Open Research Online

The Open University's repository of research publications
and other research outputs

Exploiting conceptual spaces for ontology integration

Conference or Workshop Item

oro.open.ac.uk

Exploiting Conceptual Spaces
for Ontology Integration

Stefan Dietze, John Domingue
Knowledge Media Institute,
The Open University,
MK7 6AA, Milton Keynes, UK
{s.dietze, j.b.domingue}@open.ac.uk

Abstract. The widespread use of ontologies raises the need to integrate distinct conceptualisations. Whereas the symbolic approach of established representation standards – based on first-order logic (FOL) and syllogistic reasoning – does not implicitly represent *semantic similarities*, *ontology mapping* addresses this problem by aiming at establishing formal relations between a set of knowledge entities which represent the same or a similar meaning in distinct ontologies. However, manually or semi-automatically identifying similarity relationships is costly. Hence, we argue, that representational facilities are required which enable to implicitly represent similarities. Whereas *Conceptual Spaces (CS)* address similarity computation through the representation of concepts as vector spaces, CS provide neither an implicit representational mechanism nor a means to represent arbitrary relations between concepts or instances. In order to overcome these issues, we propose a hybrid knowledge representation approach which extends FOL-based ontologies with a conceptual grounding through a set of CS-based representations. Consequently, semantic similarity between instances – represented as members in CS – is indicated by means of distance metrics. Hence, automatic similarity detection across distinct ontologies is supported in order to facilitate ontology integration.

1 Introduction

The widespread use of ontologies [17] together with the increasing availability of representations of overlapping domains of interest, raises the need to integrate distinct ontologies. This becomes particularly apparent when considering the exploitation of formally specified knowledge on the *Semantic Web (SW)* which by its distributed nature consists of heterogeneous representations. Following the symbolic representational approach based on first-order logic (FOL) and syllogistic reasoning [15] – as applied by established representation standards such as OWL[1] or RDF-S[2] – requires that heterogeneities across distinct formalisations are resolved through mappings [20][25]. With respect to [2][31], *ontology mapping* is defined as the process of establishing formal relations between knowledge entities which represent the same or a similar semantic meaning in distinct ontologies [8][9][35]. In that, the ontology mapping task strongly relies on identifying *similarities* [1] between entities across different ontologies, what appears to be a necessary requirement to support

---

[1] http://www.w3.org/OWL/

[2] http://www.w3.org/RDFS/

interoperability between multiple heterogeneous ontologies. However, with respect to this goal, several issues have to be taken into account. The symbolic approach leads to ambiguity issues and does not entail meaningfulness, since meaning requires both the definition of a terminology in terms of a logical structure (using symbols) and grounding of symbols to a conceptual level [3][27]. Therefore, concept representations across distinct ontologies – even those representing the same real-world entities – hardly equal another, and hence, similarity is not an implicit notion carried within ontological representations. But manual or semi-automatic identification of similarity relationships – based on linguistic or structural similarities across ontologies [7][24][10][16] – is costly. Consequently, representational facilities which enable to implicitly describe similarities across ontologies are required to fully facilitate ontology interoperability.

Conceptual Spaces (CS) [14][15] follow a theory of describing entities at the conceptual level in terms of their natural characteristics to avoid the symbol grounding issue [3][27]. In that, CS consider the representation of concepts as vector spaces which are defined through a set of quality dimensions. Describing instances as vectors enables the automatic calculation of their semantic similarity by means of spatial distance metrics. However, several issues still have to be considered when applying CS. For instance, CS provide no means to represent arbitrary relations between concepts or instances, such as part-of relations. The fact that the particular scope of each dimension can not be restricted poses further issues when attempting to base entire knowledge models on CS.

In order to overcome the issues introduced above, we propose a two-fold knowledge representation approach which extends FOL-based ontologies with a conceptual grounding by refining individual symbolic concept representations as particular CS. The resulting set of CS is formally represented as part of the ontology itself. Consequently, semantic similarity between instances, represented as CS members, i.e. vectors, is indicated by means of distance metrics such as the Euclidean distance whereas additional knowledge represented within the ontology, e.g. through relations and axioms, is still maintained. In that, similarity becomes an implicit notion of the representation itself by overcoming the need for ontology mapping.

