International Journal of Human – Computer Studies 103 (2017) 63–76

Contents lists available at ScienceDirect

Int. J. Human–Computer Studies

journal homepage: www.elsevier.com/locate/ijhcs

Combining information on structure and content to automatically annotate natural science spreadsheets

Martine de Vos[a,*], Jan Wielemaker[a], Hajo Rijgersberg[b], Guus Schreiber[a], Bob Wielinga[a,1], Jan Top[a,b]

[a] Computer Science, Network Institute, VU University Amsterdam, De Boelelaan 1081, 1081HV Amsterdam, The Netherlands
[b] Wageningen University and Research Centre, Food and Biobased Research, P.O. Box 17, NL-6700 AA Wageningen, The Netherlands

ARTICLE INFO

Keywords:
Domain Model
Implicit knowledge
Methodology
Spreadsheets
Units of measure
Vocabularies

ABSTRACT

In this paper we propose several approaches for automatic annotation of natural science spreadsheets using a combination of structural properties of the tables and external vocabularies. During the design process of their spreadsheets, domain scientists implicitly include their domain model in the content and structure of the spreadsheet tables. However, this domain model is essential to unambiguously interpret the spreadsheet data. The overall objective of this research is to make the underlying domain model explicit, to facilitate evaluation and reuse of these data.

We present our annotation approaches by describing five structural properties of natural science spreadsheets, that may pose challenges to annotation, and at the same time, provide additional information on the content. For example, the main property we describe is that, within a spreadsheet table, semantically related terms are grouped in rectangular blocks. For each of the five structural properties we suggest an annotation approach, that combines heuristics on the property with knowledge from external vocabularies. We evaluate our approaches in a case study, with a set of existing natural science spreadsheets, by comparing the annotation results with a baseline based on purely lexical matching.

Our case study results show that combining information on structural properties of spreadsheet tables with lexical matching to external vocabularies results in higher precision and recall of annotation of individual terms. We show that the semantic characterization of blocks of spreadsheet terms is an essential first step in the identification of relations between cells in a table. As such, the annotation approaches presented in this study provide the basic information that is needed to construct the domain model of scientific spreadsheets.

1. Introduction

In this article we propose several approaches for automatic annotation of natural science spreadsheets using a combination of structural properties of the tables and external vocabularies.

Scientists from domains other than computer science typically use spreadsheets to store and manipulate data collected during research (Chen and Cafarella, 2013; Maguire et al., 2013). This is especially true for scientists from the domain of natural science, e.g. biology, medical science and physics (Wolstencroft et al., 2011; Rijgersberg et al., 2011; Rayner et al., 2006; McDonald et al., 2012). While designing their spreadsheets they inevitably make choices with respect to the entities and processes to be included, and the way in which these are organized in tables. The domain model of the scientists is implicitly reflected in the content and structure of the spreadsheet tables. Domain scientists may give a textual explanation about their ideas and choices in their publications, but the actual domain model remains hidden. As this domain model is essential to understand the meaning and context of the spreadsheet data, it is currently hard to unambiguously interpret these data for people other than the original developers. The overall objective of our research is to explore new ways to make the underlying domain model of scientific spreadsheet data explicit.

An important cause of the limited semantic specification of scientific spreadsheet data is the free format of spreadsheets, which gives researchers a great deal of freedom in how they enter and manipulate their data. Moreover, domain scientists create their spreadsheets for their own use to analyze new problems and improve domain understanding (Segal and Morris, 2008). They may therefore be less interested in how the data is understood by peers who want to reuse or review such data. Semantic markup tools like RightField

[*] Corresponding author.
E-mail addresses: martine.de.vos@vu.nl (M. de Vos), j.wielemaker@vu.nl (J. Wielemaker), hajo.rijgersberg@wur.nl (H. Rijgersberg), guus.schreiber@vu.nl (G. Schreiber), j.l.top@vu.nl (J. Top).

http://dx.doi.org/10.1016/j.ijhcs.2017.02.006
Received 29 March 2016; Received in revised form 8 February 2017; Accepted 17 February 2017
Available online 20 February 2017

(Wolstencroft et al., 2011), OntoMaton (Maguire et al., 2013), and spreadsheet based formats like MAGE-Tab (Rayner et al., 2006) and ISA-Tab (Sansone et al., 2012) may be used to develop templates for domain scientists to enter their data. These templates encourage standardization of spreadsheet data and as such facilitate interpretation and reuse. But setting up these templates requires time and technical knowledge, and the freedom for domain scientists to enter their data is constrained. We argue that automatic annotation of scientific spreadsheet data with concepts from the underlying domain model could facilitate the interpretation, evaluation and reuse of scientific spreadsheets.

This paper extends previous work (De Vos et al., 2012) by proposing a set of approaches for automatic annotation of spreadsheet tables. Basic assumption in these approaches is that the underlying domain knowledge can be represented as the collection of concepts in a set of spreadsheet tables, and how these are related to each other. We propose to extract these concepts and interrelations by iteratively combining information on the structural properties, and information on the content of scientific spreadsheet tables. The information on the structure is expressed as heuristics and rules on table layout, and grammar rules on quantities and units of measure. The content information consists of two external vocabularies, i.e., one vocabulary related to the scientific domain of the spreadsheets and one vocabulary on units of measure, quantities and related concepts.

The focus of this paper is on spreadsheets that are used in natural science; scientists working in the domain of natural science are hereafter referred to as domain scientists. Natural science spreadsheets often represent laboratory or field measurements, and typically consist of numerical data, quantities and units of measure (Assem et al., 2010), and information on the associated objects and events. As such, these spreadsheets typically possess structural properties that pose a challenge for automatic interpretation and annotation. These properties are (1) the presence of blocks within a table, (2) the presence of units of measure, (3) the presence of quantities, (4) the presence of different type of blocks, in terms of content position, and role in the table, and (5) the grouping of similar domain concepts. For each of these properties we suggest an annotation approach, that may help to overcome the challenges. In a case study with a set of existing natural science spreadsheets, we evaluate each approach by comparing the annotation results with a baseline based on purely lexical matching.

In this paper we make the following contributions: (i) we provide the algorithms, and heuristics that are used in our approaches (Section Appendix A) for automatic semantic annotation of spreadsheet tables, (ii) in our case study experiment (Section 4) we prove that including information on structural properties of spreadsheet tables results in higher precision and recall of annotation of individual terms (Sections 5.1–5.3) and (iii) we show that the semantic characterization of blocks (Section 3.5) is an essential first step in the identification of relations between cells in a table (Section 5.3).

2. Related work

Many studies have focussed on improving the accessibility of tabular data to facilitate search and integration. We observe two types of approaches. One approach is to convert tabular data into formats that are more suitable for automatic processing. There are several systems that convert tabular data into OWL ontologies, like, Anzo suite[1] and Mapping Master (Connor et al., 2010) or other semantic web formats, e.g., RDF123 (Han et al., 2008), XLWrap (Langegger and Wolfram, 2009) and TabLinker (Meroño-Peñuela et al., 2013). Other studies have developed systems to convert tabular data into XML (Shu et al., 2015), or relational data (Chen and Cafarella, 2013; Cafarella et al., 2008). MAGE-Tab (Rayner et al., 2006), ISA-Tab (Sansone et al., 2012), and BIOM (Rayner et al., 2006; McDonald et al., 2012) are tabular formats that use an underlying data model with relevant metadata from scientific experiments. These formats can be used to either directly enter data, or as a template for mapping other spreadsheet files onto one structure. The ISA-Tab format can also be converted to RDF which enables annotation with concepts from external ontologies (González-Beltrán et al., 2014).

Another approach is to annotate tabular data with concepts from vocabularies. Some tools (Limaye et al., 2010; Mulwad et al., 2012) use existing generic ontologies, like Yago, DBPedia, while other tools (Wolstencroft et al., 2011; Maguire et al., 2013), use existing domain ontologies for semantic markup. Some approaches develop their own ontology, either manually (Shu et al., 2015) or by extracting concepts and relations from the web (Venetis et al., 2011), to annotate tabular data.

All the abovementioned studies acknowledge that a correct interpretation of tabular data is essential for conversion or annotation. In order to derive a correct interpretation, the studies use different strategies to infer the semantics from tabular data and dissolve ambiguities. Some of these strategies rely on manual mapping specifications constructed by users (Shu et al., 2015) or human analysts with sufficient knowledge of applying semantic web techniques (Han et al., 2008; Langegger and Wolfram, 2009; Wolstencroft et al., 2011; Meroño-Peñuela et al., 2013; Connor et al., 2010). Others compare their tabular data with large collections of example data, e.g., large vocabularies like Yago or DBPedia, or generic databases extracted from the Web, and rely on probabilistic reasoning methods to find the best suitable annotation or interpretation for table cells and columns (Cafarella et al., 2008; Venetis et al., 2011; Limaye et al., 2010; Mulwad et al., 2012). And, many studies use knowledge on the structural properties of a table to derive a correct interpretation of its content. Several studies created a library on commonly used layout patterns in tabular data (Garcia-silva et al., 2008; Hermans et al., 2010; Rocha Bernardo et al., 2013). Abraham and Erwig (2006) developed a framework to automatically classify roles of cells in a table based on the spatial layout of a spreadsheet. Van Assem and colleagues Assem et al. (2010) introduced disambiguation strategies for units of measure and quantities (Assem et al., 2010) based on the way these are notated in table cells. And Chen and Cafarella (2013) use heuristics and rules on spreadsheet layout and implicit metadata structure to automatically extract relational data from spreadsheets.

