A Situation Awareness Question Generator to Determine a Crisis Situation

Teitsma, M.; Sandberg, Jacobijn; Wielinga, Bob; Schreiber, Guus

Publication date
2015

Document Version
Final published version

Link to publication

Citation for published version (APA):
Teitsma, M., Sandberg, J., Wielinga, B., & Schreiber, G. (2015). *A Situation Awareness Question Generator to Determine a Crisis Situation*. 129-133. Paper presented at 5th International Workshop on Information Systems for Situation Awareness and Situation Management , Valencia, Spain. https://zh.booksc.eu/book/51982591/b13b7d

2015 26th International Workshop on Database and Expert Systems Applications

A Situation Awareness Question Generator to determine a crisis situation

Marten Teitsma
Amsterdam University of Applied Sciences
Email: m.teitsma@hva.nl

Jacobijn Sandberg
University of Amsterdam

Bob Wielinga
and Guus Schreiber
VU University Amsterdam

***Abstract*—In this paper we present a system that generates questions from an ontology to determine a crisis situation by ordinary people using their mobile phone: the Situation Awareness Question Generator. To generate questions from an ontology we propose a formalization based on Situation Theory and several strategies to determine a situation as quickly as possible. A suitable ontology should comply with human categorization to enhance trustworthiness. We created three ontologies, i.e. a pragmatic-based ontology, an expert-based ontology and a basic-level ontology. Several experiments, published elsewhere, showed that the basic-level ontology is most suitable.**

I. Introduction

To mitigate the effects of a crisis it is important to gather information as fast as possible. Ordinary people involved in a crisis often have been neglected by emergency services as a source of information. Nowadays they are more and more regarded as the true 'first responders' [6]. In this paper a framework is proposed, to gather trustworthy information about a crisis situation from ordinary people. This framework has been implemented as an application: the Situation Awareness Question Generator (SAQG). SAQG automatically generates questions from an ontology and uses the answers to determine a situation. Because in this framework number, form and content of questions is determined by the specific ontology in use, several experiments have been conducted to characterize the ontology which is most suitable for this task.

To formalize situations we used Situation Theory which has a richer structure than related formalisms and thus is much more fine-grained [12]. Section II gives an overview of Situation Theory. To map Situation Theory to OWL we used Situation Theory Ontology [5] and revised it, creating Situation Theory Ontology Revised (STOR) as shown in Section III. Several strategies to generate question are presented in Section IV. The architecture of SAQG is shown in Section V and how we characterize a suitable ontology in Section VI. We end this paper with a discussion and conclusion in respectively Section VII and Section VIII[1].

II. Situation Theory

A situation is a limited part of the world in which various, abstract or physical, entities stand in relation to each other. Situations are ubiquitous in our world. We are always in some situation or other. Despite the vagueness surrounding situations, humans are good at recognizing a specific situation when needed. We know what is important and what not, have knowledge about consequences of specific facts and handle such fluidity and vagueness well. When something complex such as a flooding happens, it becomes the situation which urges us to take proper action without knowing all details. Being aware of such a situation means, according to Endsley [2], one is aware of specific elements in an environment, has a grasp of its meaning and how this constellation will evolve.

Within Situation Theory different types of situations are being distinguished because of their role in the description of the world. Communicating about a situation is done within a specific situation, i.e. the utterance situation. The situation which has the attention and is talked about is called the focal situation. This is the situation which an utterance is about. Often a reference to another situation is made when doing an utterance. When referring to 'a man seen before' one is referring to a resource situation, i.e. the situation in which one has seen that man.

In Situation Theory the smallest entity of information is called an infon which is formally described as a tuple of the form:

$$<< R, a_1, ..., a_n, 0/1 >>$$

where R is a n-place relation, $a_1, ..., a_n$ represent objects appropriate for R and $0/1$ refers to whether or not the infon represents a fact in the real world.

To describe a real situation we most often also refer to a location where and a time when something took place:

$$Flood_{Goeree\ Overflakkee\ 1953} \models$$
$$<< flooded, street, Oude\ Tonge,$$
$$February\ 1st\ 1953, 1 >>$$

where $\models$ should be read as 'supports' instead of the traditional 'makes true'. Situations are only a part of the world and in that part of the world this infon is true. This infon refers to a street which is flooded in a place called Oude Tonge at February 1st, 1953 and is supported by the situation describing the flooding of a part of the Netherlands (Goeree Overflakkee) in 1953.

