# Determining Information Usefulness in the Semantic Web: A Distributed Cognition Approach

Santtu Toivonen[1], Tapio Pitkäranta[1], and Oriana Riva[2]

[1] VTT Information Technology
P.O.Box 1203, FIN-02044 VTT, FINLAND
{santtu.toivonen,tapio.pitkaranta}@vtt.fi
[2] Helsinki Institute for Information Technology
P.O.Box 9800, FIN-02015 HUT, FINLAND
oriana.riva@hiit.fi

**Abstract.** Determining the usefulness of domain-specific information in the Semantic Web is a critical operational precondition that must be addressed in order to realize the Semantic Web's potential. We approach the problem through the notion of distributed cognition, which emphasizes the inclusion of external elements in agents' thinking processes. We concentrate on multi-agent scenarios of distributing cognition, meaning that a single externalized piece of distributed cognition can be internalized and utilized by multiple agents. We decompose the problem of determining information usefulness into the problems of understanding the information and subsequently determining its relevance.

## 1 Introduction: Distributing Cognition with Semantic Notes

Since the Semantic Web is an extension of the current Web with information in a machine-accessible form, it is an environment for both human and software agents [2, 1]. We consider the Semantic Web as a platform for both kinds of agents to distribute their cognition by externalizing and internalizing domain-specific pieces of information we call *Semantic Notes*. In this paper we consider the internalization part of distributing cognition. In particular, we concentrate on how an agent can determine whether some piece of information is useful enough to be internalized or not.

The theory of distributed cognition emphasizes the involvement of external elements in cognitive processes. The research subjects of distributed cognition have traditionally been humans, and the external elements taking part in cognitive processes have been any entities that are outside the human brain. Examples are books, calculators, rulers, maps, other humans, and so on [4]. Releasing the cognitive load has traditionally been identified as the main reason for distributing cognition.

The work reported in this paper extends the scope of distributed cognition research, since software agents in addition to human beings are seen as creatures distributing their cognition. This differs from the traditional conception of computer involvement in distributing cognition, where human has always been the "center" of cognitive processes, and computer programs have only assisted (see for example [3]). This brings about the key difference between distributing cognition in its traditional sense and distributing cognition in the Semantic Web, which is the media through which the cognition is

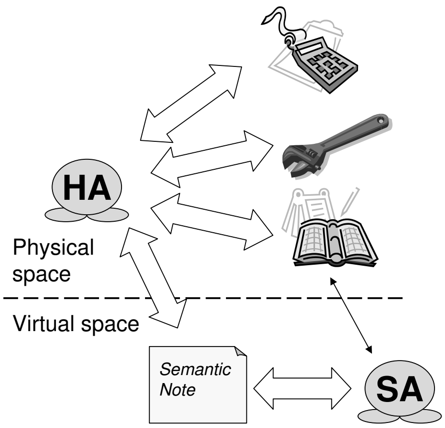

**Fig. 1.** Means of distributing cognition

distributed. In the physical world, anything conceivable to a thinking creature can be used for distributing cognition. In the Semantic Web, instead, the distribution media are more restricted, as Figure 1 depicts. Human agents (HA) can distribute their cognition to calculators, notebooks, tools, and so on, but software agents (SA) only to media accessible from the virtual space they reside in.

In principle also software agents could use physical structures for distributing cognition, for example by printing on paper, as depicted by the narrow arrow in Figure 1, but a more typical scenario is that software agents distribute their cognition in a digital form. We use the term *Semantic Note* to refer to these kinds of entities. A Semantic Note stores and transmits some meaningful piece of information, such as a definition of some complex concept or instructions for completing a procedure. The domain of information stored in Semantic Notes is unrestricted, meaning that a Semantic Note can contain a definition of a complex concept from any area. That is why Semantic Notes are defined functionally as being representations of one or more entities potentially of use in carrying out a domain-specific task. In the following sections we limit the definitions to cover only the Semantic Note, since it is the atomary unit of distributing cognition in the Semantic Web, and hence enough for our purposes. However, the definitions could be applied to other information content, too.