The remaining paper is organised as follows. The following Section 2 introduces current approaches to ontology integration and discusses CS as a possible solution. We propose our hybrid representation approach in Section 3 and discuss its application to ontology integration in Section 4. A prototypical implementation and evaluation of our approach is introduced in Section 5. Section 6 concludes the paper.

2 Ontology Interoperability

In order to illustrate our motivation, here, we define key terminology and introduce the ontology interoperability problem.

2.1. A formal approach to Ontologies

An ontology is described as the explicit, formal specification of a shared conceptualisation [17]. Such formal conceptualisations aim at representing a certain domain of interest by defining semantics through stating necessary and sufficient conditions for something to be an instance of a class. Individuals (instances) and

classes (concepts) of formal ontologies are processed by syllogistic reasoning [15]. Following [8][9], we define a populated *ontology* as a tuple:

$$O = \{(C, I, P, R, A)\}$$

With C being a set of *n concepts w*here each concept C_i is described through *l(i) concept properties pc*, i.e.:

$$PC_i = \{(pc_{i1}, pc_{i2}, ..., pc_{l(i)}) | pc_{ix} \in C_i\}.$$

I represents all *m instances* where each instance I_{ij} represents a particular instance of a concept C_j and consists of *l(i) instantiated properties pi* instantiating the concept properties of C_j:

$$PI_{ij} = \{(pi_{ij1}, pi_{ij2}, ..., pi_{l(i)}) | pi_{ijx} \in I_{ij}\}.$$

Hence, the properties P of an ontology O represent the union of all concept properties PC and instantiated properties PI of O:

$$P = \{(PC_1, PC_2, ..., PC_n) \cup (PI_1, PI_2, ..., PI_m)\}$$

Given these definitions, we would like to point out, that properties here exclusively refer to so-called data type properties. Hence, opposed to [9], we define properties as being distinctive to relations R. The latter describe relations between concepts and instances. In addition, A represents a set of *axioms* which define constraints on the other introduced notions.

2.2. Ontology Integration based on Ontology Mapping

The widespread use of ontologies together with the increasing availability of distinct ontologies representing overlapping domains of interest, raises the need to resolve heterogeneities between distinct conceptualisations [20], i.e. to integrate different ontologies by partially mapping these. With respect to [2] and [31], we define *ontology mapping* as the creation of structure-preserving relations between multiple ontologies. I.e. the goal is, to establish formal relations between a set of knowledge entities E_1 from an ontology O_1 with entities E_2 which represent the same or a similar semantic meaning in a distinct ontology O_2 [8][9][35]. The term *set of entities* here refers to the union of all concepts C, instances I, relations R and axioms A defined in a particular ontology. In that, the ontology mapping task strongly relies on identifying *semantic similarities* [1] between entities across different ontologies. Hence, the identification of similarities is a necessary requirement to solve the mapping problem for multiple heterogeneous ontologies [29]. However, with respect to this goal, the following issues have to be taken into account.

I1. *Symbolic representations lack grounding at the conceptual level*: the symbolic approach, i.e. describing symbols by using other symbols, without a grounding in the real world, of established SW representation standards, leads to ambiguity issues and does not entail meaningfulness, since meaning requires both the definition of a terminology in terms of a logical structure (using symbols) and grounding of symbols to a conceptual level [3][27].

I2. *Lack of implicit similarity representation*: Due to *I1*, describing the complex notion of any specific concept in all its facets is a costly task and may never reach semantic completeness. While concept representations across distinct ontologies – even those representing the same real-world entities – hardly equal another,

semantic similarity is not an implicit notion within ontological representations. But manually or semi-automatically defining similarity relationships is costly. Moreover, such relationships are hard to maintain in the longer term.