The related work described in this section provides multiple approaches that can be used to translate natural science spreadsheets into more appropriate representations. The automatic interpretation and annotation of the content, however, remains an issue, as many of the abovementioned approaches are not suitable for natural science spreadsheets. The use of probabilistic reasoning methods to annotate tabular data requires a large collection of example data. The content of natural science spreadsheets is too domain specific to be annotated with entities from commonly used vocabularies or generic databases extracted from the Web. Furthermore, probabilistic reasoning methods are mainly used to annotate textual values in tables. As natural science tables for a large part consist of numerical values, the success of probabilistic reasoning methods to annotate these tables is probably limited. On the other hand, the tables in natural science spreadsheets may be more structured, and demonstrate less variety in their layout than arbitrary tables in documents or on the web. Although natural science spreadsheet tables are designed for human consumption, and contain implicit information, we argue that the structural properties of these tables could be very useful to inform automatic interpretation and annotation. As such the approaches described by Mittermeir and Clermont (2002), Hipfl (2004), Abraham and Erwig (2006), Assem et al. (2010), Chen and Cafarella (2013), Garcia-silva et al. (2008), Hermans et al. (2010), and Rocha Bernardo et al. (2013) offer a useful basis to build upon.

[1] Anzo suite,http://www.cambridgesemantics.com/)

Table 1
Characteristic properties of quantitative natural science spreadsheet tables and approaches for automatic annotation. Rules, heuristics, and algorithms, are described in the appendix (Section Appendix A).

	property	annotation approach
1	Observational data and associated context are located in rectangular blocks within the table	Identify table body and context blocks by recognizing string and float blocks (Table 11, Algorithm 2)
2	Units of measure are common, and represented as symbols or short strings	Annotate unit cells by applying unit grammar + exact string matching (Table 12, Algorithm 3)
3	Quantities are common, but only implicitly mentioned through the associated unit	Annotate quantity cells by applying quantity grammar (Table 13, Algorithm 4)+ lexical matching or deduction from unit of measure
4a	The blocks in a table differ in terms of content, position and role	Identify Unit, Quantity and Phenomenon blocks by applying unit-quantity grammar and heuristics on table design (Table 14, Algorithms 5, 6)
b	These blocks contain semantically related individual terms	Annotate terms by applying vocabulary selection: OM for unit and quantity terms, and domain vocabulary for phenomenon and quantity terms (Table 15, Algorithm 7)
5a	Terms referring to similar domain concepts are grouped	Annotate phenomenon blocks with a common denominator, i.e. block term (Table 16, Algorithm 8)
b	The context of domain terms is essential for correct interpretation	Select per domain term the annotations that are related to the corresponding block term (Table 16, Algorithm 8)

3. Method

As mentioned in the introduction section, tables in natural science spreadsheets possess structural properties that may pose challenges to automatic interpretation and annotation, and at the same time, provide additional information on the content. In this section we describe five such structural properties of these tables (Table 1). For each of these five properties we propose an approach, that may improve the results of automatic annotation and interpretation. We consider lexical matching with concepts from a domain vocabulary as a baseline method for automatic annotation of natural science spreadsheet tables. Our annotation approaches for the five properties are additions to this baseline method The general idea is described in the next subsection, the individual properties and corresponding approaches are described in more detail in the following subsections.

3.1. Basic principles

In the development of their spreadsheet tables, domain scientists obviously include domain knowledge in the content of the table cells. Besides that, they apply implicit rules that shape the design of their tables, e.g. they group semantically related terms in blocks. In each of our annotation approaches (Table 1) we combine these two types of information.

For content identification we use concepts from selected vocabularies. Natural science spreadsheets often represent laboratory or field measurements, and therefore typically consist of numerical data, quantities and units of measure (Assem et al., 2010), and information on the associated objects and events. The vocabularies used for annotating content therefore consist of at least one vocabulary that covers the domain of the considered spreadsheets, and a dedicated vocabulary on quantities and units (see Section 4.2). Both vocabularies contain labeled concepts that are part of a graph structure.

For structural information we use classification rules and heuristics on the properties of natural science spreadsheet tables. The structural properties of natural science spreadsheets, and corresponding rules and heuristics, are derived from literature (Abraham and Erwig, 2006; Assem et al., 2010; Chen and Cafarella, 2013), and manual analysis of spreadsheet tables from the domain of natural science and engineering (Fisher and Rothermel, 2005; De Vos et al., 2012). The rules and heuristics, and the way these are implemented in algorithms, are described in the appendix of this paper (Appendix A section). All algorithms are developed using SWI prolog[2] and are publicly available.[3]

Although our annotation approaches are presented separately, these are not independent from each other. The approaches do not only build upon each other, but are also combined in an iterative way (Fig. 1). For example, the annotation of unit and quantity blocks builds upon the recognition of individual unit and quantity terms, while the annotation of individual unit and quantity terms can be more precise by annotating only the terms that are present in unit and quantity blocks.

3.2. Property 1: blocks within a table

The first property of natural science spreadsheet tables, or spreadsheet tables in general, is that data are organized in rectangular blocks. More specifically, string data and float values are grouped in separate blocks, which are placed together to make up a table. The float block

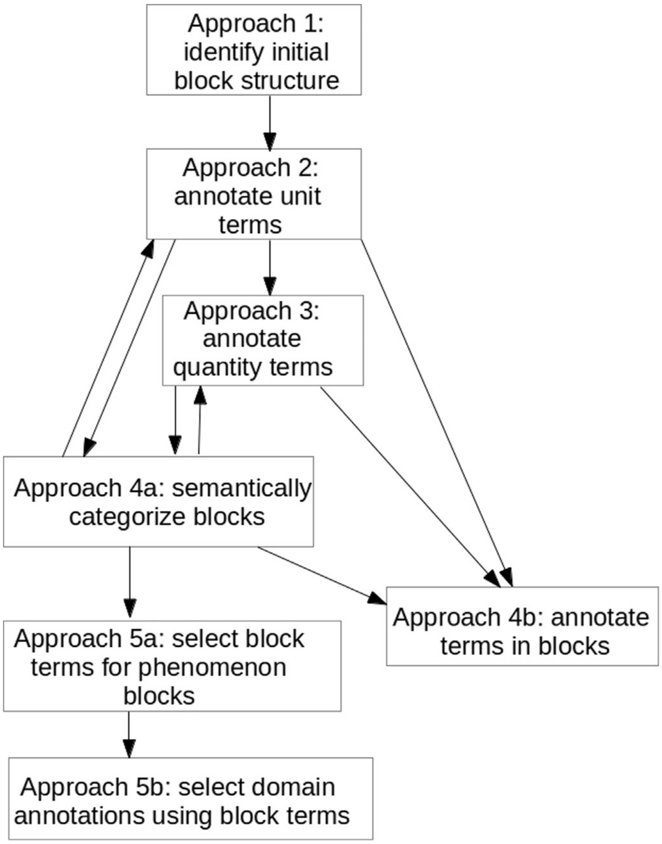

Approach 1: identify initial block structure
Approach 2: annotate unit terms
Approach 3: annotate quantity terms
Approach 4a: semantically categorize blocks
Approach 4b: annotate terms in blocks
Approach 5a: select block terms for phenomenon blocks
Approach 5b: select domain annotations using block terms

Fig. 1. Dependencies between the annotation approaches presented in this study (NB do not read this diagram as a workflow).

[2] SWI-prolog,http://www.swi-prolog.org/
[3] GitHub repository,https://github.com/MartineDeVos/Spreadsheets/tree/work/annotation

contains the values of observations or measurements in a table, and the surrounding string blocks describe the context of these measurements (Abraham and Erwig, 2006; Chen and Cafarella, 2013).

We have developed an approach to recognize string and float blocks in a spreadsheet (Table 11, algorithm 2). We annote the float blocks as table bodies, and the surrounding string blocks as context blocks.

Algorithm 1. Lexical matching.

```
for A string term in a cell do
    Break term into tokens (i.e., words), ignore stopwords, numbers and punctuations.
    for Each token do
        for Each word in the vocabulary that (partly) matches the token do
            if that word is a concept represents a label or a symbol and isub distance between token and vocabulary word is $\geq 0.85$ then
                Corresponding domain vocabulary concept is used for annotation
            else
                pass
            end if
        end for
    end for
end for
```

This first property of natural science spreadsheet tables has a special status, as we consider it a starting point in our approach, and a prerequisite for the next steps. We assume that the annotation of body and context blocks is straightforward and we do not evaluate this step separately in the case study experiment.

3.3. Property 2: units of measure

Terms representing units of measure in natural science spreadsheet tables often have a typical structure. They consist of very short strings containing one or more symbols, optional brackets and slashes, and optional some free text. For example, the term “J/kg” represents the unit “joule per kilogram”, and “m” represents the unit “metre”. Straightforward lexical matching methods (e.g., Algorithm 1) are not suitable for recognition and annotation of very short strings, as the number of matches is often large, but the corresponding isub distances (Stoilos et al., 2005), i.e., measure of the quality of these matches, are low. For example, the term “ha”, may be matched to the unit “hectare”, but also to “hectoampere”, and many more concepts, all with the same low isub distance of 0.1.

Algorithm 2. Annotation of table bodies and context blocks.

```
for Each spreadsheet do
    Identify blocks (rectangles $\geq 2$ cells) with string terms and float terms
    Sort float blocks by size
    repeat
        Build table body
        for the largest float block do
            repeat
                build a larger block by merging with a neighbouring float block
                empty cells may be included, string cells not
            until no neighbouring blocks are available
            annotate resulting block as table body
            Retract all original blocks that are included in table body
        end for
    until No float blocks are left
    Sort string blocks by size
    repeat
        Build context blocks
        repeat
            build a larger block by merging with a neighbouring string block
            empty cells may be included, float cells not
        until no neighbouring blocks are available
        if resulting block is aligned with a table body then
            force width (top-aligned) or height (left/right aligned) to be the same as that table body
            Annotate resulting block as context block
            Retract all original blocks that are included in context block
        else
            pass
        end if
    until No string blocks are left
end for
```

We have developed a set of grammar rules to recognize terms that represent units of measure (Table 12, Algorithm 3). These grammar rules recognize the typical structure of unit terms. And as an additional check, at least one symbol in the term should exactly match with a unit symbol from a dedicated vocabulary. In this study we use the OM Ontology for units of Measure and related concepts (Rijgersberg et al., 2011).