## 2 Determining the Usefulness of a Semantic Note

A Semantic Note can be decomposed into its constituents, namely statements. Statements are opinions about states-of-affairs, such as *The web site 'http://www.vtt.fi/tte/proj/dynamos' is created by Santtu Toivonen*. The terms in a statement can be organized in the subject-predicate-object model of RDF, and conform to concepts in an ontology. This kind of machine-accessibility is especially important for software agents. Using RDF, the above statement could be defined as follows:

```
<rdf:Description rdf:about="http://www.vtt.fi/tte/proj/dynamos/">
```

```
  <dc:creator>Santtu Toivonen</dc:creator>
</rdf:Description>
```

Of the above RDF excerpt's terms, only the predicate (`dc:creator`) explicitly refers to an ontology, namely that of the Dublin Core metadata elements [5]. Combining the notion of statements and the approach adopted in [6], an agent can be said to understand a statement found in a Semantic Note as follows:

**Definition 1.** *An agent (a) understands a statement (s), iff all the terms (t) constituting it conform to concepts ($\phi$) found in an ontology (o), which is accessible to a:*

$$\textit{understands}(a,s) \leftrightarrow \forall t : (t \in s \rightarrow \exists \phi : (\textit{conforms}(t,\phi) \wedge \phi \in o \wedge \textit{access}(a,o))).$$

We assume that one statement is either understood or not understood by an agent. In principle a more specific definition could be given based on the understanding of the terms constituting the statement. However, for our purposes a statement is on a more appropriate level of granularity. By applying a function $und$ we assign the statements values, denoted by $s_u$, as follows:

$$und(s) = s_u = \begin{cases} 1 & \text{if all terms } (t \in s) \text{ are understood} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

$n_u$ represents the agent's level of understanding of the Semantic Note ($n$). Let $S_n$ be the set of statements included in $n$ so that $s_1, s_2, ..., s_k \in n$, where $k = |S_n|$. $n_u$ receives values between 0 and 1 based on the number of understood statements ($s_{u1}, s_{u2}, ..., s_{uk} \in n$) divided by the number of all statements in the Semantic Note ($|S_n|$) as follows:

$$\begin{aligned} 0 \leq & n_u = \frac{1}{|S_n|} * \sum_{i=1}^{|S_n|} s_{ui} \leq 1 && S_n \neq \emptyset \\ & n_u = 0 && S_n = \emptyset \end{aligned} \tag{2}$$

Following [6], we assume that for an agent to understand a Semantic Note that another agent has created or modified, the statements in it conform to an ontology known by both agents. Based on that, we give the following definition for agents to share knowledge via Semantic Notes:

**Definition 2.** *A necessary condition for an agent $a_1$ to share knowledge via a Semantic Note (n) with agent $a_2$ is that n conforms to a set of ontologies (O), which is a disjunction of the ontologies accessible to $a_1$ ($O_1$) and $a_2$ ($O_2$):*

$$\textit{shares}(a_1,a_2,n) \rightarrow (\textit{understands}(a_1,n) \wedge \textit{understands}(a_2,n)).$$

This entails that the set of ontologies ($O_{1,2}$) has to be accessible to both $a_1$ and $a_2$. Notes can also be partially shared between agents. Consider a simple case with two agents ($a_1$ and $a_2$) and two partly overlapping ontologies ($o_1$ and $o_2$) so that $access(a_1, o_1)$ and $access(a_2, o_2)$. Suppose that $a_1$ has created a Semantic Note ($n$) which contains two statements ($s_i$ and $s_{ii}$). All the terms ($t_a$, $t_b$, and $t_c$) of $s_i$ conform to respective concepts $(\phi_a, \phi_b, \phi_c) \in (o_1 \cap o_2)$, and can therefore be shared between $a_1$ and $a_2$. $s_{ii}$, in contrast, has the terms $t_a$, $t_b$, and $t_d$, of which $t_d$ conforms to a concept $\phi_d \notin o_2$. Because of this, $s_{ii}$ is not shared between the agents. Based on the number of mutually understood statements, we can therefore conclude that 50% of $n$ is shared.

We define a new variable $n_{rel}$ for indicating the level of relevance the information carried by a Semantic Note has. A rule-based approach is adopted for determining the information relevance. The information content, of which the relevance is to be determined, is connected with user context via general preference rules specified by the user. The user context describes some essential details about the user's current situation, for example her location and current activity. Both the information content (i.e., the Semantic Notes) and the user context are realized as sets of statements.

