Information supply for high-level fusion services

Christian Mårtenson, Pontus Svenson
Division of Information Systems, Swedish Defence Research Agency, SE 164 90 Stockholm, Sweden
cmart,ponsve@foi.se

ABSTRACT

In this paper, we describe the problem of efficiently supplying high-level fusion services (situation and impact assessment) with adequate information using semantic technology and formulate an optimization problem version of it. We begin by discussing situation awareness and the need for computer tools that assist human analysts and decision makers with their sense-making. Such tools are necessary in part because of the vast amount of information that is available for analysis in today's command and control systems: the human operators need help to sort out the relevant parts. This kind of filtering requirement is however not limited to humans: automatic or semi-automatic fusion tools also need to limit the information they use in their processing. Simple such filtering could be done based on geographical location, but as the number of advanced fusion services used in the command and control system increases, more advanced techniques need to be used. We describe the information supply process when dealing with several (possibly heterogeneous) sources of differing quality and describe the concepts of information view and information scope. We describe how semantic queries can be used to achieve such filtering, and in particular describe this implemented for Impactorium, a framework tool for situation and impact assessment developed by FOI. The threat models in Impactorium previously relied solely on simple indicator tags for information supply. This can be done more robustly by adding semantic queries to the threat models. The paper concludes with a summary and some discussion of future work in this area.

Keywords: information fusion, information retrieval, situation assessment, threat assessment

1. INTRODUCTION

Information management is an important use of computers in many fields. Users need tools in order to be able to find information of interest. Since most users collect very large amounts of data, there is a need for fast and efficient systems for information retrieval.

For example, legal and medical records need to be processed and stored to enable quick retrieval by lawyers and doctors, and home users need to be able to find the image, video or music file that they are looking for. These two categories of users often use completely opposite ways of organizing and storing their information. In medicine, for example, it is possible to develop detailed ontologies [2] and to tag available information with terms from the ontology. The user, who is a professional who is familiar with the ontologies used, can formulate their information request in precise terms. Home users, in contrast, often store their files randomly, sometimes perhaps tagging their data with ordinary words, and on multiple disks. This problem calls for search methods that are able to search vast amounts of unstructured information, perhaps combined with some kind of recommendation system based on comparisons with other users [1].

Both problems are solved by using methods from the field of information retrieval [9], and both problems are relevant for the case of military intelligence analysis, which is the problem domain of interest to us. Intelligence analysts, like doctors, have a precise terminology that they can use in searching for relevant material that is written by other intelligence analysts. However, the vast majority of content that an intelligence analyst needs to consider is not written by fellow professionals, but is instead unstructured images from sensors, text collected from the web or from traditional media, or text reports filed by soldiers who have been on patrol. This heterogeneous set of data calls for hybrid approaches in intelligence information management systems.

The goal of information fusion [6] is to help users achieve situation awareness and avoid information overflow. However, information overflow is not a problem only for humans: computer algorithms whose running time is non-linear in the size of the input also need to restrict the amount of data available to them.

Intelligent Sensing, Situation Management, Impact Assessment, and Cyber-Sensing, edited by John F. Buford, Gabe Jakobson, Stephen Mott, Michael J. Mendenhall, Proc. of SPIE Vol. 7352 73520N · © 2009 SPIE · CCC code: 0277-786X/09/$18 · doi: 10.1117/12.818544

In this paper, we describe the problem of information supply for high-level fusion by comparing with information retrieval and introduce an optimization problem formulation of it. The paper is outlined as follows. In section 2, we give a brief background on information management and our problem domain. This is followed by section 3 and 4, where we give some motivational applications and then introduce the information supply problem, describing the concepts of information view and information scope. Section 4 also describes an optimization problem formulation of the information supply problem. Lastly, we summarize our paper and discuss possible future work in this area.

2. BACKGROUND

Information management has many components, each of which needs to work in order for an information management system to adequately fulfill the needs of the user. In this paper, we will focus on the functionality to

- Store information. The information could be of many different kinds, structured or unstructured, and should be stored in a way that allows fast retrieval. Most often, the information will be stored with metadata associated to it. Metadata is structured information that is attached to the information to provide information about, e.g., authorship, creation date, revision history. Metadata can also come in the form of tags, which attempt to categorize the information.
- Tag information, that is, add metadata to each information object. Tags can come both in the form of terms from an ontology and free-text.
- Retrieve information. When the user has an information need, it is necessary to quickly find all relevant information objects. This requires the ability for the user to express their information need in an unambiguous, clear and, preferably, succinct way. It also requires the ability to match the users information need with the scored information, and retrieve the best matches to the user.