Algorithm 3. Recognition and annotation of unit terms.

```
for string $<11$ chars do
    if string structure complies with
    “optional bracket open + unit symbol + optional separator + optional free text + optional bracket close” then
        for “unit symbol + optional separator + optional free text” do
            repeat
                match unit symbol with labels, symbols or descriptions of unit concepts in OM Vocabulary
                repeat
                    remove one character from end of the string
                    match unit symbol with labels, symbols or descriptions of unit concepts in OM Vocabulary
                until end of string
                remove characters from start of the string until separator
            until end of string
        end for
        if match of unit symbol with OM Vocabulary is found then
            Annotate string as unit term, corresponding unit concept is used for term annotation (Algorithm 7)
        else
            string is not a unit term
        end if
    else
        string is not a unit term
    end if
end for
```

3.4. Property 3: quantities

Quantities in natural science spreadsheets are often represented as a string, that contains an associated unit of measure, that is either enclosed in brackets within the term or present in a neighbouring cell. The domain scientists developing these spreadsheets often omit the quantity concept from the quantity term and only present the related domain concept. For example, a quantity term representing the mass of applied nitrate is written as "nitrate (kg)" and a quantity term representing the mass fraction of clay is written as "clay [%]".

Our grammar rules on quantities (Table 13, Algorithm 4) recognize the typical structure of quantity terms. We use lexical matching to find a match with a quantity concept in the OM vocabulary. If no match is found, we analyze the associated unit of measure. The OM vocabulary contains information on which units are commonly used by certain quantities. We use this information to deduce a suitable quantity concept for a quantity term.

Algorithm 4. Recognition and annotation of quantity terms.

```
if string structure complies with
freetext + bracketopen + unitterm (Algorithm 3) + bracketclose
then
    string is a quantity term
    if freetext matches with label of a quantity concept in OM
    Vocabulary then
        corresponding quantity concept is annotation
    else if unit concept annotation of unitterm is commonly used
    by a quantity concept in OM Vocabulary and this quantity
    concept has a common domain of application then
        corresponding quantity concept is used for annotation
        (Algorithm 7)
    else
        no annotation can be found for quantity term
    end if
else
    string is not a quantity term
end if
```

3.5. Property 4: block typology

In the development of their spreadsheet tables, domain scientists apply implicit rules that shape both the content and the design of their tables. They typically group cells that are semantically related (Mittermeir and Clermont, 2002) and use structure and layout features to distinguish between these groups (Hipfl, 2004; Chen and Cafarella, 2013). We assume that these groups of cells, i.e. blocks, are not only different in terms of their content and position, but also in terms of their role in the table. More specifically, we assume that four different types of blocks can be distinguished in scientific spreadsheet tables: (1) blocks containing measurements, (2) units of measure, (3) quantities, and (4) objects and events (De Vos et al., 2012).

We have developed an approach to semantically categorize the blocks in a natural science spreadsheet (Table 14, Algorithms 5, 6), which is an extension of the initial block structure developed in Section 3.2. The table body and context blocks are further categorized using four main concepts from the OM vocabulary: (1) Measure, (2) Unit of measure, (3) Quantity and (4) Phenomenon. The table bodies are all annotated as Measure blocks (Fig. 2). Subsequently, the unit blocks within the context blocks are recognized, annotated and separated. We define unit blocks as rows or columns within the context blocks that contain >30% unit cells. In order to avoid non-relevant matches with very specific, and rare, units from the OM vocabulary, the unit cells should represent commonly used units of measure, i.e., units that belong to the "om:commonApplicationArea". The next step is to recognize and annotate quantity blocks. These blocks are either aligned with the table body and unit block, or contain >30% unit cells. The remaining context blocks are annotated as phenomenon blocks.

Algorithm 5. Annotation of unit blocks.

```
for Each context block do
    if column/row in context block
    and # cells with unit terms (Algorithm 3) $\geq 30\%$
    and # cells with unit terms (Algorithm 3) $\geq$ # cells with
    domain terms then
        column or row is unit slice
    else
        pass
    end if
    find all unit slices
    for largest unit slice do
        Annotate slice as unit block
        Subtract slice from original block
    end for
end for
```

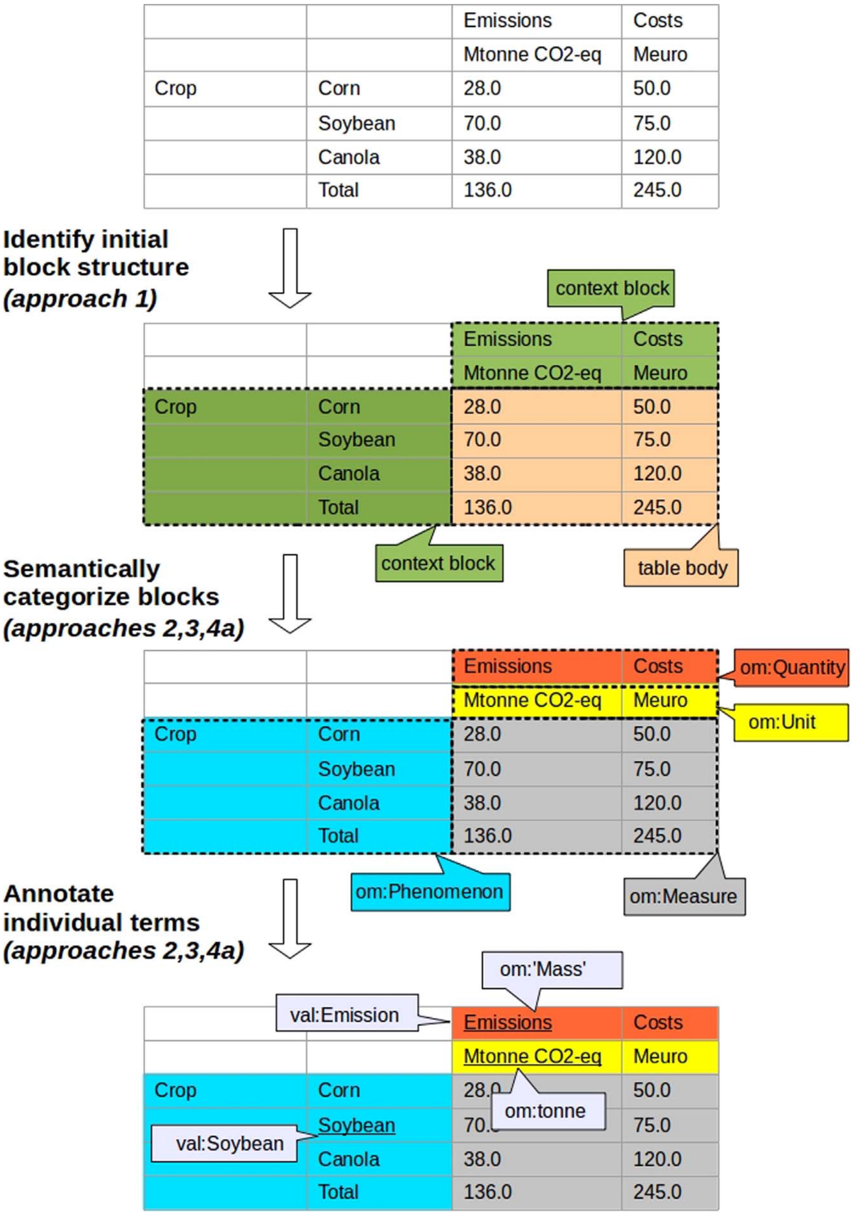
Algorithm 6. Annotation of quantity and phenomenon blocks.

```
for Each context block do
    for Each term in context block do
        if term meets grammar (Algorithm 4) or term can be
        matched with OM quantity concept then
            terms is a quantity term
        else
            term is not a quantity term
        end if
    end for
    if # cells with quantity terms $\geq 30\%$
    or context block is horizontally or vertically aligned with
    unit block and table body then
        Annotate context block as quantity block
    else
        Annotate context block as phenomenon block
    end if
end for
```

When a block is annotated with one of the abovementioned OM concepts, this annotation also applies to all the individual terms in that block. Furthermore, the semantic category of a block determines which vocabulary and concept class are selected for the annotation of the individual terms (Table 15, Algorithm 7). For some of the individual unit and quantity terms in a natural science spreadsheet our grammar rules may not apply. When these terms are located in a unit or quantity block, these terms will still be recognized and annotated as unit and quantity terms.

Algorithm 7. Annotation of individual terms.

```
for Each term in a unit block do
    for Each matching unit concept (Algorithm 3)from OM
    vocabulary do
        Annotate term with unit concept
    end for
end for
for Each term in a quantity block do
    for Each matching quantity concept (Algorithm 4) from OM
    vocabulary do
        Annotate term with quantity concept
    end for
end for
for Each term in a phenomenon block do
    for Each matching domain concept (Algorithm 1) from
    domain vocabulary do
        Annotate term with phenomenon concept
    end for
```



		Emissions	Costs
		Mtonne CO2-eq	Meuro
Crop	Corn	28.0	50.0
	Soybean	70.0	75.0
	Canola	38.0	120.0
	Total	136.0	245.0

Identify initial block structure (approach 1)
context block
context block
table body
Semantically categorize blocks (approaches 2,3,4a)
om:Quantity
om:Unit
om:Phenomenon
om:Measure
Annotate individual terms (approaches 2,3,4a)
om:'Mass'
val:Emission
om:tonne
val:Soybean

Fig. 2. Schematic overview of the annotation approaches presented in this study.

```
end for
```

3.6. Property 5: grouping of similar domain concepts

Additional to the grouping of semantically related terms in blocks, we observe another way of grouping terms. The last property we distinguish in natural science spreadsheet tables is that terms describing similar domain concepts are typically grouped (Chen and Cafarella, 2013). Furthermore, these terms often contain incomplete or ambiguous references to particular domain concepts. It is up to the reader to deduce the exact definitions of these domain concepts, for example, by considering the context of the table and the spreadsheet (Cafarella et al., 2008), or by using background domain knowledge. We assume that the grouping of terms describing similar domain concepts occurs especially in phenomenon blocks, as these blocks typically describe the domain concepts associated with the numbers and quantities in the table.

