Web Wisdom: An essay on how Web 2.0 and Semantic Web can foster a global knowledge society

Christopher Thomas *, Amit Sheth

Kno.e.sis Center, Wright State University, Dayton, OH, USA

ARTICLE INFO

ABSTRACT

Admittedly this is a presumptuous title that should never be used when reporting on individual research advances. Wisdom is just not a scientific concept. In this case, though, we are reporting on recent developments on the web that lead us to believe that the web is on the way to providing a platform for not only information acquisition and business transactions but also for large scale knowledge development and decision support. It is likely that by now every web user has participated in some sort of social function or knowledge accumulating function on the web, many times without even being aware of it, simply by searching and browsing, other times deliberately by e.g. adding a piece of information to a Wikipedia article or by voting on a movie on IMDB.com. In this paper we will give some examples of how Web Wisdom is already emerging, some ideas of how we can create platforms that foster Web Wisdom and a critical evaluation of types of problems that can be subjected to Web Wisdom.

© 2010 Published by Elsevier Ltd.

1. Introduction

Quite recently, we took our first steps on a virtual ground. We occupied phone lines to dial up to computers that allowed us access to information some universities and government organizations provided. Those days seem to be long gone. We do not dial into a network any longer, we are online and see it as an annoyance when we are in a spot where we cannot be. We have constant access to virtually limitless amounts of information at all times.

Probably the greatest change in the perception of the Web occurred when people started to reverse the information flow (see Fig. 1).[1] While the write-capability always existed, only with the advent of Web 2.0 technologies, Social Networks, Peer to Peer networking and other tools that facilitated participation did users start to take advantage of read–write capabilities of the Web on a large scale.

Tim O'Reilly, who coined the term *Web 2.0*, made an interesting observation about Web applications and knowledge accumulation:

> A true Web 2.0 application is one that gets better the more people use it. Google gets smarter every time someone makes a link on the web. Google gets smarter every time someone makes a search. It gets smarter every time someone clicks on an ad. And it immediately acts on that information to improve the

experience for everyone else. It's for this reason that I argue that the real heart of Web 2.0 is harnessing collective intelligence.[2]

There is a broad consensus that the Web is moving more and more in the direction of becoming a platform for global knowledge accumulation and intelligence gathering. However, even applications such as Google are predominantly read-only. An algorithm takes advantage of the structure of the Web and of human interaction to reflect this insight in its search results. Some other search engines drive the interaction paradigm further by providing interaction with a human search agent, for example the ChaCha search engine.[3] One can argue about the results of the expert searches, but it is an alternative step towards harnessing collective intelligence. In contrast to this immediate interaction between information seeker and provider, Wikipedia[4] allows asynchronous creation and retrieval of information. Once a topic of interest is created, several users will work on it and over time and, in the ideal case, produce a comprehensive description of the subject. Computational methods of quality assessment can help decide which topics have matured. For example Thomas and Sheth (2007) found patterns that identify when a Wikipedia article becomes sufficiently reliable. Additional work on measuring trust is emerging.

Recently the areas of human computation and social computation have attracted much interest. Intuitively, humans and computers excel at orthogonal tasks. While computers are efficient

---

 * Corresponding author.
 E-mail addresses: topher@knoesis.org (C. Thomas), amit@knoesis.org (A. Sheth).
 URLs: http://knoesis.org/researchers/topher/ (C. Thomas), http://knoesis.org/amit/ (A. Sheth).
 [1] Copied from http://www.taggly.com/tags/journal.

[2] From Tim O'Reilly's Commencement Speech at SIMS.
[3] http://www.chacha.com/.
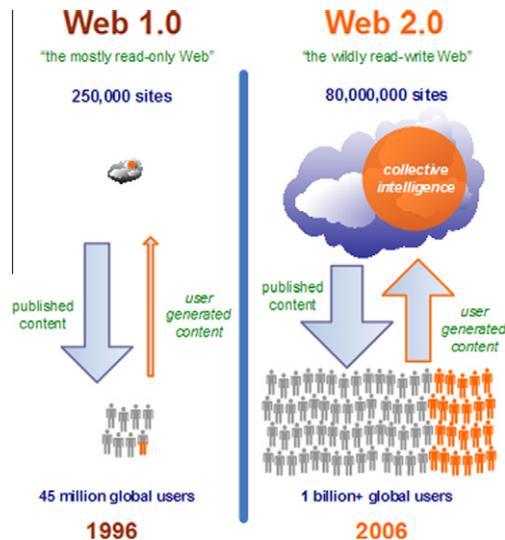[4] http://www.wikipedia.org/.

Fig. 1. Reversing the information flow in Web 2.0.

and effective in logical and mathematical analysis, humans are strong in conceptual tasks, i.e. tasks that require perception, intuition or creative thinking, etc. Thus it seems a logical step to share the workload. The WWW in general and social networks as well as the Web 2.0 paradigm in particular make it possible to analyze and harness human knowledge and human resources in an unprecedented fashion. The analysis of blogs and social networking sites has given great insight into propagation of information and the structure of social networks; sites such as Wikipedia explicitly accumulate knowledge of its community members and purposeful online games, such as the ESP game by Von Ahn and Dabbish (2004) or Google's image labeler, give recreational incentives to participate in social processes. In a different light, the annotation of images can be seen as a computational problem that, in these cases, is solved by humans acting as computers. In the Hitchhiker's Guide to the Galaxy, Douglas Adams describes the earth as a giant computer meant to give the "Ultimate Answer" to life, universe and all the rest. Whereas from some point of view this computation has probably been going on since the earth formed and computing power has significantly sped up ever since the first cell appeared, the development of the web has given another boost in this direction. Like never before can every Internet user participate in a global knowledge gathering process.

One unfortunate trait of the entirety of all that is known to mankind is that the parts of the knowledge that matter at a given moment are not possessed by me, so I have to look for it. Web searches have proven quite helpful, but the easiest way to learn is usually being taught by someone whose knowledge we trust. What if there was a system in place that, even though not intelligent enough to answer our questions directly, knew who could answer specific questions, be this a machine, an individual or a community?

