SOCA (2016) 10:365–389
DOI 10.1007/s11761-016-0197-2

ORIGINAL RESEARCH PAPER

Automating the semantic mapping between regulatory guidelines and organizational processes

Krishna Sapkota[1] · Arantza Aldea[2] · Muhammad Younas[2] · David A. Duce[2] · Rene Banares-Alcantara[3]

Received: 16 February 2016 / Revised: 4 July 2016 / Accepted: 22 July 2016 / Published online: 9 August 2016

Abstract The mapping of regulatory guidelines with organizational processes is an important aspect of a regulatory-compliance management system. Automating this mapping process can greatly improve the overall compliance process. Currently, there is research on mapping between different entities such as ontology mapping, sentence similarity, semantic similarity and regulation-requirement mapping. However, there has not been adequate research on the automation of the mapping process between regulatory guidelines and organizational processes. In this paper, we explain how Natural Language Processing and Semantic Web technologies can be applied in this area. In particular, we explain how we can take advantage of the structures of regulation-ontology and the process-ontology in order to compute the similarity between a regulatory guideline and a process. Our methodology is validated using a case study in the Pharmaceutical industry, which has shown promising results.

Keywords Semantic similarity · Ontology mapping · Information extraction · Regulatory-compliance management · Regulation · Text analysis

1 Introduction

Regulatory-compliance management (RCM) is a management process, which is implemented by an organization to ensure that every process complies with the relevant requirements and expectations. Examples of requirements are the regulatory or legal guidelines, and that of expectations are mandates, policies and guidelines. Failure to maintain the RCM in organizations generally results in heavy penalties or legal disputes or even suspension and closure. Managing compliance is an expensive process. For example, legislations, such as the Sarbanes-Oxley Act (SOX), imposed stringent compliance requirements, and organizations had to make heavy investments to meet the requirements [1].

Our research identified that early approaches to the RCM were largely manual. Managing the compliance manually is an arduous, extensive and error-prone task. It requires expertise in the field, which costs heavy capital investments for organizations. As a solution, computer-aided RCM sys-

A preliminary version of the article appeared in the Proceedings of The 10th IEEE International Conference on e-Business Engineering (ICEBE 2013) [22].

✉ Muhammad Younas
m.younas@brookes.ac.uk

Krishna Sapkota
krishnasapkota@synapseinformation.com

Arantza Aldea
aaldea@brookes.ac.uk

David A. Duce
daduce@brookes.ac.uk

Rene Banares-Alcantara
rene.banares-alcantara@eng.ox.ac.uk

[1] Synapse Information Ltd, Faraday Wharf, Innovation Birmingham Campus, Science Park, Holt Street, Birmingham B7 4BB, UK

[2] Department of Computing and Communication Technologies, Wheatley Campus, Oxford Brookes University, Oxford OX33 1HX, UK

[3] Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, UK

tems[1],[2],[3],[4] have been developed. However, these systems are still experiencing various challenges to streamline and automate the process. One of the challenges being experienced by the systems is coping with the frequent changes in regulations. With every change in the regulations, the systems should identify the affected processes. Besides, these approaches are proprietary in the sense that the knowledge about the requirements and processes is embedded within the specific codes designed for specific domains and particular purpose. The proprietary knowledge is hard to share and re-use.

The recent approaches are concentrating on using Semantic Web technologies to reduce the manual work [2–15]. These works focus on improving the steps in the regulatory-compliance such as extraction, modelling, mapping and compliance checking. In order to achieve automatic compliance, the legal concepts, for example rights and obligations, have to be extracted and represented [5,6,16], and the business process must be modelled in some meaningful format such as ontology [17–20]. The ontologies or semantic representation of the regulatory guidelines and organizational processes makes the mapping between regulatory guidelines and organizational processes effective and efficient [21–23]. Semantic modelling also helps to improve the compliance checking [2–4,13,14,24]. Although these approaches have contributed on improving the separate steps in regulatory mapping and compliance, they need to be integrated to create a holistic approach. This paper proposes a holistic approach to the mapping between regulatory guidelines and organizational processes.

Representing regulatory knowledge and process knowledge in a standard, homogeneous and interoperable format can improve updating processes and reusability. In particular, modelling the organizational processes in a process-ontology and regulatory guidelines in a regulation-ontology allows the reusability of the knowledge. However, the semantic representation of the processes and regulations needs to be updated in circumstances such as (1) changes in the existing regulatory guidelines or (2) need of the processes to conform to regulations from other regulatory bodies or in other territories. In such cases, mapping of the new regulatory guidelines with the processes constitutes an important step towards updating the affected processes. The automation of the mapping process also contributes to the overall automation of RCM.

The process of automatic mapping between regulatory guidelines and organizational processes comes with various research challenges. Firstly, there is a lack of a standard framework for mapping regulation and process ontologies. Secondly, there are ambiguities and complexities in the regulatory text. Thirdly, there is implicit information in the description of organizational processes. This paper tackles the first challenge: the design and development of an appropriate framework for the mapping. This paper describes RegCMantic framework. A preliminary description of this framework can be found in [22,25]. The contributions of the RegCMantic framework are outlined below.

1. Document-components and predicting document-structure: A document contains various document-components, which constitutes the structure of the document. Some examples of the components are the title, paragraph, headers and footers. In order to extract meaningful regulatory entities from the regulatory text, it is essential to identify the document-components that contain regulatory guidelines. The RegCMantic framework can identify these components and the document-structure.
2. Identification of the regulatory guidelines: From the document-structure, RegCMantic identifies the regulatory guidelines in the document.
3. Identification of the meaningful entities in the regulatory guidelines: Within the regulatory guidelines, this framework identifies the important regulatory entities such as the subject, object, action and obligation. Identification of the regulatory entities helps in relating the regulatory guidelines with organizational processes automatically.
4. Construction of regulatory ontology and the representation of the regulatory entities and regulatory guidelines in the ontology: An ontology to represent the regulatory guidelines and regulatory entities is essential for further processing the information in semantic means. This framework has constructed a regulatory ontology by extending an existing upper-level legal ontology.
5. Similarity between the entities of regulatory guidelines and organizational processes: In order to compute the similarity between a regulatory guideline and an organizational process, it is essential to identify the similarity between their entities. For example, determining the similarity between the subjects and the actions of a regulatory guideline and an organizational process helps in determining the similarity between the guideline and the process. This research computes the similarity between the entities in regulatory guidelines and organizational processes.
6. Similarity between regulatory-statements and organizational processes: A regulatory guideline contains one or more regulatory-statement. Before relating the regulatory guideline to organizational processes, it is essential to relate its statement with the processes. This framework computes the relatedness of a statement with processes.

[1] http://technet.microsoft.com/en-us/library/cc677002.aspx.

[2] http://www.bwise.com/grc-challenges/regulatory-compliance.

[3] http://www-01.ibm.com/software/ecm/compliance/.

[4] http://accelus.thomsonreuters.com/products/accelus-compliance-manager.

7. Similarity between regulatory guidelines and organizational processes: This research determines the relation between a regulatory guideline and an organizational process.

The rest of the paper is organized as follows. The RegCMantic framework is described in Sect. 2. Sections 3 and 4 explain in detail the extraction of the regulations and the mapping between regulations and processes. Section 5 presents and analyses the results obtained from the case study. Section 6 compares the related work and concludes the paper and identifies the future-work.

2 The framework

The RegCMantic framework comprises two main parts: extraction and mapping (see Fig. 1) [16,22,25,26]. In the extraction part, the regulatory guidelines in different document formats, such as PDF, rtf and doc, are converted into a uniform XML format by identifying their document-structures. This process is referred as document-structure analysis (DSA). In the XML document, the regulatory guidelines and the regulatory entities are annotated; this process is described as "Regulatory Entity Annotation." Finally, the annotated entities are extracted and represented in an ontology, which is described as "Regulation-Ontology Population." In the mapping part, a regulatory-statement is compared with an organized process in order to determine the level of relationship or similarity between them.

The comparison depends on three types of similarities: (i) topic-similarity, (ii) core-similarity and (iii) aux-similarity. The three types of similarities are computed from the three types of regulatory entities in a regulation: (i) the topic entities, (ii) core-entities and (iii) the aux-entities. Each step in these two parts is described in the following sections.

3 Extraction part

The extraction part is the first part of the framework and includes three steps: (i) representing the structure of the regulatory guidelines in XML format or DSA, (ii) extracting

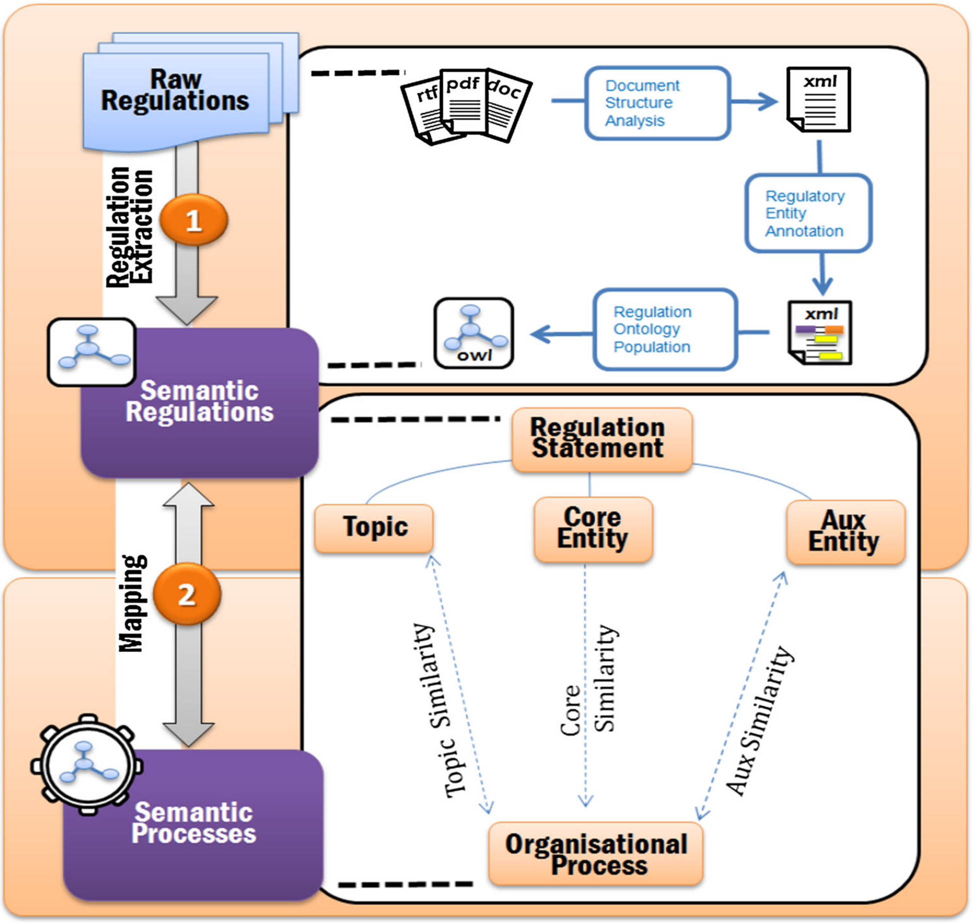


Fig. 1 The RegCMantic framework

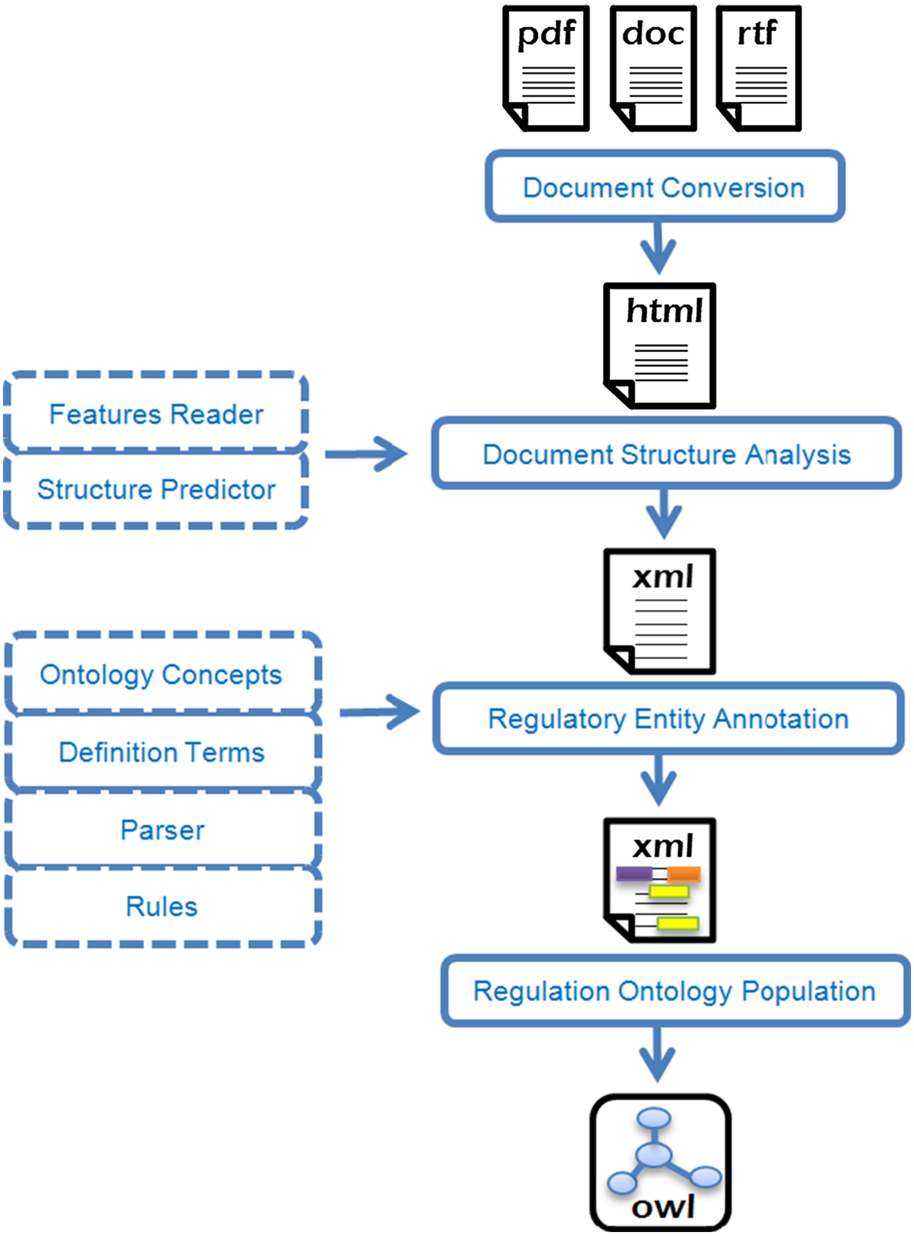


Fig. 2 Regulatory entity extraction in the RegCMantic framework

the meaningful entities from the text (see Fig. 2) and (iii) representing the regulatory guidelines in ontology.

A regulatory document contains several document-components, such as headers, footers, page numbers, footnotes, comments, titles and paragraphs. In order to extract meaningful regulatory entities from regulatory text, it is essential to identify the document-components that contain the regulatory guidelines. In particular, we need to identify regulatory-paragraphs and topics in order to extract regulatory entities. The regulatory-paragraphs or regulations are the paragraphs that impose some restrictions on organizational processes. The restrictions are usually imposed by using modal verbs, such as **must**, **should** and **may**. Once document-components are identified and regulatory entities are extracted, they need to be represented in a semantic format such as ontology. The following steps describe the process in detail.

