Kleenks: Linked Data with Applications in Research and Ambient Assisted Living

Andrei-Adnan Ismail, Razvan Dinu, Adina Magda Florea, Tiberiu Stratulat, Jacques Ferber

Kleenks: Linked Data with Applications in Research and Ambient Assisted Living

Andrei-Adnan Ismail[1], Razvan Dinu[2]
Adina Magda Florea[1], Tiberiu Stratulat[2], and Jacques Ferber[2]

[1] University Politehnica of Bucharest, 060042, 313 Splaiul Independentei
andrei.ismail@cs.pub.ro,
WWW home page: http://aimas.cs.pub.ro/people/andrei.ismail
[2] University of Montpellier 2, UMR 5506 - CC477, 161 rue Ada, Montpelllier, France
dinu@lirmm.fr,
WWW home page: http://www.lirmm.fr/~dinu/

Abstract. In the spirit of the Linked Data initiative pioneered by Tim Berners-Lee, we propose a new type of link between entities identified by URIs named kleenk. This new type of link bridges the gap between the classical structured data published as RDF and the semi-structured data formats pushed by social networks such as Facebook and Twitter.
Unlike previously published work on RDF and its extensions, our proposal promotes links to a first-class citizenship in the world of a semantic web: a kleenk has a content of itself, can be linked to recursively, and both people and machines can harmoniously collaborate in creating, evaluating and extending them.
We discuss the theoretical model of the kleenk and how it can be built on top of existing frameworks such as RDF, and identify the main challenges for adoption of the new model.
Finally, we validate our model with two real-world implementations: an online platform for spreading research results between researchers and an ambient intelligence elder tracking scenario where kleenks are used to perform sensor fusion between heterogeneous data sources.

Keywords: linked data, ambient intelligence, RDF

1 Introduction

"This is what Linked Data is all about: it's about people doing their bit to produce a little bit, and it's all connecting.,, - Tim Berners-Lee, TED 2009.

Linked data is a movement trying to expose the world's data in a structured format and to link it all together in meaningful ways. This concept has been gaining traction as more and more organizations are starting to expose their data in a structured, computer-understandable format, besides the traditional website. Until recent, the habit was this: if an organization owned some data and it wanted to expose it to the public, it created a website allowing users to explore it. However, it soon became obvious that this was not enough; humans were not the only ones interested in working with this data, sometimes even computers or software agents delegated by humans should be able to manipulate

it. In the dawn of this era, the web crawling[14] and screen scraping [15] concepts appeared. Programs that contained specific parsing code for extracting specific knowledge out of raw HTML emerged, and they were named crawlers or scrapers. Due to the technical difficulties of doing NLP (Natural Language Processing), these programs would use the underlying regularities in the HTML structure to parse the structured data. Soon, a war broke out between content owners who did not want to expose their data to machines and humans aiding the machines in extracting the data by continuously adapting the parsers to changes in HTML structure and to security additions aimed at differentiating humans from crawlers.

In the center of this war comes Berners-Lee's concept of Linked Data. Linked data is no longer data exposed by machines for machines, but it is data exposed by humans for their fellow machines. The Linked Open Data (LOD) project is leading this movement of encouraging people to expose their data to machines in a meaningful format. Most of the projects put forward by LOD are projects in which humans are in the center of the process of generating linked data. Big names in the internet industry such as Facebook agree with this vision, as confirmed by the launch of Facebook Open Graph v2 initiative at the F8 conference in 2011[3]. This announcement is about making the transition from the singular "like" action that users could perform on the social platform to a multitude of actions, even custom ones definable by application developers: read, watch, listen, cook, try out, and so on. Given the large amount of data continuously generated by users on their social networks, this step will finally expose all that data internally as structured data.

While academia has taken the pure path towards linked data adoption, industry has done the exact opposite. Social networking giants have convinced their users to first produce data in the form of small snippets of unstructured text, and then slowly introduced structure into the produced data, while keeping a vigilent eye on engagement metrics. Nowadays, users reference hashtags and other users in their tweets, allow applications to geotag their tweets in order to include location information, share pictures on Facebook and tag faces in them. While industrial examples of linked data are well-known and famous, we would like to give credit to some of the most important academic initiatives in exposing linked data:

- DBPedia[2] is a community effort to write parsers that extract structured information from Wikipedia infoboxes which are found on most pages. Triples in this database are kept in sync by using a subscription to a live feed of modifications, and they are created by automated programs as an indirect consequence of people's actions.
- Freebase [3] is an initiative supported by Google to apply the wiki concept to the world's knowledge. A user interface and a RESTful API are provided to users in order to be able to collaboratively edit a database of triples spanning more than 125 million triples, linked by over 4000 types of links, from domains as diverse as science, arts & entertainment, sports or time and space.

Domains such as information retrieval and ambient intelligence, with a huge impact in the socio-economic domains would benefit immensely from a larger adoption of

[3] https://f8.facebook.com/

linked data. In this paper we propose the **kleenk** concept, a link between two entities that is compatible with both academic and industrial approaches. Not only this model is able to leverage all existing produced data by both environments, but it is built in such a way that users and machines can both work together to build a larger graph of linked data.