When infons have parameters, they are called parametric infons. A parametric infon is not referring to an actual situation but to possible situations, i.e. it is not clear which specific referent is meant when a parameter is used in an infon. Infons can have parameters of a given type. Parameters can be of the basic types:

- TIM: the type of a temporal location. Refers to a specific time or time frame. For example, 2.13 PM.

[1] This paper is an extract of a part of a PhD-thesis by Marten Teitsma

1529-4188/15 $31.00 © 2015 IEEE
DOI 10.1109/DEXA.2015.42

- LOC: the type of a spatial location. Refers to a place such as a city, region or something else which has a location. For example, Utrecht in the Netherlands.
- IND: the type of an individual. Refers to an object which is individuated by someone. For example, the laptop computer I am writing on, also known as 'Boniface'.
- REL_n: the type of an n-place relation. For example, observing, which is a two place relation: somebody observes something.
- SIT: the type of a situation. For example a situation such as $Flood_{Goeree\ Overflakkee\ 1953}$. The type of situations referred to, are already identified.
- INF: the type of an infon. Refers to (sub-)types which can be distinguished such as elementary infons, 'parametric infons' (infons with a parameter) and 'compound infons' (a set of infons related by conjunction and disjunction operators).
- TYP: the type of a type. Every type T is a subtype of TYP: $<< of-type, T, TYP, 1 >>$.
- PAR: the type of a parameter.
- POL: the type of a polarity (0 and 1).

Information in Situation Theory, is captured by the confirmation or denial of the relation between infons and objects or situations. When, for example, someone is determining an object on fire as a ship on fire, this is information gathered by our system. When this ship, after further questioning, is determined as a cruise ship, this is also information deemed valuable for determining the situation.

To generate information from relation or causality we need constraints. Constraints are relations between types of situations which represent (natural) laws, conventions and other kinds of regularities. When there is the fact of smoke somewhere, it is because of the constraint 'fire produces smoke' that we have a clue there is fire. In Situation Theory different types of constraints are distinguished. Nomic constraints are of the kind which correspond to some natural law, e.g. 'fire produces smoke'. Necessary constraints are reflexive about a situation and tell more about the situation, e.g. 'kissing means touching'. Conventional constraints refer to social laws or rules, e.g. 'the ringing bell means class is over'.

III. Situation Theory Ontology Revised

Situation Theory is used by Kokar [5] to create an ontology which goes under the name of Situation Theory Ontology (STO). According to Kokar et. al. the two basic elements of Situation Theory are objects and types. The construction of STO is then based on the idea that an ontology of Situation Theory should have two meta-levels representing objects, i.e. things in the world and types which are abstractions. Furthermore, they interpret a class as a set of instances associated with this class. The first meta-level is representing objects which all are subordinate to the class *Object*. An example of such a subordinate concept is *Individual*. The second meta-level is representing types such as *IND* and *RELn* as subordinate concepts of the concept *TYP*. Instances of this class are classes themselves, e.g. an instance of *IND* is the class *Individual*. In STO each kind of object has two associated classes: a class which is a set of instances of the given class and a class which is an instance of a subtype of TYP.

To develop Situation Theory Ontology Revised (STOR) we distinguish several types of ontologies: domain, generic and representation ontologies [13]. The domain ontology expresses conceptualizations specific for a particular domain. A generic ontology contains concepts which are considered to be generic across several domains. A representation ontology consists of concepts which are the primitives for the formalization of the concepts as described in the generic and domain ontologies.

