY EVALUATION

In this section we present our methodology for the evaluation of tools solely on the basis of their support (provision) regarding any functional needs and associated priorities for a number of distinct user profiles that are identified in the early stages of the project – the requirements analysis phase. The methodology is sufficiently generic, so that it be readily adoptable in other projects, and is quite broad in scope, so as to be readily portable to other situations in which some sort of multi-criteria evaluation or analysis is to be performed.[1]

After we present our methodology, we illustrate how we used this in the context of our project to pursue part of its objective. Given space and confidentiality constraints, we will however not go into too much detail. We end this section with a discussion of our proposed evaluation model.

3.1 Methodology

Our evaluation model assumes the following information is available.

- A set of tools to be evaluated $\mathcal{T} = \{T_i\}_{i=1}^{t}$.
- A set of relevant functionalities $\mathcal{F} = \{F_i\}_{i=1}^{f}$ with fixed semantics and identifying labels.
- A hierarchy $\mathcal{H}$ defined over the functionalities in $\mathcal{F}$ according to the inclusion relation $\supset$ (read: subsumes); $\mathcal{H} = \{(i,j) \mid F_i \supset F_j \wedge \neg\exists k \neq i,j : F_i \supset F_k \supset F_j\}$.
  Although not mandatory for our evaluation model, $\mathcal{H}$ puts an order upon a potentially large set $\mathcal{F}$ through the identification of atomic (indivisible) functionalities and their grouping to more general functionalities. As will become clearer further in this text, $\mathcal{H}$ allows us to proceed in a more methodical and systematic manner.
- For each tool T_i a *support tree* ST_i. This concept is worked out in definition 1 (see below).
- A set of *use cases* $\mathcal{U} = \{U_i\}_{i=1}^{u}$. Formally, every use case represents a logical grouping of related functionalities $U_i = \{F_{u_{i,j}}\}_{j=1}^{u_i}$.
  In practice, a use case represents some particular task which comprises several functional components, in turn consisting out of logically related functionalities. Common components pertain to data preprocessing, the support for accomplishing the task, visualisation and interaction, import-export capabilities, etc.

[1] We refer to the application of these and similar techniques in police domain for e.g. the prioritization of criminal investigations and the assessment of threats based on offender (group) profiles or environmental conditions.

The provision of multiple use cases allows the coverage of as many of the functionalities in $\mathcal{F}$ with a selection of any number of tools, given the faint likelihood of having one supertool; a tool that supports most tasks for everyone the best.

- A set of user profiles $\mathcal{P} = \{P_i\}_{i=1}^{p}$ with identified priorities regarding each functionality in $\mathcal{F}$.
- For each use case U_i and user profile P_j a *requirements tree* $RT_{i,j}$. This concept is worked out in definition 2.

Definition 1 (support tree) *The support tree of tool T_i, noted ST_i, is a tree structure corresponding $\mathcal{H}$, wherein the node representing F_j carries as attributes the label of F_j for identification, as well as an indication of the degree to which the tool supports F_j.*

Definition 2 (requirements tree) *The requirements tree of use case U_i for user profile P_j, noted $RT_{i,j}$, is a tree structure corresponding $\mathcal{H}$ restrained to $\{F_{u_{i,k}}\}_{k=1}^{u_i}$. In this structure, the node representing $F_{u_{i,k}}$ carries as attributes the label of $F_{u_{i,k}}$ for identification, as well as an indication of the degree to which $F_{u_{i,k}}$ is desired by users of profile P_j.*

Given this information, we now aim to evaluate how well each tool conforms to every use case in $\mathcal{U}$, and this for every user profile in $\mathcal{P}$ individually. As every combination of use case and user profile is reflected in a unique requirements tree, we thus want to compute the conformity between every tool and requirements tree. For this, we define an abstract operator τ that evaluates the "degree of support" of definition 1 with respect to the "degree of desire" of definition 2, given a particular support tree ST and a requirements tree RT.[2]

$$\tau : ST \times RT \rightarrow \mathbb{R}$$

The repeated operation of τ for each tool on all requirement trees then produces an array of conformity scores, which can optionally be combined (through weighing e.g.) to global scores, or used to filter away dominated tools. This latter option can be achieved by retaining only those tools in $\mathcal{T}$ for which there is at least one requirements tree for which they give the best result (among the other tools in $\mathcal{T}$). The selection of non-dominated tools is given by the following formula.

$$\bigcup_{j,k} \{T_i \mid \tau(ST_i, RT_{j,k}) = max_r\ \tau(ST_r, RT_{j,k})\} \quad (1)$$

In the formula, the union is taken of all best tools for every requirements tree.