Perhaps the most common example today of using tagging in order to enable better searches is for music. Electronic music is automatically tagged with metadata such as artist, album, and genre and can also be tagged with user-generated tags that enable the users to quickly search for and find content.

Information management and all its components are also an integral part of the intelligence analysis process. Intelligence analysts have to quickly handle large amounts of information, determine what parts of the available information is useful, analyze it, and present their findings to their customers. Intelligence analysts also require computer tools that help them perform their analysis. Such help can come from information fusion [6] tools. Information fusion deals with the sorting, filtering and combination of data from heterogeneous information sources. Ideally, it could provide a correct situation picture of what is going on in the world to the user. However, it is not currently possible to automate the information fusion process sufficiently so that this can be done completely without human intervention[1]. Instead, information fusion tools should be considered decision support systems that help the human operators and analysts to create pieces of situation pictures and achieve situation awareness.

In the processing chain of information, information fusion and information management are intertwined in several ways. The results of the fusion need to be stored for later retrieval, and the fusion processed need to get access to correct information, both background knowledge (stored in slowly-changing databases) and real-time results from collection resources.

As such, there is a need to adapt the use of the information management system to the needs of the information fusion system. The problem of determining what information to look for is not restricted only to human analysts. Computer tools too need to restrain the amount of information they use. Part of the reason for why this is necessary is related to the computational complexity of the information fusion algorithms: if a situation assessment algorithm is exponential in the size of its input, reducing the number of items fed to it could be a necessary condition in order to receive a result in time for it to be useful.

Information supply for intelligence analysis is, in some ways, a simpler problem than information retrieval from, for example, the web. The facts that the data collection can be more guided and all data should be used for one purpose

[1] And it is arguable whether such automation would even be desirable for intelligence analysis purposes.

contribute to making it easier. But other factors contribute to making the problem considerably more difficult: the data often comes from many disparate sources, and has been collected by different persons. To this should be added the problem of information quality: for intelligence purposes, all information needs to be assessed as to its degree of correctness. It should be possible to combine data from different sources of differing reliability into a more trustworthy fusion result. Information fusion, which can arguably be defined as the processing and combination of uncertain data, can help in this: there are a very large number of fusion algorithms that fuse uncertain data. However, as in most uses of ICT technology, it is not the algorithms themselves that is the big problem, but rather the models needed in order to use them. For quality markings of information to be useful, there must be a well-defined and shared interpretation of what different qualities mean.

Information supply is not a big problem for most low-level fusion systems, since the data that the processing needs is often either available locally or (such as in distributed tracking and sensor networks) is homogeneous. Both of these simplifications make it easier to supply the fusion process with information.

For high-level information fusion, however, the data and information that is needed as input is often both heterogeneous and distributed. Since many relevant information fusion problems are computationally hard (e.g., clustering is an NP-complete problem), it is important to reduce the amount of data that needs to be processed.

Consider a corpus of documents and an intelligence query asking us to find and summarize all documents relevant for a particular topic (or a set of topics). This problem can be sub-divided into several distinct phases

- Re-formulating the query (which might have been posed to an intelligence analyst in natural language) so that it is possible to use it in the computer system. This step might also include breaking down the query into several related ones, or translating the query into another language.
- Running the query against the corpus of documents, providing a ranked list of documents that satisfy the query.
- Each of these documents might need further processing in order to be useful. A non-trivial example is translation of a document in a foreign language. Another example is to semantically tag the words and sentences in the document and perform entity extraction. This is important, for example, if we want to be able to use information extracted from the document in a social network analysis systems: in this case, we want to extract the possible relations between entities (such as persons or organizations) described in the document.
- In order to do fusion, we must also process several documents simultaneously. This could be done both before and after selection of documents. In order to fuse, we must first determine which documents that are related and which are not.