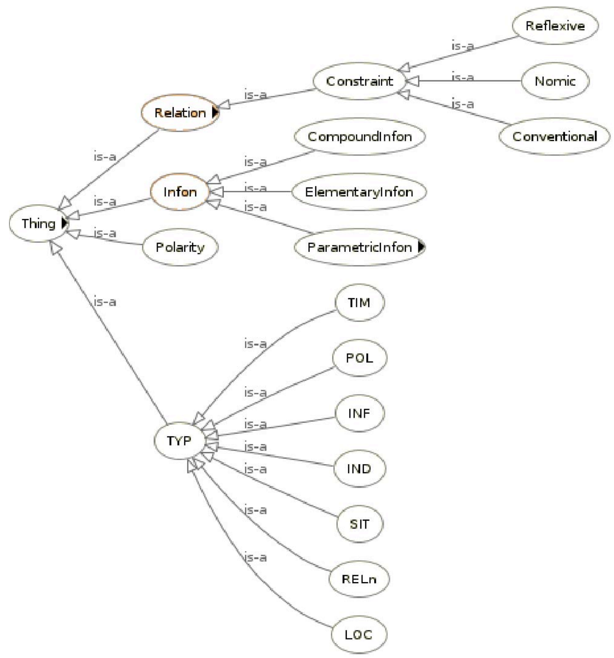

Fig. 1. The representational part of STOR.

The basic types (see Section II) should be part of the representation ontology. All the basic types are subconcepts of the type *TYP*. In STO the concept *Attribute* primarily is a superclass of the concepts *Location* and *Time* which are used to represent respectively instances of location and time. But it is also used as a superclass of other attributes such as *velocity*. Attributes are, in our view, compound infons or concepts of the type *IND*. For example, *velocity* is a compound infon combining two infons referring both to the same instance of *IND* but with a distinct pair of instances of *LOC* and *TIM*. Thus *ATTR* is rejected as type in STOR and *LOC* and *TIM* are basic types (as in Situation Theory). The same can be said of the types *DIM* and *VAL*. In STO *ATTR* is also used to add properties to a *Situation* while in STOR this is done by the property *hasAttributingInfon*.

Relation and *Infon* are concepts fundamental to Situation Theory and are part of the representation ontology because these concepts have a specific formal representation. The concept *Infon* is a superclass of *ElementaryInfon*, *CompoundInfon* and *ParametricInfon*. The concept *ParametricInfon* is used for the generation of questions: we ask for the actual anchoring of a parameter, i.e. each parameter gives rise to a question. Instead of *Rule* as in STO, we use the concept *Constraint*, as subclass of *Relation*, with its subconcepts *Nomic*, *Conventional* and *Reflexive*. In STOR the constraints are represented using the type *RELn*, i.e. relation, to define a relation between situations.

Situation is superordinate to domain-specific situations and as such part of the generic ontology. The representational part of the ontology is shown in Fig. 1

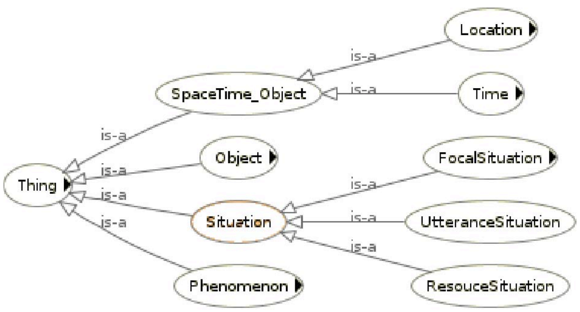

Fig. 2. The generic part of STOR.

The generic ontology consists of concepts which are valid for all the domains. All of these concepts have subclasses which are specific for a domain and part of the domain ontology. The generic ontology consists of the concept *Situation*, *Object*, *Phenomenon*, *SpaceTime_Object* (with subclasses *Location* and *Time* which represent the same concepts as in Situation Theory). The concept *Individual* and its subclass *RealIndividual* is omitted from STO and in STOR represented as instances of a concept which is more in line with representation as designed with OWL.

Fig. 3. An example of the domain part of STOR.

We distinguish in STOR three types of *Situation*: *FocalSituation*, *ResouceSituation* and *UtteranceSituation*, as is done in Situation Theory and STO. The *UtteranceSituation* is interpreted differently in STOR than in STO and in conformance with Situation Theory. In STO an *UtteranceSituation* is used to denote the situation of the user of a situation awareness system which incorporates STO. An utterance in such a system is the utterance of a user who wants to generate information and, as such, queries the system. Such a user may be interested in a specific location and retrieves all the infons related to that location from the system. In STOR an *UtteranceSituation* is the *Situation* in which information is provided. This difference gives us the opportunity to annotate gathered information with additional data. When, for example, someone reports about a *FocalSituation* from a long distance, such information should be used differently than a report from someone who is nearby the same *FocalSituation*. The difference can be found by determining the *UtteranceSituation* which is supported by infons showing the location. Because we are working in the domain of crisis management further information can also be of interest such as the physical condition of the speaker and his relation with the *FocalSituation*. The information provider is represented in the system as a *Person*, i.e. a subtype of *Object*, within a specific *UtteranceSituation*.