In our annotation approach (Table 16, Algorithm 8), we build upon the semantic characterization of blocks described in the previous section (Section 3.5). We assume that it is possible to describe a group of terms in a phenomenon block with a common denominator, i.e., one single, often higher level, domain concept. Our annotation approach collects all domain concepts that are hierarchically, i.e., “skos:broader”, or property-wise, i.e.,“skos:related”, related to the annotations of individual terms in a phenomenon block. The domain concept that is related to most terms in the block is selected as common descriptor of that block, the so called block term (Fig. 3). Such a block term may provide the context that is needed to select the right annotations for terms in a phenomenon block. It may also be used to provide information on terms in phenomenon blocks that did not get an annotation on an individual level.

Algorithm 8. Annotation of block terms.

```
for Each phenomenon block do
    for Each individual term in the phenomenon block do
        Find all skos: related or skos: broader domain concepts that are associated with the annotations of that term
    end for
```

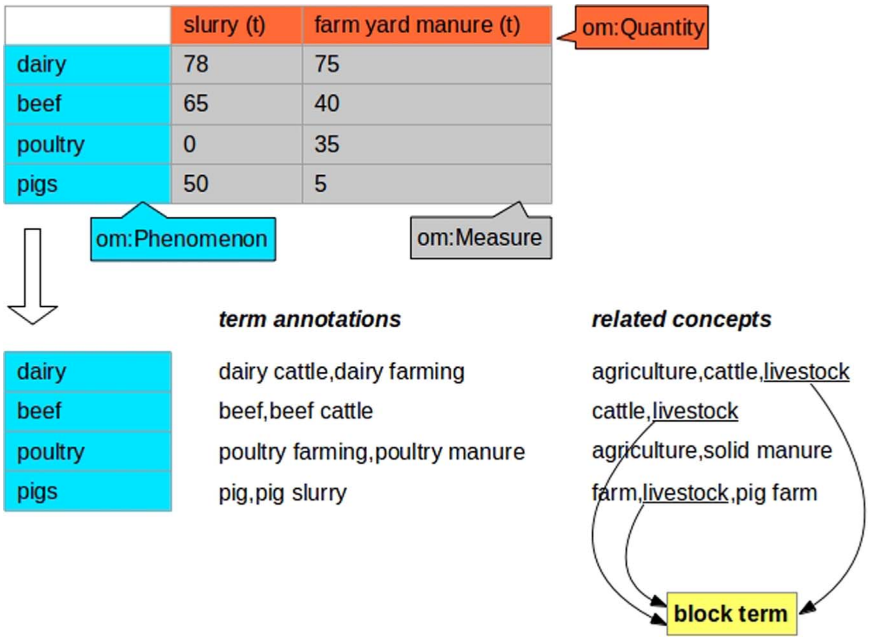


Fig. 3. Selection of block terms for phenomenon blocks.

```
    for For each domain concept related to a term in the phenomenon block do
        Determine the total # terms in the phenomenon block to which it is related
    end for
    Collect all domain concepts that are related to ≥30% of the terms the phenomenon block
    From this set, select the domain concept, that is related to the highest # terms, as block term
    Annotate block with block term
end for
for Each individual term in a phenomenon block do
    Annotate term with block term
    for Each domain annotation of that term do
        if block term is skos: related or skos: broader to the annotation then
            keep annotation
        else
            delete annotation
        end if
    end for
end for
```

4. Case study

We test our approaches for automatic annotation by applying it to a set of existing spreadsheets (Tables 2 and 3). We compare the results of our approaches with the results of a baseline method that consists of purely lexical matching (see Section 4.4). We determine the quality of the automatic annotations in terms of precision and recall relative to a manually constructed ground truth annotation.

4.1. Data set

We select case study spreadsheets that fall within the scope of our research, i.e., natural science spreadsheets that consist of numerical data, quantities and units of measure, and information on the associated objects and events. To this end, we perform a literature search using the Google Scholar web search engine. We select journal papers, that use spreadsheets to perform research analyses, and that have made these spreadsheets publicly available online as supplementary data. As several authors of this paper are working in the domain of agriculture and environmental science, we select journal papers, and a vocabulary (Section 4.2), from these particular domains. The availability of spreadsheets is checked by including the keywords “supplementary information” or “supplementary data”, and “spreadsheet” or “.xls”. The subject of the papers is defined by either the keyword “agriculture” or “environment”.

We inspect the spreadsheets of 15 journal papers and select 5 of these for our case study. The rejected spreadsheets either do not fall into the scope of our research, or are discarded because our algorithms are not yet able to handle their complicated design, e.g., as these spreadsheets contain nested tables, macros and pivot tables. Our final case study data set consists of 12 spreadsheets from 5 different journal papers containing a total of 61 tables. These spreadsheets contain data and analyses on farm profitability (Delbridge et al., 2013), life cycle assessment in agriculture (Bellon-Maurel et al., 2014), greenhouse gas emissions by dairy cropping systems (Malcolm et al., 2015), environmental services in agro-ecosystems (Ibarra et al., 2013), and life cycle models of biofuel production (Plevin, 2009). To enable a fair comparison between our approaches, and the baseline method, we remove all titles and comments that are surrounding the tables in the spreadsheets. The spreadsheets, as used in our case study, are publicly available.[4]

Table 2
Hypotheses to evaluate the suggested annotation approaches.

property	hypothesis
2	The unit grammar improves precision and recall of annotation of unit terms
3	The quantity grammar improves precision and recall of annotation of quantity terms
4	The majority of blocks can be correctly retrieved and annotated using the block annotation method
	The block annotation improves precision and recall of annotation of individual table terms
5	The majority of the block terms identified by the block term method is relevant
	Selection of domain annotations using block terms results in higher precision

Table 3
Evaluation methods for the suggested annotation approaches.

property	test	compare with	metric
2	Annotate unit terms using unit grammar + exact string matching	baseline	precisison, recall
3	Annotate quantity terms using quantity grammar + lexical matching or quantity deduction	baseline	precisison, recall
4a	Semantically categorize blocks		precisison, recall
b	Annotate individual terms in semantically categorized blocks using vocabulary selection	baseline	precisison, recall
5a	Select block terms for phenomenon blocks		precisison, recall
b	Annotate individual terms in phenomenon blocks, select annotations using block terms	baseline	precisison, recall

4.2. Vocabularies

All annotation methods, i.e., our approach, the baseline method and the ground truth, use the Valerie ontology[5] to annotate spreadsheet content with domain knowledge concepts. The Valerie ontology is developed to annotate and organize knowledge in research documents from the domains of agronomy and forestry. All annotation methods use the OM Ontology[6] (Rijgersberg et al., 2011) to annotate spreadsheet content with units of measure concepts, quantity concepts and related concepts.

4.3. Ground truth annotation

For all tables a ground truth is constructed by manually annotating both the individual terms and the blocks in the tables with the concepts from the two selected vocabularies. This process requires both knowledge of the corresponding vocabularies, and knowledge of the tables and their context. Therefore we perform the manual annotation in cooperation with domain experts, who are familiar with the selected publications and spreadsheet files. We first create a draft annotation, i.e., per block or term in a table we manually select the possible matching concepts from one or both vocabularies. For individual terms we first try to find a concept that describes the term as a whole. If such a concept is not available, we annotate parts of the term separately with matching concepts. We discuss our draft selection with the domain experts, and subsequently, choose for each block and term the most suitable concept(s) for annotation.

Results of the ground truth annotation: A majority of 81% of the string terms in the sheets can be manually annotated with concepts from the two selected vocabularies. Terms that can not be manually annotated were, for example, terms referring to computations, like “Total”, “Value” and “calculated”, and terms referring to specific models or scenarios that are used in the corresponding research project.

Half of the annotated terms are annotated with concepts from both the OM Vocabulary and the Valerie vocabulary (Table 4). The majority of these terms is referring to quantities. Domain scientists usually do not write just the quantity, but also include domain knowledge in these terms, e.g., “Volume of adjacent channels” or “Energy to press crop (MJ/yr)”. Frequently, they do not even mention the quantity at all (see Section 3.4). Besides, domain knowledge may also be included in unit terms, e.g., “kg CO2 eq” or “lb/bu soybean”.

About 36% of the phenomenon blocks can be manually annotated with a block term (Table 5). For the remaining 64% of the phenomenon blocks no block term can be selected, as the corresponding string terms can not be matched to a domain concept (see above), or because the string terms cover such a wide variety that it is not possible to find a common denominator. Several phenomenon blocks, especially those located in the same spreadsheet, are annotated with the same block term. This can be explained by the spreadsheet developers using table templates, i.e., tables with a fixed layout and fixed combinations of string terms in the table headers.

Table 4
Ground truth annotation of string terms: number of annotations.

# Unit annotations	101
# Quantity annotations	326
# Domain annotations	665
# Total annotations	1092

Table 5
Ground truth annotation of blocks.