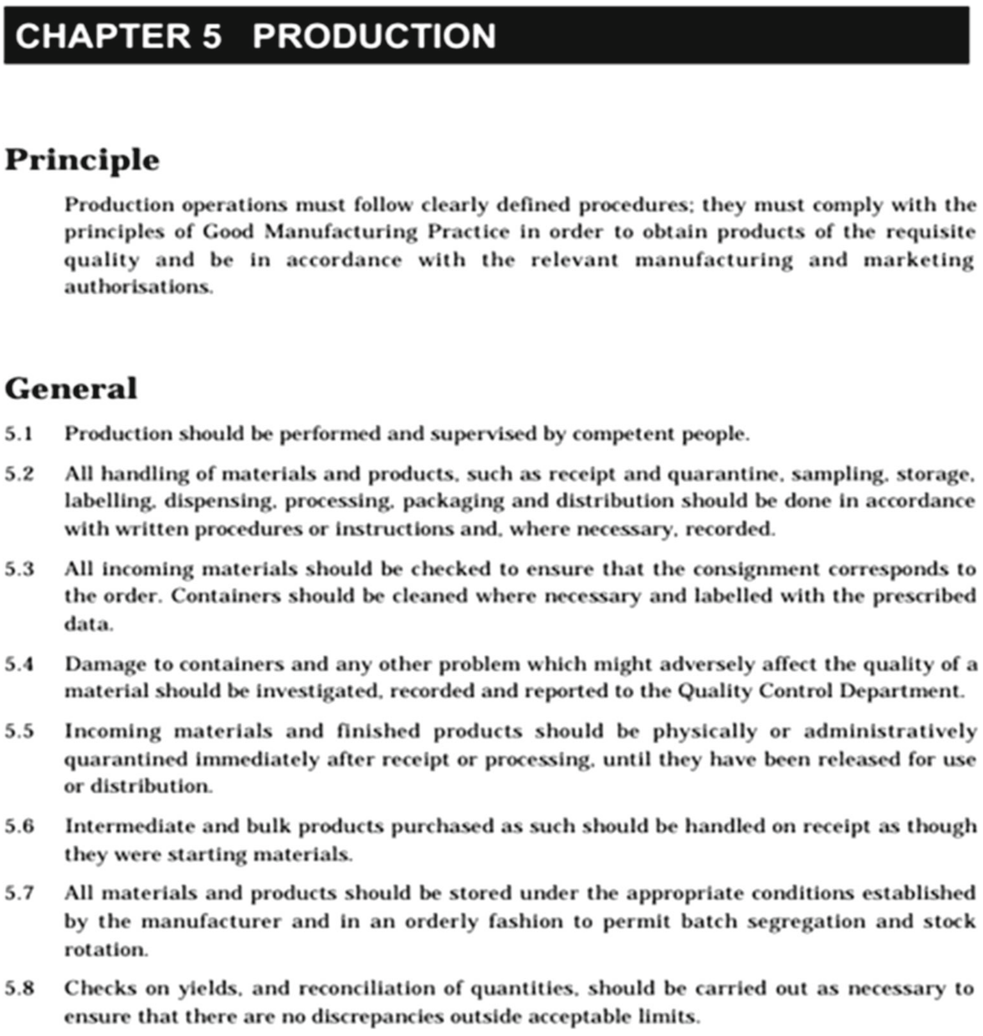
3.1 Document conversion

The regulatory guidelines are available in various document formats, such as PDF, DOC, HTML and XML (e.g. UK,[5] EU[6] and USA[7] regulations for the Pharmaceutical industries). Instead of developing processors for each format, the

[5] http://www.mhra.gov.uk/home/idcplg?IdcService=SS_GET_PAGE&nodeId=613&nodeId=613.

[6] http://ec.europa.eu/health/documents/eudralex/cd/index_en.htm.

[7] http://www.fda.gov/.



CHAPTER 5 PRODUCTION

Principle

Production operations must follow clearly defined procedures; they must comply with the principles of Good Manufacturing Practice in order to obtain products of the requisite quality and be in accordance with the relevant manufacturing and marketing authorisations.

General

5.1 Production should be performed and supervised by competent people.

5.2 All handling of materials and products, such as receipt and quarantine, sampling, storage, labelling, dispensing, processing, packaging and distribution should be done in accordance with written procedures or instructions and, where necessary, recorded.

5.3 All incoming materials should be checked to ensure that the consignment corresponds to the order. Containers should be cleaned where necessary and labelled with the prescribed data.

5.4 Damage to containers and any other problem which might adversely affect the quality of a material should be investigated, recorded and reported to the Quality Control Department.

5.5 Incoming materials and finished products should be physically or administratively quarantined immediately after receipt or processing, until they have been released for use or distribution.

5.6 Intermediate and bulk products purchased as such should be handled on receipt as though they were starting materials.

5.7 All materials and products should be stored under the appropriate conditions established by the manufacturer and in an orderly fashion to permit batch segregation and stock rotation.

5.8 Checks on yields, and reconciliation of quantities, should be carried out as necessary to ensure that there are no discrepancies outside acceptable limits.

Fig. 3 Example regulatory guidelines in the PDF file format

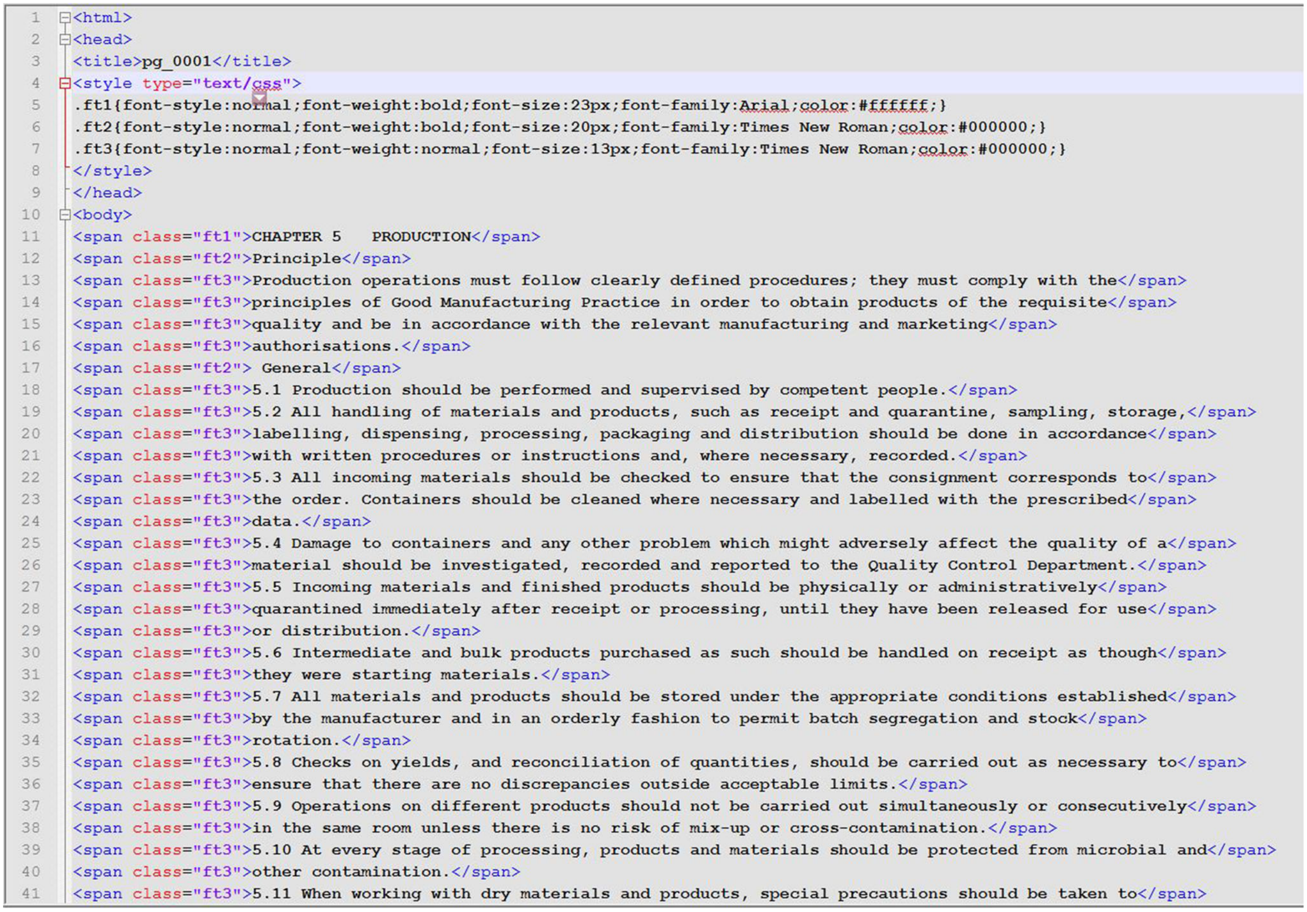
RegCMantic approach is to convert them into a single uniform processing format: HTML. An example of converting regulatory guidelines from PDF file to HTML file is provided in Figs. 3 and 4. There is a fair amount of tools, which convert documents into HTML format. In addition, there are tools available that convert documents into XML formats as well. However, in the RegCMantic framework (see Fig. 2), the documents are first converted from various file formats to HTML and then to XML. They are not directly converted into XML because the direct conversion only converts the document into the XML file format; it does not identify the document-components. The RegCMantic framework represents the structure of a document explicitly, where each document-component is clearly identified and labelled. Converting the files into HTML format preserves the original information such as font-features and the location of the text, which helps in the identification of the document-components. Once identified, the document-components are represented in an explicit (and meaningful) format such as XML.

Figure 4 represents regulatory guidelines in the HTML format, which was created by using an off the shelf HTML converter tool. In this figure, some spaces and tags have been removed to make it clearer to understand in this paper.

3.2 Document-structure analysis (DSA)

In this step, the structure of the regulatory document is identified.

A document contains different types of text having different font-features such as font-size, font-style, font-strength and font-colour. In this framework, the type of the text is called **Text-Type**. A document contains a set of text-type: $T = \{t_1, t_2, \ldots, t_n\}$. For example, the font-size of the title of a document is larger than that of the text in the body; therefore, they can be regarded as two different text-types. For each text-type, a score is computed considering all the font-features and is called **Feature-Score**. The



```
<html>
<head>
<title>pg_0001</title>
<style type="text/css">
.ft1{font-style:normal;font-weight:bold;font-size:23px;font-family:Arial;color:#ffffff;}
.ft2{font-style:normal;font-weight:bold;font-size:20px;font-family:Times New Roman;color:#000000;}
.ft3{font-style:normal;font-weight:normal;font-size:13px;font-family:Times New Roman;color:#000000;}
</style>
</head>
<body>
<span class="ft1">CHAPTER 5   PRODUCTION</span>
<span class="ft2">Principle</span>
<span class="ft3">Production operations must follow clearly defined procedures; they must comply with the</span>
<span class="ft3">principles of Good Manufacturing Practice in order to obtain products of the requisite</span>
<span class="ft3">quality and be in accordance with the relevant manufacturing and marketing</span>
<span class="ft3">authorisations.</span>
<span class="ft2"> General</span>
<span class="ft3">5.1 Production should be performed and supervised by competent people.</span>
<span class="ft3">5.2 All handling of materials and products, such as receipt and quarantine, sampling, storage,</span>
<span class="ft3">labelling, dispensing, processing, packaging and distribution should be done in accordance</span>
<span class="ft3">with written procedures or instructions and, where necessary, recorded.</span>
<span class="ft3">5.3 All incoming materials should be checked to ensure that the consignment corresponds to</span>
<span class="ft3">the order. Containers should be cleaned where necessary and labelled with the prescribed</span>
<span class="ft3">data.</span>
<span class="ft3">5.4 Damage to containers and any other problem which might adversely affect the quality of a</span>
<span class="ft3">material should be investigated, recorded and reported to the Quality Control Department.</span>
<span class="ft3">5.5 Incoming materials and finished products should be physically or administratively</span>
<span class="ft3">quarantined immediately after receipt or processing, until they have been released for use</span>
<span class="ft3">or distribution.</span>
<span class="ft3">5.6 Intermediate and bulk products purchased as such should be handled on receipt as though</span>
<span class="ft3">they were starting materials.</span>
<span class="ft3">5.7 All materials and products should be stored under the appropriate conditions established</span>
<span class="ft3">by the manufacturer and in an orderly fashion to permit batch segregation and stock</span>
<span class="ft3">rotation.</span>
<span class="ft3">5.8 Checks on yields, and reconciliation of quantities, should be carried out as necessary to</span>
<span class="ft3">ensure that there are no discrepancies outside acceptable limits.</span>
<span class="ft3">5.9 Operations on different products should not be carried out simultaneously or consecutively</span>
<span class="ft3">in the same room unless there is no risk of mix-up or cross-contamination.</span>
<span class="ft3">5.10 At every stage of processing, products and materials should be protected from microbial and</span>
<span class="ft3">other contamination.</span>
<span class="ft3">5.11 When working with dry materials and products, special precautions should be taken to</span>
```

Fig. 4 Regulatory guidelines converted into an HTML file format

main influencing factor for the feature-score is the font-size. This means that the larger the font-size, the higher the feature-score. A document contains a set of feature-scores: $S = \{s_1, s_2, \ldots, s_n\}$. A level is defined for each text-type based on its feature-score and is called **Text-Level**. This means that the higher the feature-score, the higher the text-level. A document contains a set of text-level: $L = \{l_1, l_2, \ldots, l_n\}$ for a set of text-type. In the set of the text-levels, the order of the levels is: $l_1 > l_2 > \cdots > l_n$.

Example In the text in Fig. 3, there are three text-types $t1$, $t2$ and $t3$ representing chapter, section and paragraph, respectively. The first line of text "Chapter 5 Production" has the highest feature-score:

$$\begin{aligned} s1 &= \text{font-size} \times 10 + \text{font-bold} \\ &= 23 \times 10 + 2 \\ &= 232 \end{aligned}$$

The text in "Principal" and "General" has the second highest feature-score:

$$\begin{aligned} s2 &= \text{font-size} \times 10 + \text{font-bold} \\ &= 20 \times 10 + 2 \\ &= 202 \end{aligned}$$

The text in the paragraphs starting with some numbers has the feature-score lower than the above two:

$$\begin{aligned} s3 &= \text{font-size} \times 10 + \text{font-normal} \\ &= 13 \times 10 + 0 \\ &= 130 \end{aligned}$$

We have three feature-scores $s1$, $s2$ and $s3$ for three text-types $t1$, $t2$ and $t3$,respectively. Now we can assign levels: $l1$, $l2$ and $l3$ for $t1$, $t2$ and $t3$, respectively.

Similarly, a document has a set of **Document-Components**: which are denoted by $C = \{c_1, c_2, \ldots, c_n\}$ such as chapter, section, subsection, paragraph and page numbers. The document-components specify the structure of a document. Usually, they follow a hierarchical structure depending on the text-level of each text-type. In summary, each text-type is labelled with a text-level considering its feature-score, and each text-level is labelled with a document-component considering the document-component prediction algorithms.

When the document-components are identified, they are represented in an XML file. In order to create the XML file, two processors are implemented: **Feature Reader** and **Structure Predictor** as shown in Fig. 2.

The **Features Reader** identifies the document features such as font-style, font-weight, font-family, font-colour and text-content. Reading the sufficient amount of document features helps in processing the index for each document-component.