This paper is organized as follows: in section 2 we formally define the problem kleenks are trying to solve: bridging the gap between generating structured data using machines and semi-structured data using social interactions. In section 3, we present related works in both the area of formal modelling of online knowledge and in ambient intelligence. In section 4, we contribute 2 scenarios that have been implemented by the authors in order to validate the kleenk model. In section 5, we present the formal Kleenk model and how it can be built on top of other existing models such as RDF and RDFS. In section 6 we present http://kleenk.com, a novel platform for spreading research results and linking between them based on the kleenk paradigm. In section 7 we present ElderMonitor, a novel distributed person tracking platform for building ambient intelligence applications. We show how the kleenk concept can be successfully applied to storing and processing sensor data in real-time in order to determine the location of the person. In section 8 we draw conclusions and expand on future research directions for the kleenk concept.

2 Problem statement

As we have seen in the previous section, there is a growing need for exposing the world's data in a structured format, as confirmed by industry giants and academia alike. There are a number of efforts trying to bridge this gap. Only to name a few:

- crowd-sourcing structured data from users; examples are Freebase and OpenStreetMap
- crowd-sourcing unstructured data from users, in a nearly-structured format; examples are Wikipedia and Facebook before the launching of Facebook Open Graph v2
- crawling / scraping data from unstructured data; this includes shopping, travel and news aggregators
- extracting entities and links from unstructured text using NLP (Natural Langauge Processing); one eloquent example of this is OpenCalais [4]

However, current efforts for structuring the web's data are mostly concentrated around describing entities and their properties, as shown in [2]. This is also the nature of the information usually found in web pages: in Wikipedia, each page is dedicated to one entity, and none to relations between entities. Also, most of the current approaches generate data through automated means, by parsing online data sources or exposing legacy databases in RDF format. This has two shortcomings: the only relations present in Linked Data are those detectable by a computer program (so only explicit relations can be detected), and also the decision of whether the data is correct or not is left to the computer. Moreover, the current quantity of available linked data in the largest such

[4] http://www.opencalais.com/

database was 4.7 billion RDF triples [2], compared to over 1 trillion of web pages in 2008 [5]. This tells us that the current approach of exposing the web's data in a structured form is not scalable enough when compared to the explosive growth of social content since the advent of Web 2.0 and the social web: tweets, statuses, blogs, wikis and forums are all very hard to understand for a computer program.

Therefore, it is our strongly held belief that general linked data would benefit from a social component, allowing creation and editing to be crowdsourced among enthusiasts. This way, people and machines can collaborate in order to generate more semi-structured content of better quality (because it has been validated by both parties).

We envision that people should be able to easily create links between any two online entities identifiable by a unique URI and to associate extra information to these links (note that online does not necessarily mean public in this case). The current process of creating linked data lacks transparency into the process, evaluation of the results by human beings, and human contributions. If people can create such links easily in the same format as machines do, an interesting validation phenomena can happen:

- humans can validate the content generated by machines
- machines can validate the content generated by humans by using different correlation and information retrieval algorithms

The key difference between our proposal and existing crowd-sourcing approaches is that the dual feedback cycle (human $\rightarrow$ machine $\rightarrow$ human) allows scaling the generation of content while maintaining good quality. We will showcase the importance of this feedback mechanism in both proposed scenarios. We have chosen the Ambient Intelligence in order to validate our model outside the classical applications of linked data for several reasons:

- sensor data is inherently linked and correlated; however, usually machine learning algorithms are used to detect correlations between this data
- such automatically detected correlations always need human feedback (as showcased by any online recommendation system: they all have a feedback mechanism such as vote up/down)
- sensor data can be easily accessed online by using identifiers; a recent development in this direction is http://www.pachube.com; humans should be able to link sensor data together as well, and machines can use this as training data

Our scenario is related to the Ambient Assisted Living[5] paradigm, the vision that elders should be assisted by equipment integrated seamlessly into their homes in order to prolong their independent healthy life as much as possible.

To sum up, the problem the **kleenk** is trying to solve is **scalable generation of linked data content by machines and humans collaborating together**.

3 Related works

Here, we have chosen a few relevant works that treat the same problems as mentioned previously: adding a social dimension to the web of data, using crowdsourcing to build

[5] http://googleblog.blogspot.com/2008/07/we-knew-web-was- big.html

up the web of data, or ways to open up linked data to the big public, which might be the only fighting chance of keeping up with the growth rate of online content.

ConceptWiki[6] tries to apply the wiki concept to linked data. It contains a list of concepts as specific as ”an unit of thought”. Any person with an account on the website can edit the concepts and there are two main sections on the website right now: WikiProteins (which contains information about proteins) and WikiPeople (which contains information about authors in the PubMed database). The WikiPeople exemplifies the importance of automatically generated data in bootstrapping a human community. However, due to slow human adoption, the machine-extracted data is not proving to be enough.

OpenStreetMap[8] is a success story in machine-generated data from public sources complemented by communities of enthusiasts around the world wishing to create an open and accessible geographical database. A notable design choice in this community was to only enable contributions from registered users, the opposite of Wikipedia. In addition to geographical coordinates of important landmarks, the database stores key-value pairs for each graphical coordinate.