Aside from the aspect of putting knowledge out there purely for the purpose of it being absorbed and put to use by others, there is a growing tendency towards directly employing web users to solve small problems. The grid, which used to be aimed at computers solving computational problems in now opening up to any kind of agent that can perform any sort of task that again has a bearing on other services and users on the grid or the web. *Seti@Home* was a pioneer-task for the computational view, Amazon.com's *Mechanical Turk* is probably the largest-scale implementation of the human-centric view. Platforms that could combine both are likely to appear, soon.

Despite the advances in computational algorithms, artificial intelligence and processing speed, many problems remain unsolved and may be computationally unsolvable. Limited by Turing-computability and complexity of both the algorithm and the necessary background knowledge, truly intelligent machines seem out of reach. The types of problems that can be mastered algorithmically are quite different from those that require extensive world-knowledge, creativity and the ability to purposefully abstract and intuit. Tasks that humans perform without even thinking about them (such as recognizing and grasping objects, planning a route through dense traffic while anticipating the behavior of others or stepping on a pile of books to change a light bulb) require very specialized algorithms that usually only perform well in the exact settings for which they were developed. One of the main drawbacks of traditional Turing-computation is the lack of interaction Wegner, 1997. Many tasks we solve require tedious computations that we like to outsource to machines that are built for these tasks and occasionally require human input to make or confirm a decision. Operating systems are a very common example of these kinds of tasks. Workflows in modern scientific experiments that involve massive computations alternating with human judgment exemplify further reaching interaction tasks. The Web 2.0 paradigm allows us to solve these kinds of tasks on a larger scale with many participants making judgments, decisions or contributing information. Google showed with its page rank algorithm Brin and Page, 1998 one of the first large scale deployments of independent distributed decision making. Others are looking at generalizing this idea beyond ranking web pages Rodriguez and Steinbock, 2006.

Is some information more valuable than other? Do some people make more significant contributions than others? Do some compute incorrectly? The answer to all is of course "Yes". The experience with Wikipedia and image labeling games has shown, though, that social processes weed out the bad seeds. Social networks deal with these phenomena by implicitly assigning trust insofar as the members of a discussion group or a smaller sub-community have some idea of how trustworthy their friends or collaborators are.

New problems arise, when we try to find a general way of tapping this potentially vast human resource. Looking at it from a service-oriented or agent-centered point of view, individuals and communities on the web provide services that can be discovered or advertised or they act as agents in the realm of the web. These services can be black boxes, we might not always know whether humans, computers or both are behind the offered service. And we do not need to know. A larger architecture that takes all possible computational resources into account wants complete transparency when it comes to choosing the appropriate service. However, the outcome matters, so we care about quality and time constraints.

A few years into the existence of the World Wide Web, its spiritual father Tim Berners Lee introduced a vision of a Web not only for humans to use, but also for machines; a more meaningful, a Semantic Web Berners Lee, Hendler, and Lassila, 2001. Many a researcher has worked hard on making this vision come to life just to be outsmarted by a bunch of folks who decided to bring more meaning to web resources without a formal underpinning. We like to follow the aforementioned thought by Liu and Maes (2007) that this is a beautiful development, because it shows that the Web has become a part of our social world and is evolving regardless of academic interference. However, we are not happy just letting things go unanalyzed and untouched. Here, we want to turn the tables a little bit to take some of the usurped area back into the hands of more rigorous semantics while taking the gold with us that was left behind in the form of tagging, collaborative ideas, Web 2.0 etc., by asking the question "(How) can Semantics improve the mechanisms that are in place for collaborative strategies towards

more reliable problem solving?" We have shown before how different kinds of web-semantics can be used to describe different levels of formalization and agreement (Sheth, Ramakrishnan, & Thomas, 2005). Using this insight, we want to investigate how we can describe, analyze and invoke services offered on the web in a unified way, whether they are offered by individuals, communities, based on human intelligence or computational methods.

The remainder of this essay is organized as follows. In Section 2 we will give some examples of problems to which Web Wisdom can contribute and define classes of problems that require different types of strategies to be solved. Section 3 will discuss epistemological and social implications of Web Wisdom. In Section 4 we discuss how to motivate problem solving and Section 5 deals with integrating Web Wisdom with the Semantic Web. Section 6 is concerned with drawbacks and possible failures of the approach, and Section 7 finally concludes the essay.

2. Solving difficult problems

In the introduction we described the need and potential for collaborative problem solving. Here, we aim to find a unified framework for solving problems that would traditionally be interesting for the field of Artificial Intelligence. We show a few problems that human computation can be and has been applied to and methodologies for making these problem solving services more accessible.

2.1. Problems to be solved

The following list identifies some problem classes that require different approaches. Some problems have multiple classifications.

- Problems that currently have low precision in their solution
  - Web search
- Problems that currently have low precision and low recall in their solution
  - Fact extraction
  - Object recognition
- Problems that currently can be solved only in limited domains
  - Computer vision problems
    - Image annotation
    - Object recognition
  - Quality control
- Problems for which computational solutions are not feasible or insufficient
  - Encyclopedia creation
  - Aesthetic quality of e.g. art/literature
  - Literature translation
- Problems that could be learned but for which not enough training data is available
  - Computer vision problems
  - Environment training, e.g. for robots
- Problems that are intractable
  - Traditional NP-hard problems
  - Problems that are not Turing-computable

As briefly shown in the introduction, Web search already takes clear advantage of human interaction by using page rank, click stream data, query logs and, more recently, having users explicitly promote or demote search results. Games are used for fact extraction and object recognition. We will subsequently give a few examples. It becomes more problematic when the recognition needs to be in real time, as is necessary for quality control applications. However, a big problem in object recognition tasks is the lack of well-annotated training data. Games such as *Squigl* and *Peekaboom* help provide such training data.

2.2. A few examples

The examples we show are all from the different classes of problems shown above. Here we are going to separate them by different criteria, namely whether the intended goal, i.e. the generated information resembles the human computation process or whether the computation is an artefact of a different activity.