The difference between *Object* and *Phenomenon* is that the subclasses of *Object* are simple concepts such as *Ship*, *Building* or *RoadVehicle* while the subclasses of *Phenomenon* are complex concepts such as *Sight* or *Weathertype*. These complex concepts are often more difficult to determine than subclasses of *Object* because they are more vague or even subjective. Subclasses of the concepts *Object* and *Phenomenon* are part of the domain ontology. *Location* and *Time* are part of the generic ontology because in every domain specifications of locations and categorization of time occur. The generic part of STOR is shown in Fig. 2.

Concepts which are part of the domain ontology are specific for a particular domain. A knowledge representation of crisis management consists of *CrisisSituation* and concepts representing objects that are part of the crisis. In Fig 3 several examples of *CrisisSituation* are shown such as *CarAccident* and *Fire*. A representation of a specific domain can be more fine grained when appropriate. Subclasses of *Object* are a representation of individuals being part of a *Situation*. Examples of these representations are *Streetobject*, *Building* or *Person*. Instances of these concepts refer to real objects or people which somehow have a relation with the *Situation* we want to determine.

In a domain ontology several categorizations for *Location* and *Time* can be appropriate. For example, when we want the location of a mobile phone user the GPS-coordinates are appropriate and when we ask someone to tell us where he is concepts such as *near* and subclasses of *Landmark* should be part of the ontology. Subclasses of *Time* are also domain-specific and depending on what kind of categorization of time is appropriate.

IV. Strategies to Generate Questions

The use of ontologies for automatically generating questions about a crisis situation which are asked to a large number of people requires that the number of questions is minimal, the questions as informative as possible and easy to answer. To shorten the time for determination of the situation a selection and ordering of the most informative questions has to be made. For the selection and ordering we use the informational and truth value of possible answers, i.e. the amount of information of an answer and whether these answers can be true or not, to compute the best question at a given moment. The best series of questions is the sequence of questions which renders the most information with the least number of questions about a particular situation.

To find the best series of questions we used Floridi's Theory of Strongly Semantic Information which is based on truth values and thereby precludes the Bar-Hillel-Carnap semantic paradox in which a contradiction has more information content than a true statement [3]. To compute the informativeness of statements Floridi takes two factors into account: *a*) the truth value of the statement and *b*) the degree of discrepancy between the statement and the actual situation. With these two factors Floridi distinguishes between falsehood and abstraction in various degrees with respect to a particular situation. In our strategies we use the distance between an abstract situation and the actual situation. The degree of abstraction, i.e. degree of accuracy, gives a value to the informational entities representing the amount of information of such an entity. The most efficient question is that question which, when answered, reduces the discrepancy with the truthful description of the situation the most. An answer to such a question should always generate information whether it is a positive or negative answer when asking a polar question or a specification when asking a multiple choice question. By asking for infons supported by a situation we determine situations using several strategies based on properties of the ontologies describing the domain.

We developed several strategies because of differences in the description of situations as represented in ontologies in the domain. Situations are described using multiple infons which may have relations with each other or not. When there are relations, the properties of these relations influence the choice of the specific strategy we can use.

The first and most simple strategy is a strategy which asks after each possible situation. This strategy is used when no assumption about the description of the possible situations is made. Such is the case when the possible situations are described with different infons, i.e. the infons which are supported by each situation do not relate to each other in any way. With this strategy a representation of information gain per question based on information value is low: each time a question is asked, it will take less time to get to the right description of the actual situation.

The second strategy assumes that all situations are described with the same set of independent parametric infons. An infon is independent from another infon when there is no relation between these infons, i.e. the accuracy of one infon has no influence on the accuracy of another infon. For example, the color of an object is independent of the shape of an object. Using this strategy, the information gain per question will depend on the number of possible answers, i.e. the number of referents of the parameter which are mutual exclusive. The second strategy is used when a situation consists of infons which have multiple possible answers. The possible answers are elements of the set of subordinate concepts of the attribute which represents the parameter in the parametric infon.