In order to overcome these issues, recent research approaches address semi-automatic similarity detection across ontologies, mostly based on identifying linguistic commonalities and/or structural similarities between entities of distinct ontologies [28][2]. Work following a combination of such approaches in the field of ontology mapping is reported in [24][10][16][20]. The PROMPT suite [28] exploits the content of concept and instance labels together with structural information in order to support ontology merging. GLUE, proposed in [7], follows a similar approach to enable ontology mapping, but also incorporates machine learning techniques to enable similarity detection. The work proposed in [21] follows a pure linguistic approach to ontology mapping. Moreover, similar related work had been carried out to facilitate database schema matching [22][23]. However, it can be stated that the approaches reported above rely on the idea of (semi-)automating the similarity detection process which in all the above cases requires manual intervention and hence, is a costly and error-prone process. In that, we argue that representational facilities for implicit representation of similarities are required to overcome the need for explicit ontology mappings.

2.3. Conceptual Spaces - A viable Alternative?

Conceptual Spaces (CS), introduced by Gärdenfors [14][15], follow a theory of describing entities at the conceptual level in terms of their natural characteristics similar to natural human cognition in order to avoid the symbol grounding issue. CS consider the representation of concepts as multidimensional geometrical spaces which are defined through a set of quality dimensions. Instances are supposed to be represented as vectors, i.e. particular points in a CS. For instance, a particular color may be defined as point described by vectors measuring the quality dimensions hue, saturation, and brightness. Describing instances as points within vector spaces where each vector follows a specific metric enables the automatic calculation of their semantic similarity by means of distance metrics such as the Euclidean, Taxicab or Manhattan distance [19] or the Minkowsky Metric [34]. Hence, in contrast to the costly formalisation of such knowledge through symbolic representations, semantic similarity is implicit information carried within a CS representation what is perceived the major contribution of the CS theory. However, although CS aim at solving SW-related issues such as the symbol grounding problem, several issues still have to be taken into account:

I3. Lack of representational facilities to base knowledge models on CS;
I4. Lack of expressiveness to represent arbitrary relations;
I5. Undefinable scope of particular dimensions;
I6. Reliance on quantifiable measurements, even for qualitative characteristics.

CS do not provide any representational mechanism enabling the application of CS for knowledge representation (*I3*) in order to solve the aforementioned issues *I1* and *I2* (Section 2.2). Moreover, the CS theory does not provide any notion to represent any

arbitrary relations (*I4*) [33], such as *part-of* relations which usually are represented within FOL-based knowledge models. In this regard, it is even more obstructive that the scope of a dimension is not definable (*I5*), i.e. a dimension always applies to the entire CS [33]. Nevertheless, similarity computation as major contribution of CS particularly requires the description of concepts through quantifiable metrics (*I6*), even in cases of rather qualitative characteristics.

3 Conceptual Groundings for Ontological Concepts

With respect to issues *I1-I6* (Section 2), we claim that basing knowledge models on just one theory is not sufficient, and hence, a combination of both representation approaches appears to be better suited. Moreover, it can be argued, that representing an entire knowledge model through a coherent CS might not be feasible, particularly when attempting to maintain the meaningfulness of the spatial distance as a similarity measure. Hence, we claim that CS represent a particularly promising model when being applied to individual concepts instead of representing an entire ontology in a single CS.

3.1. Conceptual Groundings for Ontological Concepts

We propose a two-fold representational approach – combining FOL ontologies with corresponding representations based on CS – to enable similarity computation across ontologies. In that, we consider the representation of a set of *n* concepts *C* of an ontology *O* through a set of *n* Conceptual Spaces *CS*. Instances of concepts are represented as members in the respective CS. The following Figure 1 depicts this vision:
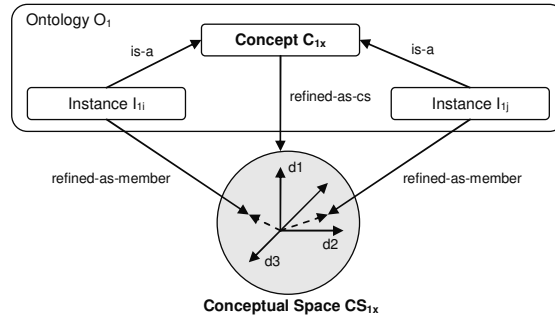

Fig. 1. Representing FOL-based concepts through Conceptual Spaces.