2.2.1. Examples in which the process resembles the goal
2.2.1.1. Tagging. The tagger wants to be able to retrieve information based on an individual keyword. Halpin, Robu, and Shepherd (2007) showed that even in an uncoordinated environment in which any tag can be given, many users agree on few tags for each item whereas many uncommon tags are rarely used. The frequency of tags follows the power law distribution that is also expected from distribution of word frequencies in natural language.

2.2.1.2. Encyclopedia creation. The online encyclopedia Wikipedia depends on collaborative editing of content. Also, the structure of Wikipedia can be used in a limited form for computational purposes. It is a good example for the tradeoffs between ease of participation and computational merits. However, over time, contributors have added semi-structured descriptions in the form of templates. Most used are the *InfoBoxes* that have been extracted as Linked open Data into DBPedia Auer et al., 2007.

2.2.2. Examples in which the process is different from the goal
2.2.2.1. Image tagging in the ESP-game. The image labeling game ESP (or the Google image labeler) exemplifies the potential of human–computer collaboration. Humans do what they do best – identify objects. The algorithm then analyzes the tags and decides whether information has been gained by the given tag. If specific tags are regularly used and do not provide any new information, they are set as taboo-tags to force the users to get more specific in their descriptions.

2.2.2.2. Image segmentation in Squigl and Peekaboom. Whereas the ESP-game was meant to provide high-quality tags for images, *Squigl*[5] is a game in which objects inside an image are demarcated by drawing a line around them. The tag output from the ESP-game are presented to the players along with the image and both players draw a line around the object denoted by the tag. Bonus points are given when both players identify the same area of the image. The results are then collected to compute averages. The output of this game can either serve search engines to give better results or serve as training data for object recognition algorithms. *Peekaboom* Von Ahn, Liu, and Blum, 2006 was a game (it has since been discontinued) in which one player slowly reveals objects inside an image to another player who guesses the label of the object. It is easier and faster to play than Squigl and also gives insight into what the most discriminating features of an object are, in order for humans to recognize it.

2.2.2.3. Fact extraction in Verbosity. With a similar strategy to the party game Taboo, Verbosity asks one player to state properties of some concept to let another player guess what is being referred to. The goal is to collect facts for a large common-sense knowledge base. Data generated from Verbosity has been used in the Analogy-Space project Speer, Havasi, and Lieberman, 2008 to identify similarity of concepts and relationships between concepts.

---

[5] http://www.gwap.com/gwap/gamesPreview/squigl/.

2.2.3. Platforms

2.2.3.1. Amazon's Mechanical Turk. The Mechanical Turk differs from the previous examples insofar as it is rather a marketplace for services. The incentive to participate here is monetary compensation. Service seekers can post requests and suggest a price. Problems that are posted using web forms or an API are so called human intelligence tasks (HITs). The HITs are described using tags, a natural language description and formal descriptions, such as allotted time, reward and expiration date. Problem solvers are given reputation points based on reliability/accuracy and can take qualification tests that make them eligible for specific tasks.

2.3. Defining AI-interesting problems

As mentioned above, many of the problems that were discussed traditionally belong to the field of Artificial Intelligence. Computational solutions, however, are thus far limited to specialized and narrow domains, give insufficient results, or are intractable. Hence, we will call them AI-interesting problems. This definition is informal and more general than that in Von Ahn, Blum, and Langford (2004) of AI-hard problems.

Definition 1. A problem is AI-interesting if there is no algorithmic procedure that solves it efficiently, but humans can solve it with sufficient success in a reasonable amount of time or if it is currently perceived to be more accurately solvable by humans.

The definition is intentionally kept vague, because it is not only the "hard" problems that are generally perceived to be better solved by humans. An example is document tagging. While classifiers have a comparably high precision when assigning document classes, tagging is often intentionally biased. An Amazon page selling the new PlayStation, for instance, can easily be classified into a category *video game consoles*, but not as easily into the *"things my kids want that I'll never buy, but they'll nag their grandparents until they'll get it anyway"* category.

As mentioned before, humans and computers have different strengths and weaknesses in their abilities to compute. Many human-computation tasks are thus not restricted to humans, but enforce or promote interaction between man and machine. Let the computer do the tedious work and let the human intervene where the computer fails.

2.4. Interaction

On an abstract level, the web is a large collection of data as well as human and computational agents, connected via static and dynamic relationships, such as hyperlinks, search engines, Web Services, etc. All these agents or services constantly perform some tasks as sensors or effectors of their virtual environment. When we use this classification to categorize activities on the web, it can be seen as a giant interactive identity machine (IIM), as defined by Wegner (1997):

Definition 2. IIMs are simple transducers that realize nonalgorithmic behavior by harnessing the computing power of the environment.

The architecture of many contemporary Web Applications is built on interaction. Computation is done client- and server-sided after or during the user's interaction with the system. For many of these interaction tasks, the participating services are predefined, but other more complicated or more uncertain problems may require the discovery of previously unknown services. In order to find the right service, service directories exist with descriptions of the capabilities of the services. For this discussion, the quality constraints of a service are of particular interest.

2.5. Service descriptions

We want to limit the discussion to services that can be seen as atomic, i.e. taking a set of inputs and giving back a set of outputs without requiring the invoking agent to specify other services that go along with it. The inner workings of an atomic service do not concern us here. It might well be a complex process composed of multiple services. In order to unify the view of problem solving or of services on the web, some classes of problems are defined in Fig. 2. It shows how problems may be decomposable into subproblems of different kinds. The solution seeker will have a number of quality constraints to solve each of these problems. The problem description together with the quality constraints are then input to a service. The quality of service for a Web Service can be described as a set of assertions that can be numbers, strings, functions such as fuzzy class membership functions or probability distributions. A human service can, for example, be annotated with a success distribution and a time distribution. When the service is discovered, it is then possible to see that it will e.g. take 5 h on average to execute with a standard deviation of 2 h. The probability of a correct answer could be 95%. It is also feasible to have combined functions of different constraints, if e.g. there appears to be a strong correlation between the time taken to solve a task and the accuracy of the solution. Much work has been done in the area of Web Service QoS as exemplified in a survey by Dustdar and Schreiner (2005). Work in modeling of uncertain constraints is rare, but Hwang, Wang, Tang, and Srivastava (2007) have studied probabilistic models using random variables for QoS constraints. Allowing for random variables, intervals or functions to describe a QoS measure, allows a discovery agent to be ignorant as to whether a human or a computer is behind the service. All that matters is that the functional description of the service matches the problem and the QoS parameters are within the given tolerance interval(s).