[2]In the workout, we show how we defined the operator τ.

3.2 Workout

3.2.1 User Profiles

In association with the Belgian police, we first identified four user profiles for the tools being sought after. These profiles are quite general in nature and are equally found in other police organisations, even though they may go by different names.

Administrator Collects, manages, structures, and sometimes already relates facts described in official documents (case reports e.g.), dispatching the gathered or derived information to other services upon request or as part of the information flow.

Investigator Conducts criminal investigations. Her task is to compile a comprehensive report (a legal case file) describing all acts and elements part of the investigation, which will be the main source of evidence used by judicial authorities for prosecution.

Operational analyst Examines, supports and assists criminal investigations, especially more complex ones. New hypotheses, alternatives, links, contextualisations, schematisations, etc. can be suggested or provided.

Strategic analyst Analyze safety problems; their tendencies, trends, patterns, processes, novelties, etc. Such analysis serve as the basis for strategic (long-term) decision making, pinpointing the main security problems and giving insights into their nature and characteristics. This allows allocating limited police resources for top efficacy.

3.2.2 Functionalities and Priorities

We compiled an extensive list of functional requirements, partly technical requirements of a more prerequisite nature that we as technical researchers were able to identify ourselves, and partly functional needs of the user group, which we gathered through questionnaires, meetings and work sessions held throughout the different police departments. A topical, high-level overview follows.

- Tool Configuration
  - Document content indexing process
  - Security and access control
  - Support for multiple languages and document formats [3]

[3]A prerequisite is the support for the three official languages in Belgium, namely Dutch, French, and German, along with English for open sources. Given the emerging threat of terrorism and the organised crime wave coming from the East, interest in Arabic and Asiatic languages is growing.

  - Inclusion of metadata
  - Automatic clustering or classification
- Search & Retrieve
  - Metadata search: document id, url, title, type, language, origin,...
  - Free text search: crosslingual, fuzzy, conceptual search,...
  - Entity search: crosslingual, phonetic, morphological search,...
  - Similarity search: crosslingual search-by-example
  - Taxonomy search: category or cluster selection
  - Multi-modal search
  - Monitoring: automated signaling of relevant, new or updated information, e.g. through user profiling and proactive search agents
- User-System interaction
  - Assisted formulation of search queries
  - Filtering of search results through successive formulation of queries
  - Relevance feedback and query refinement
  - Repeated search and search history
  - Visualisation, exportation, manipulation, and browsing of search results
  - Automated clustering or classification of the search result
- Qualitative Analysis
  - Discovery of relations between terms, concepts, entities, or any combination
  - Assisted annotation of documents, also known as text coding
  - Support for creation of graphical schemes
  - Automated recognition and classification of entities
  - Visualisation and exportation of analysis results

Functionalities were hierarchically ordered and presented in clear language to police officers of the identified user profiles. By having them score the functionalities to their active needs, we were able to associate real-valued priority values to $\mathcal{F}$ for each profile.

3.2.3 Use Cases and Requirement Trees

Out of $\mathcal{F}$ we were able to distinguish ten distinct use cases. As an example, consider the use case "free text search". As all others, this use case is made up of several functional components, including tool configuration, document indexing, text search, and various interaction functions. Given the hierarchical ordering of our functionalities we set up the corresponding requirements tree.

3.2.4 Tools and Support Trees

For each of the tools considered for evaluation, we will construct their corresponding support tree.[4] The implementation is done through the specification of *support values* for each of the functionalities in $\mathcal{F}$. Concrete, the support value of tool T_i for F_j, noted $\sigma(T_i, F_j)$, is a real number in unit interval giving expression to the "degree of support" of definition 1. A value of 0 indicates no support, 1 indicates full support, and partial support might be mapped along the continuum.[5]

$$\sigma : \mathcal{T} \times \mathcal{F} \rightarrow [0, 1]$$

3.2.5 Conformity Matching

In order to match a requirements tree with the support tree of a tool, we implement the matching operator τ through the specification of *objective functions* at every single node in the requirements tree. These functions take as arguments the support values of the tool, the requirement priorities of a user profile, and some extra, profile-independent parameters. Each objective function produces as a result a real-valued conformity score with respect to the functionality associated to the node having the function attached. Through repeated and systematic evaluation of these functions - starting at the leaf nodes, tracing intermediate nodes, and ending at the root node - one obtains a global conformity score for each tool on every use case and for every user profile.

