An Ontology-based Adaptive Reporting Tool

Christian Mårtenson, Andreas Horndahl
Swedish Defense Research Agency (FOI)
Stockholm, Sweden
firstname.lastname(at)foi .se

Ziaul Kabir
The Royal Institute of Technology (KTH)
Stockholm, Sweden
mzkabir(at)kth.se

***Abstract*— Intelligence gathering by human observers is important for acquiring indirect and non-physical information. The drawback is that it is often delivered as free text which is not well-suited for further exploitation through automatic processing. In this paper we present a concept for structured human reporting based on an ontology-driven adaptive user-interface. The concept lays the foundation for the implementation of a possibly hand-held in-field reporting system, which can adapt to the context of the reporting situation as well as to possible information needs of other agents in the intelligence system.**

Keywords-semantic technologies; ontologies; adaptive user interfaces; context aware interaction

I. INTRODUCTION

In spite of constant technological advances, the nature of today's conflicts has increased the importance of intelligence gathering by human observers. Automatic sensing systems do a good job detecting and monitoring physical features like vehicle or human movements, but for acquiring indirect information and information referring to the cognitive domain humans are still the main asset. This kind of information is often referred to as soft data. The advantage of soft data is its high informational value; the drawback is that it is often delivered as free text, which though human friendly is less suitable for exploitation through automatic processing. Hence an important issue in managing soft data is the transformation of unstructured free text into structured content adhering to a formalized information model. Techniques for automatic structuring of text include linguistic and statistical approaches for entity and relation extraction. Such techniques are computational intense, often require a lot of training data and are never completely accurate. In a human reporting system these are limiting factors and alternative approaches are of interest.

One might argue that speaking or writing in your native tongue is the most intuitive method for delivering a human message, and that issues regarding human reporting will be solved when language processing has been cultivated to perfection or near perfection. However, the opposite approach, forcing the human reporter to directly input structured information can have other benefits:

- The language is more precise, which can prevent the user from making unintentional fuzzy statements
- The format is more compact, implying a potential for faster input
- The underlying information model is based on a shared understanding, which can prevent misunderstandings and increase interoperability on a semantic level

However, the main argument for exploring the topic of structured data input is that it has the potential to deliver completely accurate input already today. In addition, a direct correspondence between the manual input and the information model used by the input device greatly improves the conditions for accomplishing a computer based dialogue system.

In this paper we present a concept for structured human reporting based on an ontology-driven adaptable user-interface. The concept lays the foundation for the implementation of a possibly hand-held in-field reporting system, which can adapt to the context of the reporting situation as well as to possible information needs of other agents in the intelligence system. More specifically we put the following requirements on the system:

- It should be intuitive to a non-expert, who is neither an ontology engineer nor a domain expert.
- It should be domain independent, i.e. the system should work with ontologies from different domains.
- The output should be rdf-triples adhering to the ontology.
- It should be adaptable to the context of the reporting situation (who is reporting, what is the role of the reporter, where is the reporter, what time).
- It should be adaptable to the information needs of other agents in the intelligence system.

Fig. 1 gives an overview of how the system is intended to adapt to capture external information needs. The user observes an event and enters event information in the reporting system. The output of the reporting system is semantic statements. These statements are matched with information needs from other parts of the systems, which also are expressed as semantic statements. If there is a match, the information need is presented to the user as prioritized information to enter.

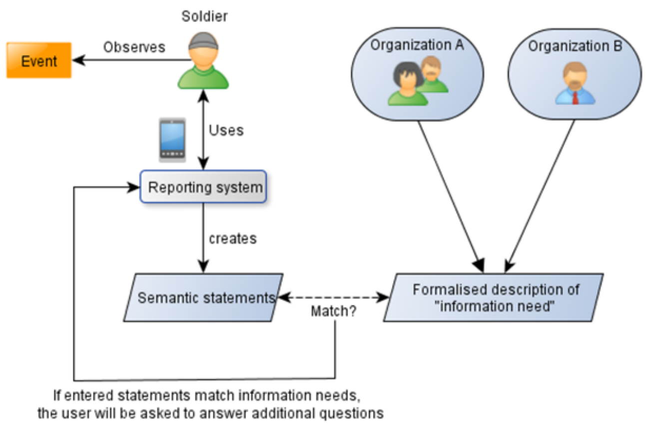

Figure 1. An overview of the process for capturing external information needs.