2.6. Putting it all together

In order to use human and computational problem solving agents in an efficient way, a good description of both the problem and the services that can solve it is necessary. We outlined types of problems that require human computation and unified ways to describe services that offer problem solving of any kind, abstracting from the kind of service providing agent. What matters in the end is the quality of the solution, not the process. Web process composition frameworks Verma et al., 2005, for example, can then optimize on the best services that are offered. Even for discovery of the more lightweight services usually used in Ajax-based web applications, these descriptions are helpful, because timely execution and reliability are important. The Mechanical Turk already provides rudimentary support for this kind of annotation and an API for computational access to the solutions. More streamlined notification and more formal descriptions would allow for a close to real-time invocation of these human computation services. That said, the Mechanical Turk works solely on the basis of monetary compensation. Later on, we will explore different kinds of incentives for problem solvers and why solution can be trustworthy.

3. Problem solving in a social world

Having identified some types of problems that could benefit from large-scale problem solving, this chapter discusses epistemological implications of Web Wisdom. The obvious questions being about truth, verifiability, reliability and trust. The previous chapter showed ways to formalize these parameters. Here, we are discussing reasons why this might just work. The times in which lone thinkers mused over problems in complete isolation are over.
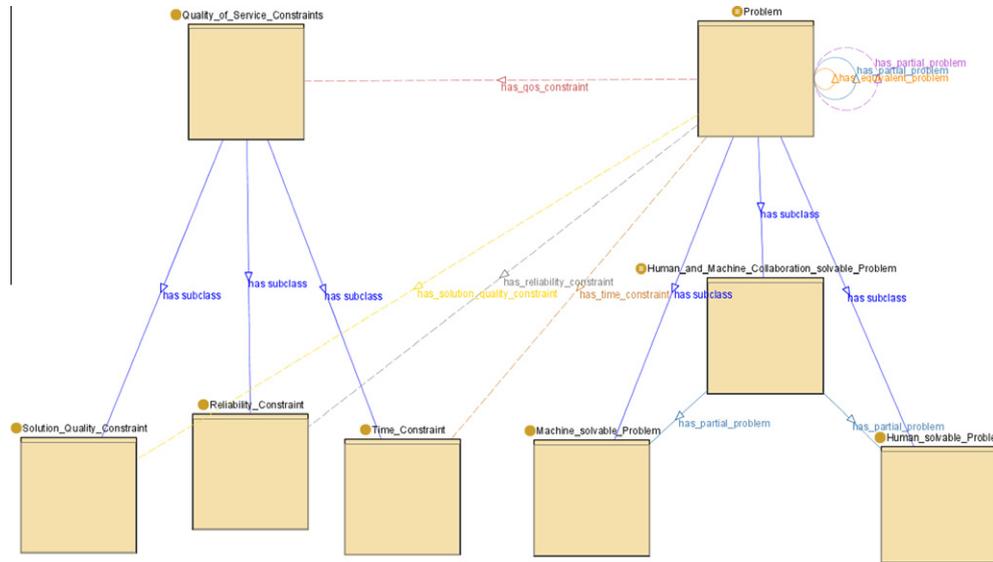
Fig. 2. Classes of problems and some QoS dependencies.

The underlying assumption here is thus that knowledge discovery and aggregation is (almost) always a social activity. This starts with language acquisition, which not only gives us the means to communicate, but arguably puts us already in an epistemic context Lakoff and Johnson, 1980. It continues during education and socialization and does not end when we get out of school, job training or university. We gain knowledge through our peers (friends, teachers, society) and for our peers.

3.1. Individuals and crowds

Goldman (1999) identifies four stages of social distribution of knowledge (see Fig. 3):

(1) Discovery
(2) Production and transmission of messages
(3) Message reception
(4) Message acceptance

These four stages are still all focused on individual choices, but are quite helpful, since the individual is at the first and last stage of knowledge accumulation. Stage 1 describes the individual discovery of information, regardless of how this information was produced. Stage 2 describes the transformation of the information to knowledge and what the individual decides to do with it. The individual can decide not to communicate the information at all, in which case the path of information ends. If the person wants to communicate she can decide to exaggerate or falsify the information.

The message receiver plays another crucial role. In humans, reception is more than the physical task of seeing, hearing or feeling stimuli; the signals that are perceived need are processed and categorized by the recipient's world view. Whether the message has been received as it was intended must often be clarified through further communication between sender and receiver. In the acceptance stage the recipient then has to decide whether to actually incorporate the information into his world view as a true belief.

3.2. About the Wisdom of crowds

Whereas Goldman (1999) analyzes the individual choices in knowledge gathering and communication of this knowledge, Surowiecki (2005) in "The Wisdom of Crowds", identifies complementary social knowledge aggregation techniques that promise to be successful. It is possible to substitute a community for an individual in Goldman's four stages, making justified communication more likely.
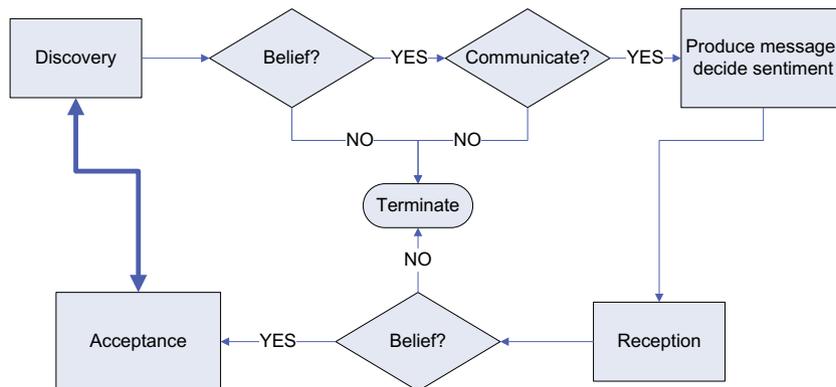

Fig. 3. Goldman's levels of communication.