In addition, next to the detailed intermediate results, which can give useful insight as to why and at which points some tools fail, we build two clear and concise contracted tables which we can easily derive through priority composition. One table gives the conformity of each tool for each use case (combined over all profiles), whereas the other table gives the conformity of each tool for every user profile (combined over all use cases).

3.3 Discussion

3.3.1 Considerations

To safeguard the proper application of the proposed evaluation model with a sound interpretation and use of the produced results, a few conditions and remarks should be made.

[4] To prevent any market influence and to safeguard the confidentiality of our research, we choose not to make the tools publicly known, at least not at this stage.

[5] Whenever no (reliable) information can be obtained about the degree of support, we safely assume support is missing; $\sigma = 0$.

First of all, support values should be obtained by confident means so as to resemble the tools' true support, otherwise results are deemed to be meaningless and therefore useless. Through own experience, we found that software vendors have a tendency to badge their products as being extremely versatile and applicable to the specific task at hand.[6] As a researcher, one should therefore strive to establish these values through objective and motivated means, possibly skimming any documentation that describes the tools' capabilities and features, attending demo presentations, installing evaluation versions, looking for related studies conducted by trustworthy third-parties, through personal use or prior knowledge, etc.

Second, it is the nested objective functions which serve to produce the absolute conformity scoring values. As these functions capture the very semantics of the evaluation (matching) taking place, they should be deviced with great care and precision. Judicious use of mathematical operators (additive, multiplicative, fuzzy logical,...) and overall consistency in design are primal points of attention.

Third, interpretation of results should primarily be based on a relative comparison of tools by identifying any significant differences in conformity scores, as the absolute scores may depend heavily on the somewhat arbitrary structuring of the tree, composition of objective functions, and parameter settings.

As a last remark, we observed a marked difference in prioritizing functionalities among different user profiles. Whereas some profiles cautiously distributed priorities as if they were given some fixed amount of priority points, others rated the majority of functionalities equally and sufficiently high. Judicious use of normalizing operators in objective functions at different levels in the requirements tree prevents the model from being biased by these different prioritizing behaviors. The successive application of small-scale normalization will give the desired effect of conformity scores being somewhat more tailored for profiles having defined more balanced priority schemes, provided those scheme reflect actual gradations in desirability of functional needs.

3.3.2 Possible Uses

Given accurate support values and priorities, one could use this procedure to make a selection of tools on the basis of functional conformity, as suggested by (1). Such selection would allow to identify tools that are promising and suitable candidates for further, more thorough testing. Since the number of tools on the market is usually quite large, and time is limited in research projects, this early kind of preliminary evaluation may turn out to be an interesting, efficient and effective exercise.

As we had little prior knowledge about the tools under study and too little time to perform a full-scale support analysis of the tools, we decided to make a preselection motivated through early conformity impressions drawn from tool documentation, demo presentations and personal contacting, and retaining the conformity evaluation procedure until a later stage of our project.

[6] As an example, tools claiming certain functional capabilities merely by the provision of some general-purpose macro language or *Application Programming Interface* (API) cannot be considered meeting our interest in directly applicable, off-the-shelf tools.

4 RELATED WORK

In the past decade, many IT implementation projects have been conducted in collaboration with police forces throughout the world. Most of these projects revolve around the centralisation and consolidation of various digitized information sources, for the purpose of information fusion, information sharing, improved availability (ubiquitousness) of information, and advanced exploitation for criminal analysis. Among the more renowned (pilot) projects we mention the trendsetting and since 1997 vigorously growing COPLINK project of Chen et al. ((Hauck et al., 2001; Atabakhsh et al., 2001; Chen et al., 2002; Chen et al., 2003; Chen et al., 2004)) in the state of Arizona, US, the CLEAR (*Citizen Law Enforcement Analysis and Reporting*) project in Chicaco, the FLINTS (*Forensic Led Intelligence System*) project developed since 1999 by West Midlands police under the auspices of R. M. Leary ((Leary,)), the OVER project of Oatley, Ewart en Zeleznikow ((Oatley et al., 2004)), in association with West Midlands police since 2000, and the expanding KDD-PN (*Knowledge Discovery from Databases – Police Netherlands*) project including the *DataDetective* tool since 2001.