Facebook Open Graph (v2) is a recent development of the social networking giant, allowing people to publish their online social activity as something very similar to RDF triples. They can now connect themselves to other entities by verbs like watch, read and listen, instead of the traditional like. Friends can afterwards rate and comment these actions, therefore this approach has also a very strong community evaluation component. However, this platform lacks in two respects: the first is generality, as it only connects people with entities, and through a pretty limited amount of actions (Facebook has to approve all new actions, giving it complete control over the ontology of predicates that appear); the second is aggregated visualisation capabilities, which is actually what makes the web of data interesting for the regular user: the ability to discover new content by navigating from content to content.

Pachube[9] is a platform for recording and distributing sensor data from around the world. It is based on a set of APIs that are used to store and retrieve sensory data from the selected feeds. Thousands of enthusiasts have connected sensor for temperature, wind speed and many others and are feeding data into this online sensor data brokerage platform. This platform, developed by a successful startup, exemplifies the importance of linked data in all sensor-related activities, including Ambient Intelligence.

SensorML[4] is an approved Open Geospatial Consortium standard. It provides models and encoding schemes (for example, XML) that can be used for describing process measurements in an interoperable way. It supports a wide range of sensors, on both stationary and dynamic platforms.

The Semantic Sensor Web[19] is an initiative aiming to integrate and adnotate data sources around the web in order to make up for a machine query-able and discoverable array of measurements that can be integrated with other data sources. Interestingly enough, this is similar to our proposed application in Ambient Intelligence, with the observation that it lacks the double feedback loop we’re proposing in order to scale the meta-data of the system and ensure its correctness.

[6] http://conceptwiki.org/

The fact that there are a number of projects solving the same problem as us, some even approached by internet giants or academia gives us the strength to believe that we are working on the right problem. However, our proposed solution is unique, in that it lets both humans and machines users easily create their own linked data, both in structured and semi-structured format, with a great potential for powerful visualisations, as we will shortly see in the next sections.

4 Working Scenarios

We will use two scenarios in this article: one is related to discovery of scientific articles by a young researchers, and one is related to tracking a single elder living at home, a scenario prototyped at University Politehnica of Bucharest[11].

4.1 Research Discovery

Rob is a PhD student in computer science and he is reading a lot of books and papers related to his subject, which is artificial intelligence. He is testing applications and algorithms to see how they perform in different scenarios. He would like to discuss his findings with other researchers to have their opinions and also make his results easily accessible. In addition to discussing with his friends, he publishes multiple articles, but he feels that the feedback is limited and delayed (at least a few months from an article submission to its publication). Rob also has some younger friends that study the same topic. Whenever they find a new interesting article or application they ask Rob about it: What's important about this article? How does this application relate to application Y? Rob could tell them to read his articles but that may take a lot more time and his friends may get confused and get lost in other information they might not need. He gives them the answer but he knows that there may be more students out there that would benefit from those answers. How can he structure this information, and where to put it, so that it can be easily found by all interested researchers?

4.2 Elder Tracking

Mary is 80 years old and has been retired for a decade. Her children have moved to the U.S. to found a successful start-up and her husband recently died of a heart-attack. Now she is alone but doesn't want to leave the house she has lived in for all her life. Also, she doesn't want to ask her children to come back to Europe and take care of her. So she is researching on the internet for a system that is able to monitor her vital signs non- intrusively and alert her physician in case of danger. The system should not require her to wear any equipment , and would ideally allow her to easily communicate with her children without having to use an actual computer. Also, it's very important that the system works well at night, since her vision is weakened and sometimes stumbles against the furniture. How can sensors together with linked data processed in a private cloud ensure that she is safe and sound? Also, her physician and children need to be able to look into the insights drawn by the system, visualize them in an intuitive way, and provide corrections as training data. (This is actually the double feedback loop we were mentioning earlier on).

5 Kleenks

In this section we will propose a solution to the problem stated in section 2. We start by considering a simplified model of the Web of Data which allows us to explain the role of our approach and how it fits in the existing landscape. We finish by identifying the main challenges for implementing our proposal.

5.1 Web of Data

We consider a simplified model for the Web of Data which consists of the following elements: *contents*, *entities*, *links*, *software agents*, *humans* and *ontologies*.

Contents represent any type of unstructured data such as text, images, sounds or videos and they may, or may not have, an URI that uniquely identifies them. Entities can represent anything such as places, people or articles and they are uniquely identified by URIs. Links connect two entities, have an associated type and they can represent any relation between entities. By software agent we understand any software application (desktop, web or mobile) that uses the Web of Data. Also, we consider that humans can access entities and links directly, making abstraction of the browser or any application in between. Finally, ontologies can be used by both humans and software agents to understand the links between entities.

5.2 A new perspective, a new type of links

Inspired by the explosion of content in Web 2.0, we believe that the Web of Data could also use an internal perspective in which links are first class citizens, and not just express a relationship between content. This would create an interesting phenomena, where links can be part of other links as either the target or the source. We believe that the Web of Data needs a new social, unstructured and collaborative dimension that would bring people, unstructured content, entities and links closer to each other (Figure 1).

We argue that this can be achieved through a new type of links, that we call *kleenks* (pronounced “clinks”), which are collaborative links created, evaluated and consumed by the users of the Web of Data in collaboration with machines (just like in the example ofOpenStreetMap, where some public data was automatically imported in the system and afterwards enhanced by users). A kleenk (Figure 2) is a directed connection and consists of the following (below the words “entity”, “content” and “link” have the meaning considered in the simplified model of the Web of Data from the beginning of this section):

1. **Source.** The source of a kleenk is an entity.
2. **Target.** The target of a kleenk is another entity.
3. **Type.** The type is a verb or expression that summarizes the link from the source to the target.
4. **Contents.** The contents represents the most important elements of a kleenk and they can have different roles:
   - *Description*. Descriptive contents can be simple text paragraphs, other media contents such as images and videos or even domain specific. They provide more details about the connection and they are added by the creator of a kleenk.