A major problem for intelligence analysis is to determine the quality (in terms of reliability and credibility) of given information. For processed or fused information that relies on several different original information objects, this is an even worse problem.

3. MOTIVATIONAL APPLICATIONS

In this section, we will describe some applications, taken from recent information fusion research at the Swedish Defence Research Agency, that motivate our focus on information supply.

In an experiment performed jointly with the Swedish Armed Forces Joint Concept Development and Experimentation Centre, we tested the ability of platoon commanders to formulate information requests with enough detail so that our semantic reasoner could find the relevant set of available information and display it to them [13]. This experiment showed the importance (and difficulty) of formulating relevant queries when selecting what information to display to users. The users were able to refine and change their queries at fixed times in the scenario, eventually converging. In the experiment, we did not perform fusion of the information retrieved by the queries, but in the future we plan to do this, as well as test different ways of automatically generating good queries using, for example, evolutionary algorithms.

Impactorium [14] is a framework for threat analysis and soft/hard information fusion, mainly applicable in OOTW (operations other than war) type scenarios. It uses the concept of “threat model” (currently Bayesian belief networks) and “indicators” to fuse observation reports and calculate the probability of future events of interests (mainly threats). An

indicator is an *observable* event that can influence our belief in the probability of a future event or state. For instance, the opponents plan to do X causes them to take action Y, which we observe and use as an indicator that they are planning X. Fig 1 below shows a threat model. In this model, there are several indicators that must be observed in order for the system to believe that the threat is about to occur.

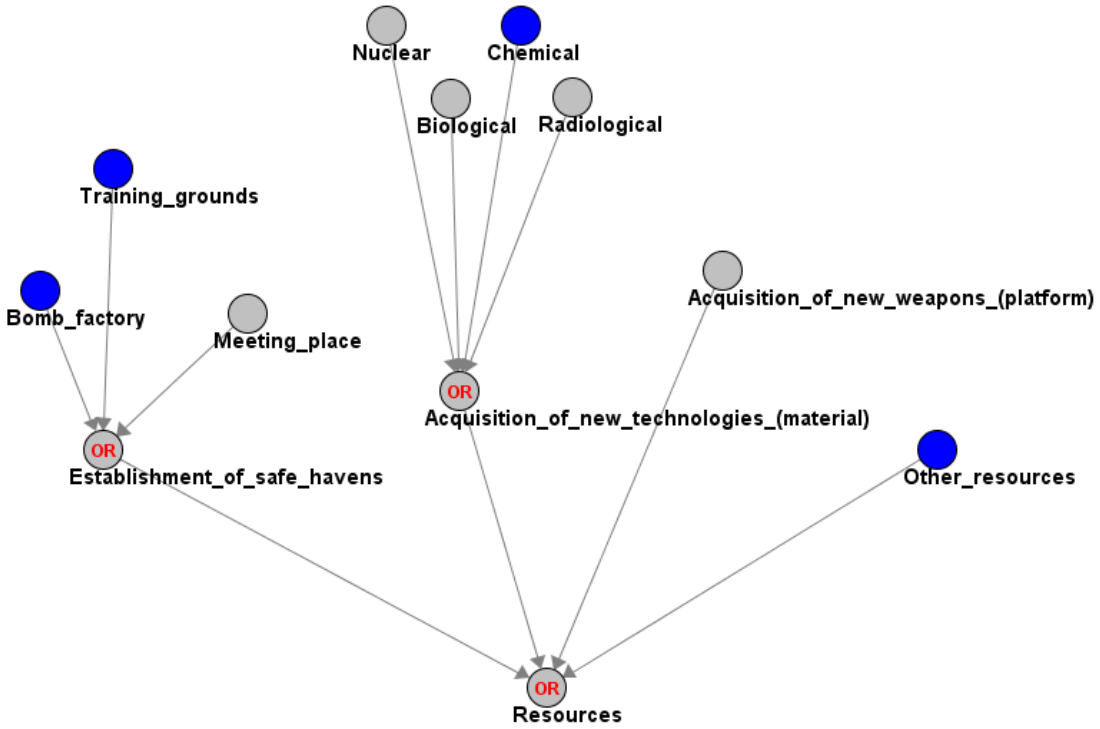

Fig 1. An example threat model from Impactorium. The indicators are represented by the top nodes, for example "Training_grounds".