Based on the document features, the **Structure Predictor** infers the components of the document. The paragraph is the main document-component, which helps determine the regulation. Therefore, among the document-components, at first, the paragraph is identified. Then, the other components are investigated based on their preceding text or label. A series of algorithms is implemented in order to predict the structure of the document; the structure is presented in a user interface, where a user verifies the suggested structure.

3.2.1 Paragraph prediction

In the set of text-levels L, each text-level $l$determines (i) how much text it contains, (ii) how many sentences it has, (iii) how many obligatory words, such as **must** and **should**, has and (iv) how far its font-size is from the standard font-size of a paragraph text.

The prediction of a text as a paragraph requires computing the paragraph index of the text. Moreover, it needs to compute the indices of sentence, text, obligation and deviation. A sentence index is the percentage of the sentences in a text-level. The text index of a text-level is the percentage of its text-content. The obligation index of a text-level is the percentage of the obligatory words in the text. The deviation index of a text-level is the percentage of the distance of the text-level from the text-level of a standard paragraph. In general, the font-size of a paragraph is 12px, and it is not bold and italic. A paragraph index prediction is the average value of the weighted values of these four indices. The text in the text-level that has the highest paragraph index is regarded as the paragraph (see Algorithm 1).

Example Following from the previous example, there are three text-types in Fig. 3: $t1$, $t2$ and $t3$. The feature-score of a typical paragraph is computed as

$$\begin{aligned} s_p &= \text{font-size} \times 10 + \text{font-weight} \\ &= 12 \times 10 + 0 \\ &= 120. \end{aligned}$$

In this case, the closest feature-score to the paragraph is that of $t3$ (i.e. 130). This suggests that $t3$ is most likely to be a paragraph. Similarly, three other factors also suggest that $t3$ is a paragraph: the amount of text in $t3$ is the highest; $t3$ has the highest number of sentences; and there are more modal verbs in $t3$.

ALGORITHM 1. Paragraph Prediction

```
Input:     L  is a set of text-level in the document.
Output:  L is a new set of text- levels with the  predicted text-level for the paragraph
Function:  PREDICT-PARAGRAPH(L) returns L

    i = 0, $l_k$ = null
    L  = {$l_1$, $l_2$, .. , $l_n$}
    for each $l_i \in L$
        j = COMPUTE-PARA-PREDICTION-INDEX($l_i$)
        if (j >i) then
            $l_k = l_i$
            i = j
        end if
    end for
    $l_k$.SET-COMPONENT(paragraph)
    return L
```

3.2.2 Indicator-based prediction

When the paragraph prediction is completed, the next process will predict the rest of the text-levels based on its preceding label or text also referred to as indicators. In many cases, the document-components with higher text-level, such as **part**, **chapter** and **section**, are preceded with the relevant text such as "Chapter 5 Production" and "Sect. 5.3 Starting Materials." When a text-level with an indicator is found, the document-component of the text-level is determined by the indicators. For example, if the text in the text-level l_1 starts with "Chapter," then the document-component of the text-level l_1 will be set to chapter (see Algorithm 2).

Example Following from the previous example, the $t3$ has been suggested as the paragraph in Fig. 3. Now, we need to identify the document-component of $t1$ and $t2$. The text-type $t1$ is preceded with an indicator term "Chapter," which suggests that $t1$ is a chapter.

```
ALGORITHM 2.   Paragraph Based on the Indicator Text
Input: C is a set of document-components (document-structure). L is a set of text- level in the document.
Output: L is a new set of text- levels in the document with document structure values computed from the preceding
text
Function: PREDICT-COMPONENT-WITH-INDICATOR(C, L) returns L
    $C = \{c_1, c_2, .., c_n\}$
    $L = \{l_1, l_2, .., l_n\}$
    for each $l_i \in L$
        $c_i$ = GET-COMPONENT($l_i$)
        $text$ = GET-INDICATOR-TEXT($l_i$)
        if ($c_i = null$) then
            for each $c_j \in C$
                if ($text = c_j$) then
                    $c_i = c_j$
                end if
            end for
        end if
    end for
return L
```

```
ALGORITHM 3.   Predicting the Remaining Structure of the Document
Input: C is a set of possible document-component s (document-structure). L is a set of text-levels in the document.
Output: L is a new set of text-levels in the document with document structure values computed from the preceding
text
Function: PREDICT-REMAINING-COMPONENT(C, L) returns L
    $C = \{c_1, c_2, .., c_n\}$
    $L = \{l_1, l_2, .., l_n\}$
    for each $l_i \in L$
            $c_i$ = GET-DOCUMENT-COMPONENT($l_i$)
            $c_{i+1}$ = GET-DOCUMENT-COMPONENT($l_{i+1}$)
            if ($c_i = null$) then
                $c_1 \in C$
                $c_i = c_1$
            end if
            if ($c_i \neq null$ or $c_{i+1} = null$) then
                for each $c_j \in C$
                    if ($c_i = c_j$) then
                        $c_{i+1} = c_{j+1}$
                    end if
                end for
            end if
    end for
return L
```

3.2.3 Prediction based on empirical values

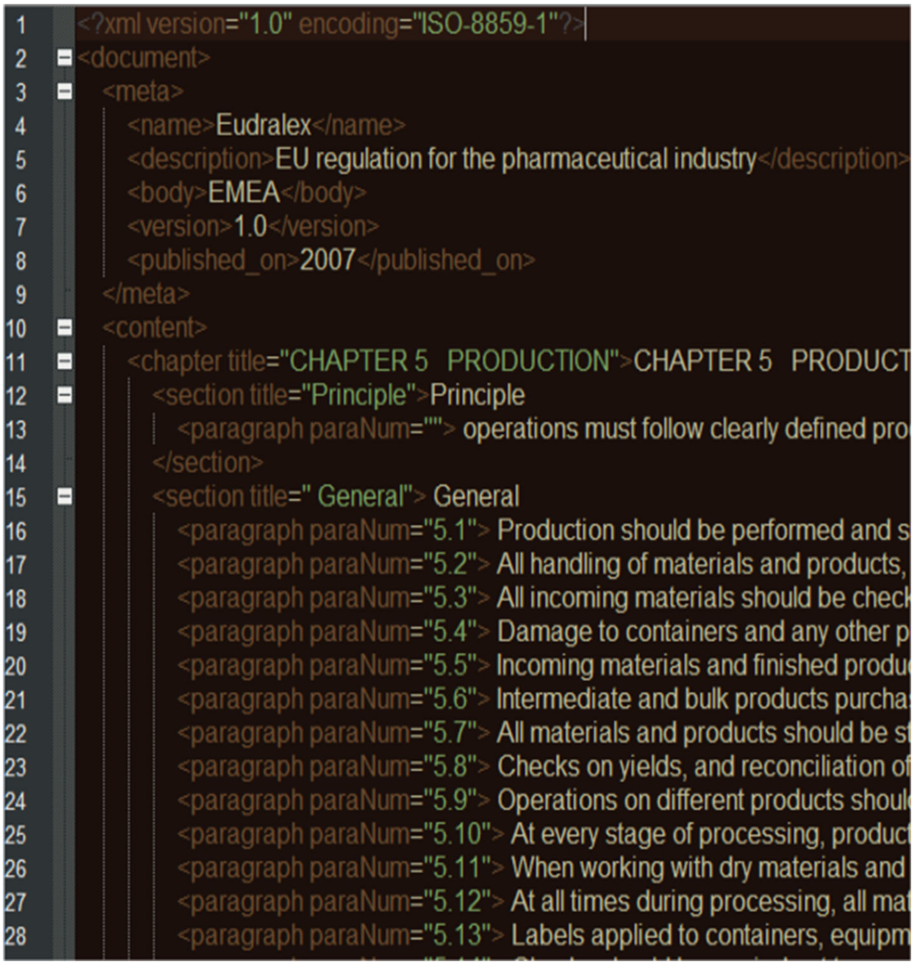
The predictions of the text-levels that have not been completed yet are computed based on the proximity of empirical values (see Algorithm 3). Based on the proximity, the algorithm predicts the closest document-component with respect to an empirically created hierarchical component set $C = \{c1, c2, \ldots, cn\}$. When there are many possible document-components for a text-level, the document-component of the text-level is determined as the closest one to the highest predicted document-component.

Example Following from the previous example, in Fig. 3, $t1$ and $t3$ have been suggested as `chapter` and `paragraph`, respectively. Now, we need to identify the document-component of $t2$. The empirical value suggests that the document-components between `chapter` and `paragraph` are `section` and `subsection`. In this case, the document-component closest to chapter is a section. Therefore, it suggests that $t2$ is a `section`.

The predicted document-structures are presented to users via a GUI. Users, then, are able to select, analyse and modify the suggested document-structures.

3.2.4 XML regulation

Following the earlier steps, the HTML document is converted into XML (see Fig. 5). The conversion is an important step since it identifies a different document-components in a document and represents the document-components in an explicit format. When the document-components are explic-



```
<?xml version="1.0" encoding="ISO-8859-1"?>
<document>
  <meta>
    <name>Eudralex</name>
    <description>EU regulation for the pharmaceutical industry</description>
    <body>EMEA</body>
    <version>1.0</version>
    <published_on>2007</published_on>
  </meta>
  <content>
    <chapter title="CHAPTER 5  PRODUCTION">CHAPTER 5  PRODUCT
      <section title="Principle">Principle
        <paragraph paraNum=""> operations must follow clearly defined pro
      </section>
      <section title=" General"> General
        <paragraph paraNum="5.1"> Production should be performed and s
        <paragraph paraNum="5.2"> All handling of materials and products,
        <paragraph paraNum="5.3"> All incoming materials should be check
        <paragraph paraNum="5.4"> Damage to containers and any other p
        <paragraph paraNum="5.5"> Incoming materials and finished produ
        <paragraph paraNum="5.6"> Intermediate and bulk products purcha
        <paragraph paraNum="5.7"> All materials and products should be st
        <paragraph paraNum="5.8"> Checks on yields, and reconciliation of
        <paragraph paraNum="5.9"> Operations on different products shoul
        <paragraph paraNum="5.10"> At every stage of processing, product
        <paragraph paraNum="5.11"> When working with dry materials and
        <paragraph paraNum="5.12"> At all times during processing, all mat
        <paragraph paraNum="5.13"> Labels applied to containers, equipm
```

Fig. 5 An example of the regulatory guidelines represented in the XML representation format

itly labelled or represented, it helps in the extraction of specific entities from specific document-components. Note that, in rare situations, if regulators publish the regulation-documents in a standard and explicit format, the previous two steps may not be necessary. However, this is not a common practice; those stages constitute an important part of the process.

The most important document-component is paragraph because the regulatory guidelines are represented in paragraphs. A regulation-document contains several paragraphs; however, not all the paragraphs are regulatory guidelines. In this framework, a paragraph containing regulatory guidelines is called **regulation** or **regulation-paragraph**; a sentence within in a regulation-paragraph is called **regulation-statement**.

3.3 Regulatory entity annotation

A regulation-statement contains regulation-entities, such as subject, obligation and action, which help express regulatory requirements. A **subject** is a regulation-entity, upon which the requirements are imposed. For example, in a regulation-statement "*Equipment should be cleaned after processing*," the word **Equipment** is the subject. In a regulation-statement, a subject can be equipment, substance, person, document or a process. The text in a regulation-document contains some modal verbs such as **should**, **must** and **shall**. These modal verbs are the means of expressing the requirements of a regulatory guideline and are called **obligations**. The strength of the obligations may also vary from soft and medium to strong; for example, **shall**, **should** and **must** are the soft, medium and strong obligations, respectively. An **action** is a regulation-entity that has to be performed in order to comply with some requirements and expectations. Usually, the action is the main verb in a sentence; however, sometimes the verb may be modified to different grammatical forms such as nouns and adjectives. In the example described above, **cleaned** is the action. The three entities subject, obligation and action are called core-entities. Beside the core-entities, there are other entities that express time, place, reason and quality, and they are called **auxiliary-entities** or **aux-entities**.

In the process of regulatory entity annotation, the RegCMantic framework identifies the regulatory constraints in organizational processes. The first task in this process is to identify the regulation-statements. In each regulation-statement, it annotates the regulation-entities. For the annota-

Table 1 An example of a parsed text

Natural text	*Starting materials should only be purchased from approved suppliers named in the relevant specification and, where possible, directly from the producer*

Parsed text (typed dependencies)	`amod(materials-2, Starting-1)`
	`nsubjpass(purchased-6, materials-2)`
	`aux(purchased-6, should-3)`
	`advmod(purchased-6, only-4)`
	`auxpass(purchased-6, be-5)`
	`root(ROOT-0, purchased-6)`
	`prep(purchased-6, from-7)`
	`amod(suppliers-9, approved-8)`
	`pobj(from-7, suppliers-9)`
	`partmod(suppliers-9, named-10)`
	`prep(named-10, in-11)`
	`det(specification-14, the-12)`
	`amod(specification-14, relevant-13)`
	`pobj(in-11, specification-14)`
	`cc(specification-14, and-15)`
	`dep(possible-18, where-17)`
	`dep(specification-14, possible-18)`
	`conj(specification-14, directly-20)`
	`prep(named-10, from-21)`
	`det(producer-23, the-22)`
	`pobj(from-21, producer-23)`

tion, it uses four main components: natural language parser, ontology concepts, definition terms and IE rules.

3.3.1 Natural language parser

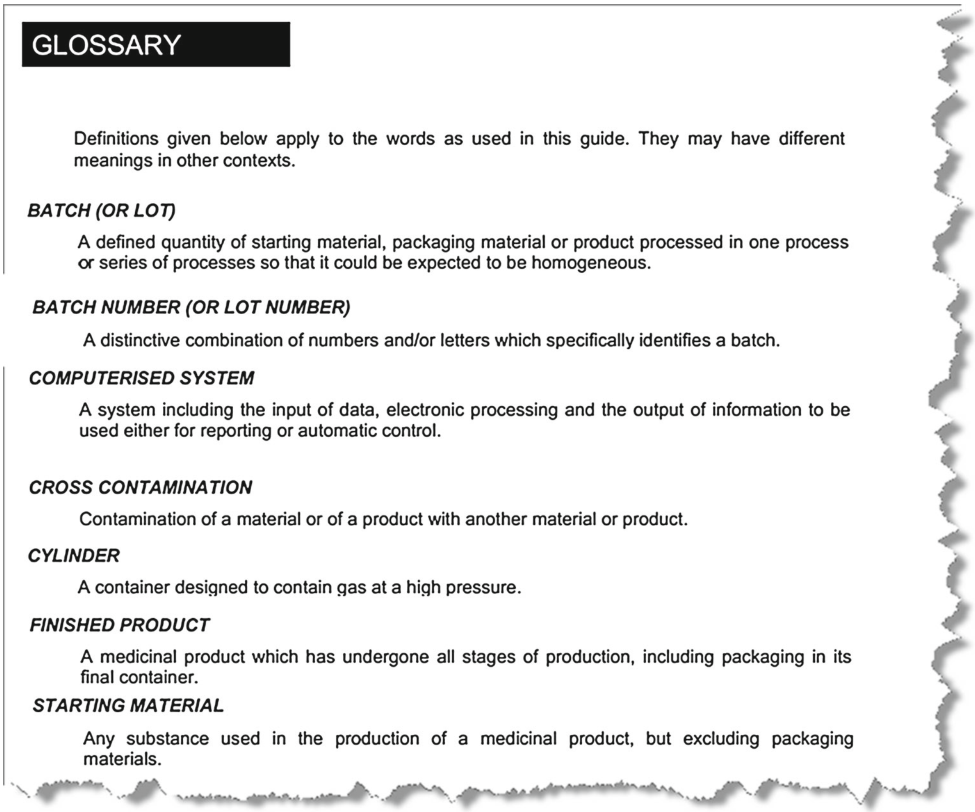
Natural language parsers interpret a sentence in terms of its grammatical structure. In particular, it identifies grammatical units and their relationship in the sentence such as subject, verb, object, preposition and determiners (see Table 1). Breaking down a regulation-statement into subject-containing chunk, object-containing chunk, action-containing chunk and complementary chunk helps in identifying the regulation-entities in a sentence accurately. For example, if a concept or a term is identified in a regulation-statement, and the position of the concept or the term is located within a subject-containing chunk, it verifies that it is a subject. In this process, a parser is used with some rules to identify the special chunks such as condition-chunk, subject-chunk, obligation-chuck, action-chunk, complement-chunk, where-chuck, when-chunk, why-chunk and how-chunk.