# Unit blocks	34
# Quantity blocks	60
# Phenomenon blocks	56
# Body blocks	85
# Total blocks	235

4.4. Baseline annotation

We compare the results of our approaches with the results of a baseline method. In this baseline method all string terms in the case study spreadsheets are automatically annotated with concepts from the OM and Valerie vocabularies using a lexical matching method (Algorithm 1). The lexical matching method compares spreadsheet terms to labels or symbols from the two selected vocabularies, and expresses the similarity between the two as isub distance (Stoilos et al., 2005). The maximum isub distance is 1, meaning two strings are exactly the same. When the isub distance between a spreadsheet term and a label or symbol in a vocabulary is ≥0.85, the corresponding concept is selected by the baseline method for annotation.

5. Results

In this section we describe the results of our case study experiment according to the evaluation methods described in Table 3. It should be noticed that the annotation results are not yet part of the spreadsheet files, but rather have the form of raw prolog output (Fig. 4). However, we also collected these results in csv files, which are publicly available.[7]

5.1. Annotation of unit terms: approach 2

The unit grammar has a higher recall than the baseline method (Table 6). More than 80% of the manual annotations of unit terms is missed by the baseline method, and about 25% by the unit grammar method. The unit grammar also has a higher precision than the baseline method, but still generates about 28% incorrect annotations.

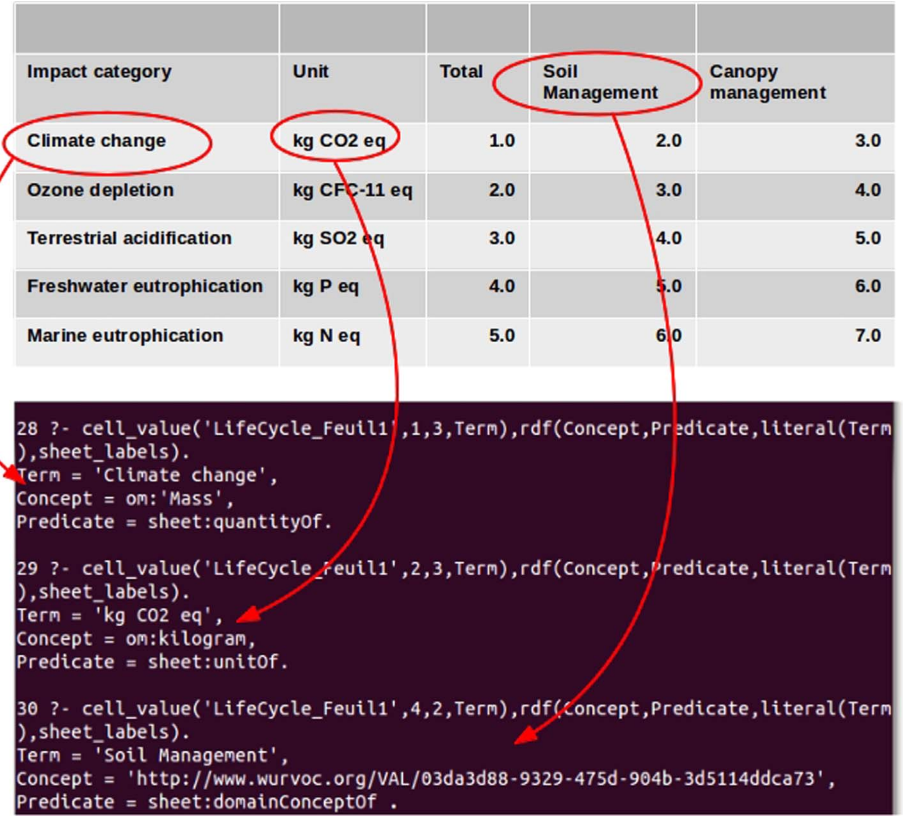
The lexical matching method applied in the baseline does not perform well on the unit terms in the case study spreadsheets (Section 3.3), which causes the recall to be low. The majority of these unit terms

[4] GitHub repository, https://github.com/MartineDeVos/Spreadsheets/tree/work/annotation

[5] Valerie ontology, http://www.foodvoc.org/page/Valerie

[6] OM ontology, http://www.wurvoc.org/vocabularies/om-1.8/

[7] GitHub repository, https://github.com/MartineDeVos/Spreadsheets/tree/work/annotation



Impact category	Unit	Total	Soil Management	Canopy management
Climate change	kg CO2 eq	1.0	2.0	3.0
Ozone depletion	kg CFC-11 eq	2.0	3.0	4.0
Terrestrial acidification	kg SO2 eq	3.0	4.0	5.0
Freshwater eutrophication	kg P eq	4.0	5.0	6.0
Marine eutrophication	kg N eq	5.0	6.0	7.0

```
28 ?- cell_value('LifeCycle_Feuil1',1,3,Term),rdf(Concept,Predicate,literal(Term
),sheet_labels).
Term = 'Climate change',
Concept = om:'Mass',
Predicate = sheet:quantityOf.

29 ?- cell_value('LifeCycle_Feuil1',2,3,Term),rdf(Concept,Predicate,literal(Term
),sheet_labels).
Term = 'kg CO2 eq',
Concept = om:kilogram,
Predicate = sheet:unitOf.

30 ?- cell_value('LifeCycle_Feuil1',4,2,Term),rdf(Concept,Predicate,literal(Term
),sheet_labels).
Term = 'Soil Management',
Concept = 'http://www.wurvoc.org/VAL/03da3d88-9329-475d-904b-3d5114ddca73',
Predicate = sheet:domainConceptOf .
```

Fig. 4. Screenshot example of some case study results in the form of raw prolog output.

Table 6
Automatic annotation of unit terms: comparing lexical matching (baseline) with a method using unit grammar rules.

Annotation	Precision	recall
Baseline annotation	0.35	0.16
Unit grammar annotation	0.72	0.75

consist of either short strings, e.g., “MJ” and “kg”, which are not recognized by the lexical matching method, or series of short strings, e.g., “MJ/ha” and “kg CO2 eq”, which are annotated as separate terms.

Part of the unit terms that are missed by the unit grammar method contains sloppy notations of unit symbols. Domain scientists may make, intentionally or not, typing errors, e.g., by writing “BTU” instead of “Btu” for “om:’BritishThermalUnit’ or “gr” instead of “g” for “om:gram”. They also may not realize that some of the unit symbols they use in their spreadsheets require additional specification in order to be interpreted correctly, e.g., the symbol “gal” may refer to “om:dry-Gallon-US”, “om:gallon-Imperial” or “om:gallon-US”. And sometimes these sloppy notations are commonly used notations among domain scientists, e.g., the unit “om:second-Time” is often represented with the symbol “sec” instead of the official symbol “s”. As the sloppy notations do not exactly match the labels or symbols of units concepts in the OM vocabulary, the corresponding terms are not considered as unit terms, or not all included symbols are correctly annotated.

The difference in precision between the two methods can be explained by the fact that the lexical matching method applied in the baseline is quite tolerant. It does not distinguish between upper and lower case, and also considers partial matches to a symbol or label from the OM vocabulary. The unit grammar method is more strict, as it only considers a term as a unit term, when it complies to the grammar rules, and when the included symbol exactly matches with a symbol or label from the OM vocabulary. Terms like “Corn Grain”,“Percent reported” and “MJ/m3” are incorrectly annotated by the baseline method with, respectively, “om:grain”,i.e., a unit of mass, “om:percent”, and “om:megajoulePerSquareMetre”. The unit grammar method, on the other hand, does not consider the first two terms as unit terms, as “Grain” is written with a capital “G”, and the string “Percent reported” is longer than ten characters. The term “MJ/m3” is annotated with both “om:megajoule” and “om:cubicMetre”.

The false positives generated by the unit grammar method are mainly caused by the presence of abbreviations. Domain scientists often use abbreviations of domain terms, e.g., “N” for nitrogen, or code names for experiments or dates, e.g. “rye_cg” for a crop rotation method with rye and corn grain. The grammar rules recognize these abbreviations as unit symbols, e.g., ‘N” is recognized as symbol for “om:newton”, and the “cg” in “rye_cg” as symbol for “om:centigram”.

5.2. Annotation of quantity terms: approach 3

The baseline method shows a very low recall (Table 7). The recall of the quantity grammar method is higher, but still low on an absolute level. The two annotation methods show a similar precision; and both generate about 60% incorrect annotations.

The sets of individual spreadsheet terms that are annotated with a quantity concept differ between the two annotation methods. The baseline annotates a term as a quantity term when the lexical matching method yields a match with a quantity concept. The majority of the

Table 7
Automatic annotation of quantity terms: comparing lexical matching (baseline) with a method using quantity grammar rules.

annotation	Precision	Recall
Baseline annotation	0.38	0.05
Quantity grammar annotation	0.39	0.23

quantity concepts, though, is not explicitly mentioned in the corresponding terms, which explains the very low recall. The quantity grammar method, on the other hand, recognizes a term as a quantity term when its structure complies to the quantity grammar, which is the case for half of the quantity terms in the test spreadsheets. However, only half of these recognized quantity terms can be correctly annotated with a quantity concept.

The high percentage of incorrect annotations generated by the baseline method is mainly caused by the lexical matching yielding multiple matches, which are not all correct. For example, the quantity term "Volume" is correctly annotated with "om:'Volume", but also wrongly annotated with "om:'MolarVolume" and "om:'VolumeFraction".

More than half of the incorrect annotations generated by the quantity grammar method can be explained by the fact that the units of measure associated with some quantity terms do not comply with the unit grammar. Most of these units have a domain concept included in the symbol, e.g., "Diesel fuel (kg CO2e/ha)". These units can not be correctly annotated by our method, and as a consequence, the correct quantity concept can not be deduced. For example, the term "Diesel fuel (kg CO2e/ha)" is manually annotated with the quantity concept "om:AreaDensity". The quantity grammar method does not recognize the unit symbol "kg/ha", but only the separate symbols "kg" and "ha" and therefore annotates the term with the corresponding quantities "om:Mass" and "om:Area".