In previous work, incoming reports were manually tagged with indicators. In the next version of the system, new functionality for management of indicators will be added. The values of indicators will be determined by "Analysis objects" that, among other things, connect the present value of the indicator to the set of information objects that form the basis for giving the indicator the specified value. It will also be possible (and even necessary) to update the values of indicators as time passes.

In addition to being able to create indicators based on fused reports, it should also be possible to replace an indicator in a threat model with a query, or a set of queries, along with a function that determines how the value should be calculated from the results of the query.

Social network analysis is a set of methods for calculating properties of groups based on the way that they group communicates. A basic problem when doing social network analysis is to determine what the proper network to analyze should be. When should a communication between two persons be interpreted as a link, and when is the communication just noise? There are vast amount of relational data available in, e.g., phone and email databases, and it is important to filter this data before subjecting it to network analysis. Semantic queries and graph matching are interesting possible ways of solving these problems. Our current social network analysis tool [4] uses an open source graph query system (Proximity[2]) along with some heuristical methods for selecting the interesting parts of the data.

[2] http://kdl.cs.umass.edu/software/proximity.html

4. THE INFORMATION SUPPLY PROCESS

High level fusion services by definition deals with combinations of objects. As argued in Section 2, for a human analyst making a situation assessment it is only possible to handle situations with very few objects, as the number of possible relational combinations increases exponentially. A human can try to cope with this mainly by picking out the most likely cases that adhere to her experience and focus the analysis on this subset. For this strategy to be successful it is central that the amount of information presented to the analyst is sufficiently small [10].

Computer programs generally lack the human capability of experience based abstract reasoning. On the other hand they can manage a much larger set of relational combinations by brute force. This means that automation of a certain fusion service implies that a larger volume of input information is possible, and in most cases also necessary to maintain quality of the fused results. However, we argue that the same filtering procedures that are used in manual analysis also are necessary for limiting the input to many automated fusion algorithms. This can be easily seen for the case of a fusion algorithm whose running time is exponential in the size of the input. Reducing the number of input items that need to be considered can then have a dramatic effect on the running time. The same effect is also present, albeit not so dramatically, for fusion algorithms with polynomial running time.

4.1 Information Supply model

No matter if a fusion service is manual or automatic it always has an information need. The information need can be expressed as a number of *queries* on a collection of information objects that the fusion service has access to. We call the (time-dependant) result of these queries (the answers) the *information view* of that particular fusion service. If the information need extends over time the queries are transformed to subscriptions that will catch matching new information objects as they arrive in the system. This means that even if the queries are fixed over time the information view might change.

Due to resource constraints, further described in Section 4.2, it might not be possible for the fusion service to account for all information in the view. The different queries must be prioritized and their results should be presented as *ranked lists*, see Fig 2. This makes it possible for the fusion service to process information in a greedy manner until it is out of resources, and still be sure that the result will be the best possible. The number of information objects that are chosen from each list depends on the quality of each list and the expected value for the fusion service of the set of chosen objects. We call the parameter settings that determine the shape and content of the information view the *filtering configuration*.

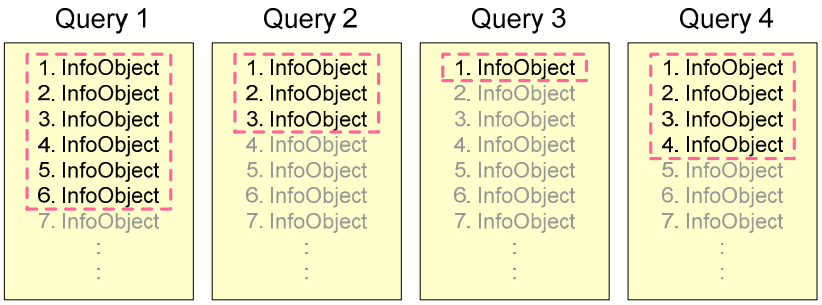

Fig 2. The figure illustrates an information view consisting of ranked result lists from four queries. The information objects inside the red dashed boxes are those that will be processed by the fusion service.