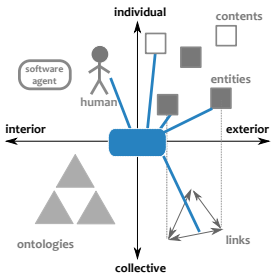

Fig. 1. Social, Unstructured, Collaborative dimension to the Web of Data

- *Feedback*. As with descriptive contents, feedback contents can take any form but they are added by other participants to the kleenk (other people or software agents).
- *Evaluation*. Evaluation contents must provide means to obtain quantitative data about the quality of a kleenk and they can take the form of ratings, like or thumb up/down buttons etc.

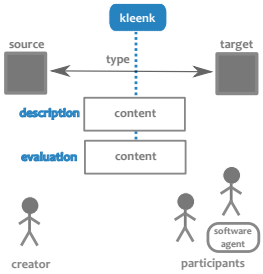

Fig. 2. Elements of a kleenk

Kleenks are collaborative links because new content can be added to a kleenk at any time by its creator or by another participant (be it human or machine).. Kleenks have an unstructured dimension because the content added to a kleenk is in an unstructured form. Finally, a kleenk is social because it provides a mechanism for users to express their position (like, agree, disagree, etc.) with respect to it. One important point is that this position can even be expressed by automated means: for example, a spam filtering module could vote with ”disagree” on a content created by a human.

The term “kleenk” is actually a short version for collaborative link with a slightly different spelling since the term “clink” has been used in other works such as Project Xanadu [7] and we wanted to avoid confusion.

[7] http://www.xanadu.com

Let's take an example. Rob, from our first working scenario, reads a paper X that talks about an efficient implementation of an algorithm described in another paper Y. He will create a kleenk from the article X to article Y with the type "efficient implementation of". Also, if the implementation is accessible on the internet he can also create a second kleenk from X to the implementation with the type "implemented here". As a description of the first kleenk he will provide a few details about what exactly makes the implementation efficient. Other researchers can express their opinion about the implementation directly on the kleenk, and comment for instance that the performance improvement is visible only on a particular class of input data. Other implementations can be kleenked to the same article X and the implementations can also be kleenked between them. Now, whenever an younger friend of Rob finds paper X he will quickly see the most important implementations of the algorithm and the relations with other important papers and they can continue their research without interruption.

Also, in order to get the community started on the usage of Kleenks (so that Rob can explore something that already exists), a large body of "A cites B" type of kleenks can be created automatically by scanning all the papers in the domain and importing them as kleenks. As the content of these kleenks, the paragraph in which the citation occurs can be used, so that people can rate on the relevance of the citation itself or not. This could also lead to interesting research in the scientometry field, where citations from a paper to another usually cannot be distinguished by importance.

5.3 Benefits and quality of kleenks

One main feature of kleenks is the ability to add unstructured content, in any form, to structured links. They are therefore compatible with both the existing RDF databases already available online and the semi-structured content added by users in social networking websites. As a side-note, in this section, we will mainly insist on the advantages of humans finally being able to contribute links between content in an easy way; the advantages of machines already doing that are well known.

Kleenks have multiple benefits for both the user and the Web of Data. First because kleenks are richer in content than simple links, this makes them important on their own. Up until now, in the Web of Data, it is rare that links are very important on their own but rather in sets that describe an entity or a topic. We believe that making each link important on its own will engage people more in creating meaningful links. Second, allowing people to create links with content will also facilitate the apparition of new links of high abstraction level that otherwise would have been impossible to extract automatically.

Allowing people to contribute to existing kleenks with new content is meant to make kleenks become more accurate and complete. However, as it has been seen in many projects such as Wikipedia and StackOverflow, an explicit evaluation system for user contributed content is necessary. The design of rating systems has been widely studied in computer science [12]. An overview of techniques that can be used to heuristically assess the relevance, quality and trustworthiness of data is given in [1]. This automatic rating system which also takes into account human intervention is what can close the feedback loop. An important note is that in the absence of human feedback (which can

be scarce on some deep topics, just like those found in scientific research), an automatic algorithm has to complement this lack of feedback.

Also, allowing social validation through mechanisms such as likes, agree/disagree or ratings allows important kleenks to step ahead of the less important ones guiding the users through what is important and what is less important. Of course, the best way to validate a content can differ from domain to domain and each platforms that uses kleenks is free to choose the method that is more suitable.

5.4 Challenges

Implementing a system based on kleenks, be it targeted to a specific domain or as a general platform, raises a few challenges that must be properly addressed in order to be successfully used.

Access to entities A kleenk, as an RDF triple, is a link that connects two entities and in addition it adds more content to the link. Letting regular users create such kleenks raises an important question: *"How will a user quickly select the entities he's interested in kleenking?"*.

The answer to this question depends on the type of platform: domain specific or general. In case of a domain specific platform it means that the user will kleenk entities he's working with. Usually these entities are already gathered in some databases and the kleenk platform only needs to integrate with these databases to provide quick search of the entities the user wants to kleenk.

