Semantic Web Services
Assignment 1

Stefan Haselwanter
Juliette Opdenplatz

March 21, 2018

1 Evolution of the Web

Web 1.0 – Traditional Web

The first stage of the web era was mainly made up of web pages which provided static content (e.g., text, images) wrapped in HTML. Different websites were connected by hyperlinks and there was little to no interactive component or way to contribute content. Therefore, users were limited to consuming the content presented by the browser. That is why Web 1.0 is sometimes referred to as *Read-Only Web*. The client-server communication over HTTP was basically uni-directional and resources like files were accessed using URIs. Even though most websites were static, a few of them already provided some kind of interaction. Those were mostly e-commerce websites with basic shopping cart applications.

Web 2.0 – Social Web

The transition from Web 1.0 to Web 2.0 happened more gradually and introduced two-way interactions between users and machines. With the establishment of interactive web applications (AJAX) and services (APIs), the user became both the content consumer and content provider. New platforms like social media, wikis, blogs, boards, etc. arised to communicate and exchange information with each other over the web and changed the attitude from media for individuals towards media for communities (*Read-Write Web* or *Social Web*). In addition, the concept of social networking (blogging, tagging, sharing) enabled a strong collaboration between users on the Internet (e.g., Wikipedia). Even users without any technical background were able to contribute. Besides the human-to-human interaction, human and machine computing were integrated as a further extension. For example, in *Amazon Mechanical Turk*, users are able to post jobs or tasks (e.g., processing photographs, collecting information, cleaning data) that are then browsed and completed by other users in exchange for money.

Web 3.0 – Semantic Web

Since the web mainly consists of information written in human readable languages, they are most likely not easy to interpret for machines. Web 3.0 tries to fill this communication gap and make data accessible to machines by using semantic web technologies such as RDF, OWL, SKOS, or SPARQL. Those techniques provide an environment where applications can query data, draw inferences using vocabularies, link and integrate different kinds of information, or automate processes (*Read-Write-Execute Web* or *Semantic Web*). Semantic markup (annotation, ontology) enriches the data with meta data for context which empowers the meaning of web services. Web services are software systems designed to support machine-to-machine interaction over the Internet.

Examples for Web 3.0

One example for Web 3.0 is a search engine integrated in systems like Amazon's Alexa or Google Allo. Instead of narrowing down the search into its individual parts (e.g., 1. Searching for a

movie, 2. Searching for a place to eat, 3. Searching for a club to party), the search engine will fetch the answer based upon the complex sentence typed in, e.g., "I want to go for an action movie, then eat at a good Chinese restaurant, and then listen to some techno music. My options are?". A second example is a health search where a patient wants to figure out what he or she is suffering from based on a set of provided symptoms. This may result in an inaccurate answer as many diseases show common symptoms.

Wolfram Alpha is a computational knowledge engine which uses its own internal knowledge base with its own semantics and ontology. For example, one can enter "3 bretzels vs. 1 bread" and you will get comparisons of ingredients and calorie contents. This is done by taking a large amount of raw information and performing computations using those data. It produces pages of new information that have never existed on the Internet. On the other hand, Google provides you with links to pre-exisiting data. In a nutshell, Google delivers links to websites which might provide answers while Wofram Alpha provides actual answers. It does so by generating data (e.g., statistics) by processing data provided by different databases which contain structured data.

2 Computer Science of the 21st Century

While computer science is all about how a single computer works, web science is a completely different, multi-disciplinary field. Researchers try to understand what the web is, how it affects us, and how we can prepare its future. These are questions that cannot be answered by computer science alone. In web science, many different disciplines (e.g., computer science, mathematics, economics, law, sociology, etc.) have to be combined to let us design and build more complex human-oriented systems in order to understand the web and its relationships to humans and machines. Also, computer science is synthetic. It is made up by humans and about designing formalisms and algorithms to solve specific problems by using mathematical models which are themselves made up by humans. To the contrary, natural sciences such as chemistry, biology, and physics are analytic disciplines that aim to find laws that explain observed phenomena. Since the web is such a heterogeneous, decentralized, fast changing, and huge scaling system, there have to be at least some scientific methods which are combinations of both paradigms.

3 Data Science, Web and Social Considerations

Any decision process (human or algorithm) that decides what to include, exclude, or emphasize in information processing has the potential to include biases in the output. It is clear that algorithms can not understand concepts like racism. But, for example, if measurable criteria are used to exclude some sort of information which accidently correlates with race divides, racial biases might appear.

Example 1: *Google News* was one of the first platforms attempting to aggregate and personalize news with algorithms. Its algorithm ranks and groups news articles by frequency of appearance, source, freshness, location, relevance, and diversity. Unless explicitly programmed, an algorithm will never favorite one thing over another just because it likes the one thing more than the other. However, Google News will show you news according to its criteria. For example, it might tell you more about specific political parties if one presidential candidate appears more often in the media (frequency of appearance). Therefore, the site might seem biased even though the algorithm is not (at least not intentionally). When thinking about the two party system in the USA, there are actually more parties from which you hardly ever hear about unless you are specifically searching for them.

Biases can also occur in algorithms if it processes data that is already biased. For example, data created by humans is most likely biased. Hence, using this data as input to algorithms will often produce biased output.

Example 2: Consider a data about regional crime rates to compute where more police patrols are needed. However, there might already be biases in police recorded data due to racial profiling. Using this data can lead to a result which recommends increasing police activity where the majority of the population is of a certain race.