We found that the majority of projects are quite similar in scope and nature, involving the application of data mining and subsequent visualisation techniques on information that is implicitly assumed to be electronically available in structured, clean, preprocessed, and unprotected (readily accessible) form. Among the more inspiring technologies are decision tree building, offender profiling, social network analysis, spatio-temporal statistics and visualisation techniques including hot spot analysis. Applications and numerous case studies can be found in a recent book of J. Mena on the subject matter ((Mena, 2003)). Kumar ((Kumar et al., 2006)) & De Beer ((De Beer et al., 2006)) has discussed in detail on the quality of commercial information retrieval and text mining tools. Rijsbergen ((Van Rijsbergen, 1979)) has discussed evaluation techniques for measuring

the performance of information retrieval tools. Related studies can be found from Lancaster ((Lancaster, 1968)), Cooper ((Cooper, 1973)), and Ingwersen ((Ingwersen, 1992)) on functional use assessment, relevance assessment, and quality evaluation, while the evaluation methodologies suggested by Elder and Abbot ((Elder and Abbott, 1998)), Nakhaeizadeh, and Schnabl ((Nakhaeizadeh and Schnabl, 1997)), Collier et al. ((Collier et al., 1999)) are notable.

5 CONCLUSION

Through our research project with the Belgian police, we encountered many interesting aspects that are not readily found or touched upon in literature on the subject, most noticeably on the issues of privacy, security, legal aspects such as the evidential value of generated results, data preprocessing and cleaning, integration, flexibility, adaptability, and performance of exploitation tools in practical settings. In this paper we presented our proposed evaluation methodology for conformity testing of software tools, which fits in a larger framework of tool evaluation. We hope our work may prove useful, inspire or ponder other field workers on these topics, as we believe the success and promising future of these tools heavily depends on their careful consideration.

ACKNOWLEDGMENTS

The authors would like to thank the Belgian police for their interest and active collaboration, in particular Kris D'Hoore, Martine Pattyn and Paul Wouters. This work was supported by the Belgian Science Policy Office through their AGORA research programme. AG/01/101

REFERENCES

Atabakhsh, H., Schroeder, J., Chen, H., Chau, M., Xu, J. J., Zhang, J., and Bi, H. (2001). Coplink knowledge management for law enforcement: Text analysis, visualization and collaboration. In *Proceedings of the National Conference for Digital Government Research*, volume 1.

Chen, H., Chung, W., Xu, J. J., Wang, G., Qin, Y., and Chau, M. (2004). Crime data mining: a general framework and some examples. 37(4).

Chen, H., Schroeder, J., Hauck, R., Ridgeway, L., Atabakhsh, H., Gupta, H., Boarman, C., Rasmussen, K., and Clements, A. (2002). Coplink connect: information and knowledge management for law enforcement. 34(3):271–285.

Chen, H., Zeng, D., Atabakhsh, H., Wyzga, W., and Schroeder, J. (2003). Coplink managing law enforcement data and knowledge. 46(1):28–34.

Collier, K., Carey, B., Sautter, D., and Marjaniemi, C. (1999). A methodology for evaluating and selecting data mining software. In *Proceedings of the International Conference on System Sciences*.

Cooper, W. S. (1973). On selecting a measure of retrieval effectiveness. *Journal of the American Society for Information Science*, 24(2):87–100.

De Beer, J., Kumar, N., Moens, M.-F., and Vanthienen, J. (2006). Assessing the state of the art of commercial tools for unstructured information exploitation.

Elder, J. F. and Abbott, D. W. (1998). A comparison of leading data mining tools. Technical report.

Hauck, R. V., Schroeder, J., and Chen, H. (2001). Coplink: Developing information sharing and criminal intelligence analysis technologies for law enforcement. In *Proceedings of the National Conference for Digital Government Research*, volume 1, pages 134–140.

Ingwersen, P. (1992). *Information Retrieval Interaction*. Taylor Graham, London.

Kumar, N., De Beer, J., Vanthienen, J., and Moens, M.-F. (2006). A study on the quality of enterprise search tools.

Lancaster, F. W. (1968). *Information Retrieval Systems: Characteristics, Testing and Evaluation*. Wiley, New York.

Leary, R. M. The role of the national intelligence model and flints in improving police performance. Online at http://www.homeoffice.gov.uk/docs2/resconf2002daytwo.html.

Mena, J. (2003). *Investigative data mining for security and criminal detection*. First edition.

Nakhaeizadeh, G. and Schnabl, A. (1997). Development of multi-criteria metrics for evaluation of data mining algorithms. In *Proceedings KDD-97*. AAAI Press.

Oatley, G. C., Ewart, B. W., and Zeleznikow, J. (2004). Decision support systems for police: Lessons from the application of data mining techniques to 'soft' forensic evidence.

Van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworths London, second edition.