On the other hand, a general platform is faced with a much more difficult question due to inherent ambiguities. If a user wants to use "Boston" as the source of a kleenk the platform has to decide whether it's about the city, the band or the basketball team. In this context we believe that semantic searches and large open databases such as DBpedia and Freebase will help in the disambiguation process.

Also, the user might want to kleenk things that don't yet have an URI and the platform must be able to create such URI's on the fly.

The ontologies for kleenks Even though kleenks contain unstructured content, their type, as with RDF links, will still be a predicate in an ontology, allowing computers to have at least a basic understanding of what a kleenk means and use them in new ways. However, allowing users to create any type of links between entities means that it is very hard to develop a comprehensive ontology from the start. A kleenk platform would have to provide a mechanism that would allow users to define ontologies, such as in Freebase, or it must integrate with platforms that allow users to build ontologies such as MyOntology.

Also, for kleenks automatically created by machines, these ontologies should serve as the authoritative sources of types of kleenks available. Humans create the ontologies, machines use types from the ontologies to create kleenks automatically, humans evaluate the kleenks by providing rating and feedback, and machines take as much feedback as they can (and understand) and make sure that in the future they create better kleenks. In this respect, kleenks can serve as a semi-structured way for humans to interact with machines.

Visualization and privacy Allowing users to create kleenks between any two entities has the potential of creating a very big number of kleenks. Users must be able to handle a big number of kleenks related to the entities that are of interest to them. Since kleenks form a graph structure, we can use visualisation techniques for graphs and create interactive ways of navigating the kleenks. We believe that since kleenks contain more content on the "edges" between the nodes, than just a simple predicate, more interactive and engaging visualizations can be built.

Since kleenks contain more content than simple RDF links and since most of this content will be based on the user's experience, the problem of the visibility of a kleenk must not be neglected. A user might want to create a kleenk between two entities and allow only a limited number of persons to see it. Also, kleenks can be used to collaboratively build some data (i.e. state of the art on a topic) which might, at least on its early stages, be visible only to a limited number of people. So, a kleenk platform must also provide proper mechanisms for kleenks' visibility.

Also, in our scenario regarding elder tracking, we are referring to kleenks extracted in real-time from sensor data and processed within a private cloud. This means that not only the kleenks themselves can be private or public, but also the kleenk platform themselves.

5.5 Modeling kleenks

In this section we will look at the theoretical and technical aspects of modeling kleenks using existing techniques in semantic web. We will first analyze different alternatives and motivate our chose for one of them. Finally, we will give an example of what a kleenk might look like.

Theoretical model Basically, the kleenk model could be seen as an extension of the RDF model with support for unstructured data. In the semantic web many extensions of the RDF model have been proposed during the last years. There are extensions dealing with temporal aspects [7], with imprecise information [13], provenance of data [6] or trust [18]. In [21] a general model based on annotation is proposed which generalizes most of the previous models.

All the above mentioned techniques are based on the named graph data model, a well known technique in semantic web to attach meta-information to a set of RDF triples. Even though these techniques could be applied to model kleenks, that would require that each kleenk has its own named graph (with its own URI), in order to associate the unstructured content with it.

A different technique, known under the name of RDF Reification, is described in the RDF specification [10]. This technique has well known limitations and weak points such as triple bloat and the fact that SPARQL queries need to be modified in order to work with reified statements. However, we believe that this techniques is the most suitable for modeling kleenks because a kleenk needs many different types of meta-information associated with it: creator, description content, feedback content (i.e. comments), evaluation content (i.e. ratings) and possibly other domain specific data.

Next, we provide an example kleenk by using reification:

```
kleenk:123  rdf:type          rdf:Statement
kleenk:123  rdf:subject       entity:234
kleenk:123  rdf:predicate     research:implements
kleenk:123  rdf:object        entity:444

kleenk:123  klnk:description  "Some text describing
 in detail the link between the source and
 the target. This can be used in order to motivate
 the chosen type for the kleenk."

kleenk:123  klnk:creator      user:33

kleenk:123  klnk:comment      comment:1023
comment:123 klnk-comment:txt  "The opinion of a
 user about the kleenk"
comment:123 klnk:creator     user:123

kleenk:123  klnk:rating       rating:2231
rating:2231 klnk-rating:txt   "4.5"^^xds:decimal
rating:2231 klnk:creator      user:344
```

We believe that modeling kleenks this way provides maximum interoperability with existing infrastructures and allows kleenks to be implemented on top of practically any triple store.