First, Surowiecki identifies three categories of problems:

(1) Cognition problems: questions with one answer or questions with a preferred answer.
(2) Coordination problems that require coordination of actions between members of a community.
(3) Cooperation problems involve the challenge of getting members of a group involved in tasks or duties that seem contrary to their self-interest, such as paying taxes.

Several field experiments have shown large groups succeeding in cognition problems. Surowiecki gives a few compelling examples.

- *Estimating the number of Jelly Beans in a jar.* Averaging over all individual guesses always got very close to the actual number of Jelly Beans in a jar. For each experiment, only few individual estimates were closer. Furthermore, the best guesses did not come from the same guessers, but were most likely lucky shots
- *Betting on election outcomes.* Surowiecki cites the Iowa electronic markets (IEM) that allow stock-market style purchases of election options. The price for an option reflects the current standing of a candidate. Even though disproportionately many Iowans participate and the number of participants is over all comparably small, the IEM tends to produce better estimates than other polling data.

The first one seems quite straightforward. When the group is large enough, the individual estimation errors cancel each other out, leaving a small margin of error. If the better guesses were not just coincidental, the "estimation experts" would have been able to repeat them. The second one is trickier, because also in polls large, representative groups are asked for an independent opinion. The difference between the polls and the betting is that participants think about how the others might act, without knowing how they will actually act. So it seems that simply taking "the others" into consideration, improves our estimation capabilities. This is an example of people solving an extremely complex problem with many variables in an intuitive way. The argument being that intuition, normalized over large crowds, will yield good results. This is a field where machines lack capabilities. When the number of variables gets too large, computation takes a long time; when the available data is too sparse, prediction becomes unreliable.

According to Surowiecki, the requirements for good precision in answering questions, making predictions or making decisions are

(1) Diversity of Opinion
(2) Independence
(3) Decentralization
(4) Mechanisms for aggregation of individual results

On the web, the first three are, for the most part, given by definition. However, the last point, while theoretically independent of the others, can in practice impede them, because a tighter community will likely have aggregation mechanisms more easily in place than a more loosely coupled community, thus diminishing the value of the first three requirements. In some instances, tight communities can easily sabotage the outcome of a distributed decision making process by infusing it with pre-decided answers. A previous version of the Google page rank algorithm was particularly susceptible to this kind of misinformation-attack. So-called link-farms took advantage of the fact that Google indexes not only on the words on the pages themselves but also on anchor text used in the *href* tag that links to the page. These so-called Google-bombs [6] would force Goo-

---

[6] URL: en.wikipedia.org/wiki/Google_bomb.

gle to return e.g. the Biography page of President George W. Bush as first result of the search "miserable failure" or the website of his 2004 rival John Kerry for the search term "waffles". Undoubtedly, these kinds of attacks are a problem for large scale distributed problem solving, especially when the results are not evaluated but immediately serve as the input to another, independent and potentially mechanic, problem solving or decision process.

The ESP game or Google's image labeler are examples of cognition problems that have preferred answers, but where the preference is not easily gradable. Success is measured by 2 participants using the same term to describe an image at the same time. A drawback of this method is that it leads to correct, but more generic answers. It seems that users will first enter more specific descriptions of the image, but then, when the time runs out and no consensus is reached, use more general tags, such as *Man, Woman, Car, Tree*, etc. Here, the kind of competition used is counter-productive to annotation-depth. This drawback was later addressed by using general tags as *taboo words* in order to force the players to be more specific.

In general, it seems that crowds are quite good at solving problems that can be answered in a single sentence, an estimate or a thumbs up. How many pickles are in this jar? Which horse is going to win the race? Should I buy this company's stock? Should the gladiator live? One cannot say that these are simple questions, but the answer is always simple. Even though many factors weigh into the decision for a particular answer, there seems to be an immediate relationship between subject and object of the question.

The more difficult problem is how to map this to more complex problems that require a chain of associations that defy immediate intuition. Often, these are the *how* and *why* questions as opposed to the *"is it the case that"*, *"how many"* or *"should I"*, etc. questions. "How does smoking cause cancer?", "what causes stem cells to differentiate at some point?". Generally, these questions require high scrutiny and scientific exactness to be answered; a skill that cannot be expected from random crowds. Some of these problems might be decomposable into simple subproblems of the *"is it the case that"*-kind. Also, questions can be truthfully answered with varying degrees of depth. Common sense versus the scientific method – it turns out that both have their place.

3.3. Truth, verifiability and trust

Surowiecki (2005) makes a compelling case for the problem solving capabilities of anonymous masses. In the end, however, it is not enough to know that some social entity is capable of solving problems. We need to know whether there is a good chance that it will solve the problem and that the solution is correct, at least to the best of the participants' knowledge. Along these lines Goldman (1999) asserts that there is a "propensity towards truth." According to Goldman, it is in general more likely to get a correct answer than a false one. From an epistemological point of view, it will always be easier to agree on a justified, conventional or prevalent statement than on an ad-hoc invented lie. As such, the larger the body of participants gets the more likely it is that the answer received from a crowd is at least not a maliciously invented lie.