While still benefiting from implicit similarity information within a CS, our hybrid approach allows overcoming CS-related issues (Section 2.3) by maintaining the advantages of FOL-based knowledge representations. In order to be able to refine and represent ontological concepts within a CS, we formalised the CS model into an ontology, currently being represented through OCML [26]. Hence, a CS can simply be instantiated in order to represent a particular concept.

Referring to [15][32], we formalise a CS as a vector space defined through quality dimensions d_i of *CS*. Each dimension is associated with a certain metric scale, e.g.

ratio, interval or ordinal scale. To reflect the impact of a specific quality dimension on the entire CS, we consider a prominence value p for each dimension [15]. Therefore, a CS is defined by

$$CS^n = \{(p_1 d_1, p_2 d_2, ..., p_n d_n) | d_i \in CS, p_i \in P\}$$

where P is the set of real numbers. However, the usage context, purpose and domain of a particular CS strongly influence the ranking of its quality dimensions. This clearly supports our position of describing distinct CS explicitly for individual concepts. Please note that we do not distinguish between dimensions and domains [15] but enable dimensions to be detailed further in terms of subspaces. Hence, a dimension within one space may be defined through another CS by using further dimensions [32]. In this way, a CS may be composed of several subspaces and consequently, the description granularity can be refined gradually. Dimensions may be correlated. For instance, when describing an apple the quality dimension describing its sugar content may be correlated with the taste dimension. Information about correlation is expressed through axioms related to a specific quality dimension instance.

A particular member M – representing a particular instance – in the CS is described through valued dimension vectors v_i:

$$M^n = \{(v_1, v_2, ..., v_n) | v_i \in M\}$$

With respect to [32], we define the semantic similarity between two members of a space as a function of the Euclidean distance between the points representing each of the members. Hence, with respect to [32], given a CS definition CS and two members V and U, defined by vectors $v_0, v_1, ..., v_n$ and $u_1, u_2, ..., u_n$ within CS, the distance between V and U can be calculated as:

$$dist(u, v) = \sqrt{\sum_{i=1}^{n} p_i ((\frac{u_i - \bar{u}}{s_u}) - (\frac{v_i - \bar{v}}{s_v}))^2}$$

where $\bar{u}$ is the mean of a dataset U and s_u is the standard deviation from U. The formula above already considers the so-called Z-transformation or standardization [4] which facilitates the standardization of distinct measurement scales utilised by different quality dimensions in order to enable the calculation of distances in a multi-dimensional and multi-metric space.

3.2. Representing Ontological Concepts through Conceptual Spaces

The derivation of an appropriate space CS_i to represent a particular concept C_i of a given ontology O is understood a non-trivial task which aims at the creation of a CS instance which most appropriately represents the real-world entity represented by C_i. We particularly foresee a transformation procedure consisting of the following steps:

S1. Representing concept properties pc_{ij} of C_i as dimensions d_{ij} of CS_i.
S2. Assignment of metrics to each quality dimension d_{ij}.
S3. Assignment of prominence values p_{ij} to each quality dimension d_{ij}.
S4. Representing instances I_{ik} of C_i as members in CS_i.

Given the formal ontological representation of the CS model (Section 3.1), we are able to simply instantiate a specific CS by applying a transformation function

$$trans : C_i \Rightarrow CS_i$$

which is aimed at instantiating all elements of a CS, such as dimensions and prominence values *(S1 – S3)*. *S1* aims at representing each concept property pc_{ij} of C_i as a particular dimension instance d_{ij} together with a corresponding prominence p_{ij} of a resulting space CS_i:

$$trans : \left\{ \left(pc_{i1}, pc_{i2}, ..., pc_{in} \right) \middle| pc_{ij} \in PC_i \right\} \Rightarrow \left\{ \left(p_{i1}d_{i1}, p_{i2}d_{i2}, ..., p_{in}d_{in} \right) \middle| d_{ij} \in CS_i, p_{ij} \in P \right\}$$