6 kleenk.com

6.1 Description

kleenk.com [8] is an online collaborative platform for linking scientific content. The project's motto is: "Smart-connecting scientific content". It is allows users to link scientific contents, revealing other relations than citations, such as:

- paper P1 implements the algorithm in paper P2 (relation: "implements algorithm in")
- diagram D1 is an explanation for the theory in paper P2 (relation: "explains the theory in")
- algorithms A1 and A2 solve the same problem (relation: "solves the same problem as")

This kind of relation is not easy to extract neither by an automated program, and nor by humans that are just starting their research in a certain area. That is why it is crucial that these two means complement each other. In Europe, the first year of a PhD program is usually dedicated to researching the state of the art, which consists of reading many

[8] http://app.kleenk.com

scientific contributions by other authors and creating mental links like those mentioned previously. Given the exploding number of scientific works, conferences and journals it is hard to keep up-to-date even for a scientific advisor, which makes the work of a starting researcher even harder. Kleenk actually solves this problem by allowing the community to create and visualise kleenks between the contents. Machines aid in this process by automatically extracting relationships between papers in the following ways:

- extract citations automatically between different papers - this will ensure a healthy amount of content for the community to get started and to suggest possible pairs of papers that might be interconnected in other ways, not just by way of citation
- extract n-grams automatically from different papers and create links between papers that speak about the same concepts, especially when the elements of the n-grams are very rare

While the machine-extracted content is surely limited in quality, making sure that machines aggresively generate kleenks of "good enough" quality that are afterwards reviewed by the community is key for this platform's adoption. Given the observation found during the development of OpenStreetMap[8], that up to 99.8% users of a wiki will only consume its content, making sure that the rest of 0.2% users are encouraged to easily contribute is key to the success of a wiki.

This platform is aimed at the following groups of persons:

- PhD students which need community guidance in order to read the most relevant and up-to-date materials related to their subject
- professional researchers who need to stay in touch with the vibrant scientific community's developments
- other people interested in quickly gaining an overview of a scientific domain

The platform allows the easy selection of content to kleenk from a number of sources by manually adding it, importing it from web pages (such as ACM or IEEE public pages of articles) and even by importing BibTeX bibliography files. Once all the content a user wants to kleenk is available in the platform, the user can start creating kleenks by selecting a source and destination content.

After they are created, kleenks can be shared with research fellows or made public, and grouped around meaningful ideas using tags. Every time a new content is created or updated, the interested users are notified using their personal news feed. Therefore, changes to a kleenk or any comment reach out across the entire community instantly.

Authors have the chance to kleenk their own papers to existing ones, and by subjecting these kleenks to the community scrutiny, the platform makes it possible for them to obtain early feedback for their ideas. In today's society, when the internet allows information to be propagated from one end of the world to another in seconds, the traditional peer review system is becoming more and more criticized due to the number of months passed from submitting the work to actual post-publication feedback from the scientific community. Our service aims to complement the quality and thoroughness of the peer review system with the opinion of the crowd. One important observation is that the opinion of the crowd is not necessarily misinformed, as proven lately by the tremendously successful service for programmers StackOverflow [9]. This website is a

[9] http://www.stackoverflow.com

collaborative question answering system, with world renown experts easily connecting and answering each others' questions. We think that the scientific community would benefit from a low-latency alternative to obtaining feedback for a piece of work.

6.2 Implementation of the theoretical framework

Having earlier detailed the kleenk model and characteristics, we will now underline which instantiation of the general principles was used in order to implement this knowledge sharing platform. First of all, in our particular case, the kleenk has the following elements:

- **the source, destination and type** - these are also present in the general model
- **the description** - this is specific to this pair of content, and represents a more detailed explanation of the type. It should be used in order to motivate the choice of type and to give more relevant results
- **comments** - since each kleenk has its own set of comments, these can be used in order to discuss the relevance of the link and to give extra information by anyone who can see it. These are similar to Wikipedia's talk pages, which are used by contributors to clarify informations in the main page
- **ratings** - together with ratings, these allow the community to evaluate the quality of a kleenk. In the visualisation, kleenks with better community score (which is computed from the ratings, number of comments, number of views and a number of other metrics) are displayed with a thicker connecting line, signifying a greater importance. Ideally, an user who is interesting in exploring the web of scientific articles will first navigate the most important kleenks.
- **privacy level** - as already mentioned in the general model, there should be a privacy setting associated with each kleenk. This allows users to first try out their own ideas in a personal incubator before promoting them to the whole community. In our implementation, there are 3 privacy levels: private (visible only to the owner), public (visible to anyone) and shared (visible to research fellows, which can be added through a dedicated page, given that they also agree).
- **tags** - each kleenk can be part of one or more tags. This is actually a mechanism for grouping tags related to the same idea or topic under a single name. For example, when writing this article, the authors created a "Kleenk Article" tag which contained the relevant bibliographic items and the kleenks between them.

The visualisation of the graph induced by the kleenks is done, as mentioned in the description of the general model, using consacrated layout methods. Specifically, in our case, we use an attraction-force model.

kleenk.com is a linked data application, conforming to Berner-Lee's vision of the future of the web. Contents, kleenks and tags all have persistent URIs that can be dereferenced in order to obtain linked data. One other interesting side-effect of this is that interesting scientific applications can emerge on top on the data contributed by the users to kleenk. For example, new scientometric indicators based on kleenks could be computed by a 3^{rd} party application.

6.3 Use case example

Obtaining feedback for a recently published article Alice is a fresh PhD student in Semantic Web, who is overwhelmed by the vast amount of publications on this topic. Being a first year student, she has to complete a document describing the state of the art by the end of the year. As a Facebook user, it's easy for her to create an account using one click on kleenk.com, since it features integration with Facebook's login service. Once logged in, she adds her colleagues who already have a Kleenk account as research fellows and now can easily see their shared tags. She studies the visualisations and grows to see a few important articles which are in the center of most tags, and starts reading them. Since she pays close attention to her news feed, she can easily see in real time what connections her colleagues are creating, and they all obtain quick feedback from their advisor, via comments and ratings.