**Definition 3.** *If there exists a term ($t_{ctx}$) in a statement found in the user context, as well as a term ($t_n$) in a statement found in the Semantic Note so that both of those conform to respective concepts ($\phi_{ctx,n}$) which are navigable from the concepts ($\phi_{r1,r2}$) found in the rule ($r$), the rule is said to be applicable ($r_a$):*

$$\exists t_{ctx} : \mathit{conforms}(t_{ctx}, \phi_{ctx}) \wedge \exists t_n : \\ \mathit{conforms}(t_n, \phi_n) \wedge \mathit{navigable}(\phi_{r1}, \phi_{ctx}) \wedge \mathit{navigable}(\phi_{r2}, \phi_n) \rightarrow r_a$$

where *navigable(x,y)* means that there exists a network of concepts and relationships, realized as one ontology or several connected ontologies, that enables navigating between $x$ and $y$. A positive match indicates that an applicable rule is found, as well as suitable values to satisfy it. Negative match means that there exists an applicable rule, but that the statements plugged in it do not have suitable values. In order to assign relevance values for the Semantic Notes utilizing the applicable rules, we define the following abstract function:

$$app(r_a) = r_m = \begin{cases} 1 & \text{positive match} \\ 0 & \text{negative match} \end{cases} \tag{3}$$

The function $app$ is realized as various concrete rules, that determine the relevance assignment ($r_m$, where $m$ comes from "match"). The applicable rules ($r_a$) as well as the match value ($r_m$) are utilized in the relevance equation for Semantic Notes. Let $R_a$ be the set of applicable rules so that $r_{a1}, r_{a2}, ..., r_{ak}$, where $k = |R_a|$. The Semantic Note relevance ($n_{rel}$) can receive values between 0 and 1 as the ratio between the sum of the match values ($r_{m1}, r_{m2}, ..., r_{mk}$) and the number of applicable rules ($|R_a|$):

$$\begin{aligned} 0 \leq & n_{rel} = \frac{1}{|R_a|} * \sum_{i=1}^{|R_a|} r_{mi} \leq 1 && R_a \neq \emptyset \\ & n_{rel} = 0 && R_a = \emptyset \end{aligned} \tag{4}$$

We define the usefulness of a Semantic Note for an agent to consist of both understanding the note and considering it relevant. The information usefulness variable ($n_{use}$) also receives values between 0 and 1, and is formalized as follows:

$$n_{use} = a * n_u + b * n_{rel} \tag{5}$$

where $0 \leq a + b \leq 1$ and $a, b \in \mathbf{R}^+$. Parameters $a$ and $b$ in Equation 5 indicate the weights that are assigned to the understanding ($n_u$) and relevance ($n_{rel}$), respectively. The emphasis on these weight parameters depends on the application.

## 3 Conclusions and Future Work

We described an approach for determining information usefulness in the Semantic Web from a single agent's point of view. Information usefulness is formed based on the levels of *understanding* and context-dependent *relevance* of the information. We introduced a notion of *Semantic Note* to refer to the meaningful unit of information for an agent acting in the Semantic Web. Determining information usefulness forms a part of a broader approach, namely applying the theory of distributed cognition in the Semantic Web. Since the Semantic Web is an environment for software agents in addition to humans to operate, both were considered as "cognition distributors".

Among our future work is to consider various context-aware filters with our model. In addition to the most typical context attributes, namely location and time, activities and user interests associated with them could be taken into account when evaluating the relevance of content. Other future work includes developing a more refined classification of content creators—ranging from individual users to commercial parties, public administration, and virtual communities—and considering their impact in the information usefulness determination. In our current implementation, developed in terms of the DYNAMOS project[3], we have support only for dividing between service providers and individual users, but we plan to extend this. We will also pay more attention to the interrelationships and relative importances of various statement kinds in Semantic Notes, as well as to the rules that connect the Semantic Notes with users' current contexts.

**Acknowledgements.** The work reported in this paper was conducted in terms of the DYNAMOS project, funded by Tekes[4], TeliaSonera, Suunto, ICT Turku, and VTT.

## References

1. T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):34–43, May 2001.
2. J. Hendler. Agents and the semantic web. *IEEE Intelligent Systems*, 16(2):30–37, 2001.
3. J. Hollan, E. Hutchins, and D. Kirsh. Distributed cognition: Toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction*, 7(2):174–196, 2000.
4. E. Hutchins. *Cognition in the Wild*. MIT Press, Cambridge, MA, 1996.
5. Dublin Core Metadata Initiative. *Dublin Core Metadata Element Set, Version 1.1*, 1999. Specification identifier: http://dublincore.org/documents/2004/12/20/dces/.
6. A. Williams and Z. Ren. Agents teaching agents to share meaning. In *AGENTS '01: Proceedings of the fifth international conference on Autonomous agents*, pages 465–472, Montreal, Quebec, Canada, 2001. ACM Press.

---

[3] Dynamic Composition and Sharing of Context-aware Mobile Services, URL: http://www.vtt.fi/tte/proj/dynamos/

[4] National Technology Agency of Finland