Please note that we particularly distinguish between data type properties and relations. While the latter represent relations between concepts, these are not represented as dimensions since such dimensions would refer to a range of concepts (instances) instead of quantified metrics, as required by *S2*. Therefore, in the case of relations, we propose to maintain the relationships represented within the original ontology O without representing these within the resulting CS_i. In that, the complexity of CS_i is reduced to enable the maintainability of the spatial distance as appropriate similarity measure. The assignment of metric scales to dimensions (*S2*) which naturally are described using quantitative measurements, such as size or weight, is rather straightforward. In such cases, interval scale or ratio scale, could be used, whereas otherwise, a nominal scale might be required. *S3* is aimed at assigning a prominence value p_{ij} – chosen from a predefined value range – to each dimension d_{ij}. Since the assignment of prominences to quality dimensions is of major importance for the expressiveness of the similarity measure within a CS, most probably this step requires incremental ex-post re-adjustments until a sufficient definition of a CS is achieved.

With respect to *S4*, one has to represent all instances I_{ki} of a concept C_i as member instances in the created space CS_i:

$$trans : I_{ik} \Rightarrow M_{ik}$$

This is achieved by transforming all instantiated properties pi_{ikl} of I_{ik} as valued vectors in CS_i.

$$trans : \left\{ \left(pi_{ik1}, pi_{ik2}, ..., pi_{ikn} \right) \middle| pi_{ikl} \in PI_{ik} \right\} \Rightarrow \left\{ \left(v_{ik1}, v_{ik2}, ..., v_{ikn} \right) \middle| v_{ikl} \in M_{ik} \right\}$$

Hence, given a particular CS, representing instances as members becomes just a matter of assigning specific measurements to the dimensions of the CS. In order to represent all concepts C_i of a given ontology O, the transformation function consisting of the steps *S1-S4* has to be repeated iteratively for all C_i which are element of O. The accomplishment of the proposed procedure results in a set of CS instances which each refine a particular concept together with a set of member instances which each refine a particular instance. Please note that applying the procedure proposed here requires additional effort which needs to be further investigated within future work.

4 Enabling Ontology Interoperability

In order to illustrate the actual contribution of our hybrid representation method with respect to ontology integration and interoperability, we define a simplified scenario. Please note, the following simplifications are not requirements for the utilisation of our approach in general but just aim at describing an environment to formalise and compare our contribution.

i.   Two ontologies O_1 and O_2 represent the same domain of interest.
ii. Each concept of O_1 can be mapped, i.e. is similar, to one of the concepts of O_2 and vice versa.
iii. Each instance of O_1 can be mapped to one of the instances of O_2 and vice versa.
iv. All relations R and axioms A constraining concepts of O_1 apply to the similar concepts C_2 of O_2 and vice versa.
v.   Spatial distance in a CS is perceived to be a valid similarity measure.

With respect to *i-v* and [8], we define the ontology integration problem as a problem of (a) identifying the most similar concept C_{2j} for a given concept C_{1i} and (b) identifying the most similar instance I_{2j} for a given instance I_{1i}. Please note, that ontologies in many cases are not completely heterogeneous. While, on the one hand, the increasing use of upper-level ontologies such as DOLCE [13], SUMO [30] or OpenCyc[3] supports a certain degree of commonality between distinct ontologies, on the other hand, ontologies are often used in rather closed environments, for instance, virtual organisations, where a common agreement to a certain extent is ensured. Therefore, we distinguish between three cases with respect to the extent of common agreement required from the involved parties.

4.1. Case 1 – Shared Ontology at the Concept Level

This case considers two parties ("agents" in Figure 2), which share an ontology at the conceptual level, but not at the instance level.
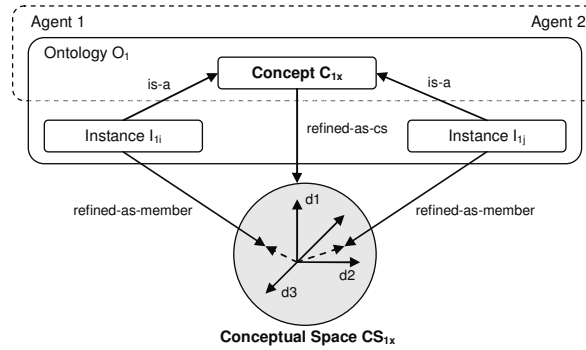

Fig. 2. Two parties sharing a common ontology at the concept level.