4. Recruiting problem solvers

With all kinds of problems waiting to be solved, who is going to solve them? Should we trust the problem solver and why should someone solve a problem for us in the first place? The problem solver can potentially be anyone or anything on the web. The more focused my search radius is though, the more knowledge I will have about the solver and hence the more trust. Of course the equation is not quite that simple, but it is a start. In general, we
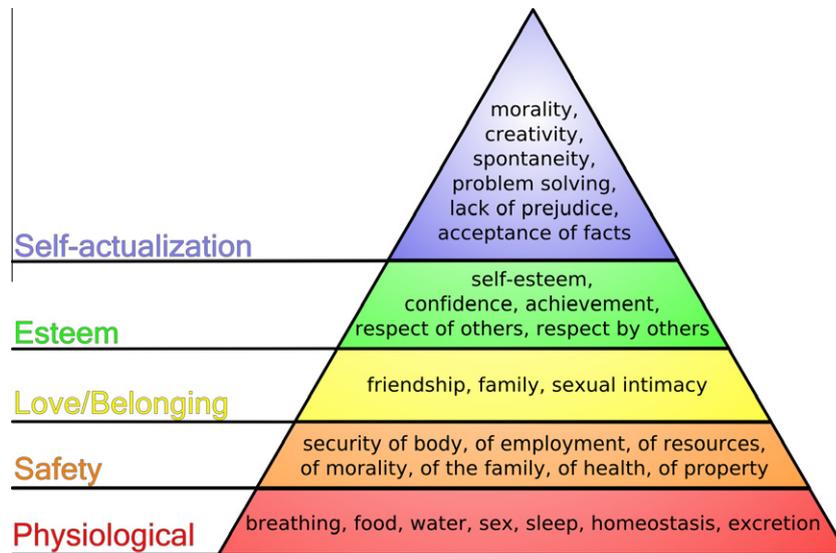
Fig. 4. The Maslow hierarchy of needs.

can say that the golden rule, for numerous reasons, also applies in the context of problem solving on the web. Having stated this belief in the kindness of strangers, discretion is of course the better part of valor. That is why most problem solving tasks apply some sort of verification of the solutions. In a social setting, this is usually done by checking for agreement between agents. In the ESP game Von Ahn and Dabbish, 2004, image labels are only trusted when both randomly chosen players have entered them in the same round. This makes for a high precision with the price of a low recall. More generally, Game Theory knows quite a few mechanisms that can raise the likelihood of a stable outcome. Combined with the propensity towards truth and the wisdom of crowds, we are set.

Why would someone want to help me? There are numerous reasons why someone should or should not solve a stranger's problem. Since there are no immediate repercussions for not helping out, the negative version of the golden rule does not apply.

Interestingly, as so often on the web, it was the adult entertainment industry that first made large scale use of human computation. When email providers installed CAPTCHAsVon Ahn, Blum, Hopper, and Langford, 2003 to prevent bots from automatically creating thousands of spam email addresses, adult web-site providers copied the CAPTCHAs they encountered on the email provider's site to the users' sessions, asked them to solve the problem and offered to see more free pictures as reward.

In the case of image labeling applications, online games became quite famous. the ESP game and Google's image labeler draw players because they are well developed and fun to play. Another approach is taken by Amazon.com with its *Mechanical Turk* project. Users are paid small amounts of money to solve problems on the web. Amazon.com termed this "Artificial Artificial Intelligence".

Looking at these approaches, the good old *bread and circuses* saying comes to mind. We are happy to be exploited when in the end we can fulfill our desires for food and entertainment.

The above exemplify low commitment types of problems. The user's involvement ends with the end of the session. In other cases, users make much stronger commitments. Contributors to Wikipedia or to Sourceforge[7] devote substantial amounts of their time to improving articles or algorithms without reimbursement. For these kinds of problem solving communities the *bread and circuses* metaphor does not seem to apply. Maybe we can take advantage of more sophisticated analyzes of human motivation.

The Psychologist Abraham Maslow proposed in his influential paper "A Theory of Human Motivation"Maslow, 1943 a hierarchy of needs humans aim at satisfying (see Fig. 4). This hierarchy shows a level of basic physiological needs that must be satisfied first. The higher up in the hierarchy a need is the more abstractly it will be satisfied. It must be said here, that Maslow's hierarchy is not undisputed, especially when it comes to the ordering of the needs in the hierarchy or to whether there is an order at all. This aside, for our purposes the last three levels in the hierarchy are of importance and an order is secondary.

Further criticisms include that it is not a fundamental human hierarchy of needs, but geared towards a western culture. It presupposes a mostly selfish human psyche that acts altruistically only to fulfill individual needs.

We can take this position as a worst case scenario by asking" what does an invitation to help out on a web-scale problem have to look like in order for the selfish user to participate? In the case of high-commitment tasks, however, the link seems vague. The Wikipedia community is to a large extent anonymous or at least very low profile. It seems that contributing to Wikipedia is nothing to brag about or to be highly regarded for by one's peers. The individual contributor matters less than the communal outcome.

Overall, to explain the motivation for participating in problem solving tasks on the web, moralistic approaches seem to fail. Karma, the golden rule, "what goes around, comes around", etc. usually involve negative assumptions that leave the protagonist imagining what happens if a bad deed is done.

Another approach to the explanation of why people act altruistically is given in the field of self determination theory (SDT). See here e.g. Ryan and Deci, 2000. SDT looks at human actions from a less subconscious and selfishly need-driven point of view in that it contemplates the degree to which people endorse their actions at the highest level of reflection and engage in these actions with a full sense of choice. This field has not been extensively analyzed. However, Ludford, Cosley, Frankowski, and Terveen (2004) did research on factors for online participation.

At first glimpse it seems that many factors play a role in whether a task or a community are deemed worth participating in. This can range from aesthetic appeal of the site or interest in the problem to the draw by members of other networks who are already participating. Probably what draws most people to e.g. My-

---

[7] URL: http://www.sourceforge.net.

space.com is that many people are using Myspace.com. Its features are not outstanding, its user-friendliness leaves a lot to be desired, there are few options for privacy settings, etc. Wikipedia, as mentioned above, offers very few rewards other than seeing a communal effort being accomplished. So maybe dynamic structure of the web, with the possibility to join and leave communities, where interaction is location independent and hence can draw very diverse communities from all over the world, and with its "free of charge" philosophy provides is some sense of "the greater good".

Recapitulating this chapter, we can assume that people are willing to engage in problem solving activities provided that the incentive is, analogous to Freudian categories: fun, monetary benefit, or the prospect of contributing to something sensible and bigger than the individual's contribution.

5. Web Wisdom and the Semantic Web

The previous chapters have outlined the kinds of web-solvable problems there are, how to formalize them, how to solve them, why we can trust the solution and why our questions may be answered in the first place. This chapter will attempt to put these capabilities and insights in the context of the Semantic Web.