To enable the information view filtering, a collection process must first fill the system with relevant information objects to query. It can be directed to deliver different scopes at different qualities by alteration of the *collection configuration*.

In a complex fusion system there are many collecting resources of many different types. This means that collected observations often are not represented in a manner that allows immediate query access. In such cases it is necessary to add a processing step to transform the output of a specific resource to a common representation. Fig 3 describes the entire information supply process in four steps, collection, processing, filtering and fusion, with two feedback loops, scope refinement and view refinement.

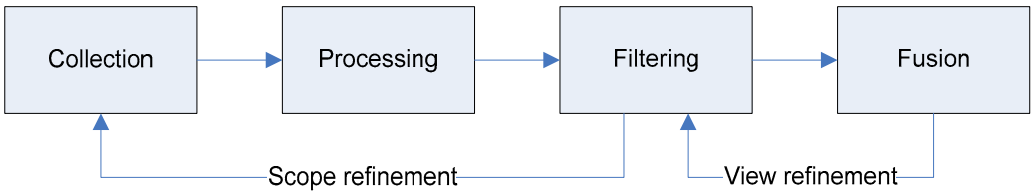

Fig 3. The Information Supply process.

The view refinement loop informs the filtering process of the latest information needs of the fusion service and returns feedback on how well previous filtering performed according to the goals of the fusion service. This might result in an update of the filtering configuration. The scope refinement loop redirects the collection process in order to stay in line with current queries. It also forwards updated quality requirements for the different parts of the scope, derived from information in the view refinement feedback. Fig 4 illustrates the relationship between scope and view.

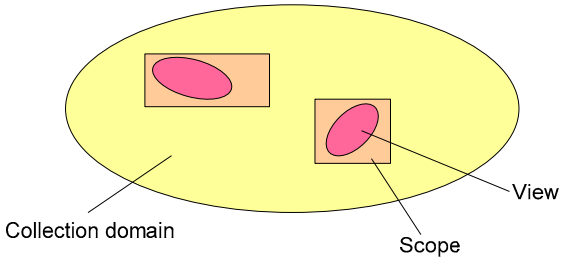

Fig 4. The collection scope and the filtering view are tightly coupled. The scope should be as small as possible but still cover the view in order to deliver a base for satisfactory query answering. Note that in the figure the view is represented in the collection domain as it would appear if it was translated backwards through the Processing step.

4.2 Information Supply constraints

Each step in the information supply process chain has its own specific set of costs:

- *Collection cost* is a function of the scope and quality requirements, and which resources collect what observations.
- *Processing cost* is a function of the number and sizes of observation objects, their types and which transformation operations that should be performed.
- *Filtering cost* is a function of total number of current information objects and the complexity of the filtering queries.
- *Fusion cost* is a function of the number of query results objects to consider and the nature of the fusion operation.

As an example, consider a case where we want to create a social network over *N* actors of interest using semantic entity and relation extraction. The collection costs could be the resources needed to collect relevant documents from the web using a standard web search engine. Just searching on the actors of interest one at a time would give a low quality output. Searching on all pairs of actors would probably increase the quality, but would certainly also increase the cost as there are *N*^2-*N* such pairs. The processing cost consists of the computational resources needed to run an automatic extraction tool on the retrieved documents, and possibly also an additional manual quality check afterwards. Filtering costs emerge as the semantic queries are executed on the collected and processed information objects. This cost could increase if the queries require semantic inference to be performed. Finally, if we want to fuse similar extracted relations to get a more certain result, the resources for this operation would add to the fusion cost.

4.3 Optimization

In this section we define the problem setting in a more formal way. We assume that we have an optimization problem that can be described by a utility function *U* over all possible situation state estimates (SSEs). Furthermore we assume knowledge of an information fusion service (IFS) that given a set of information objects (IOs) updates the situation state

estimate to SSE'. As argued above, we have two main operators that determine which information objects that are selected to be fed into the IFS. Firstly, the collection configuration *c* determines which resources are to collect IOs in which scope and at what quality. These IOs are then transformed to a common representation format that can be processed by the IFS. Secondly, the filtering configuration *f* defines a number of queries and how many results from each query that will be passed on to the IFS.