This occurs, for instance, in cases where two parties subscribe to a common schema, e.g. database or ontology schema, to represent institutional knowledge, which is then instantiated independently. As described above, we assume this to be a common case. Whereas no mappings at the concept level need to be defined in this case, similarity-detection at the instance level is obsolete, since it is indicated by means of Euclidean distances within the respective CS (Section 3). Hence, even though the ontological commitment of both parties just applies to the concept-level, similarity at the instance level becomes an implicit notion when following the proposed representation

---

[3] http://www.opencyc.org/

approach. In contrast, without our representational model, all instances I_{1ki} of a concept C_{1k} in O_1 would have to be compared with all instances I_{2lj} of a similar concept C_{2l} in O_2. Generally speaking, assuming n being the number of similar concepts C_i in O_1 and O_2 where each concept has m_j instances I_k, leads to the following number x of required similarity comparisons:

$$x(I(O_1), I(O_2)) = \sum_{j=1}^{n} (m_j)^2$$

Even though such similarity comparisons could be semi-automated, we assume that manual involvement is required in any case.

4.2. Case 2 – Distinct Ontologies, shared Conceptual Spaces

The case of distinct ontologies which still subscribe to a common set of CS (Figure 3) is likely, in cases where two parties created distinct ontologies O_1 and O_2 and decide ex-post to represent concepts following the procedure described in Section 3.2 in order to take advantage of implicit similarity computation.
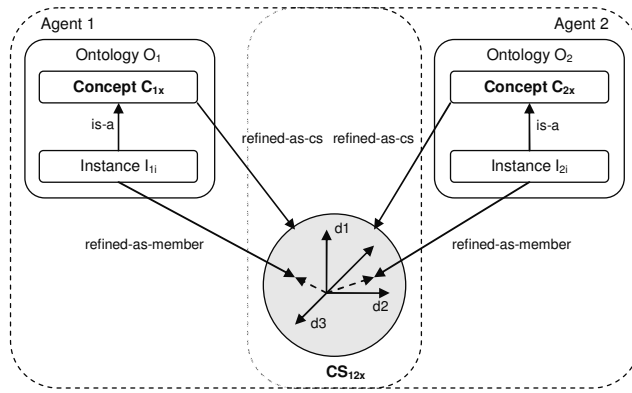

Fig. 3. Two parties with distinct ontologies sharing equivalent sets of Conceptual Spaces.

Given such a scenario, concept similarity is implicitly defined through concept refinement in the equivalent CS, i.e. two concepts agreeing on the same CS representation necessarily are similar, if not equivalent. Instance similarity is computable by means of the spatial distance. In case a CS-based representation as shown in Figure 3 is not provided already, beforehand the ontological concepts have to be mapped in order to be able to agree on a common CS for each concept. Hence, creating the requirements for this case from a set of distinct ontologies, would require n^2 similarity comparisons, with n being the number of concepts within each O_1 and O_2. In contrast, following traditional ontology mapping approaches would require additional comparisons to map instances (Section 4.1), in order to fully enable mapping between both ontologies:

$$x(I(O_1), I(O_2)) = n^2 + \sum_{j=1}^{n} (m_j)^2$$

4.3. Case 3 – Distinct Ontologies, (partially) heterogeneous Conceptual Spaces

Case 3 introduces another degree of heterogeneity by assuming distinct ontologies together with partially overlapping (Figure 4.i) or even completely heterogeneous CS (Figure 4.ii). Such a case is particularly likely either where two agents independently create ontologies and corresponding CS representations or in cases where two initially equivalent sets of CS evolve through time, and consequently, develop heterogeneities.
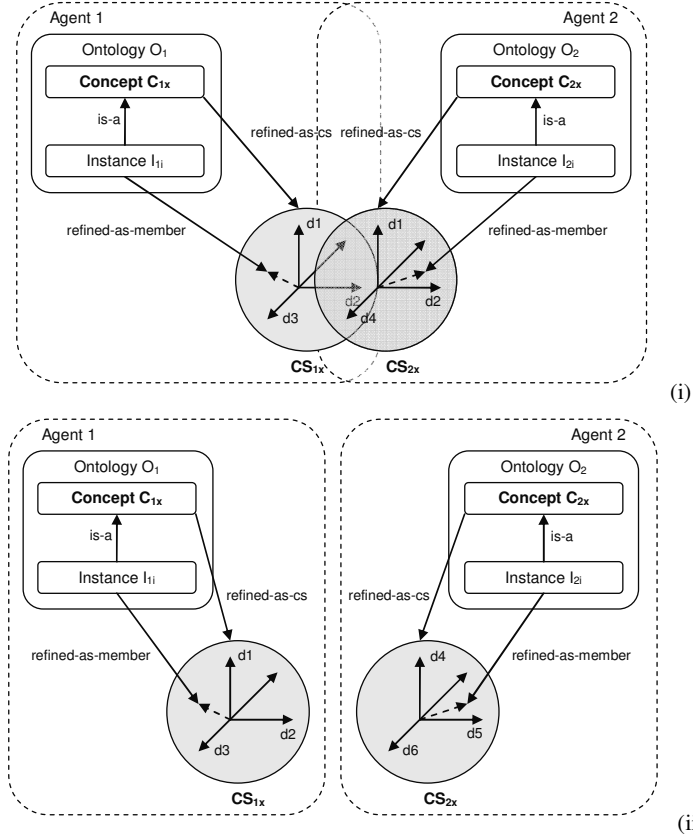

(i)

(ii)

Fig. 4. Two parties with distinct ontologies sharing (i) partially overlapping and (ii) completely distinct Conceptual Spaces.

In the cases shown in Figure 4, certain restrictions regarding the CS-based similarity-detection between instances would apply. First, traditional ontology mapping methods (Section 2.2), could be applied in order to identify similar dimensions between heterogeneous spaces to increase the amount of overlapping dimensions. Subsequently, similarities, i.e. distances, could be computed between members in two partially overlapping CS by just considering the overlapping dimensions. Let us assume a member U in CS_1 with n dimensions d_n and a member V in a partially overlapping CS_2 with m dimensions d_m. The assumption that both CS partially overlap

implies the existence of a set of l dimensions in CS_3 with $CS_3 \subset CS_1$ and $CS_3 \subset CS_2$. Hence, by disregarding the non-overlapping dimensions, similarity could still be computed utilizing the overlapping dimensions as follows:

$$d(u,v) = \sqrt{\sum_{i=1}^{l} p_i(((\frac{u_i - \overline{u}}{s_u}) - (\frac{v_i - \overline{v}}{s_v}))^2}$$

In contrast, following traditional approaches to ontology mapping between two completely independent ontologies would require the same amount of similarity-comparisons as proposed in Section 4.2.

5 Evaluation

To evaluate the applicability of our approach, initial proof-of-concept prototype applications were provided [5][6] which apply the hybrid representational approach proposed in this paper to enable similarity-based matchmaking between distinct representations of Semantic Web Service (SWS) [12] capabilities. There, an environment as described in Section 4.2 (case 2) was established by enabling a SWS provider (agent 1) to refine symbolic SWS capability descriptions through CS. In that, by following the approach proposed here, concept instances as part of SWS capability descriptions had been individually represented within CS-based representations. On the other hand, heterogeneous user data (agent 2) is dynamically represented as members in the same CS. Instead of (semi-automatically) mapping distinct ontologies utilized by both agents, similarities are computed as proposed in Section 3 by automatically calculating Euclidean distances between a set of CS members. Due to space restrictions here, we would like to refer the reader to [5] and [6] for further details.

In order to further evaluate the contribution of our approach, in the following, we provide an attempt to compare the required number of similarity computations, following our approach on the one hand, and following traditional FOL-based ontological representations on the other. However, please note that the authors are aware that providing representations following our two-fold approach requires additional effort to provide the representations enabling to benefit from the contributions discussed here. In the following, we distinguish between cases 1-3 (Section 4) and define a set of additional simplifications which further detail a concrete ontology interoperability scenario:

vi. Ontologies O_1 and O_2 each consist of n concepts C_j (C_k) with m_j instances I_j (I_k) each.
vii. Distinct degrees of heterogeneity between O_1 and O_2 are considered with respect to the case differentiation proposed in Section 4.