3.3.2 Ontological concepts

The ontological concepts defined in a domain are useful for IE. For example, in the Pharmaceutical industry, some concepts in the process-ontology are *Equipment*, *Substance* and *Filtering*. Using these concepts, and their synonyms and hyponyms, the RegCMantic framework can identify meaningful entities in the regulatory guidelines. In order to achieve this, a list of concepts is created from the process-ontology. Misleading concepts or the parts of the concepts should be removed. In this framework, these concepts are referred to as “Domain Specific Stop-Words.” Some examples of the domain specific stop-words in the Pharmaceutical industry, as in the OntoReg ontology, are *Action*, *Module, Entity and Domain* in *Equipment_Module*, *Physical_Entity*, *Abstract_Entity* and *Process_Domain*, respectively. The stop-words are removed from the list of ontological concepts before using them for the annotation.

3.3.3 Definition terms

Regulatory guidelines are usually provided with definition terms. The definition terms in regulatory documents are also known as introductory terms or glossary, and they are provided at the beginning of the documents. The terms are provided with their definition and the context in which they are being used (see Fig. 6). These terms help in understanding the semantic of the regulatory guidelines and the annotation of the regulatory entities in the text. Similar to the list of



GLOSSARY

Definitions given below apply to the words as used in this guide. They may have different meanings in other contexts.

BATCH (OR LOT)

A defined quantity of starting material, packaging material or product processed in one process or series of processes so that it could be expected to be homogeneous.

BATCH NUMBER (OR LOT NUMBER)

A distinctive combination of numbers and/or letters which specifically identifies a batch.

COMPUTERISED SYSTEM

A system including the input of data, electronic processing and the output of information to be used either for reporting or automatic control.

CROSS CONTAMINATION

Contamination of a material or of a product with another material or product.

CYLINDER

A container designed to contain gas at a high pressure.

FINISHED PRODUCT

A medicinal product which has undergone all stages of production, including packaging in its final container.

STARTING MATERIAL

Any substance used in the production of a medicinal product, but excluding packaging materials.

Fig. 6 An example of definition terms

ontological concepts, a list of definition terms is created for the annotation.

3.3.4 Information extraction rules

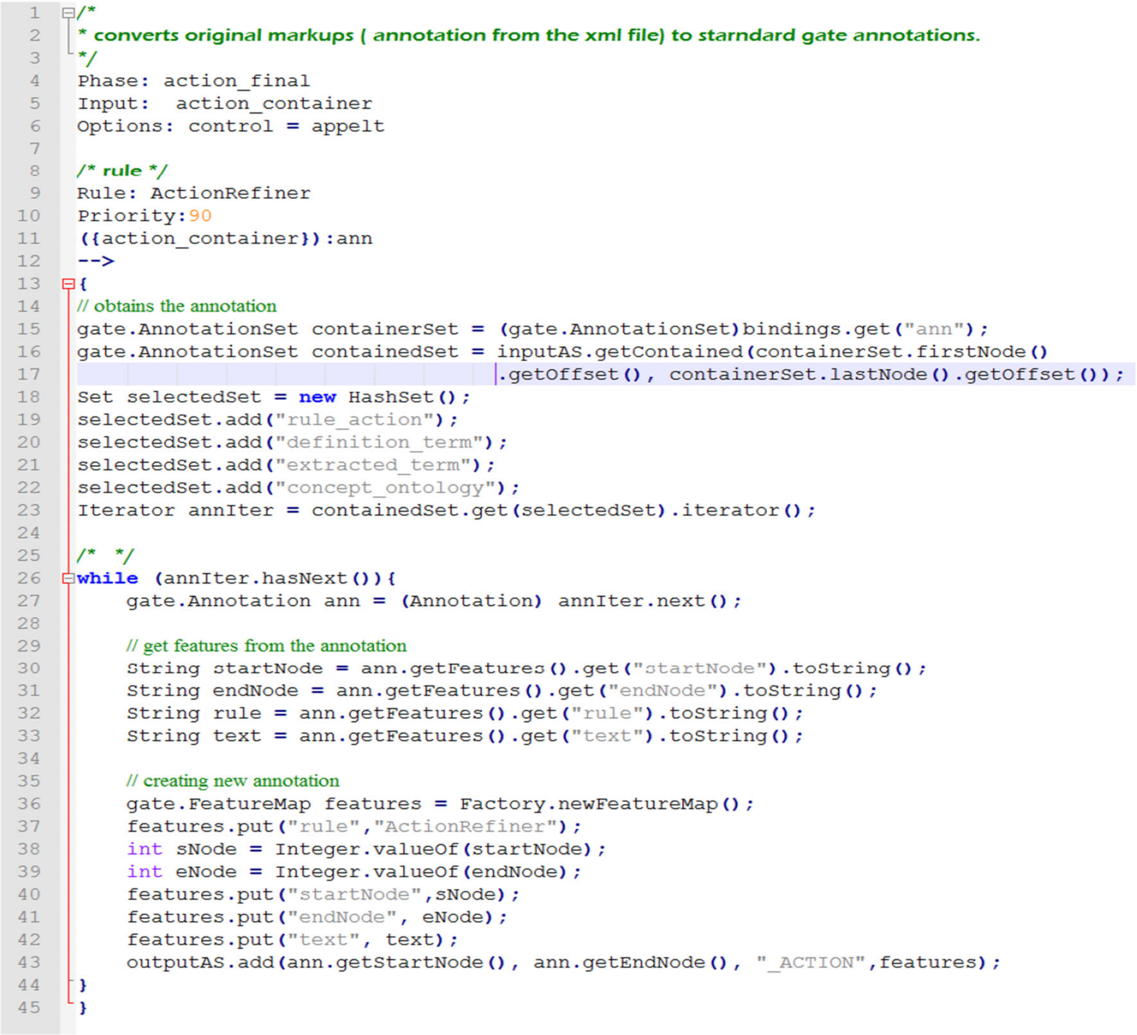
Application of pattern matching rules is regarded as an established IE technique [27]. As an advancement on the regular expression technology, some rule specification languages are being used as state-of-the-art tools such as Common Pattern Specification Language (CPSL) [28]. Java Annotation Pattern Engine (JAPE) [29] is an example of implementation of the CPSL (see Fig. 7). These rules typically have patterns on the left-hand side (LHS) as their conditions, and actions to be performed on the right-hand side (RHS). A typical example of the actions on the RHS is the annotation.
Therefore, the application of these rules helps annotate the text if a specified pattern is met. In this step, the rules incorporate all the above annotations and create a new set of annotations and/or confirm the existing annotations.

In Fig. 7, line 5 indicates that it takes input the annotation called “action_container.” Line 6 determines what type of option is applied to the rule. Line 9 defines the rule name, and line 10 defines the priority of the rule. In this example, it takes “action_container” as the annotations to process from the LHS. In the RHS, the annotations are processes using Java. Lines 15–16 accept the annotations passed from the LHS. Similarly, lines 18–22 define the names of the annotations that need to be processed. Finally, lines 26–43 process the annotations and output the results.

In summary, ontological concepts help to identify the synonyms and hyponyms of the concepts in regulatory guidelines. Rules such as JAPE [29] help in specifying the grammar for pattern matching and incorporating the entities identified by ontological concepts. Similar to ontological concepts, the definition terms, provided by the regulatory document creators, can help in the identification of the regulatory terms, their synonyms and hyponyms. A lexical parser can be used to separate different grammatical units in a sentence; this helps in the identification of the important chunks in a sentence such as subject-containing chunk and action-containing chunk.

3.4 Semantic representation of regulatory guidelines

The semantic representation is the population of regulatory ontology with the extracted regulatory entities such as subject, action, obligation and modifiers. Representing regulatory guideline in semantic models such as ontology helps in the automation of RCM. For the population, ontology with appropriate concepts is required. The ontology creation and population processes are described below.



```
/*
* converts original markups ( annotation from the xml file) to starndard gate annotations.
*/
Phase: action_final
Input:  action_container
Options: control = appelt

/* rule */
Rule: ActionRefiner
Priority:90
({action_container}):ann
-->
{
// obtains the annotation
gate.AnnotationSet containerSet = (gate.AnnotationSet)bindings.get("ann");
gate.AnnotationSet containedSet = inputAS.getContained(containerSet.firstNode()
                                  .getOffset(), containerSet.lastNode().getOffset());
Set selectedSet = new HashSet();
selectedSet.add("rule_action");
selectedSet.add("definition_term");
selectedSet.add("extracted_term");
selectedSet.add("concept_ontology");
Iterator annIter = containedSet.get(selectedSet).iterator();

/*  */
while (annIter.hasNext()){
    gate.Annotation ann = (Annotation) annIter.next();

    // get features from the annotation
    String startNode = ann.getFeatures().get("startNode").toString();
    String endNode = ann.getFeatures().get("endNode").toString();
    String rule = ann.getFeatures().get("rule").toString();
    String text = ann.getFeatures().get("text").toString();

    // creating new annotation
    gate.FeatureMap features = Factory.newFeatureMap();
    features.put("rule","ActionRefiner");
    int sNode = Integer.valueOf(startNode);
    int eNode = Integer.valueOf(endNode);
    features.put("startNode",sNode);
    features.put("endNode", eNode);
    features.put("text", text);
    outputAS.add(ann.getStartNode(), ann.getEndNode(), "_ACTION",features);
}
}
```

Fig. 7 An example of a JAPE rule

3.4.1 Regulation-ontology creation

In order to represent the regulatory guidelines semantically, a regulatory ontology called SemReg is created. It is recommended [30] that the ontology engineer should employ the concepts of the existing ontologies in a similar domain and that of the upper ontologies. Therefore, the LKIF-Core ontology [31,32] is considered for the SemReg engineering. The LKIF ontology is the recent development in the legal domain, and it has defined the appropriate level of concepts. These concepts are extended to the application-level concepts and populated with the extracted entities. Although it is a core ontology, in order to adapt the concepts in the pharmaceutical domain, further concepts are created. Among the concepts are *Subject*, *Obligation*, *Action*, *Regulation*, *Statement*, *Time*, *Place*, *Intention* and *Evaluative Expression*. Figure 8 shows the extension of the LKIF-Core concepts in the SemReg ontology. In this figure, big boxes with dark borders are the extended concepts and the other boxes are the concepts in LKIF-Core ontology (please refer to [33] for detailed information about this ontology).

3.4.2 The SemReg ontology population

Ontology population is a process where ontological classes are populated with instances. After the identification and annotation of the regulatory entities in the regulatory guidelines, they are converted into the instances of the SemReg ontological classes (see Fig. 9); the regulatory guidelines are called semantic regulations. In other words, the semantic regulations are the regulations represented in an ontology. Semantic representation helps process the regulations efficiently. The process of converting regulatory guidelines from text to semantic format has also been described in [26].

Figure 9 displays the SemReg ontology.