5.3. Block typology

5.3.1. Semantic categorization of blocks: approach 4a

The vast majority of the unit, quantity and measure blocks, and about half of the phenomenon blocks is correctly annotated by the automatic block annotation method (Table 8). The automatic annotation method is able to retrieve the vast majority of the manually annotated phenomenon and measure blocks. However, only 56–65% of the unit, and quantity blocks can be retrieved.

The majority of the unit blocks that are not automatically annotated consists mainly of unit terms that are not recognized as common units of measure, i.e., these units do not belong to the "om:commonApplicationArea". This includes, for example, unit terms like "kg/ha" representing the unit "om:kilogramPerHectare and "MJ" representing "om:megajoule".

The majority of the quantity blocks that are not automatically annotated consists mainly of quantity terms that are not recognized as such. These terms can neither be lexically matched to a quantity concept, nor can these be associated with a unit of measure. In some of these cases the quantity is only represented in the title of the table, while the supposed quantity cells contain associated domain concepts In some of these cases there is a unit block present, but this unit block is not recognized by the algorithm (see previous section).

Context blocks that cannot be annotated as unit or quantity block are annotated as phenomenon blocks. The low precision for phenomenon blocks is therefore directly influenced by the low recall of unit and quantity blocks.

Table 8
Semantic categorization of blocks.

semantic category	Precision	Recall
Unit blocks	0.88	0.65
Quantity blocks	0.92	0.56
Phenomenon blocks	0.54	0.80
Measure blocks	0.98	0.98

Table 9
Automatic annotation of individual terms with vocabulary concepts. Comparing lexical matching of all terms at once (baseline), to lexical matching of terms using vocabulary selection (blocks).

Vocabulary concept		Precision	Recall
Unit concept			
	Baseline	0.35	0.16
	Blocks	0.44	0.12
Quantity concept			
	Baseline	0.38	0.05
	Blocks	0.42	0.02
Domain concept			
	Baseline	0.34	0.17
	Blocks	0.37	0.15

5.3.2. Annotation of individual terms in semantically categorized blocks: approach 4b

For all concepts the precision of annotation is slightly higher for the method using semantically categorized blocks than for the baseline method, but the recall is slightly lower (Table 9).

In the baseline method all string terms can be annotated with domain concepts and concepts from the OM Vocabulary. In contrast, in the method using semantically categorized blocks only terms in Quantity and Phenomenon blocks are annotated with domain concepts, and only terms in Quantity and Unit blocks are annotated with concepts from the OM Vocabulary. The baseline method thus creates additional annotations on top of the set created by the method using blocks, which results in a higher recall.

The baseline method creates more annotations than the block method, including more incorrect annotations, some of which are annotations with concepts from the wrong vocabulary. Examples of these incorrect annotations are the annotation of the domain term "Corn Grain" with the unit concept "om:grain", and the annotation of the quantity term "Energy" with the domain concepts "energy production" and "energy source". The higher precision of the block annotation can partly be explained by its use of vocabulary selection, which reduces this type of incorrect annotations. The vocabulary selection works best for phenomenon and unit blocks, where the amount of annotations with concepts from the wrong vocabulary is reduced with about 10%.

5.4. Grouping of similar domain concepts: approach 5a

For only 27% of the automatically categorized phenomenon blocks, the block terms can be compared to the manually selected block terms. For none of these blocks the manual and automatic annotation use the same block terms (Table 10).

For a majority of 73% of the automatically categorized phenomenon blocks the block terms can not be compared to the manually selected block terms. About half of these blocks are in fact quantity or unit blocks that are falsely recognized as phenomenon blocks. Therefore the terms in these blocks either contain a limited amount of domain knowledge, or the blocks contain such a wide variety of terms that it is not possible to select a common block term. The other half consists of

Table 10
Comparison of block terms used for the same phenomenon blocks in the manual and automatic annotation.

manual block term	automatic block term
Crop	FAO700 Row crop Management strategy
Wheat	Cereal crop
Farm management	Management strategy
Agro-ecosystem	Beneficial effect

Table 11
Classification rules and heuristics implemented in Algorithm 2 Annotation of table bodies and context blocks.

Classification rules

1. Cells of the same data type, i.e., string and float, are clustered in rectangular blocks
2. Blocks with string or float cells may also contain empty cells
3. The table body consists of one block of quantitative (float) observations
4. The table body is surrounded by blocks describing the context of the observations
5. The context blocks consist of string terms

Heuristics

1. Tables are designed to be symmetric. Context blocks have the same height or width as the table body

correctly recognized phenomenon blocks. However, as these blocks contain multiple terms that can not be matched to a domain concept, the selection of a common block term is not possible.

Some of the automatic block term are semantically related to the manual 'counter part', e.g., "row crop" is a narrower term of "crop" and "cereal crop" is a broader term of "wheat". As such, these block terms might be considered relevant. However, as the set of relevant block terms is very small, it can not be used to select relevant domain annotations for individual terms in phenomenon blocks.

6. Discussion

6.1. Discussion of the annotation approach per property

Annotation of unit terms: approach 2. The case study results show that the unit grammar rules improve the precision and recall of the annotation of unit terms. The grammar rules allow very short strings to be matched to relevant unit concepts in the OM vocabulary, and the strict matching method applied by the grammar rules avoids many false positives. A disadvantage of this strict matching method is that unit terms may be missed, as domain scientists are sometimes sloppy in their notations of unit symbols. These sloppy notations may be caused by typing errors, ignorance, or these may be commonly used notations among domain scientists. A limited set of these commonly used 'unofficial symbols' are already included in the OM vocabulary.

Another disadvantage of the current grammar rules is that domain specific abbreviations in the spreadsheet tables are mistaken for unit symbols. These abbreviations often yield unit annotations that are not relevant to the domain of the spreadsheets. Allowing only unit annotations from an OM application area that is relevant to the spreadsheet set (see Section 6.1), like 'agriculture' or environmental science', would probably yield in a higher precision of unit annotations.

Annotation of quantity terms: approach 3. Results of the case study show that the quantity grammar rules improve the recall of the annotation of quantity terms, but not the precision. The grammar rules assume a specific structure of quantity terms, which helps to recognize quantity terms, and results in a higher recall than the baseline method. However, as half of the quantity terms in the spreadsheets do not meet this structure, the recall in absolute terms is still low. The quantity grammar and the baseline method each annotate a different set of individual terms as quantity terms, and can be considered complementary. Both precision and recall of the annotation of individual quantity terms may be improved if both methods are merged. This implies that a term is considered a quantity term, if it can either directly be matched to an OM quantity concept, or if the term structure meets a prescribed quantity structure. In fact, this procedure is already applied in the recognition of quantity terms in order to annotate quantity blocks (Algorithm 6).

Table 12
Classification rules implemented in Algorithm 3: Annotation of unit terms.

1. Unit terms are short strings (<11 characters)
2. Unit terms mainly consist of one or more unit symbols
3. A unit symbol is a short string that exactly matches a symbol from a unit concept in the OM Vocabulary

Table 13
Classification rules and heuristics implemented in Algorithm 4 Annotation of quantity terms.

Classification rules

1. Quantity terms contain a unit term (Table 12, Algorithm 3) enclosed in brackets

Heuristics

1. Annotation concepts for quantity terms can be deduced from the included unit term

Table 14
Classification rules and heuristics implemented in Algorithms 5, and 6: Annotating blocks.

Classification rules

1. The table body is annotated with "om:Measure"
2. The context blocks are annotated with "om:Quantity", or "om:Phenomenon", or "om:Unit of Measure"
3. When a block is annotated with a particular om concept, this annotation applies to all terms in that block

Heuristics

1. The context blocks in one table consist of at least one Phenomenom, and only one Quantity block.
2. If units of measure are present they occur either in a separate Unit block, or included in the Quantity block.
3. A unit block consists of either a single row or a column
4. A unit block contains >30% unit terms (Table 12, Algorithm 3)
5. A unit block contains more unit terms than domain terms
6. The Quantity block is vertically or horizontally aligned with the Unit block and Measure block
7. A Quantity block contains >30% quantity terms (Table 13, Algorithm 4)

The quantity grammar rules use the associated unit of measure to deduce a suitable quantity concept for a quantity term. If the structure of the associated unit of measure does not comply to the grammar rules, the unit annotation and, consequently, the quantity deduction are incorrect. The use of terms with 'custom made' units, e.g., "Diesel fuel (kg CO2e/ha)", may be common among domain scientists. The unit grammar may be adapted for this, e.g., by filtering out domain concepts prior to matching the term to symbols and labels from the OM vocabulary.

Block typology: approaches 4a and 4b. The majority of the blocks in the case study spreadsheets can be correctly retrieved and annotated

Table 15
Classification rules and heuristics implemented in Algorithm 7: Annotating terms.

Classification rules

1. Terms in Phenomenon blocks are annotated with concepts from a domain ontology
2. Terms in Unit blocks are annotated with concepts from the OM ontology
3. Terms in Quantity blocks are annotated with concepts from both the OM and a domain ontology

Heuristics

1. Annotation concepts for terms in Quantity blocks can be deduced from the associated unit term, which is either located within the quantity term, or in the neighbouring Unit block

using the block annotation method. However, the precision for phenomenon blocks is lower than for the other types of blocks. The same holds for the recall for unit and quantity blocks.