II. Related Work

There is not much work reported on supporting manual input of semantic data (i.e. ontology instances). Standard ontology editors, such as Protégé, allow instance creation but require advanced user knowledge both regarding the domain and ontology engineering. The Disciple-RKF system [1] supports semantic user input through "knowledge elicitation scripts", which specifies natural language queries to be shown to the user and then how to process the user's answer semantically. This gives a good input support for a non-expert user, but requires an extensive manual work for the system engineers when defining the scripts as the logic of the GUI is defined there rather than in the ontology itself.

More effort has been put into developing user friendly systems for the querying of semantic repositories, although as stated in [2] the works are mainly for ontology engineers and not meant to assist domain experts or novice users. Semantic querying share common ground with semantic data input as it includes the creation of semantic statements, which are used as templates for matching the repository content. There are at least four approaches to support users in constructing semantic queries: natural language, controlled natural language, graphical editors and forms.

- Natural language query interfaces for semantic querying is a daunting task as it involves all issues related to natural language processing plus the additional constraint that the output must comply with a specific ontology. Its usability for querying large semantic web database is discussed in [3].
- Controlled natural language (CNL) defines a restricted form of natural language (e.g. English). It is used in a number of tools [4][5][6] developed for editing and querying ontologies. The disadvantage of CNL is that although the user can write and understand queries there is still an issue with learning the specific rules and boundaries of that particular CNL.
- Graphical ontology query tools are visual query systems that provide graphical notations to pictorially express semantic queries to retrieve data from semantic repositories. A number of scientific prototypes exist [2][7][8], which all however require the users to have knowledge about ontologies.
- The final approach for semantic query construction support is to use forms. In its simplest form it is just a predefined template, like an instance template in Protégé. More advanced support can include auto-completion, filtering and model checking [9].

In this paper we have due to the limitations of the other approaches chosen to build on the ideas of "smart" forms, extending them with more advanced methods for adaptation to context and external information needs.

III. Scenario

The following scenario illustrates the usage of the suggested system:

An army patrol is visiting a village. An officer of the patrol talks to the village leader who explains that the village was visited by a group of Talibans the week before. The village leader further describes the group as consisting of approximately 100-150 people and that they were threatening the population in order to get food.

The officer uses the reporting tool to enter information about the event. After manually choosing "threatening" as the main event type the tool automatically asks for related information, e.g. generic attributes as event "date" and "location", but also attributes and relationships specific to "threatening" like who is the "perpetrator" and "victim". The tool stores the information as triples in an rdf-repository. Once there, it is matched to external *requests for information* (RFIs) which have been posted by other people in the system. In this case there happens to be an RFI from the headquarter asking for information about what kind of weapons the Talibans possess. The statements of the report that our patrol officer is entering match this RFI as they are both about Talibans. The match triggers the reporting tool to present the RFI, so that the officer can make additional queries to the village leader.

IV. Conceptual Design

A. Overview

The overall idea of the reporting system is that it should adapt the interface based on what the user is reporting and take external information needs into consideration. In the event reporting scenario described above, the system should be loaded with a suitable military reporting ontology with attributes from e.g. the JC3IEDM. As an entry point the reporter is encouraged to report some basic event information consisting of the event type, time and place and information about the source (Fig. 2).

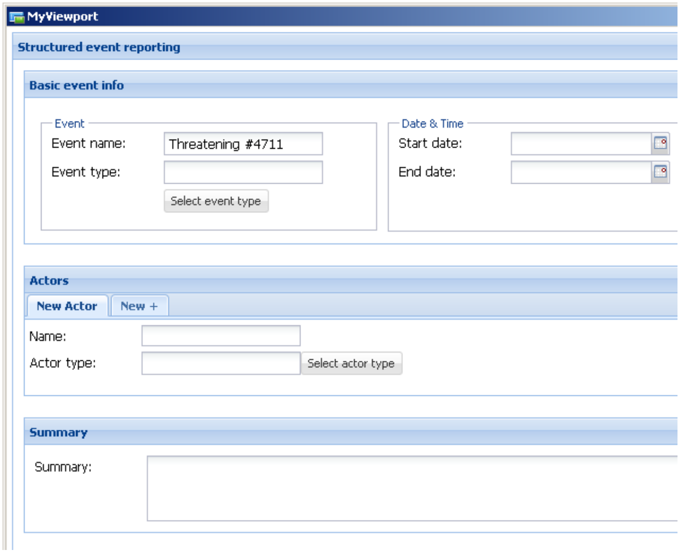

Figure 2. Initially the interface only includes fields for basic event information.