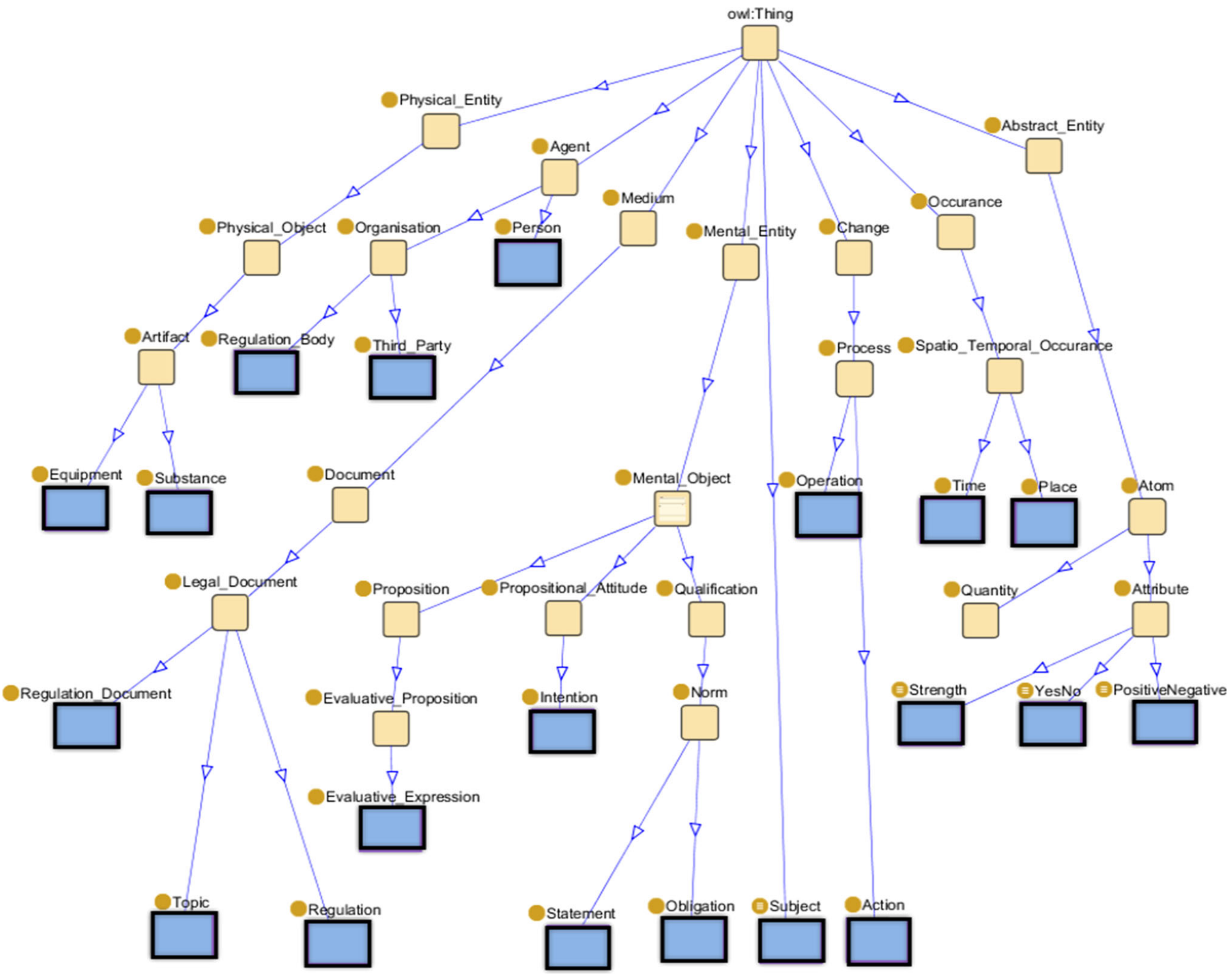


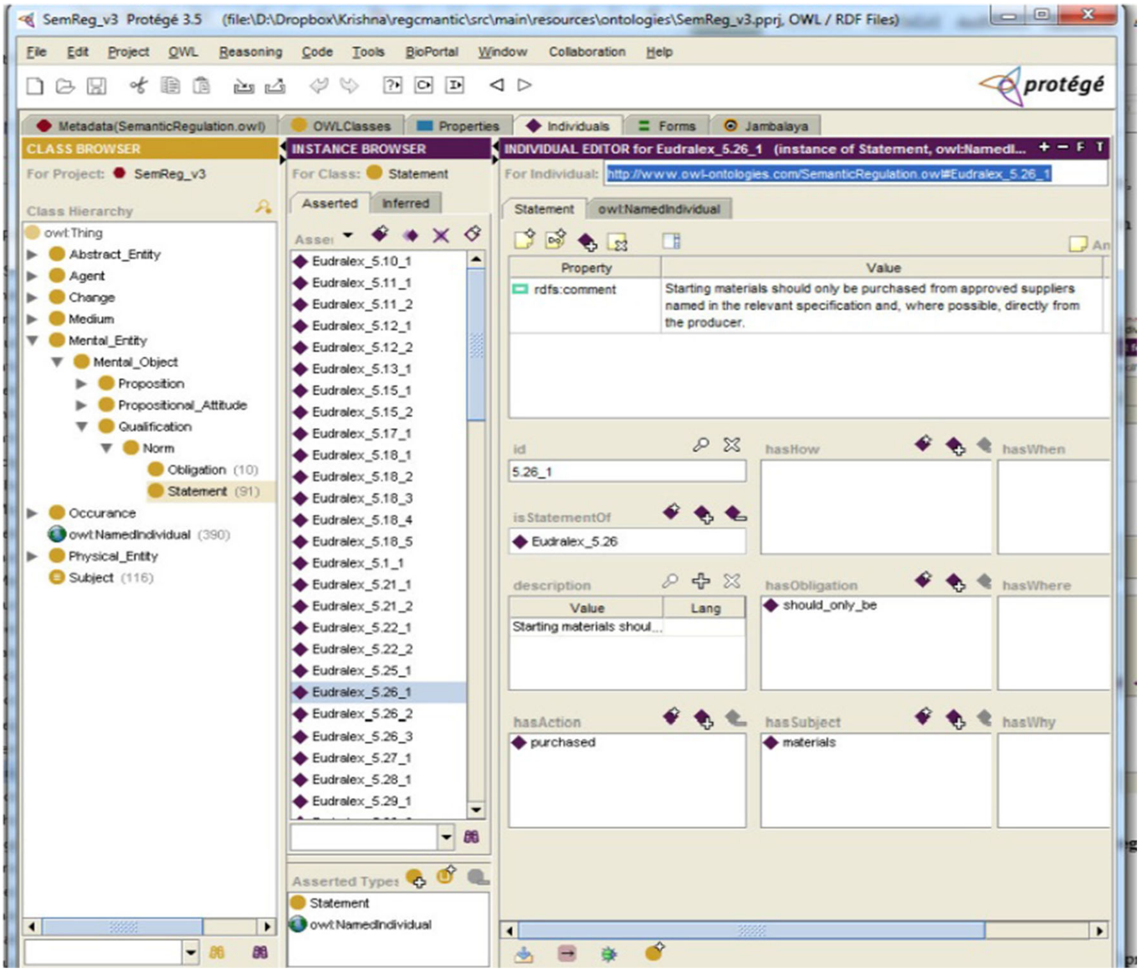
Fig. 8 Concepts in the SemReg ontology

On the left panel or class browser, it is showing hierarchies of classes preceded with circles. The classes also indicate the number of individuals they contain. For example, in the selected class *Statement*, there are 91 individuals. On the middle panel or instance browser, it is enlisting the individuals of the class *Statement*, which are indicated by purple diamonds. On the right panel or individual editor, it is displaying the properties of the individual *Eudralex_5.26_1* such as *id*, *description*, *isStatementOf*, *hasSubject*, *hasObligation* and *hasAction*.

4 Mapping part

It is the second part of the framework, which identifies the relationship between the regulatory guidelines and the organizational processes by using the regulatory entities extracted from the first part of the framework. In particular, it needs two ontologies: a regulation-ontology representing regulatory guidelines and a process-ontology representing organizational processes. The development of a process-ontology was not the scope of this research, and therefore, a process-ontology, OntoRegd, developed by the Engineering Science Department in the University of Oxford [34] has been used. In the OntoReg ontology, a validation-task (Task) is the smallest unit of an organizational process that is used for compliance checking. The two most important concepts associated with a validation-task are subject (Sub) and action (Act). Figure 11 displays a validation-task `S101_PurchasingTask`, which is associated with a subject, `SalicyclicAcid` and an action, `Purchasing101`, respectively.

In the mapping part, three similarity scores are computed: (1) topic-similarity, (2) core-entity similarity and (3) auxiliary-entity similarity. Figure 10 shows the computation of the three types of similarities. Figure 11 depicts a mapping between a regulation and a validation-task in the regulation-ontology SemReg and the process-ontology OntoReg. The



For Individual: http://www.owl-ontologies.com/SemanticRegulation.owl#Eudralex_5.26_1

Property	Value
rdfs:comment	Starting materials should only be purchased from approved suppliers named in the relevant specification and, where possible, directly from the producer.

id: 5.26_1

isStatementOf: Eudralex_5.26

description:

Value	Lang
Starting materials shoul...	

hasObligation: should_only_be

hasAction: purchased

hasSubject: materials

Fig. 9 An example of the population of a regulatory ontology in Protégé

steps involved in the similarity computation are described separately in the following subsections.

4.1 Conceptual distance computation

In the similarity computation, the similarity between an individual in the regulatory ontology and an individual in the process-ontology is identified. Although some concepts look like very similar to each other in a general context, they can be different from each other in terms of their intentions in a specific context. For example, the concepts *substance* and *equipment* are closely related in the WordNet ontology, whereas in the OntoReg ontology, they are defined as different from each other. In the RegCMantic framework, the distance between two concepts in the OntoReg ontology is computed considering the axiom `disjointWith`. Currently, the value becomes 1 or 0 considering their disjointness, but in the future, we aim to consider the semantic distance computation algorithm [35] to determine the value. After the conceptual difference computation, a table is created; each row in the table is represented by $<c_1, c_2, \delta>$, where c_1 and c_2 are two concepts in the ontology and δ is the difference-value between the concepts.

4.2 Three types of similarity score computation

In a regulation-ontology, regulations (Reg) are placed under a hierarchy of topics (Topic) such as part, chapter, section and subsection. A regulation contains one or more regulation-statement (Stmt). A regulation-statement comprises core-entities (Core) and auxiliary-entities (Aux). The core-entities represent subject (Sub) and action (Act); the auxiliary-entities represent extra information such as time, place and purpose. An example of the regulatory text depicting topics, core-entities and auxiliary-entities, such as action modifier, is presented in Fig. 12.

In this framework, three types of similarities are computed: (1) topic-similarity (Topic vs. Task), (2) core-entity similarity (Core vs. Task) and (3) auxiliary-entity similarity (Aux vs. Task).

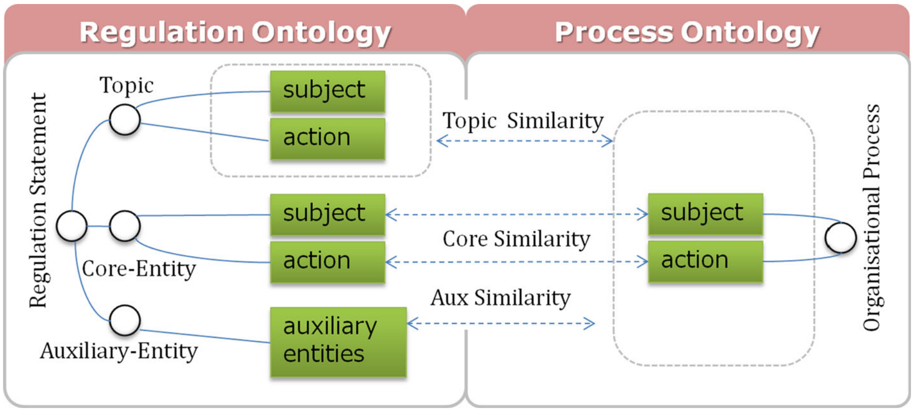


Fig. 10 Three different types of similarity computations in the RegCMantic framework

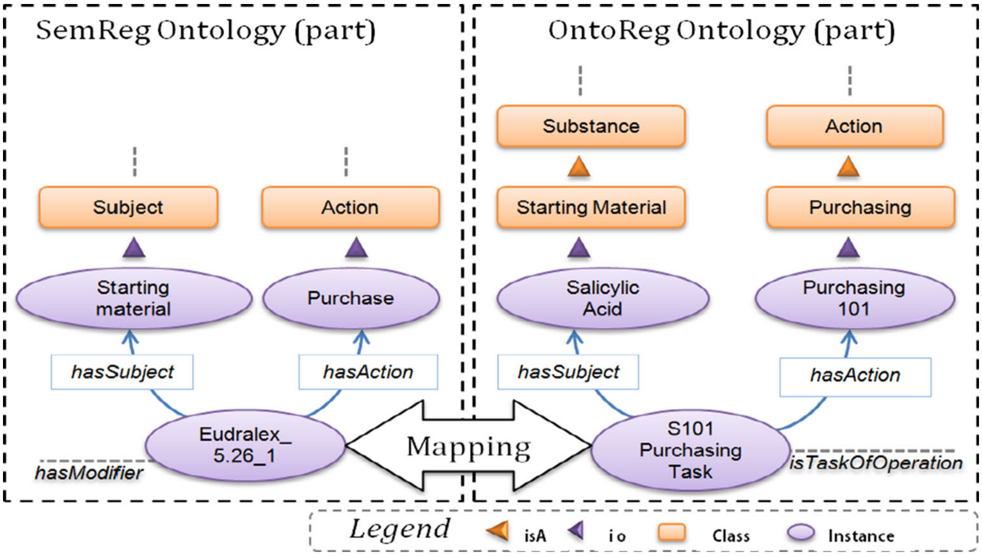


Fig. 11 Mapping between a regulation and a validation-task (process) using regulation and process ontologies

In the core-entity similarity, each individual in a regulation-statement is compared with that of a validation-task. Since the individuals are associated with their subjects and actions, the similarity scores for the subjects and the actions are computed separately. The similarity score between two words is computed using the popular Lin-similarity [36]. The Lin-similarity considers the hierarchical structure of the terms in a lexical ontology, WordNet [37] and information content value (IC) of the terms in large corpora. It identifies the lowest common subsumer (LCS) between two compared words, computes the depth of the LCS from the root, measures the distance between the two compared terms via the LCS and applies the IC values obtained from large corpora to compute the similarity measure. The subject-score computation results into a set of similarity scores. The highest similarity score among them is selected as the similarity score of the subjects.

Algorithm 4 shows the similarity computation between a regulation-subject and a process-subject. Initially, the score is set to zero, which will be updated with the computed value. Consider there are two sets of subjects: S_r from the regulation-statement and S_t from the validation-task. Now, we compare each word in these sets. The difference-value δ is obtained from the difference-table, which is created from the process-ontology. If the two words are not defined as different in the process-ontology, only then, the similarity score between them is computed.

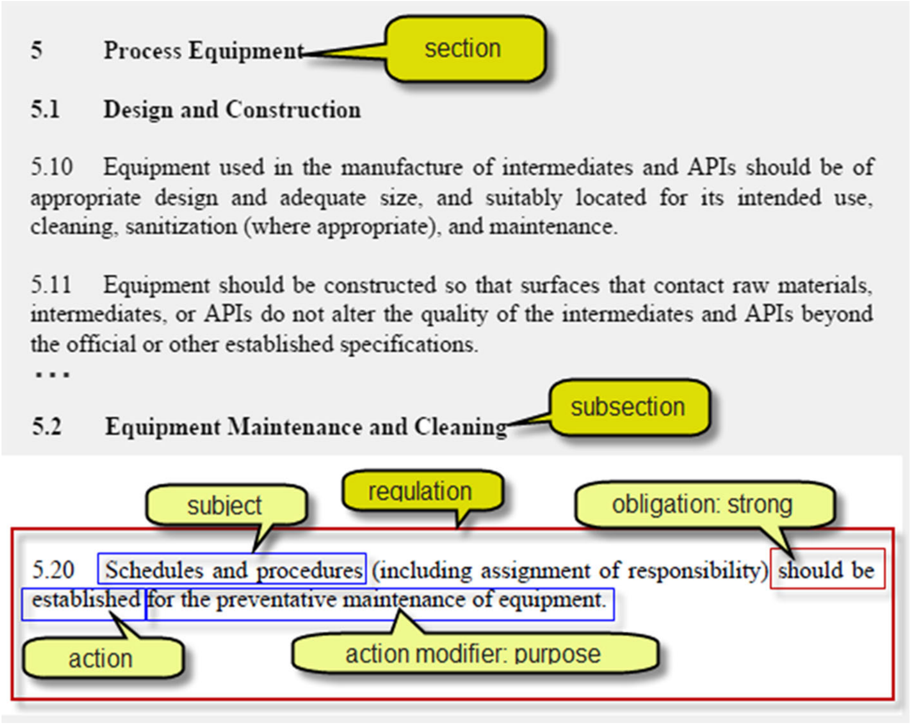
Similarly, the action similarity is computed by comparing the action words associated with a regulatory-statement and a validation-task. After these two similarity scores are computed, the core-entity similarity is determined as the average of the subject-score and the action-score. The topic-similarity is computed by comparing each word in the topic of a regulatory guideline with the subject and the action of a validation-task. Similar to the topic-similarity computation, the auxiliary similarity score is computed by comparing each word in the auxiliary-entities of a regulation-statement with the subjects and the actions of a validation-task.

ALGORITHM 4. Computing the Subject Similarity

```
Input: r is a regulation and t is a validation task.
Output: score is the similarity score
Function: GET-SUBJECT-SCORE(r, t) returns score
        score = 0
        $S_1$ = {$s_1$ | $s_1$ is_a_subject_in stmt}
        $S_2$ = {$s_2$ | $s_2$ is_a_subject_in task}
        for each $s_i \in S_1$
            for each $s_j \in S_2$
                $\delta$ = GET-DIFFERENCE-VALUE($s_i$, $s_j$)
                if ($\delta < \theta$ ) then
                        score' = SIMILARITY-SCORE($s_i$, $s_j$)
                        if (score' > score) then
                                score = score'
                        end if
                end if
            end for
        end for
    return score
```

ALGORITHM 5. Computing Aggregate Similarity Score

```
Input: $S_{topic}$, $S_{core}$ and $S_{aux}$ are topic-score, core-score and aux-score respectively
Output: $S_{agg}$ is the aggregate similarity score of the three scores.
Function: GET-AGGREGATE-SCORE($S_{topic}$, $S_{core}$, $S_{aux}$) returns $S_{agg}$
        $S_{agg}$ = 0
        $S_{tc}$ = MAX ($S_{topic}$, $S_{core}$)
        if ($S_{tc} \geq S_{aux}$) then
                $S_{agg} = S_{tc}$
        else
                $S_{agg} = (S_{tc} + S_{aux})/2$
        end if
    return $S_{agg}$
```



5 Process Equipment

5.1 Design and Construction

5.10 Equipment used in the manufacture of intermediates and APIs should be of appropriate design and adequate size, and suitably located for its intended use, cleaning, sanitization (where appropriate), and maintenance.

5.11 Equipment should be constructed so that surfaces that contact raw materials, intermediates, or APIs do not alter the quality of the intermediates and APIs beyond the official or other established specifications.