The algorithm for the annotation of unit blocks only considers units of measure from the “om:commonApplicationArea”. This restriction avoids matches with ‘exotic’ units of measure that are only used by specific application areas in the OM vocabulary, e.g., “om:radiometryAndRadiobiology” and “om:astronomyAndAstrophysics”, that do not match the application area of the spreadsheets test set. However, some units of measure that are relevant to the application area of the tested spreadsheets are also avoided, e.g., “om:kilogramPerHectare and “om:megajoule”. This can be explained by the fact that the class “application area” in the OM vocabulary is not yet sufficiently populated. One third of the units of measure in the OM vocabulary does not belong to any application area, and the vocabulary only contains a selection of 17 application areas from the field of pure and applied physics (Rijgersberg et al., 2011). Selecting units that belong to an application area that is more relevant to the spreadsheet set, like ‘agriculture’ or ‘environmental science’, would probably yield in a higher recall of unit block annotations, and consequently, in a higher recall of quantity block annotations and a higher precision of phenomenon block annotations.

The results of the case study show that the block annotation, and corresponding vocabulary selection, slightly improves the precision of the annotation of individual terms, but slightly reduces the recall.

Grouping of similar domain concepts: approach 5a. The case study results show that the majority of the automatically selected block terms for phenomenon blocks is not relevant, and can not be used to select domain annotations for the individual terms within these blocks.

Our assumption that individual terms from the same phenomenon block can be related to a common denominator concept seems not apply to this case study, as only one third of the phenomenon blocks can be manually annotated with a block term. However, related studies were indeed able to find common denominators for groups of string terms in their spreadsheet tables. Some of these used structural properties of a table to determine a class hierarchy, and populated it with the original string terms as found in the table (Abraham and Erwig, 2006; Hermans et al., 2010; Chen and Cafarella, 2013). These studies showed that many phenomenon blocks already have a common denominator that is present within the table, i.e., located above or left from the phenomenon block. Our approach does not consider these terms, as these are not included in the identified context blocks. It may be useful to include the original string terms in our approach and consider these as block term candidates. We submit, however, that the use of vocabulary concepts is more appropriate, as the original string terms may be ambiguous, or hard to interpret.

Other studies inferred annotations for the header labels by considering the string values in the corresponding row or column (Venetis et al., 2011; Limaye et al., 2010), i.e., an approach that is similar to our method of finding block terms for phenomenon blocks. These studies used probabilistic reasoning, instead of the majority method, to find suitable annotations for the header labels. The use of probabilistic reasoning is related to the use of large, common vocabularies, since these will yield header labels that are too generic to be relevant, when the majority method is used. This issue does probably not apply to the use of more specific domain vocabularies, like the one used in our study.

Table 16
Classification rules and heuristics implemented in Algorithm 8 Annotating block terms.

Classification rules
1. A block term is either hierarchically or property-wise related to ≥30% of the terms in a phenomenon block 2. When a phenomenon block is annotated with a particular block term, this annotation applies to all terms in that block
Heuristics
1. All terms in a Phenomenon block can be semantically related to one block term, i.e., one single domain concept that serves as a common denominator 2. Terms in a phenomenon block should only be annotated with concepts that are related to the corresponding block term

6.2. General discussion

The annotation approaches presented in this study do not only build upon each other, but are also combined in an iterative way. The case study results show that these dependencies yield advantages, as the approaches inform each other, but this construction is also fragile, as mistakes made in one approach, negatively influence the annotation results of the depending approach(es). The correct annotation of unit terms forms a critical first step, and improving this step would probably result in better results for all of the following steps.

The availability of suitable vocabularies is essential in most of our approaches. The Valerie vocabulary covers the domain of the spreadsheets, and its graph structure allows automatic selection of related, higher-level domain concepts as common denominator for the terms in phenomenon blocks. The graph structure in the OM vocabulary allows us to check whether a term is correctly recognized as a unit or a quantity term. We used four main concepts from the OM vocabulary to semantically categorize the blocks. And, in our method we use several other types of information provided by the OM vocabulary, i.e., (1) which units are commonly used by certain quantities, (2) which units and quantities are commonly used by certain application areas, and (3) which unofficial notations for units and quantities are commonly used by domain scientists. There are alternative vocabularies available to annotate unit or a quantity terms, e.g., QUDT (Hodgson et al., 2014). However, these vocabularies do not include the phenomenon concept, which we argue is essential to formally distinguish domain knowledge, nor do they offer the additional types of information that are mentioned above.

The vocabularies that are used in the study are selected manually. As our annotation approaches are fully automated, automatic vocabulary selection would be a logical step in the annotation process. The OM vocabulary can be used as a standard, generic vocabulary to annotate quantities and units of measure in all sorts of natural science spreadsheets. The automatic selection of a domain vocabulary could be done by collecting a larger set of suitable (see previous paragraph) vocabularies within the natural science domain. For a given spreadsheet, a pilot annotation based on lexical matching could indicate which vocabulary from this set is most suitable to annotate the domain terms in the spreadsheet. Another option would be to develop a more interactive tool, that allows domain scientists to supply or choose a relevant vocabulary. Newly proposed vocabularies could then be used to enrich our ‘vocabulary database’.

We tested our automatic annotation approaches on natural science spreadsheets that (1) consist of numerical data, quantities and units of measure, and information on the associated objects and events, and (2) have a simple design. Although this type of spreadsheets is typical for the domain, we are aware that a considerable part of the natural science spreadsheets may not have such a simple, homogeneous structure. It is quite common for domain scientists to create tables that contain no or very limited information on the quantities, units or the associated objects and events, or to create tables with a complicated design. Our annotation approaches may also be applied to more complicated or heterogeneous tables, but we expect that the recall and precision for the annotation of these types of tables will be lower, compared to the results of present case study. On the other hand, the spreadsheets used in our case study are officially published as supplementary research data. We believe that this type of spreadsheets is a small set compared to the number of natural science spreadsheets used in the informal area. The reuse of spreadsheet data would be valuable in this area in particular, which would make it an interesting area for application of our method.

7. Conclusions and future work

The ultimate goal of this paper is to make the underlying domain model of natural science spreadsheet data explicit. Ideally, the presentation of this domain model is independent of the spreadsheet syntax, i.e. a collection of concepts and the way these are related to each other, so that it can actually be consumed computationally. Our annotation approach for individual terms provides information on the concepts that are present in the tables. The annotation of blocks with the four OM concepts Measure, Unit, Quantity, and Phenomenon is an essential first step in the identification of relations between cells in a table. As such, the annotation approaches presented in this study provide the basic information that is needed to construct the domain model of natural science spreadsheets.

As mentioned above, we expect that our annotation approach may also be applied to tables that do not meet the requirements of complete information, or simple design. In future work we plan to investigate to what extent and in what ways existing natural science tables deviate from our 'ideal situation'. We also plan to investigate how these deviations may be addressed by heuristics, and to what extent automatic annotation of these tables is still possible.

Furthermore, we see several promising directions to extend and improve our approach. The annotation results presented in this study could be visualized within the spreadsheet tables, together with additional explanation from the vocabularies, e.g., the concept definition, alternative labels and related concepts. This would help users to understand and correctly interpret natural science spreadsheet data. Furthermore, we would like to explore ways to actually reconstruct the domain model. Our present approach for the annotation of individual terms may yield multiple concepts per term. A logical next step would be to develop a method to select the best, single concepts to be included in the domain model. And also, to identify and annotate the relations that exist between concepts. We may further analyze the relations that exist between concepts that are present in the semantically categorized blocks, e.g., relations between different phenomenon concepts, and between phenomenon and quantity concepts. In such an analysis, the formulas in spreadsheet tables could form and additional and useful source of information (de Vos et al., 2015).

Acknowledgments

We wish to thank Frits van Evert and Hein ten Berge (Wageningen University and Research centre) for their useful comments on our draft manual annotation. This publication was supported by the Data2Semantics project in the Dutch national program COMMIT.

Appendix A

This section provides a more detailed description of the heuristics and classification rules used in our approaches, and how these are implemented in our algorithms.

References

Abraham, R., Erwig, M., 2006. Inferring Templates from Spreadsheets. In: Proceedings of the 28th International Conference on Software Engineering. ACM, Shanghai, China, pp. 182–191.

Assem, M.V., Rijgersberg, H., Wigham, M., Top, J., 2010. Converting and annotating quantitative data. In: Patel-Schneider, P. (Ed.), ISWC2010. pp. 16–31.

Bellon-Maurel, V., Short, M.D., Roux, P., Schulz, M., Peters, G.M., 2014. Streamlining life cycle inventory data generation in agriculture using traceability data and information and communication technologies - Part I: Concepts and technical basis. J. Cleaner Prod. 69, 60–66.

Cafarella, M.J., Halevy, A., Wang, D.Z., Wu, E., Zhang, Y., 2008. WebTables: exploring the power of tables on the web. Proceedings of the VLDB Endowment 1 (1), 538–549, URL ⟨http://dl.acm.org/citation.cfm?doid=1453856.1453916⟩.

Chen, Z., Cafarella, M., 2013. Automatic web spreadsheet data extraction. In: Proceedings of the 3rd International Workshop on Semantic Search Over the Web - SS@ '13. pp. 1–8. URL ⟨http://dl.acm.org.prox.lib.ncsu.edu/citation.cfm?id=2509908.2509909⟩.

Connor, M.J.O., Halaschek-wiener, C., Musen, M.A., 2010. Mapping master : a flexible approach for mapping spreadsheets to OWL. In: The Semantic WebISWC. Springer, Berlin, Heidelberg, pp. 194–208.

De Vos, M., Van Hage, W.R., Ros, J., Schreiber, A., 2012. Reconstructing semantics of scientific models: a case study. In: Proceedings of the OEDW Workshop on Ontology Engineering in a Data Driven World, EKAW 2012, Galway, Ireland.

De Vos, M.G., Wielemaker, J., Wielinga, B., Schreiber, G., Top, J., 2015. A methodology for constructing the calculation model of scientific spreadsheets. In: Proceedings of the 8th International Conference on Knowledge Capture.