The Semantic Web, as envisioned by Berners Lee et al. (2001) is a vast collection of formalized knowledge and agents that are capable of understanding these formalizations and acting upon them. The formalized knowledge is available in ontologies that use logic or graph-based formalisms on top of unambiguous pointers to web resources in order to represent knowledge in a computer accessible manner. The semantic web paradigm is in some sense perpendicular to that of the Web 2.0 termed collections of human-generated networks and tags that are ambiguous, arbitrary and chaotic. These tags and connections are created on-the-fly, reflecting the taggers' associations. By definition, ontologies contain true or at least highly justified knowledge and need to be logically consistent for agents to derive sound conclusions. Some drawbacks of this rigor have been addressed in Thomas and Sheth (2006) and Sheth et al. (2005); however, in order to have a completely reliable chain of inferences, this rigor is of great value. For this reason, Ontologies need to be created with immense scrutiny, both in common-sense domains Noy and McGuinness, 2001 as well as in scientific domains Sahoo, Thomas, Sheth, York, and Tartir, 2006; Thomas, Sheth, and York, 2006. While it is theoretically possible to have this kind of scrutiny reflected on domain-specific, expert-curated database-driven websites, such as Gene and Protein databases, the kind of grassroots semantics of Web 2.0 takes its steam from the fact that it is easy, quick and still extremely useful. These paradigms are not easy to combine. One is meant for machine consumption, the other for humans. Even in the Semantic Web camp, the opinions diverge. While Jim Hendler coined the phrase "A little semantics goes a long way", Chris Welty asserts "A little semantics is killing the Semantic Web." The paradoxical situation is that both may be exactly right. Web 2.0 has shown that a little semantics goes for the time being further than a lot of it. However, it has very little to do with the Semantic Web vision.

The question then remains, how can these two paradigms that seem so close intuitively, yet so far ideologically and technically, be combined to achieve the kind of man–machine–hybrid computing power that the web has the potential of providing? First of all, who said there cannot be anything between these two paradigms? We can take "a little semantics" and gradually improve it. We can relate tags to concepts in ontologies. We can take informal assertions and convert them into formal ones. How can this be achieved? It is a problem on the web, so let the web solve it. Many communities might have a strong interest in simply tagging resources. Let them tag! Others need rigorous formal representations, and they will develop them out of their needs. Given these

semantic bottom and top layers, other communities will have an incentive to relate them. The Wikipedia Infoboxes are an example of users seeing a need to present bits of information in a structured way. Information in the Infoboxes can be easily harvested by extraction algorithms. The DBPedia Infobox project Auer et al., 2007 has done that and makes this information available in RDF. In addition to manual creation of references to formal concepts, computational algorithms can find commonalities between similar pages and similar tags. It has been shown that free tagging converges to a stable semantic state where most taggers use few terms for the same resource and few taggers use diverging terms that form a long tail of tags Halpin et al., 2007. The development of Web 2.0 parallel to the Semantic Web but with more speed and larger acceptance makes apparent the futility of pushing web users in a specific direction. Any attempt to marry Web 2.0 and Semantic Web must respect this lesson learned. The Semantic Web must provide platforms that facilitate the use of semantics, that hide the formalisms from those who do not want and do not need to see them, that connect the things that are interesting to everyone to those that are interesting only to Semantic Web visionaries.

Web 2.0 is not a technological revolution. It marks a stage in a normal progression of the Web. Communication between client and server has always been possible and it has been used. Larger bandwidths and larger acceptance of the Internet by more and more users makes it the ideal playground for social networking sites and collaboration between users. Hence Web 2.0 has to be and is able to accommodate all the imperfections of the Web: the smorgasbord of conflicting opinions, inconsistencies even on single pages, different data formats etc. This is achieved by ignorance and apathy. The web simply does not know or care about these problems. So it goes on in its evolution and allows better and better searches and categorization, delivery and communication. At the same time, lacking formal requirements of knowledge representation, it cannot give us the kinds of decision making tools and question answering systems we are also looking for.

Let us not kid ourselves. The reason why the Semantic Web as an overarching structure is failing is because it cannot deal with these inconsistencies. Humans use the Web, humans can deal with and consistently create ambiguous statements that would drive every automatic reasoner mad. The beauty of the Web lies in this ability to cope with all these logical fallacies. Ambiguities are signs of beauty. The human Web is the Poetry and Prose that is challenged by the flawlessness of the Semantic Web with its resemblance to the language of a legal bill. For the most part we think that bills are necessary and that there is a reason why these bills are cumbersome. We usually let those who have dedicated their lives to understanding government speech deal with them. However, we do engage in politics on many different levels through direct involvement in meetings, petitions, voting, consumer behavior, etc. These examples were ordered by a diminishing degree of personal involvement. This analogy can also hold for the web. Intelligent applications can analyze our behavior and learn from it.

6. Some criticisms

One may wonder whether the kind of Web Wisdom that is promoted in this paper would, instead of Wisdom, foster nothing but great mediocrity. A positive response to this criticism could be that, following Surowiecki, crowds are also good at handling cooperation problems. This could mean that a large enough crowd will outsource problems that need expert solutions even if this outsourcing to experts involves certain sacrifices, such as monetary loss, loss of recognition, etc. Still, the success of a problem solving task involving large crowds seems to be determined by the aggregation function of the individual solutions. We cannot blindly trust

a solution, just because it was found by a large number of people. The method needs to meet the criteria put forth in Section 3.2 and good practices (such as the scientific method) for finding verifiable outcomes discussed for example by Goldman (1999). Differentiating these aggregation methods may well follow an evolutionary selection process. Different forms of organizations can coexist on the Web. For many areas of the web infrastructure, a total knowledge democracy might not be feasible or not even desirable. The hubs and authorities paradigm Kleinberg and Raghavan, 2005 can be seen as a web metaphor for a meritocratic form of organization. Different from a form of government, web users can change their trusted authorities at any point in time. Because of this freedom of choice, web authorities will be inclined to find the kind of organization that best benefits their peers.

Another point of criticism is ethical. If the participation in online tasks and communities is not based on selfish reasons, taking personal advantage without giving back would constitute exploitation. Hence, soliciting services free of charge from volunteers should also be for the greater good and not be used for monetary gain.