...

5.2 Equipment Maintenance and Cleaning

5.20 Schedules and procedures (including assignment of responsibility) should be established for the preventative maintenance of equipment.

Fig. 12 An excerpt from the Eudralex regulation showing regulatory entities

4.3 Aggregation of similarity scores

Once the three similarity scores have been computed, the overall similarity between the regulation and the validation-task is determined by computing the aggregate similarity score from the three similarity scores.

The similarity aggregation algorithm (see Algorithm 5) emphasises the importance of the topic-similarity and the core-similarity, as these similarities are more meaningful as compared to the aux-similarity. The aux-similarity considers every annotated word in the regulatory text, such as the annotations within exceptions, which can be sometimes misleading.

In the aggregation algorithm (see Algorithm 5), the maximum score between topic-score and core-score is chosen as the aggregate score. However, if the aux-score is the highest of all, the highest of the topic-score and the core-score is computed. Then, the average between the highest score and the aux-score is regarded as the aggregate score. The aggregation of the similarity scores has been simplified from its previous implementation [22]; it has shown improved results.

4.4 Statement similarity to regulation similarity computation

The three types of similarity scores computed above are between a regulation-statement and a validation-task, not between a regulation-paragraph and a validation-task. As mentioned earlier, a regulation is composed of one or more statements. The overall similarity computed above is the similarity of a statement with a validation-task in the process-ontology. Now, if a regulation contains more than one statement, it also contains a set of similarity scores; the maximum score in the set, i.e. $\mathrm{SimReg} = \mathrm{MAX}(\mathrm{Sim}_{s1}, \mathrm{Sim}_{s2}, \ldots, \mathrm{Sim}_{Sn})$, is regarded as the similarity score between the regulation and the validation-task.

4.5 Baseline framework versus extended framework

The framework has evolved during its implementation. In this paper, the initial framework is called Baseline Framework (BF) and the evolved framework is called Extended Framework (EF).

The extraction phase of the BF used only two components: ontological concepts and rules, whereas that of EF used two additional components: lexical parser and definition terms. Use of lexical parser helps separate the different chunks of the text in a sentence. These chunks help to identify the entities more accurately. The definition terms have been used to identify the entities more accurately. The mapping phase of the BF used only the core-similarity, whereas the EF used two additional similarities: topic-similarity and aux-similarity. It has been observed that the results of the EF outperformed that of the BF.

5 Results and evaluation

5.1 Experimental setup

In order to test the framework, we have used a case study in the Pharmaceutical industry in the EU, which is one of the most heavily regulated domains. The regulation governing this domain in the EU is the Eudralex[8],[9],[10],[11] regulation. As described earlier, the framework requires two ontologies: one for regulatory domain called SemReg and the other for process domain called OntoReg. The research group of chemical engineers in the University of Oxford that developed OntoReg has been regularly consulted for the requirements and validation of the framework.

In order to explain the results in this paper, a regulation, `Eudralex_5.22` in the SemReg ontology and a validation-task, `FilterCleaningTask` in the OntoReg ontology have been selected.

Among the tools and technologies used for the framework are NLP and Semantic Web technologies. The interactions to the ontologies with JAVA have been carried out with the help of Jena API [38]. Jena has been used with Pellet reasoner to trace the property values and infer new knowledge from the implicit knowledge in the ontologies. General Architecture for the Text Engineering (GATE) has been used for the NLP tasks.

5.2 Extraction

This section presents the results and analysis of the extraction part of the framework. In particular, it analyses how the regulatory entities displayed in Fig. 14 have been extracted from the regulatory guidelines in a PDF file in Fig. 13.

The regulation, `Eudralex_5.22` (see Fig. 13) [39], comprises only one regulation-statement and is preceded by an indicator number, 5.22. Each regulation is associated with some topics, which indicates the context of the regulatory guidelines. The topics, in this regulation, are "*Process Equipment*" and "Equipment Maintenance and Cleaning."

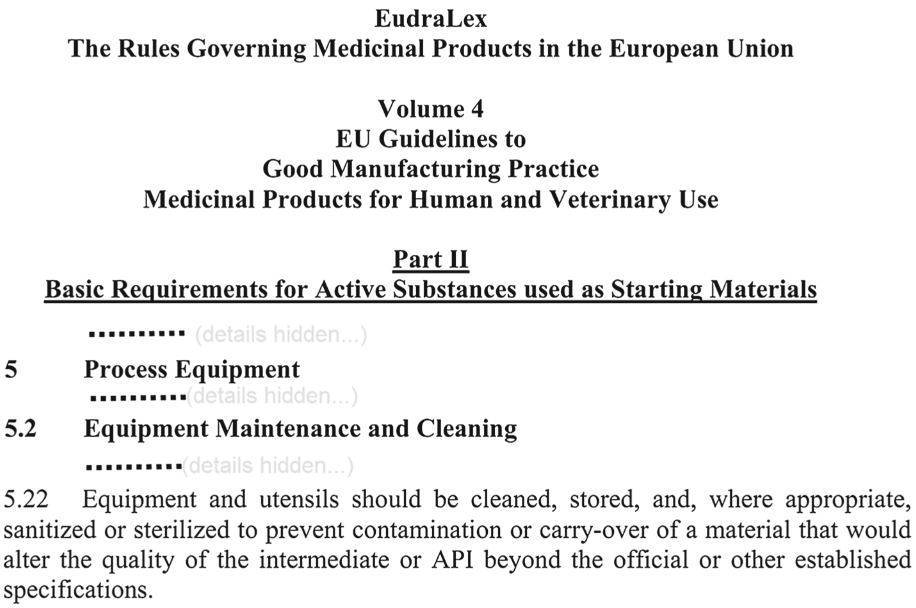
The regulation paragraphs have been annotated using the process described in the framework, and the extracted entities

[8] http://ec.europa.eu/health/files/eudralex/vol-4/pdfs-en/cap5en.pdf.

[9] http://www.europeanlawmonitor.org/what-is-guide-to-key-eu-terms/eu-legislation-what-is-an-eu-directive.html.

[10] http://findlaw.co.uk/law/government/european_law/basics_european_law/500358.html.

[11] http://www.innertemplelibrary.org.uk/news/FAQeu/DifferencesDirectives.htm.



EudraLex
The Rules Governing Medicinal Products in the European Union

Volume 4
EU Guidelines to
Good Manufacturing Practice
Medicinal Products for Human and Veterinary Use

Part II
Basic Requirements for Active Substances used as Starting Materials

.... (details hidden...)

5 Process Equipment

..........(details hidden...)

5.2 Equipment Maintenance and Cleaning

..........(details hidden...)

5.22 Equipment and utensils should be cleaned, stored, and, where appropriate, sanitized or sterilized to prevent contamination or carry-over of a material that would alter the quality of the intermediate or API beyond the official or other established specifications.

Fig. 13 Regulation text in the Eudralex 5.22 regulation

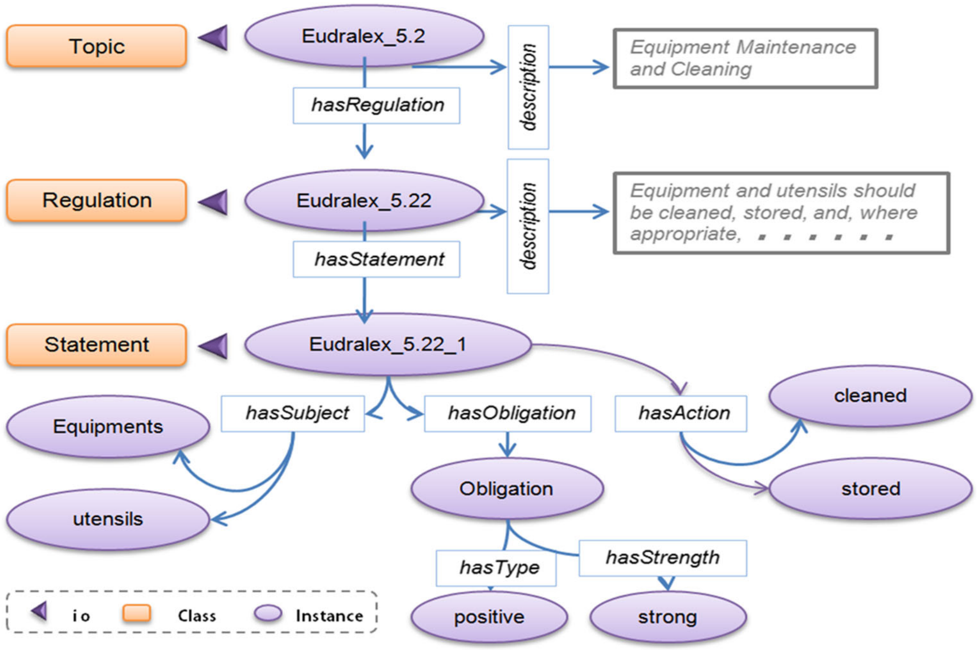


Fig. 14 Eudralex 5.22 regulation represented in the SemReg ontology

have been populated in the SemReg ontology. A graphical representation of the part of the ontology is shown in Fig. 14. In this figure, the classes are Topic, Regulation and Statement, and their individuals are `Eudralex_5.2`, `Eudralex_5.22` and `Eudralex_5.22_1`, respectively. The descriptions of the topic and the regulatory individuals are represented by a data-type property called `description`. A statement is a part of a regulation, which comprises the core- and auxiliary-entities. Among the core-entities, `Equipments and utensils` are presented as the subjects; `cleaned` and `stored` are actions. The subjects and actions relate to the statement via object properties: `hasSubject` and `hasAction`, respectively. The obligation, along with its type and strength, has very little impact in the similarity computation; however, it acts as an indicator phrase in order to identify the subjects and the actions.

Analysis of the results of the baseline and extended frameworks is presented in Table 2. The precisions of the baseline framework and extended framework were determined as 0.89 and 0.96, respectively. The recall of the baseline framework and extended framework was found 0.78 and 0.86, respectively. The f-measures of the baseline and extended

Table 2 Evaluation of the different types of annotations

Evaluation measures	Precision		Recall		F-measure	
Annotation types	BF	EF	BF	EF	BF	EF
Subject	0.89	0.96	0.78	0.86	0.83	0.91
Obligation	1.00	1.00	1.00	1.00	1.00	1.00
Action	0.88	0.96	0.90	0.99	0.89	0.97
Object	1.00	1.00	0.29	0.86	0.44	0.92
Modifier	0.58	0.88	0.27	0.54	0.37	0.67
Condition	0.50	1.00	0.22	0.67	0.31	0.80

framework were computed as 0.83 and 0.91, respectively. This means that the extended framework performed better than the baseline framework did. The comparison between the BF and the EF presented that the current version outperformed the initial version. Although there is no change on the identification of obligations, there is improvement in the identification of other core-entities: subject and action. On the extraction of auxiliary-entities such as object, modifier and condition, it showed better improvement in the extended framework.

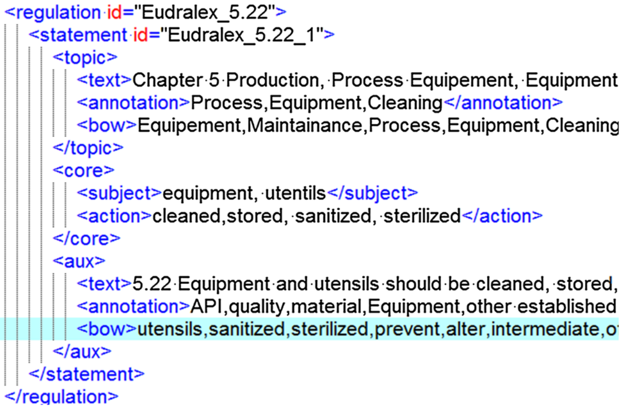
The first three rows in these tables present information about subject, obligation and action, which are described as the core-entities in this framework. The core-entities play a more important role in the regulation process mapping as compared to the auxiliary-entities. The both frameworks have identified all 52 obligations. This is because the framework has created an exhaustive list of obligatory words such as “should be,” “must” and “can be.” Regarding the actions, the extended framework showed a good f-measure, 0.97. Identification of an object, a modifier and a condition did not perform as well as that of the core-entities because the framework focuses on identification of the core-entities. A comprehensive algorithm to identify the auxiliary-entities remains recommended for the future-work of this research.

5.3 Mapping

This section analyses the results of the three types of similarity scores and their aggregation. In particular, it describes a walk thorough example of mapping between the regulatory guideline, “Eudralex_5.22” and an organizational process, “FilterCleaningTask.”

5.3.1 A regulatory guideline in SemReg ontology

In order to compute the three scores, the framework compares three types of entities: (i) topic, (ii) core-entities and (iii) aux-entities. An XML snippet representing these three types of entities, prior to the computation of the aggregate similarity score, is presented in Fig. 15.



```
<regulation id="Eudralex_5.22">
  <statement id="Eudralex_5.22_1">
    <topic>
      <text>Chapter 5 Production, Process Equipement, Equipment
      <annotation>Process,Equipment,Cleaning</annotation>
      <bow>Equipement,Maintainance,Process,Equipment,Cleaning
    </topic>
    <core>
      <subject>equipment, utentils</subject>
      <action>cleaned,stored, sanitized, sterilized</action>
    </core>
    <aux>
      <text>5.22 Equipment and utensils should be cleaned, stored,
      <annotation>API,quality,material,Equipment,other established
      <bow>utensils,sanitized,sterilized,prevent,alter,intermediate,o
    </aux>
  </statement>
</regulation>
```

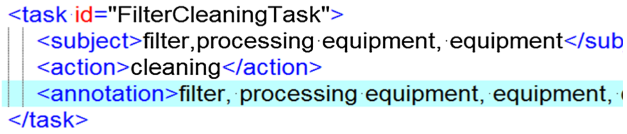
Fig. 15 Three types of entities in Eudralex 5.22 regulation

The text in the topic comprises a combination of higher and lower topics related to the statement. Annotations are the most important entities in the text in terms of their meanings and their relation to the regulation and process. All the words except the stop-words are included in the bag of words (bow). The difference between the annotations and the bow is that the earlier ones are the concepts annotated from the domain ontology and the later ones are all the words remaining after removing the stop-words. The core-entities are collected directly from the subject and action properties of the statement in the SemReg ontology. The auxiliary-entity collection is similar to the topic entity collection, where the annotations and the bag-of-words collection follow the same process. The text in the auxiliary-entity is the text of the statement.

5.3.2 An organizational process in OntoReg ontology

In the process-ontology, OntoReg, a validation-task is associated with a subject via an object-property `hasPatient`, for which we have created an equivalent property called `hasSubject` for clarity. Similarly, an action is indirectly associated with a task, which can be determined by traversing through some object properties and individuals. In the `FilterCleaningTask`, the subject is `Filter101`, which is an individual of a class `Filter`. The class `Filter` is subsumed by the classes `ProcessingEquipment` and `Equipment`. The action for the `FilterCleaningTask` is defined implicitly. Having traversed through the property `isReponsibilityOf` and `performs`, it was inferred that `CleaningIndividual` is an individual of a class `Cleaning`. The class `Cleaning` is subsumed by its superclass `Action`.