A third strategy is to ask for the infons which are supported by a situation, just as the second strategy, but now the number of possible answers is restricted to two possible answers: 'yes' or 'no'. The constellation necessary for this strategy consists of situations which are described with infons of which the polarity is unknown. The information gain is continuous and the number of questions to ask and to determine the situation is equivalent to the number of infons used to describe the situation.

The fourth strategy we call 'semantic strengthening' and is analogue to the third strategy but is further restricted by the assumption that in practice certain situations are prohibited. This strategy is used when some theoretically possible but in practice impossible situations are involved because the infons are not independent from each other. The information gain for each question differs depending on whether an answer implicates other infons. When an answer implicates another infon the information gain is larger than when an answer does not implicate another infon.

The fifth strategy searches for the most detailed description available in the ontology. In this constellation an ontology represents knowledge of a specific domain in a hierarchy of concepts which are related to each other by 'is-a relations'. Superordinate concepts have subordinate concepts which specify characteristics of the superordinate concepts. All the questions are multiple-choice questions, i.e. the question asks for a further specification and the possible answers are presented to choose from. The specific informational gain with each question depends on the specific number of subordinate concepts of each concept. In a more or less balanced ontology the relative informational gain at the start of the series of questions is the greatest because then the largest number of otherwise possible situations is disregarded. This relative gain diminishes with each question.

V. ARCHITECTURE OF SAQG

Our system consists of two sides: a server and a client residing on a mobile phone. The server sends a domain ontology to a client, matching it to STOR and uses the ontology to produce questions. The answers to these questions are preserved until a situation is determined. Then the answers are send to the central server and possibly used for further computation. In the model we propose such a server has a repository of several domain ontologies.

The reading, modeling and manipulation of the ontology we use to generate questions is done with Jena. Jena offers a comprehensive API to create functions handling the information stored in such an ontology [1], [4]. Before an algorithm to generate questions is used, first the ontology is retrieved and transformed into a Triple DataBase (TDB) for high performance. The TDB can be accessed in the same manner as the access to an ontology represented by OWL is done, using exactly the same queries, but the retrieval of data is faster.

VI. SUITABLE ONTOLOGIES

To uncover the characteristics of an ontology suitable for automatic question generation we constructed three different ontologies [10]. A pragmatic-based ontology was constructed from a vocabulary which the Amsterdam fire department uses to categorize calls for 112. An expert-based ontology was constructed from two ontologies developed by knowledge-engineers and experts in the field of art. A basic-level ontology was constructed from concepts and their attributes retrieved from ordinary people. Using the attributes we created an ontology with an algorithm based on the basic-level theory of Rosch [7].

We set up a framework to measure the suitability of these three ontologies which evaluated the ontologies on four aspects: *a*) the ontology must have a structure which is useful for automatic question generation, *b*) the construction of the ontology is efficient, *c*) the ontology must be complete and *d*) the ontology should be compliant with human categorization. For this framework we used several existing metrics such as maximum path length, number of concepts, entropy and the Ingve-Miller number. Also, we developed a new metric called 'semantic distance validation'. This metric compared the distance of concepts in terms of path length with how participants in an experiment evaluated this distance.

From these metrics we learnt that the expert-based ontology had a structure least suitable for automatic question generation because it has a large number of concepts, the longest path length, high maximum number of subclasses and a high entropy. The least efficient to construct was the basic-level ontology because of the laborious retrieval of concepts and attributes. The re-engineering of the vocabulary to create the pragmatic-based ontology did cost more time than applying an algorithm as was done to construct the expert-based ontology. The expert-based ontology is the most complete ontology and the pragmatic-based ontology is the least complete ontology. With respect to the compliance with human categorization the basic-level ontology scored best.

From these results we concluded that the most information will be gathered using the expert-based ontology when it is used by experts but the number of questions will be larger than when using other ontologies. The basic-level ontology is most compliant with human categorization but costs a lot of time to construct. The pragmatic-ontology generates the least number of questions but the amount of information is the smallest of the three ontology. These conclusions were confirmed by experiments presenting scenarios and showing videos to participants who subsequently answered questions on paper [8] or their mobile phone [9], [11].