7. Conclusion

This paper outlined a vision of a unified view of problem solving on the web. It encompasses traditional Web Services executed by computers as well as a semi-formal view of human based services. The infrastructure is already in place and extensively used. Counting on the continuation of the current trend, Semantic Web research will have a hard time keeping up with all the "little grassroots semantics" that are produced at a phenomenal pace. However, if it manages to give these efforts a formal underpinning without being a disturbing factor in this rather chaotic growth, it can greatly help to unify the informal approaches and thus create an environment where knowledge can prosper and problems can be solved on a larger scale.

In future investigations we should take closer looks at the human and social components of this unified framework. Using the criteria outlined in this paper, we can evaluate services provided by individuals and social networks with respect to their reliability, costs and accuracy. We hope that we can empirically narrow the expected parameters that different types of problem classes have in order to get a better understanding of the feasibility of invoking or requesting corresponding human or social network based services.

References

Auer, S., Bizer, C., Lehmann, J., Kobilarov, G., Cyganiak, R., & Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. pp. 11–15. Available from http://www.informatik.uni-leipzig.de/auer/publication/dbpedia.pdf.

Berners Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American, 284*(5), 34–43. Available from http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. In *WWW7: Proceedings of the seventh international conference on World Wide Web 7* (pp. 107–117). Amsterdam, The Netherlands, The Netherlands: Elsevier Science Publishers B.V..

Dustdar, S., & Schreiner, W. (2005). A survey on web services composition. *International Journal of Web and Grid Services, 1*(1), 1–30.

Goldman, A. I. (1999). *Knowledge in a Social World.* Oxford University Press. Available from http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20&path=ASIN/0198238207.

Halpin, H., Robu, V., & Shepherd, H. (2007). The complex dynamics of collaborative tagging. In *Proceedings of the 16th international World Wide Web conference (WWW'07)* (pp. 220). New York, NY: ACM Press. Available from http://www2007.org/papers/paper635.pdf.

Hwang, S.-Y., Wang, H., Tang, J., & Srivastava, J. (2007). A probabilistic approach to modeling and estimating the qos of web-services-based workflows. *Inf. Sci., 177*(23), 5484–5503.

Kleinberg, J., & Raghavan, P. (2005). Query incentive networks. In *FOCS '05: Proceedings of the 46th annual IEEE symposium on foundations of computer science* (pp. 132–141). Washington, DC, USA: IEEE Computer Society.

Lakoff, G., & Johnson, M. (1980). *Metaphors We Live By.* Chicago: Chicago University Press.

Liu, H., & Maes, P. (2007). Introduction to the semantics of people *ampersand* culture. Int. J. Semantic Web Inf. Syst.

Ludford, P. J., Cosley, D., Frankowski, D., & Terveen, L. (2004). Think different: increasing online community participation using uniqueness and group dissimilarity. In *CHI '04: Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 631–638). New York, NY: ACM Press.

Maslow, A. H. (1943). A theory of human motivation. *Psychological Review, 50,* 370–396.

Noy, N. F., & McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology. Available from http://www.ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html.

Rodriguez, M. A., & Steinbock, D. J. (2006). The anatomy of a large scale collective decision making system. Tech. Rep. LA-UR-06-2139, Los Alamos National Laboratory, Los Alamos, NM.

Ryan, R. M., & Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. American Psychologist 55.

Sahoo, S. S., Thomas, C., Sheth, A., York, W. S., & Tartir, S. (2006). Knowledge modeling and its application in life sciences: a tale of two ontologies. In *WWW '06: Proceedings of the 15th international conference on World Wide Web* (pp. 317–326). New York, NY, USA: ACM Press.

Sheth, A. P., Ramakrishnan, C., & Thomas, C. J. (2005). Semantics for the semantic web: The implicit, the formal and the powerful. *International Journal on Semantic Web and Information Systems, 1*(1), 1–18.

Speer, R., Havasi, C., & Lieberman, H. (2008). Analogyspace: reducing the dimensionality of common sense knowledge. In *AAAI'08: Proceedings of the 23rd national conference on Artificial intelligence* (pp. 548–553). AAAI Press.

Surowiecki, J. (2005). The Wisdom of Crowds. Anchor. Available from http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20&path=ASIN/0385721706.

Thomas, C., & Sheth, A. P. (2006). On the Expressiveness of the languages for the semantic web - making a case for a little more. *Fuzzy Logic and the Semantic Web*, 3–20.

Thomas, C., & Sheth, A. (2007). Semantic convergence of wikipedia articles. In *Proceedings of the 2007 IEEE/WIC international conference on Web intelligence* (pp. 600–606). Washington, DC: IEEE Computer Society.

Thomas, C. J., Sheth, A., & York, W. (2006). Modular ontology design using canonical building blocks in the biochemistry domain. In *Proceeding of the 2006 conference on formal ontology in information systems: Proceedings of the fourth international conference (FOIS 2006)* (pp. 115–127). Amsterdam (NL): IOS Press.

Verma, K., Sivashanmugam, K., Sheth, A., Patil, A., Oundhakar, S., & Miller, J. (2005). Meteor-s wsdi: A scalable p2p infrastructure of registries for semantic publication and discovery of web services. *Information Technology and Management, 6*(1), 17–39.

Von Ahn, L., Blum, M., Hopper, N., & Langford, J. (2003). CAPTCHA: Using hard AI problems for security. *Lecture Notes in Computer Science*, 294–311.

Von Ahn, L., Blum, M., & Langford, J. (2004). Telling humans and computers apart automatically. *Communications of the ACM, 47*(2), 56–60.

Von Ahn, L., & Dabbish, L. (2004). Labeling images with a computer game. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 319–326). New York, NY, USA: ACM Press.

Von Ahn, L., Liu, R., & Blum, M. (2006). Peekaboom: a game for locating objects in images. In: Proceedings of the SIGCHI conference on Human Factors in computing systems. ACM, p. 64.

Wegner, P. (1997). Why interaction is more powerful than algorithm. *Communications of the ACM, 40*(5), 80–91.