

Running head: WEB SEMANTIZATION

Web semantization

Peter Vojtáš, Jan Dědek, Alan Eckhardt

Department of software engineering, Faculty of Mathematics and Physics, Charles University

Institute of Computer Science, Academy of Sciences of the Czech Republic

Abstract

Abstract.

Web semantization

Introduction.

Problem

Searching a used car. In our work for web semantization, we started with an illustrative example of a user buying a used car. She wants to get a reliable car with a small consumption. The car should also have as many airbags as possible. There are lot of other attributes of a car that may or may not be significant for the user. There are also attributes of the seller of the car, such as the price of the car, the confidence of the seller or the geographical position of the shop, where the car is parked.

In the current state of Web, our user has to access several (many) web pages to find a car she would really like. She can have no assurance that she found the best one; she maybe missed one car seller or did not look on the right page. Web pages of sellers also often do not present all necessary information about cars they offer. She have to go to e.g. the manufacturer's page, where the given car model is described.

There exists pages that gather information about cars or other goods (<http://www.activeshopper.com>). These pages are often only a list of names of goods with their price and sellers without any further information about the product. For making use of them, the user have to first find what type of car she wants and then find the lowest price. Even then she might not be satisfied - the seller might be in other region or be unreliable etc. The price is not the only criteria.

Keyword search limitations. If the user wanted to use Google (*Google search engine*, <http://www.google.com>, n.d.) to find her a good car, she would not be any more successful. Using keywords is a good method when searching some information about cars, but not so much when you search a car with some properties. Using keywords such as "airbags" or "low consumption" may lead to a better results, but the search on the web page is necessary.

Car search in a semantized web. Let's assume that web have gone a little bit semantized - some information is stored on web pages in a machine-readable form, there are distributed lists of such pages, some ontologies exists and are widely used.

In this web, our user would have a virtual agent that will search the car for her. She only has to express her preferences in terms of an ontology that describes cars and their sellers. The agent will then search for pages that contains information about cars. It can decide, which car is more appropriate for the user based on the properties of cars and their sellers. It has to combine information about both to an overall degree of preference. It provides the user with this degree as well as other information about the car. She can rectify the agent by rating his results. In this way, the agent refine his criteria on cars and on sellers.

In this scenario, the user does not need to access any page at all. She can, if she is interested in a particular offer, to find out more detail about it.

My Idea

Agent proposal.

Ontologies.

Web search/LG.

Decathlon - conflicting objectives/AE. One of the aspects our proposed agent should be able to do is to compare different objects based on their attributes. We call this "decathlon effect", or in economy terminology "conflicting objectives". The problem is that cars have several attributes, such as number of airbags, reliability, maximum speed etc. The user have some preferences on these attributes, she wants many airbags, lower maximum speed, high reliability... These preferences are called objectives.

Some of these objectives are conflicting e.g. wanting high maximum speed and low consumption. There is no clear winner, some of the cars may be incomparable - Audi TT has high

speed but high consumption, on the other hand Ford Focus may have lower speed but also lower consumption. Regarding two objectives "high speed" and "low consumption", these two cars are incomparable.

This undecidability is solved by an aggregation function, often denoted by @. This function takes preference on attributes and combines them into one global score of the whole object (car). In this way, the agent can order all cars by their score. The aggregation function may be a weighted average, for example $@(Speed, Consumption) = (3 * Speed + 1 * Consumption) / 4$, where *Speed* and *Consumption* are the degrees to which the car satisfy the criteria.

Linguistic.

Details of my idea

Web search/LG.

Top-k search.

Experiment/JD+AE.

Related work

Conclusion and future work

References

Google search engine. <http://www.google.com>. (n.d.).

Author Note

.