Depending on what event type is chosen, new fields will emerge for the reporter to fill in. In the case of the Taliban scenario, the reporter chooses "threatening" as event type and will then be asked about which actors that were involved, there respective roles (perpetrator or victim) and additional properties that are related in the underlying ontology ("A" in Fig. 3)

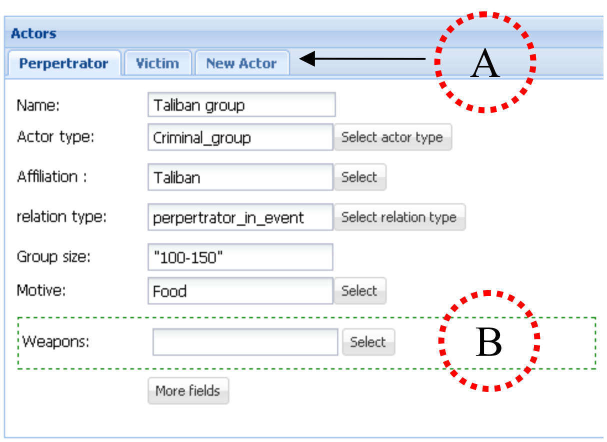

Figure 3. Depending on the user's choice of event type, related actor types emerge as new tabs (A). External information needs (B) emerge when entered information matches an RFI.

B. *Matching external information needs*

In addition to adapting the user interface by adding or removing input options based on what the user enters, the system will also match the event description with external information needs. In the Taliban scenario, an external information need had been registered in the form of an RFI, asking about the kind of weapons that the Talibans possess. The RFI is expressed as a set of semantic statements, which allows semantic matching. When the reporter enters affiliation "Taliban" for the perpetrator, this will trigger a match with the RFI. An additional field will emerge in the reporting tool asking for weapons information ("B" in Fig. 3).

A starting point is to match actors, places and event types between the event and external information need. If there is a match, the user might possess or have access to additional valuable information not reported yet. The matching process could also be done by executing a SPARQL query on the statements. If the result, with a degree of fuzziness, matches the information, the system asks the user some additional questions. A detailed description of the matching process is given in Fig. 4.

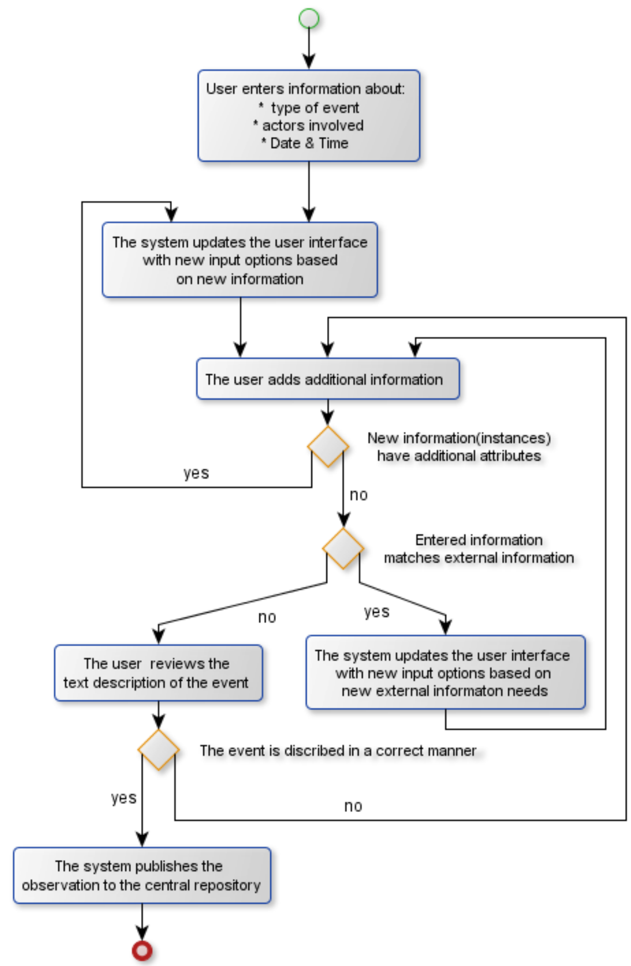

Figure 4. A detailed description of the matching process.

C. *Adaptable interface*

The ontology can be used to filter out irrelevant input fields and selection options. Besides type definitions, an ontology also defines relationship types and specifies when and how the relationships can be used. A relationship type can be restricted to only be valid from one kind of instance (domain) to another kind of instance type (range). Specifying domain and range provides means for creating a user interface with an increased level of usability since unsuitable input fields can be hidden. For instance, if the user wants to add a fact about an actor or an event, only the properties that have the corresponding domain will be accessible.