In the mapping process, the regulatory entities such as topic, core-entities and auxiliary-entities are compared with the process entities such as subject, action and annotations. Figure 16 depicts the collection of subjects, actions



```
<task id="FilterCleaningTask">
    <subject>filter,processing equipment, equipment</sub
    <action>cleaning</action>
    <annotation>filter, processing equipment, equipment,
</task>
```

Fig. 16 Subject, action and annotations in Filter Cleaning Task

Table 3 Similarity scores between regulatory and process-subjects

Regulatory subject	Process-subject	Similarity score
Equipment	Filter	0.42
Equipment	Processing equipment	0.54
Equipment	Equipment	1.00
Utensils	Filter	0.32
Utensils	Processing equipment	0.27
Utensils	Equipment	0.48
	Highest similarity score	1.00

Table 4 Similarity scores between regulatory and process actions

Regulatory action	Process action	Similarity score
Cleaned	Cleaning	1.00
Stored	Cleaning	0.00
Sanitized	Cleaning	0.00
Sterilized	Cleaning	0.84
	Highest similarity score	1.00

and annotations of `FilterCleaningTask` just before the similarity score computation. The subjects are identified by the names and labels of the subject individual, classes and super-classes. Similarly, the action is determined by the names and labels of the action individual, their classes and super-classes. The annotation is the combination of these two types of entities.

5.3.3 Three scores computation

The comparison of the regulatory entities (topic, core and auxiliary) and the process entities (subject and action) produces three types of scores, namely topic-score, core-score and aux-score.

For the core-score computation, the `subject` and `action` in the regulation-statement `Eudralex_5.22_1` were compared with the `subject` and `action` of the validation-task `FilterCleaningTask`, respectively. In particular, the terms in regulatory subject "*equipment and utensils*" were compared with the terms in the process-subject "*filter, processing equipment, equipment*." This comparison produced a set of similarity between these two subjects. After the two separate comparisons, it produced two sets of scores: subject-score set (see Table 3) and action-score set (see Table 4).

In the `subject-score` set {0.42, 0.54, 1.00, 0.32, 0.27, 0.48}, the highest score is determined as 1.00. Therefore, 1.00 was set as the similarity score between the sets of subjects in the regulation-statement, `Eudralex_5.22_1` and the process, `FilterCleaningTask`. Similarly, in the `action-score` set {1.00, 0.00, 0.00, 0.84, 1.00} the highest score was found as 1.00. Therefore, the similarity score between the sets of actions in the regulation-statement, `Eudralex_5.22_1` and the process, `FilterCleaningTask` was set as 1.00. Then, the average score between the `subject-score` and `action-score`, 1.00 was determined as the `core-score`.

In the `topic-score` computation, the terms, "*Equipment, Maintenance, Process, Equipment, Cleaning*," in the `bow of topic` in the regulation-statement, `Eudralex_5.22_1`, were compared with the terms, "*filter, processing equipment, equipment, cleaning*" in the `annotation` of `FilterCleaningTask` (see Table 5). The highest similarity score between the term "*Equipment*" in regulation and the terms "*filter, processing equipment, equipment, cleaning*" in the process was found as 1.00. Similarly, the highest similarity scores of "*Maintenance,*" "*Process,*" "*Equipment*" and "*Cleaning*" with respect to their comparison with the terms in process annotations were found as 0.73, 0.56, 1.00 and 1.00, respectively. Then, the average of these scores, 0.86, was determined as the `topic-score` between the regulation-statement, `Eudralex_5.22_1` and the process, `FilterCleaningTask`.

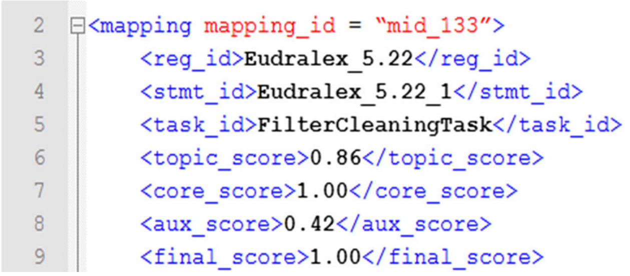
The computations of `aux-score` is similar to that of `topic-score`. In the `aux-score` computation, the terms, "*utensils, sanitized, sterilized, prevent, alter, intermediate, official, API, quality, material, equipment…*," in the bow of aux in the regulation-statement, `Eudralex_5.22_1`, were compared with the terms, "*filter, processing equipment, equipment, cleaning*," in the `annotation` of `FilterCleaningTask`. We also carried out the highest similarity score computation and the average of the highest similarity score computation. Then, the `aux-score` between the regulation-statement, `Eudralex_5.22_1`, and the process, `FilterCleaningTask`, was computed as 0.42. A part of an XML file representing the three scores computed between the regulation `Eudralex_5.22` and the process `FilterCleaningTask` is provided in Fig. 17.

5.3.4 Aggregating the similarity scores

Having computed the three types of similarity scores between the regulation and validation-task, the next step was to compute the aggregate similarity between the pairs. In the earlier section, the `topic-score`, `core-score` and

Table 5 Similarity scores between a regulatory topic and a process

Regulatory topic	Process annotation	Similarity score
Equipment	Filter	0.42
Equipment	Processing equipment	0.54
Equipment	Equipment	1.00
Equipment	Cleaning	0.06
	Highest similarity score	1.00
Maintenance	Filter	0.00
Maintenance	Processing equipment	0.12
Maintenance	Equipment	0.00
Maintenance	Cleaning	0.73
	Highest similarity score	0.73
Process	Filter	0.08
Process	Processing equipment	0.56
Process	Equipment	0.12
Process	Cleaning	0.40
	Highest similarity score	0.56
Equipment	Filter	0.42
Equipment	Processing equipment	0.54
Equipment	Equipment	1.00
Equipment	Cleaning	0.06
	Highest similarity score	1.00
Cleaning	Filter	0.00
Cleaning	Processing equipment	0.00
Cleaning	Equipment	0.00
Cleaning	Cleaning	1.00
	Highest similarity score	1.00
Average of the highest similarity scores		0.86



```
<mapping mapping_id = "mid_133">
    <reg_id>Eudralex_5.22</reg_id>
    <stmt_id>Eudralex_5.22_1</stmt_id>
    <task_id>FilterCleaningTask</task_id>
    <topic_score>0.86</topic_score>
    <core_score>1.00</core_score>
    <aux_score>0.42</aux_score>
    <final_score>1.00</final_score>
```

Fig. 17 Three types of similarity scores between Eudralex_5.22 and FilterCleaningTask

aux-score were computed as 0.86, 1.00 and 0.42, respectively. In the aggregation algorithm, the maximum score between topic-score and core-score was computed as:

$$S_{\text{tc}} = \text{MAX}\ (S_{\text{topic}}, S_{\text{core}}) = \text{MAX}\ (0.86, 1.00) = 1.00$$

where S_{tc} is the maximum score between topic-score, S_{topic} and core-score, S_{core}. In this case, the S_{tc} is greater than the aux-score, S_{aux}. Hence, the final similarity score between the regulation-statement, Eudraxlex_5.22_1, and the validation-task, FilterCleaningTask, was determined as 1.00, which was represented as the final-score. Then, an XML file, containing all the three scores and the aggregate score between regulation-statements and processes, was generated. A part of the XML file is shown in Fig. 17.

5.3.5 *Evaluation of the mapping result*

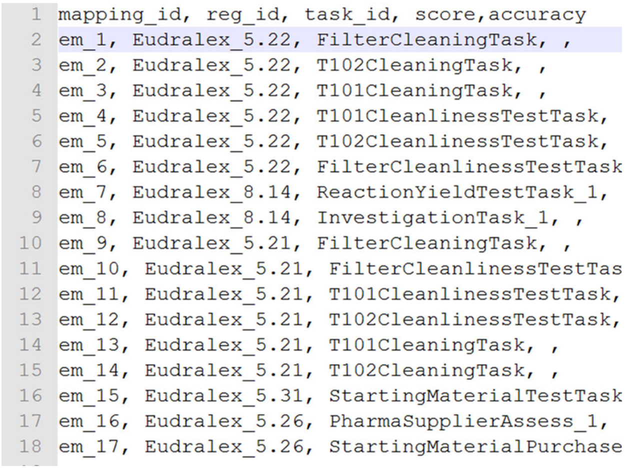
The OntoReg ontology contains a set of mapping between Eudralex regulations and validation-tasks. In particular, each validation-task is associated with one or more regulations, and each regulation is related to one or more validation-tasks, called existing mapping. The existing mappings were created by the experts manually. A subset of existing mapping collected from the OntoReg is depicted in Fig. 18, where line number 2 indicates that there is a mapping between the regulation Eudralex_5.22 and the validation-task FilterCleaningTask. The list in Fig. 18 was created by using the values of the object-property isRegulationOf of individuals under the concept Regulation.

The mappings between a regulation and a validation-task generated by the RegCMantic framework is referred to as computed mapping. A subset of computed mappings is shown in Fig. 19. The line number 8 indicates that there is a mapping between the regulation, Eudralex_5.22 and the validation-task, FilterCleaningTask.

As stated above, a regulation comprises one or more regulation-statements; the final-score computed above is the similarity score between a statement and a validation-task. Therefore, the similarity score computation created a set of final similarity scores between the regulation and the validation-task; the highest score was regarded as the similarity score between the regulation and the validation-task.

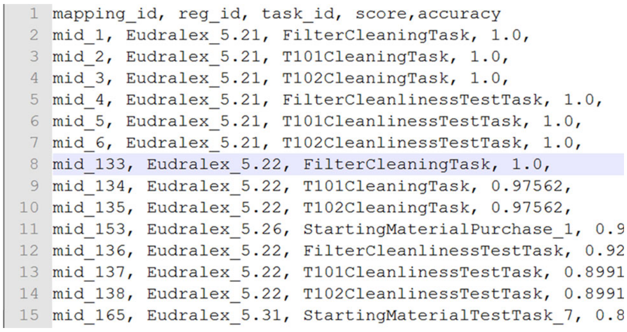
In order to evaluate the result of the algorithm, the set of manual mappings was considered as the standard mappings, which were compared with the set of computed mappings; the comparison generated three types of mappings: the correct mappings, incorrect mappings and missing mappings. These three types of mapping are used to compute the standard evaluation techniques called precision, recall and f-measure. Precision, recall and f-measure are popular in Information Retrieval (IR) and have been borrowed in several other domains, as well. Since the authors have not come across the frameworks that map regulatory guidelines with organizational processes, the evaluation of the framework was carried out by observing the precision, recall and f-measure only.

The selection of the mappings also needs to define the minimum threshold, τ. The value of τ was set as 0.85; only the mappings with the score 0.85 or above were selected as



```
mapping_id, reg_id, task_id, score,accuracy
em_1, Eudralex_5.22, FilterCleaningTask, ,
em_2, Eudralex_5.22, T102CleaningTask, ,
em_3, Eudralex_5.22, T101CleaningTask, ,
em_4, Eudralex_5.22, T101CleanlinessTestTask,
em_5, Eudralex_5.22, T102CleanlinessTestTask,
em_6, Eudralex_5.22, FilterCleanlinessTestTask
em_7, Eudralex_8.14, ReactionYieldTestTask_1,
em_8, Eudralex_8.14, InvestigationTask_1, ,
em_9, Eudralex_5.21, FilterCleaningTask, ,
em_10, Eudralex_5.21, FilterCleanlinessTestTas
em_11, Eudralex_5.21, T101CleanlinessTestTask,
em_12, Eudralex_5.21, T102CleanlinessTestTask,
em_13, Eudralex_5.21, T101CleaningTask, ,
em_14, Eudralex_5.21, T102CleaningTask, ,
em_15, Eudralex_5.31, StartingMaterialTestTask
em_16, Eudralex_5.26, PharmaSupplierAssess_1,
em_17, Eudralex_5.26, StartingMaterialPurchase
```

Fig. 18 An excerpt of the existing mappings between regulations and validation-tasks



```
mapping_id, reg_id, task_id, score,accuracy
mid_1, Eudralex_5.21, FilterCleaningTask, 1.0,
mid_2, Eudralex_5.21, T101CleaningTask, 1.0,
mid_3, Eudralex_5.21, T102CleaningTask, 1.0,
mid_4, Eudralex_5.21, FilterCleanlinessTestTask, 1.0,
mid_5, Eudralex_5.21, T101CleanlinessTestTask, 1.0,
mid_6, Eudralex_5.21, T102CleanlinessTestTask, 1.0,
mid_133, Eudralex_5.22, FilterCleaningTask, 1.0,
mid_134, Eudralex_5.22, T101CleaningTask, 0.97562,
mid_135, Eudralex_5.22, T102CleaningTask, 0.97562,
mid_153, Eudralex_5.26, StartingMaterialPurchase_1, 0.9
mid_136, Eudralex_5.22, FilterCleanlinessTestTask, 0.92
mid_137, Eudralex_5.22, T101CleanlinessTestTask, 0.8991
mid_138, Eudralex_5.22, T102CleanlinessTestTask, 0.8991
mid_165, Eudralex_5.31, StartingMaterialTestTask_7, 0.8
```

Fig. 19 An excerpt of computed mapping between regulations and validation-tasks

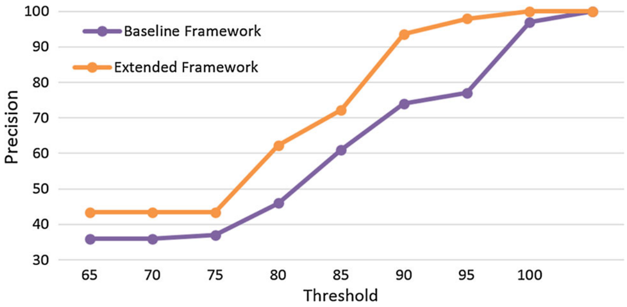


Fig. 20 Precisions of the mappings in different thresholds

the accepted mappings; and the rest of the mappings were discarded. Figures 20, 21 and 22 show the precision, recall and f-measure of the mapping results, respectively. The value of τ was set as 0.85 because it was found the optimum threshold after repeated observation, which can be seen in Fig. 22.

The base line framework refers to the similarity score computed by using only the core scores, and extended framework refers to the score generated by using the topic, core and auxiliary scores.

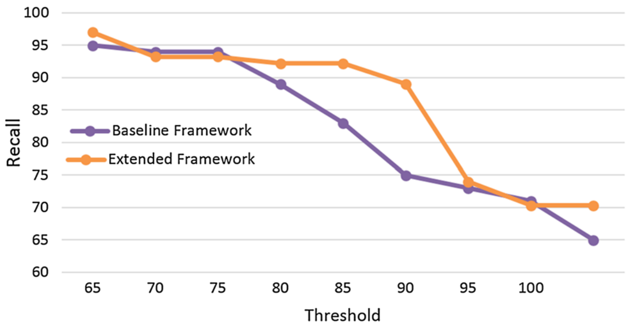


Fig. 21 Recalls of the mappings in different thresholds

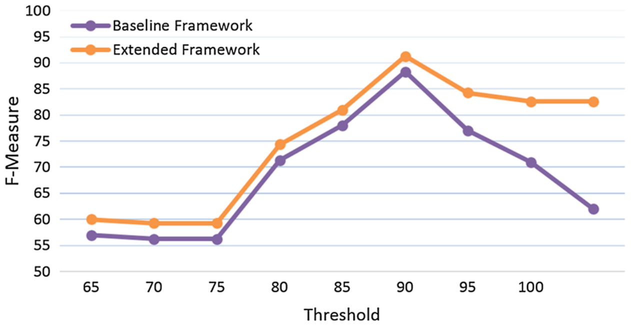


Fig. 22 F-measure of the mappings in different thresholds

6 Related work

A system similar to RegCMantic has not been found; however, there are systems that work with automatic extraction of regulatory entities and others that map regulations with organizational processes. These approaches are described in the following sections.