Since she will be writing a survey article as well, she started creating a tag specifically for the bibliography of the article. First, the tag is private, since it is a work in progress and she doesn't want to share it with anyone. As the text of the article and the bibliography mature, she changes the visibility of the tag from private to shared, so that her research fellows can express their opinion on the connections she is making. After receiving the final approval for publication, she makes the tag public and includes the visualisation of the bibliography in a presentation for her department.

7 ElderMonitor - Real-time Kleenk Creation in a Private Cloud

We have implemented the concept of Kleenk in a prototype distributed tracking system of an elder, inside a lab of University Politehnica of Bucharest. This system uses a number of 9 Microsoft Kinect[20] and 20 Arduino [17] boards equipped with microphones, light and proximity sensors in order to track an elder living alone at home[11]. This system consumes raw sensor data and turns it into kleenks by using a data structure named the **Database of Trajectories**. This data structure is comprised of tuples (x, y, z, t) together with some associated information, presented as one or more kleenks, having this tuple as the source. (x, y, z) refers to the position of the person, while t refers to the timestamp when the system has determined that the person was at the position (x, y, z). Because the system can track more persons, but only one single person at a time, a first information attached to this tuple would be the identity of the person:

```
kleenk:123  rdf:type          rdf:Statement
kleenk:123  rdf:subject       andrei.ismail@cs.pub.ro
kleenk:123  rdf:predicate     ami:has_position
kleenk:123  rdf:object        (x,y,z,t)

kleenk:123  klnk:description "This has been automatically
                             determined by node 13, based
                             on the image 24."

kleenk:123  klnk:creator      node:13.
```

```
kleenk:123  klnk:comment      comment:1024
comment:123 klnk-comment:txt  "Caregiver A doesn't this this
                              represents a valid position
                              determination based on images
                              25 and 26"
comment:123 klnk:creator     user:123

kleenk:123  klnk:rating       rating:2231
rating:2231 klnk-rating:txt   "4.5"^^xds:decimal .
rating:2231 klnk:creator      user:344
```

Secondly, to such an (x, y, z, t) tuple, the Database of Trajectories allows association of proof based on which the machine algorithms have determined this position. Such proof consists of imagery, sound samples, and sensor positions and physical parameters. For example, the proof for one particular location might be comprised of:

- 6 sound samples from 3 different pairs of microphones and their physical positions (based on which the sound source position - the person - has been triangulated)
- RGB and depth imagery from a Kinect which confirms the same position by using skeleton data derived from the depth image in order to estimate the relative position of the person to the kinect

This can be represented using kleenks as following:

```
kleenk:124  rdf:type          rdf:Statement
kleenk:124  rdf:subject       kleenk:123
kleenk:124  rdf:predicate     ami:proof_for
kleenk:124  rdf:object        img:456

kleenk:124  klnk:description "Image with id 456 is a proof that kleenk with id 1
                             represents a correct information"

kleenk:124  klnk:creator      node:13.
```

Attaching proof to the user localization has several benefits:

- interesting queries can be formulated against the Database of Trajectories: "give me the images that serve as proof that the person Y was near the kitchen in the night of 27th of May". Given that the data can be exposed in an RDF format, it can be queried using the SPARQL[16] language.
- information can be spatially and temporally correlated more easily. For example, if a person enters the room but has a new haircut and some glasses, and the system doesn't recognize the person for the first 300 seconds, but recognizes it when the person answers the phone using the voice, all the imagery samples related to the anonymous-until-then person can be fed to the system as training samples. This will ensure that the system is permanently adapted to the changing physiognomy of its users.

– proof can be manually validated by both caregivers and experts helping to train the system, introducing the double-feedback mechanism mentioned before. It is assumed that the deployment of such a system includes a training phase in which the sensors are calibrated by specialized personnel and the algorithms' training data tweaked to the specific location. If needed, specialized personnel and caregivers can intervene and manually create kleenks in order to supply further training data to the system (except for the training data it is able to generate for itself).

One interesting aspect of the usage of kleenks in this case is that it naturally uses "recursive" kleenks (that have another kleenk as the source) in order to represent semi-structured data retrieved by machine learning algorithms. Specialized users are able to provide feedback to the system, which in turn grows more and more able to generate samples for itself by using the data structure. This is a perfect exemplification of how human and machine efforts can complement each other in order to reach unprecedented scale in producing linked data.

Just like in the kleenk.com use-case, privacy is of utmost importance. Kleenks created by the nodes of the system are private and processed within a private cloud running a private kleenk system. The kleenk system provides creation, discovery and feedback mechanisms for its users, with the sole purpose of scaling the quantity and quality of linked data.

8 Conclusions and Future Works

This article discusses the current context of the Web of Data, analyzes a few of its current limitations and focuses on the need to scale linked data generation with respect to the quantity of raw data available. We propose a new approach inspired by the success of Web 2.0 techniques such as wikis, blogs and social networks, that allows both machines and humans to combine efforts into creating more and better linked data.

The main contribution of this paper is the concept of *kleenk* which is a collaborative link that contains unstructured data in addition to the classical RDF predicate. We discuss the importance of allowing users to add unstructured data to the Web of Data and how this approach could lead to the creation of links which would otherwise be impossible to automatically parse from existing datasets. In the ElderMonitor use-case, we clearly exemplify how machines can benefit from this new linked data to parse even more linked data, leading to a spiraled evolution. What is most important about kleenks, they can be evaluated, manually or automatically (by software similar to spam filters), in order to control its influence in generating further linked data based on it.