Delbridge, T.A., Fernholz, C., King, R.P., Lazarus, W., 2013. A whole-farm profitability analysis of organic and conventional cropping systems. Agricult. Syst. 122, 1–10. http://dx.doi.org/10.1016/j.agsy.2013.07.007.

Fisher, M., Rothermel, G., 2005. The EUSES spreadsheet corpus: a shared resource for supporting experimentation with spreadsheet dependability mechanisms. In: ACM SIGSOFT Software Engineering Notes. vol. 1. pp. 1–5 URL ⟨http://doi.acm.org/10.1145/1082983.1083242%5Cnhttp://dl.acm.org/citation.cfm?id=1083242⟩.

Garcia-silva, A., Gomez-perez, A., Suarez-figueroa, M. C., Villazon-terrazas, B., 2008. A Pattern based approach for re-engineering non-ontological resources into ontologies. In: The Semantic Web. No. 2. Springer Berlin Heidelberg, pp. 167–181.

González-Beltrán, A., Maguire, E., Sansone, S.-A., Rocca-Serra, P., 2014. linkedISA: semantic representation of ISA-Tab experimental metadata. BMC bioinformatics 15 Suppl 1 (Suppl 14), S4 URL ⟨http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-15-S14-S4%5Cnhttp://www.biomedcentral.com/1471-2105/15/S14/S4⟩.

Han, L., Finin, T., Parr, C., Sachs, J., Joshi, A., 2008. RDF123: From Spreadsheets to RDF. In: The Semantic Web-ISWC 2008. Springer, Berlin, Heidelberg, pp. 451–466.

Hermans, F., Pinzger, M., Deursen, A.V., 2010. Automatically Extracting Class Diagrams from Spreadsheets. In: 24th European Conference on Object-Oriented Programming (ECOOP), Lecture Notes in Computer Science, Springer-Verlag, Maribor, Slovenia, pp. 52–75.

Hipfl, S., 2004. Using Layout Information for Spreadsheet Visualization. In: Proceedings of the European Spreadsheet Risks Interest Group 5th Annual Conference. Klagenfurt, Austria.

Hodgson, R., Keller, P.J., Hodges, J., Spivak, J., 2014. QUDT – Quantities, Units, Dimensions and Data Types in OWL and XML; Version 1.1 URL ⟨http://qudt.org/⟩.

Ibarra, A.A., Zambrano, L., Valiente, E.L., Ramos-Bueno, A., 2013. Enhancing the potential value of environmental services in urban wetlands: an agro-ecosystem approach. Cities 31, 438–443. http://dx.doi.org/10.1016/j.cities.2012.08.002.

Langegger, A., Wöß, W., 2009. XLWrap–querying and integrating arbitrary spreadsheets with SPARQL. International Semantic Web Conference. Springer, Berlin, Heidelberg. ⟨http://dx.doi.org/10.1007/978-3-642-04930-9_23⟩.

Limaye, G., Sarawagi, S., Chakrabarti, S., 2010. Annotating and searching web tables using entities, types and relationships. In: Proceedings of the VLDB Endowment, vol. 3. pp. 1338–1347. URL ⟨http://portal.acm.org/citation.cfm?id=1921005⟩.

Maguire, E., González-Beltrán, A., Whetzel, P.L., Sansone, S.A., Rocca-Serra, P., 2013. OntoMaton: a Bioportal powered ontology widget for Google Spreadsheets. Bioinformatics 29 (4), 525–527.

Malcolm, G., Camargo, G., Ishler, V., Richard, T., Karsten, H., 2015. Energy and greenhouse gas analysis of northeast U.S. dairy cropping systems. Agricul. Ecosyst. Environ. 199, 407–417, URL ⟨http://linkinghub.elsevier.com/retrieve/pii/S0167880914004708⟩.

McDonald, D., Clemente, J.C., Kuczynski, J., Rideout, J., Stombaugh, J., Wendel, D., Wilke, A., Huse, S., Hufnagle, J., Meyer, F., Knight, R., Caporaso, J., 2012. The Biological Observation Matrix (BIOM) format or: how I learned to stop worrying and love the ome-ome. GigaScience 1 (1), 7, URL ⟨http://www.gigasciencejournal.com/content/1/1/7⟩.

Meroño-Peñuela, A., Ashkpour, A., Rietveld, L., Hoekstra, R., Schlobach, S., 2013. Linked Humanities Data : The next frontier? A case-study in historical census data. In: The Semantic Web: Semantics and Big Data. Springer, Berlin, Heidelberg, pp. 645–649.

Mittermeir, R., Clermont, M., 2002. Finding High-Level Structures in Spreadsheet Programs. In: Proceedings of the 9th Working Conference on Reverse Engineering, Richmond, VA, USA, pp. 221–232.

Mulwad, V., Finin, T., Joshi, A., 2012. A Domain Independent Framework for Extracting Linked Semantic Data from Tables. In: Search Computing. Springer, Berlin, Heidelberg, pp. 16–33.

Plevin, R.J., 2009. Modeling corn ethanol and climate: a critical comparison of the BESS and GREET Models. J. Ind. Ecol. 13 (4), 495–507.

Rayner, T.F., Rocca-Serra, P., Spellman, P.T., Causton, H.C., Farne, A., Holloway, E., Irizarry, R.A., Liu, J., Maier, D.S., Miller, M., Petersen, K., Quackenbush, J., Sherlock, G., Stoeckert, C.J., White, J., Whetzel, P.L., Wymore, F., Parkinson, H., Sarkans, U., Ball, C.A., Brazma, A., 2006. A simple spreadsheet-based, MIAME-supportive format for microarray data: MAGE-TAB. BMC Bioinform. 7, 489.

Rijgersberg, H., Wigham, M., Top, J., 2011. How semantics can improve engineering processes: a case of units of measure and quantities. Adv. Eng. Inform. 25 (April (2)), 276–287, URL ⟨http://linkinghub.elsevier.com/retrieve/pii/S1474034610000753⟩.

Rocha Bernardo, I., Mota, M.S., Santanchè, A., 2013. Extracting and semantically integrating implicit schemas from multiple spreadsheets of biology based on the recognition of their nature. J. Inf. Database Manage. 4 (2), 104–113.

Sansone, S.-A., Rocca-Serra, P., Field, D., Maguire, E., Taylor, C., Hofmann, O., Fang, H., Neumann, S., Tong, W., Amaral-Zettler, L., Begley, K., Booth, T., Bougueleret, L., Burns, G., Chapman, B., Clark, T., Coleman, L.-A., Copeland, J., Das, S., de Daruvar, A., de Matos, P., Dix, I., Edmunds, S., Evelo, C.T., Forster, M.J., Gaudet, P., Gilbert, J., Goble, C., Griffin, J.L., Jacob, D., Kleinjans, J., Harland, L., Haug, K., Hermjakob, H., Ho Sui, S.J., Laederach, A., Liang, S., Marshall, S., McGrath, A., Merrill, E., Reilly, D., Roux, M., Shamu, C.E., Shang, C.A., Steinbeck, C., Trefethen, A., Williams-Jones, B., Wolstencroft, K., Xenarios, I., Hide, W., 2012. Toward interoperable bioscience data. Nat. Genet. 44 (2), 121–126, URL ⟨http://www.nature.com/ng/journal/v44/n2/full/ng.1054.html?WT.ec_id=NG-201202⟩.

Segal, J., Morris, C., 2008. Developing scientific software. IEEE Softw. 25 (4), 18–20.

Shu, Y., Ratcliffe, D., Compton, M., Squire, G., Taylor, K., 2015. A semantic approach to data translation: a case study of environmental observations data. Knowl.-Based Syst. 75, 104–123. http://dx.doi.org/10.1016/j.knosys.2014.11.023.

Stoilos, G., Stamou, G., Kollias, S., 2005. A String Metric for Ontology Alignment. In: The Semantic WebISWC, 2005. pp. 624–637.

Venetis, P., Halevy, A., Madhavan, J., 2011. Recovering semantics of tables on the web. In: Proceedings of the VLDB Endowment, vol. 4. pp. 528–538. URL ⟨http://dl.acm.org/citation.cfm?id=2002939⟩.

Wolstencroft, K., Owen, S., Horridge, M., Krebs, O., Mueller, W., Snoep, J. L., du Preez, F., Goble, C., 2011. RightField: embedding ontology annotation in spreadsheets. Bioinformatics Oxford, England 27 (July (14)), 2021-2. URL ⟨http://www.ncbi.nlm.nih.gov/pubmed/21622664⟩.

Martine de Vos is a Ph.D. student at the Web & Media group of the Department of Computer Science, VU University Amsterdam, and part of the Network Institute. website: http://martinedevos.wordpress.com/

Jan Wielemaker is a Researcher at the Web & Media group of the Department of Computer Science, VU University Amsterdam, and part of the Network Institute. He is also the main developer of SWI-Prolog, a free implementation of the programming language Prolog. website: http://www.cs.vu.nl/~janw/

Hajo Rijgersberg is a Researcher at the Food and Biobased Research Department, Wageningen University and Research centre. website: https://www.wageningenur.nl/nl/Personen/H-Hajo-Rijgersberg.htm

Guus Schreiber is a Professor at the Web & Media group of the Department of Computer Science, VU University Amsterdam, and part of the Network Institute. website: http://www.cs.vu.nl/guus/

Bob Wielinga was a Professor of Social Science Informatics (SWI) in the Faculty of Psychology, University of Amsterdam. He was also a Professor at the Web & Media group of the Department of Computer Science, VU University Amsterdam, and part of the Network Institute.

Jan Top is a Professor at the Web & Media group of the Department of Computer Science, VU University Amsterdam, and part of the Network Institute. He is also Researcher at the Food and Biobased Research Department, Wageningen University and Research centre. website: http://www.cs.vu.nl/~jltop/