The available input fields can in our concept also be prioritized. In a time critical situation, it's important that the observer focus on what's important rather than trying to fill out all available fields. In a threat scenario, the victim's ethnicity may be a prioritized attribute to report, whereas in a crime investigating scenario, the shoe size may be a relevant attribute.

How the attributes are prioritized are scenario and context dependent. The priorities are also influenced by external RFI's. Consequently, the priorities are dynamic and the reporting system should be able to adapt to new priorities on the fly. In order to speed up the reporting, available contextual information should be used. This could mean automatically inserting information about time and place (by using GPS information).

Since we focus on using structured input fields which correspond to formally defined concepts we avoid using free text fields. By avoiding free text fields, there is a chance that the user thinks that the system didn't catch the meaning or some details. For this reason, the system will also provide a summary in natural language generated from the formal statements.

V. Discussion and Future Work

The tool presented in this paper is only a conceptual description. The next step is to do a proof of concept implementation and perform user tests. A setup for a thorough user evaluation could look like the following.

An ontology of a domain of interest is constructed together with a set of "observations" and a set of RFIs. The observations should consist of three parts:

- Part A contains the information that the test person should try to report, presented in either free text, or as an image or a combination.
- Part B contains additional information that the reporting agent has access to but don't enter unless someone asks for it. This could also be free text, an image or both.
- Part C contains the "correct" triples according to the test leader or some third party person/group. This part should not be revealed to the test person.

The RFIs should be in RDF-triples, where each RFI simulate the information need of another actor.

The test person is given the task to input the information presented in Part A of the observations. If the entered information matches the RFIs, the information from Part B can be used to answer any additional RFI related questions that the system presents to the test person. The resulting report is then compared to Part C and evaluated according to the following measures:

- the time to enter the information,
- the correctness of the resulting report,
- the completeness of the entered information, and
- the number of RFIs that were correctly answered.

Acknowledgment

This work was supported by the FOI research project "Tools for information management and analysis", which is funded by the R&D programme of the Swedish Armed Forces

References

[1] G. Tecuci, M. Boicu, D. Marcu, B. Stanescu, C. Boicu and J. Comello, "Training and Using Disciple Agents: A Case Study in the Military Center of Gravity Analysis Domain," in AI Magazine, 24,4, 2002, pp.51 - 68. AAAI Press, Menlo Park, California, 2002.

[2] A. Fadhil and V. Haarslev, "OntoVQL: A Graphical Query Language for OWL" Proceedings of the 2007 International Workshop on Description Logics (DL-2007), Brixen-Bressanone, Italy, June 2007, pp. 267-274.

[3] E. Kaufmann and A. Bernstein, "Evaluating the usability of natural language query languages and interfaces to Semantic Web knowledge bases," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 8, no. 4, pp. 377 - 393, 2010.

[4] A. Funk, V. Tablan, K. Bontcheva, H. Cunningham, B. Davis, S. Handschuh, "CLOnE: Controlled Language for Ontology Editing", in Proceedings of 6th International Semantic Web Conference (ISWC), Busan Korea, November, 2007

[5] V. Tablan, T. Polajnar, H. Cunningham, K. Bontcheva, "User-friendly ontology authoring using a controlled language", in Proceedings of LREC 2006 - 5th International Conference on Language Resources and Evaluation. ELRA/ELDA, Paris, 2006

[6] R. Schwitter, K. Kaljurand, A. Cregan,C. Dolbear, and G. Hart, "A comparison of three controlled natural languages for OWL 1.1", in Proceedings of OWL: Experiences and Directions (OWLED 2008 DC), Washington, DC (metro), 2008.

[7] P. R. Smart, A. Russell, D. Braines, Y. Kalfoglou, J. Bao, and N. R. Shadbolt, "A Visual Approach to Semantic Query Design Using a Web-Based Graphical Query Designer," in Proceedings of the 16th international conference on Knowledge Engineering: Practice and Patterns, Berlin, Heidelberg, 2008, pp. 275–291.

[8] C. Kiefer and A. Bernstein, "The creation and evaluation of iSPARQL strategies for matchmaking," in Proceedings of the 5th European semantic web conference on The semantic web: research and applications, Berlin, Heidelberg, 2008, pp. 463–477.

[9] C. Mårtenson, A. Horndahl, "Using semantic technology in intelligence analysis", in Proceedings of Skövde Workshop on Information Fusion Topics, Skövde, Sweden, 2008.