We also identify the main challenges of a platform allowing users to create kleenks: access to entities, collaborative ontology creation, visualization of kleenks and privacy. These challenges have to be properly addressed for a system to succeed in applying kleenks. We introduce two already implemented platforms that have kleenks at their core:

– kleenk.com, a novel platform for spreading research results and obtaining feedback on insights extracted from them in real-time. Insights are represented here as

kleenks between papers, that can be grouped together by tagging them with common tags. Kleenks have privacy levels and users can view newly created kleenks by following a newsfeed similar to that of many social networks.
- ElderMonitor, a state-of-the-art distributed indoor tracking system for single elders living at home. This system has been prototyped at University Politehnica of Bucharest [11] and uses kleenks as the backbone of its high-level information extracted from raw sensor data.

In both systems, we highlight the importance of kleenks in generation of more linked data, and the importance of both the social and unstructured aspects of our proposal.

As an extension to the already implemented systems and to our proposed kleenk model we envision:

- creation of a special query language adapted to kleenks themselves. While SPARQL[16] can be an elegant and efficient way to query RDF data, kleenks can be represented as RDF but this is an alternative representation. SPARQL lacks specialized primitives in order to touch on the unstructured data and social ratings and comments attached to kleenks
- defining scientometric metrics and algorithms to compute influence scores of authors based on new criteria after the kleenk.com platform gains enough traction
- implementation of algorithms that are able to extract generic knowledge from the Database of Trajectories in the form of public kleenks that can be reused from one deployment of ElderMonitor to another

9 Acknowledgments

Development of the ElderMonitor system has been sponsored by project ERRIC - Empowering Romanian Research on Intelligent Information Technologies/FP7- REGPOT-2010-1, ID: 264207. We are greatful to the scientific communities at University Politehnica of Bucharest and University of Montpellier for guidance and continuous exchange of constructive feedback.

References

1. Christian Bizer and Richard Cyganiak. Quality-driven information filtering using the WIQA policy framework. *Web Semant.*, 7:1–10, January 2009.
2. Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia - A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, September 2009.
3. K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.
4. Mike Botts, George Percivall, Carl Reed, and John Davidson. Ogc® sensor web enablement: Overview and high level architecture. *GeoSensor networks*, pages 175–190, 2008.

5. Ricardo Costa, Davide Carneiro, Paulo Novais, Luís Lima, José Machado, Alberto Marques, and José Neves. Ambient assisted living. In *3rd Symposium of Ubiquitous Computing and Ambient Intelligence 2008*, pages 86–94. Springer, 2009.
6. Renata Dividino, Sergej Sizov, Steffen Staab, and Bernhard Schueler. Querying for provenance, trust, uncertainty and other meta knowledge in RDF. *Web Semant.*, 7:204–219, September 2009.
7. Claudio Gutierrez, Carlos A. Hurtado, and Alejandro Vaisman. Introducing Time into RDF. *IEEE Trans. on Knowl. and Data Eng.*, 19:207–218, February 2007.
8. M. Haklay and P. Weber. Openstreetmap: User-generated street maps. *Pervasive Computing, IEEE*, 7(4):12–18, 2008.
9. O Haque. Pachube. *Online at http://www. pachube. com*, 2004.
10. http://www.w3.org/TR/rdf primer/. RDF Primer, February 2004.
11. Andrei-Adnan Ismail and Adina Magda Florea. Multimodal indoor tracking of a single elder in an aal environment. In *ISAMI 2013 - 4th International Symposium on Ambient Intelligence*.
12. Audun Jøsang, Roslan Ismail, and Colin Boyd. A survey of Trust and Reputation Systems for Online Service Provision. *Decis. Support Syst.*, 43:618–644, March 2007.
13. Mauro Mazzieri and Aldo Franco Dragoni. Uncertainty reasoning for the semantic web i. chapter A Fuzzy Semantics for the Resource Description Framework, pages 244–261. Springer-Verlag, Berlin, Heidelberg, 2008.
14. Marc Najork and Allan Heydon. High-performance web crawling. *Handbook of massive data sets*, 4:25, 2002.
15. Charles Petrie and Christoph Bussler. Service agents and virtual enterprises: A survey. *IEEE Internet Computing*, 7(4):68–78, 2003.
16. Eric PrudHommeaux, Andy Seaborne, et al. Sparql query language for rdf. *W3C recommendation*, 15, 2008.
17. E. Ramos. Arduino basics. *Arduino and Kinect Projects*, pages 1–22, 2012.
18. Simon Schenk. On the Semantics of Trust and Caching in the Semantic Web. In *Proceedings of the 7th International Conference on The Semantic Web*, ISWC '08, pages 533–549, Berlin, Heidelberg, 2008. Springer-Verlag.
19. Amit Sheth, Cory Henson, and Satya S Sahoo. Semantic sensor web. *Internet Computing, IEEE*, 12(4):78–83, 2008.
20. Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1297–1304. IEEE, 2011.
21. Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia. A General Framework for Representing, Reasoning and Querying with Annotated Semantic Web Data. *Elements*, pages 1437–1442, 2011.