6.1 Related extraction approaches

Kiyavitskaya et al. propose in [40] a system that extracts rights and obligation by the extension of the *Cerno* framework. This research aims to identify the requirements by detecting the presence of normative phrases as it is done in the RegCMantic framework. However, in contrast to the application of shallow parser in the *Cerno* framework, a deep parser is used in the RegCMantic framework because they are more useful in the more grammatically correct text such as regulation. Furthermore, the *Cerno* framework is applicable to more structured text such as legalese and needs engineers to annotate the regulatory text. In contrast, the extraction part of the RegCMantic framework can be applied to the text with no explicitly defined document-structure and the annotation process is automatic. The exception extraction by Gao et al. [41] and the regulation-entities extraction in Mu et al. [42] are also related to the RegCMantic framework. However, the former is only confined to the extraction of exception with limited indicator terms. The latter is more related to the RegCMantic approach, since it extracts a variety of regulation-entities such as subject, subject-modifier, object, object-modifier, action, location, time, manner and constraints. Furthermore, it also uses a deep parser and a list of terms. However, it has not been mentioned how to deal with the text with implicit document-structure. Moreover, the terms are defined by the experts manually, which, in contrast, is extracted automatically in the RegCMantic framework.

6.2 Related mapping approaches

This section reviews the existing work related to the RegCMantic mapping approach. Examples of the related work include the similarity techniques in Business Process Modelling (BPM), sentence similarity, word similarity, ontology mappings and conceptual distance.

6.2.1 BPM similarities

BPM represents the processes of an enterprise so that they can be easily analysed and improved. There are similarity approaches that relate a process to another process [19,21,24] or a controlled objective [8,14,43]. The controlled objectives are the objectives created by considering the standards and the regulatory guidelines related to the business processes. The similarity techniques used to relate these components could be considered as related to this work.

The similarity in the elements of two processes was determined in [21] with two kinds of matching: graph matching and pure lexical matching. The redundant or duplicate elements in processes were identified in [19] by using ontology matching technology. The similarity between two processes were identified in [24] by extracting annotations from the data schema and templates associated with the processes. However, these approaches do not relate regulatory guidelines with organizational processes.

Creating controlled objectives from the regulations and the processes, and relating the objectives were explored in [8,43]. Similarly, the regulations were represented in a rule-based logic, FCL, and the processes were represented in BPMN and annotated to align the processes with the regulations [14]. However, it has not been explained how they were related, since their focus was to determine the non-compliance in the processes.

6.2.2 Sentence similarity

In [44], sentence similarity is computed using align-heuristics where noun, verb, adjective, adverb and numbers are aligned; the approach was inspired by the popular sentence alignment algorithm in [45]. The decomposition of sentences into different entities for the similarity measure is similar to the RegCMantic framework; however, this can be only applied to compute the sentence similarity. The sentence matching based on the Bag of Words (BoW) algorithm was applied in [46] in order to determine the answer similarity. A BoW is an unordered collection of words, which does not consider the grammar and the order of the words. It has been predominantly used in Information Retrieval (IR) in order to classify the pages. In the similarity computation, each word in a BoW is compared with the words in the other BoW. The computation of similarity of words in two sentences is related to this work. However, it is only applicable to compare sentences. Similarly, a pilot for similarity in SemEval competition has described the similar algorithms for the sentence similarity which also requires training and testing sentences [47].

6.2.3 Ontological concept and relation similarity

Conceptual distance and similarity computation in ontologies are also related to this work. The use of weight allocation and node routing table in order to compute semantic distance between two concepts in an ontology [35] is related to the RegCMantic framework. In [48], a graph-based similarity is computed considering various types of ontological properties and the depth of the concepts. In [49], two ontologies have been defined in order to determine similarity of a new event with an existing event. The similarity computed using WordNet similarity is related to this work; however, it requires that both ontological concepts and individuals designed and populated by the domain expert manually. In this framework, regulatory ontology is populated automatically from the text in the regulatory guidelines.

6.2.4 Combined similarities

The work presented in [50] applies a combination of similarity approaches in order to determine similarity between contents of two television programmes. The most related part in this framework is the computation of the similarity of topics and the text in the television programme synopsis. However, it is only applicable if both compared entities contain hierarchy and text description in the sentences. The RegCMantic framework can be applied to determine the similarity where the processes are represented in ontological concepts, and the regulatory guidelines are represented in an unstructured text format, and the regulatory entities are populated in a regulatory ontology automatically [26].

7 Conclusion

Mapping regulatory guidelines with organizational processes becomes crucial when there are changes in the guidelines, or the organizational processes need to follow the guidelines from different policy makers. Various extraction and similarity algorithms are closely related to the RegCMantic framework. However, they are not directly related to the mapping between the guidelines and processes. Therefore, there is a greater need for efficient algorithms that can map regulations with processes. This paper has presented RegCMantic framework, which identifies the regulatory entities automatically in order to map the regulatory guidelines with organizational processes. It has computed three types of similarity scores: (1) topic-similarity, (2) core-entity similarity and (3) auxiliary-entity similarity. The framework considers the ontological structures in order to compute the similarity scores. The case study carried out in the Pharmaceutical industry has demonstrated some promising results.

References

1. Zhang IX (2007) Economic consequences of the Sarbanes-Oxley Act of 2002. J Account Econ 44:74–115
2. Pham TA, Le Thanh N (2016) An ontology-based approach for business process compliance checking. In: Proceedings of the 10th international conference on ubiquitous information management and communication. ACM, New York, NY, USA, pp 56:1–56:6
3. Ternai K (2015) Semi-automatic methodology for compliance checking on business processes. In: Ko A, Francesconi E (eds) Electronic Government and the Information Systems Perspective. Springer, Berlin, pp 243–256
4. Beach TH, Rezgui YR, Li H, Kasim T (2015) A rule-based semantic approach for automated regulatory compliance in the construction sector. Expert Syst Appl 42:5219–5231
5. Zeni N, Kiyavitskaya N, Mich L, Cordy JR, Mylopoulos J (2015) GaiusT: supporting the extraction of rights and obligations for regulatory compliance. Requir Eng 20:1–22
6. Goedertier S, Vanthienen J (2006) Designing compliant business processes from obligations and permissions. In: Proceedings of 2nd workshop on business processes design (BPD'06). Springer, Vienna, pp 5–14
7. Breaux TD, Vail MW, Antón AI (2006) Towards regulatory compliance: extracting rights and obligations to align requirements with regulations. In: Proceedings of 14th IEEE international requirements engineering conference (RE'06). IEEE Computer Society, Minneapolis, pp 49–58
8. Sadiq S, Governatori G (2010) A methodological framework for aligning business processes and regulatory compliance. In: vom Brocke J, Rosemann M (eds) Handbook of business process management: 2. Strategic alignment, governance, people and culture. Springer, Berlin, pp 159–176
9. Logrippo L (2008) Requirements and compliance in legal systems: a logic approach. In: Requirements engineering and law, 2008. RELAW'08. IEEE Computer Society Press, Barcelona, Spain, pp 40–44
10. Ghanavati S, Amyot D, Peyton L (2007) Towards a framework for tracking legal compliance in healthcare. In: Proceedings of the 19th international conference on advanced information systems engineering (CAiSE'07). Springer, Berlin, pp 218–232
11. Haider SI (2006) Validation standard operating procedures. Informa Healthcare, New York
12. Liu Y, Muller S, Xu K (2007) A static compliance-checking framework for business process models. IBM Syst J 46:335–361
13. Elgammal A, Turetken O, Van Den Heuvel W-J, Papazoglou MP (2012) Using patterns for the analysis and resolution of compliance violations. Int J Coop Inf Syst 21:31–54
14. Governatori G, Shek S (2012) Rule based business process compliance. CEUR 874:1–8
15. Sadiq S, Governatori G (2010) Managing regulatory compliance in business processes. In: Brocke J, Rosemann M (eds) Handbook on business process management 2. Springer, Berlin, pp 159–175
16. Sapkota K, Aldea A, Younas M, Duce DA, Banares-Alcantara R (2012) Extracting meaningful entities from regulatory text. In: Proceedings of the fifth international workshop on requirements engineering and law (RELAW'12). IEEE Computer Society Press, Chicago, pp 29–32
17. Ceci M, Gangemi A (2016) An OWL ontology library representing judicial interpretations. Semant Web 7(3):229–253
18. Ghanavati S, Amyot D, Rifaut A (2014) Legal goal-oriented requirement language (legal GRL) for modeling regulations. In: Proceedings of the 6th international workshop on modeling in software engineering, pp 1–6
19. Castellanos C, Correal D, Murcia F (2011) An ontology-matching based proposal to detect potential redundancies on enterprise architectures. In: 30th international conference of the Chilean Computer Science Society (SCCC). IEEE Computer Society, pp 118–126
20. Lu R, Sadiq S, Governatori G (2008) Compliance aware business process design. In: Proceedings of the 2007 international conference on business process management. Springer, Berlin, pp 120–131
21. Dijkman R, Dumas M, Garcia-Banuelos L, Kaarik R (2009) Aligning business process models. In: 2009 IEEE international enterprise distributed object computing conference. IEEE, pp 45–53
22. Sapkota K, Aldea A, Younas M, Duce DA, Banares-Alcantara R (2013) RP-Match: a framework for automatic mapping of regulations with organizational processes. In: The 10th IEEE international conference on e-business engineering (ICEBE 2013). IEEE Computer Society Press, Coventry, UK, pp 257–264
23. Sapkota K, Aldea A, Younas M, Duce DA, Banares-Alcantara R (2012) Semantic knowledge mapping: an extension of compendium with semantic knowledge representation. Int J Artif Intell Appl 3:1–12
24. Hashmi M, Governatori G, Wynn MT (2012) Business process data compliance. In: Bikakis A, Giurca A (eds) Rules on the web: research and applications SE-4. Springer, Berlin, pp 32–46
25. Sapkota K, Aldea A, Younas M, Duce DA, Banares-Alcantara R (2011) Towards semantic methodologies for automatic regulatory compliance support. In: Proceedings of the 4th workshop on workshop for Ph.D. students in information and knowledge management (PIKM'11). ACM Press, Glasgow, pp 83–86
26. Sapkota K, Aldea A, Younas M, Duce DA, Banares-Alcantara R (2011) Semantic-ART: a framework for semantic annotation of regulatory text. In: Proceedings of the fourth workshop on exploiting semantic annotations in information retrieval (ESAIR'11). ACM Press, Glasgow, pp 23–24
27. Sarawagi S (2007) Information extraction. Commun ACM 1:261–377
28. Appelt DE, Onyshkevych B (1998) The common pattern specification language. In: TIPSTER workshop (TIPSTER'98). Association for Computational Linguistics, Baltimore, Maryland, pp 23–30
29. Thakker D, Osman T, Lakin P (2009) GATE JAPE grammar tutorial. http://gate.ac.uk/sale/thakker-jape-tutorial/GATEJAPEmanual.pdf
30. Gómez-Pérez A, Fernández-López M, Corcho O (2007) Ontological engineering: with examples from the areas of knowledge management, e-commerce and the semantic web. Pringer, Secaucus
31. Hoekstra R, Breuker J, Di Bello M, Boer A (2007) The LKIF core ontology of basic legal concepts. In: Casanovas P, Biasiotti MA, Francesconi E, Sagri MT (eds) Proceedings of the 2nd workshop on legal ontologies and artificial intelligence techniques (LOAIT'07). CEUR-WS.org, Stanford, California, USA, pp 43–63

32. Wyner A, Hoekstra R (2012) A legal case OWL ontology with an instantiation of Popov v. Hayashi. Artif Intell Law 20(1):83–107
33. Sapkota K (2013) Semantic frameworks for regulatory compliance support. PhD thesis
34. Sesen MB, Suresh P, Banares-Alcantara R, Venkatasubramanian V (2010) An ontological framework for automated regulatory compliance in pharmaceutical manufacturing. Comput Chem Eng 34:1155–1169
35. Ge J, Qiu Y (2008) Concept similarity matching based on semantic distance. In: Proceedings of the fourth international conference on semantics, knowledge and grid (SKG'08). IEEE Computer Society, Beijing, China, pp 380–383
36. Lin D (1998) An information-theoretic definition of similarity. In: Shavlik JW (ed) Proceedings of the fifteenth international conference on machine learning (ICML'98). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 296–304
37. Pedersen T, Patwardhan S, Michelizzi J (2004) WordNet:: similarity: measuring the relatedness of concepts. In: Proceeding of the demonstration papers at HLT-NAACL 2004 (HLT-NAACL–Demonstrations'04). ACL Press, Stroudsburg, PA, USA, pp 38–41
38. Grobe M (2009) RDF, Jena, SparQL and the "Semantic Web". In: Proceedings of the ACM SIGUCCS fall conference on user services conference (SIGUCCS'09). ACM Press, St. Louis, Missouri, USA, p 131
39. Eudralex: the rules governing medicinal products in the European Union. http://ec.europa.eu/health/documents/eudralex/cd/index_en.htm
40. Kiyavitskaya N, Zeni N, Breaux TD (2008) Automating the extraction of rights and obligations for regulatory compliance. In: Li Q, Spaccapietra S, Yu E, Olivé A (eds) Lecture notes in computer science. Springer, Berlin, pp 154–168
41. Gao X, Singh MP, Mehra P (2011) Mining business contract for service exceptions. IEEE Trans Serv Comput 5:333–344
42. Mu Y, Wang Y, Guo J (2009) Extracting software functional requirements from free text documents. In: Proceedings of international conference on information and multimedia technology, 2009. ICIMT'09. IEEE Computer Society Press, Jeju Island, Republic of Korea, pp 194–198
43. Sadiq S, Governatori G, Namiri K (2007) Modeling control objectives for business process compliance. In: Alonso G, Dadam P, Rosemann M (eds) Proceedings of 5th international conference, BPM 2007. Springer, Berlin, pp 149–164
44. McCarthy D, Gella S, Reddy S (2012) DSS: text similarity using lexical alignments of form, distributional semantics and grammatical relations. In: Proceedings of the first joint conference on lexical and computational semantics (SEM'12). ACL Press, Montreal, Canada, pp 557–564
45. Barzilay R, Elhadad N (2003) Sentence alignment for monolingual comparable corpora. In: Proceedings of the 2003 conference on empirical methods in natural lnguage processing (EMNLP'03). ACL Press, Stroudsburg, PA, USA, pp 25–32
46. Mohler MAG, Bunescu R, Mihalcea R (2011) Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies (HLT'11). ACM Press, Stroudsburg, PA, USA, pp 752–762
47. Agirre E, Cer D, Diab M, Gonzalez-Agirre A (2012) SemEval-2012 task 6: a pilot on semantic textual similarity. In: Proceedings of the 6th international workshop on semantic evaluation (SemEval 2012), in conjunction with the first joint conference on lexical and computational semantics (SEM 2012). ACL Press, Montreal, Canada, pp 385–393
48. Hawalah A, Fasli M (2011) A graph-based approach to measuring semantic relatedness in ontologies. In: Proceedings of the international conference on web intelligence, mining and semantics (WIMS'11). ACM Press, New York, pp 1–12
49. Chen Y, Wang J, Cheng Z, Jing L, Zhou Y (2010) An algorithm to compute similarity between danger objects based on ontology for danger-aware systems. In: Proceedings of the 2nd international symposium on aware computing (ISAC'10). IEEE Computer Society, Tainan, Taiwan, pp 128–135
50. Yu Z, Zhou X (2009) Combining vector space model and category hierarchy model for TV content similarity measure. In: Proceedings of the third international conference on multimedia and ubiquitous engineering (MUE'09). IEEE Computer Society, Qingdao, China, pp 130–136