We denote the set of IOs that a specific combination of collection and filtering configuration will pass on to the IFS by $IO_{(c,f)}$. Inspired by [7], we express the optimal combined configuration as

$$(c,f)^* = \arg\max_{(c,f)} \left(U(IFS(IO_{(c,f)}, SSE)) \right),$$

with the cost constraint

$$Cost_{Collection}(c) + Cost_{Processing}(c) + Cost_{Filtering}(c,f) + Cost_{Fusion}(c,f) \leq Cost_{Total}.$$

Note that the collection and processing costs does not explicitly depend on the filtering configuration *f*. However, there is an implicit dependence through the scope refinement process.

Retrieval optimization using this formalism is very similar to normal sensor management in standard information fusion. For the problem of determining where to place ground sensor networks in order to track opponent units as good as possible, we have previously [12] applied a formalism based on random sets [5]. We believe that it would be fruitful to formalize the more general information supply problem presented here using random sets and plan to do so in the future.

5. DISCUSSION AND FUTURE WORK

In this paper we have discussed the need for advanced filtering techniques to support automated or semi-automated high-level fusion services with properly sized and relevant sets of information. We presented a model for high-level information supply with parameters to tune both collection and filtering, and an optimization problem formulation of the problem that is strongly related to previously introduced formalism for sensor management was introduced.

As mentioned above, we believe that a random set formulation of the information supply optimization problem, along with implementing random set or finite-set statistics [8] based methods for solving it would be useful. There are several other interesting research problems in this area. It would be interesting to develop a proof of concept system that implements all aspects of the suggested information supply model. We also believe that the model needs to be developed further. An interesting aspect not covered in this paper is how to deal with chains of dependent information fusion services. Such constructs introduce the risk of data incest, which has to be handled with care. Another related aspect that would be interesting to study is information supply for distributed fusion architectures.

REFERENCES

[1] G. Adomavicius, A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions", IEEE Transactions on Knowledge and Data Engineering, vol 17, pp 734-749, 2005

[2] J. B. L. Bard, S. Y. Rhee, "Ontologies in biology: design, applications and future challenges", Nature Reviews Genetics, vol 5, pp 213-22, 2004

[3] J. Brynielsson, A. Horndahl, L. Kaati, C. Mårtenson, P. Svenson, "Development of Computerized Support Tools for Intelligence Work", submitted to 14th ICCRTS 2009

[4] L. Ferrara, C. Mårtenson, P. Svenson, P. Svensson, J. Hidalgo, A. Molano, A. L. Madsen, "Integrating Data Sources and Network Analysis Tools to Support the Fight Against Organized Crime,, Springer Lecture Notes in Computer Science 5075, pp 171-182, 2008

[5] I. R. Goodman, R. P., Mahler, H. T. Nguyen, [Mathematics of Data Fusion*]*, Kluwer Academic Publishers 1997

[6] M. E. Higgins, D. L Hall, J. Llinas, Handbook of Multisensor Data Fusion: Theory and Practice, CRC Press 2008

[7] Johansson, R., "Large-Scale Information Acquisition for Data and Information Fusion," PhD Thesis, KTH, Numerical Analysis and Computer Science (2006).

[8] R. P. S. Mahler, Statistical Multisource-Multitarget Information Fusion, Artech House 2007

[9] C. D. Manning, P. Raghavan, H. Schütze, [Introduction to Information Retrieval], Cambridge University Press, 2008
[10] Miller, G. A., "The magical number seven, plus or minus two: Some limits on our capacity for processing information", Psychological Review, 63, 81-97 (1956).
[11] C. Mårtenson, A. Horndahl, "Using semantic technology in intelligence analysis", Proc. Skövde Workshop on Information Fusion Topics 2008
[12] C. Mårtenson, P. Svenson, "Evaluating sensor allocations using equivalence classes of multi-target paths", Proc Eighth International Conference on Information Fusion (FUSION 2005), Paper B9-1 5.
[13] C. Mårtenson, P. Svenson, "Information model for non-hierarchical information management" ,Proc. Ontology for the Intelligence Community (OIC 2008)
[14] P. Svenson, T. Berg, P. Hörling, M. Malm, C. Mårtenson, Using the impact matrix for predictive situational awareness, In Proceedings of the 10th International Conference on Information Fusion (FUSION 2007), 2007