VII. Discussion

During the research we encountered several subjects which gave rise to discussion such as what criteria for alignment with human categorization are available, whether other sources can be used to create a suitable ontology, how the information value or entropy of an ontology should be measured and how much information is gathered by microblogs, e.g. Twitter.

With respect to information gathering using microblogs, which is researched intensively, we suspect that the amount of information gathered is less than using SAQG. For this we have two reasons. Firstly, while building a gold standard we asked participants of that particular experiment to describe videos, we noticed that the amount of information elicited was small. Secondly, while microblogs address active knowledge of a situation, we adress passive knowledge which is per definition larger.

VIII. Conclusion

To determine a crisis situation we developed the Situation Awareness Question Generator that automatically generates questions on a mobile phone which are asked to users of the system. To create the Situation Awareness Question Generator we used a fine-grained formal theory about situations and improved an existing ontology based on this theory, by making it more compliant with knowledge engineering principles and the formal theory. Several strategies, based on a theory of information, were developed to detect the most informative questions. Differentiating between how domain-specific knowledge is captured in an ontology, we detected variation in accuracy of information and time needed to gather information. Suitability of an ontology depends on characteristics of the users of this system.

References

[1] Jeremy J. Carroll, Ian Dickinson, Chris Dollin, Dave Reynolds, Andy Seaborne, and Kevin Wilkinson. Jena: implementing the semantic web recommendations. In *WWW Alt. '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 74–83. ACM, 2004.

[2] M.R. Endsley. Theoretical Underpinnings of Situation Awareness: A Critical Review. In M.R. Endsley and D.J. Garland, editors, *Situation Awareness Analysis and Measurement*, pages 3–33. CRC Press, 2000.

[3] L. Floridi. *The philosophy of information*. Oxford Univ Pr, 2011.

[4] Michael Grobe. RDF, Jena, SparQL and the 'Semantic Web'. In *SIGUCCS '09: Proceedings of the 37th annual ACM SIGUCCS fall conference*, pages 131–138, New York, NY, USA, 2009. ACM.

[5] M.M. Kokar, C.J. Matheus, and K. Baclawski. Ontology-based situation awareness. *Information Fusion*, 10(1):83–98, 2009.

[6] L. Palen, K.M. Anderson, G. Mark, J. Martin, D. Sicker, M. Palmer, and D. Grunwald. A vision for technology-mediated support for public participation & assistance in mass emergencies & disasters. In *Proceedings of the 2010 ACM-BCS Visions of Computer Science Conference*, page 8. British Computer Society, 2010.

[7] E. Rosch, C.B. Mervis, W.D. Gray, D.M. Johnson, and P. Boyes-Braem. Basic objects in natural categories. *Cognitive psychology*, 8(3):382–439, 1976.

[8] M. Teitsma, J. Sandberg, M. Maris, and B. Wielinga. Using an Ontology to Automatically Generate Questions for the Determination of Situations. In *Database and Expert Systems Applications*, pages 456–463. Springer, 2011.

[9] M. Teitsma, J.A.C. Sandberg, M. Maris, and B.J. Wielinga. Automatic question generation to determine roles during a crisis. In *SOTICS 2011, The First International Conference on Social Eco-Informatics*, pages 37–42, 2011.

[10] Marten Teitsma, Jacobijn Sandberg, Guus Schreiber, Bob Wielinga, and Willem Robert van Hage. Engineering ontologies for question answering. *Applied Ontology*, 2014.

[11] Marten Teitsma, Jacobijn Sandberg, Bob Wielinga, and Guus Schreiber. Validating Ontologies for Question Generation. In *Proceedings of the 26th Benelux Conference on Artificial Intelligence*, pages 113–120, November 6-7 2014.

[12] Johan van Benthem and Maricarmen Martinez. The stories of logic and information. In Pieter Adriaans and Johan van Benthem, editors, *Handbook of the Philosophy of Information*, Handbook of the Philosophy of Information, pages 217–280. Elsevier, 2008.

[13] G.A.C.M. van Heijst. *The role of ontologies in knowledge engineering*. PhD thesis, University of Amsterdam, 1995.