Following *i-vii* and with respect to the elaborations in Section 4 we assume efforts for similarity-detection as follows (summarised in Table 1). As described in Sections 4.1 and 4.2, following our hybrid representational approach (b in Table 1) would require an additional representational effort (Section 3.2) but no additional alignment tasks, respectively similarity comparisons in cases 1 and 2. In contrast, following the current symbolic approach (a) requires to semi-automatically identify instance similarities (case 1) and to additionally detect concept similarities in case 2. The same applies to

case 3 when following approach (a). With respect to approach (b), congruent CS have to be provided in case 3. Even though similarities can be computed partially (Section 4.3), we consider the worst case scenario for both cases, i.e. the need to manually align distinct spaces. Hence, we take into account the formal alignment of both ontologies at the concept level leading to n^2 necessary similarity comparisons.

Table. 1. Formalisation of required similarity comparisons to align heterogeneous ontologies.

	Case 1	Case 2	Case 3
(a) First-order logic representation:	$\sum_{j=1}^{n} (m_j)^2$	$n^2 + \sum_{j=1}^{n} (m_j)^2$	$n^2 + \sum_{j=1}^{n} (m_j)^2$
(b) Two-fold representation:	-	-	n^2 (or effort to align CS)

For instance, assuming the likely case 3, we can apply the formalisations (Table 1) and assume 2 ontologies O_1 and O_2 each consisting of n concepts C_j and C_k with m_j instances I_{jl} of each concept, to calculate the number of required similarity comparisons under the assumptions *i-vii*. Figure 5 depicts the expected number of similarity comparisons for the traditional approach (a) in case 3. Since in the case of our proposed solution (b) instance similarity is an implicit notion, the first row ($m_j=0$) of the table also indicates the respective number of similarity comparisons following approach (b).
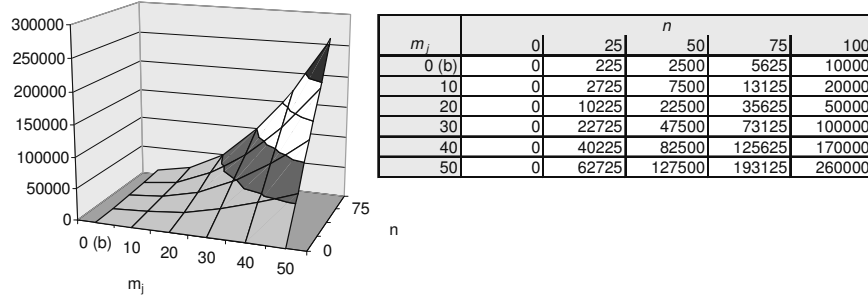

m_j				n		
		0	25	50	75	100
0 (b)		0	225	2500	5625	10000
10		0	2725	7500	13125	20000
20		0	10225	22500	35625	50000
30		0	22725	47500	73125	100000
40		0	40225	82500	125625	170000
50		0	62725	127500	193125	260000

Fig. 5. Required similarity comparisons to map between two ontologies O_1 and O_2 in case 3 dependent on number of concepts n and number of instances m_j.

As shown in Figure 5, solution (b) significantly reduces the amount of required similarity comparisons, which increase with a growing number of instances m_j when following solution (a). Even though an additional effort is required to apply our representational model (Section 3.2) this reduction is perceived to be the major contribution of solution (b). Whereas the majority of assumptions *i-vii* just aims at describing a formal and comparable environment one might particularly doubt the validity of *v*. However, within previous work [5][6], the authors already proved the appropriateness of distance metrics in a CS as similarity measure.

Consequently, adopting our approach enabled similarity detection across heterogeneous ontologies instead of manually aligning individual instances. It is apparent that an initial effort has to be made to represent heterogeneous concepts in common CS and to represent instances as corresponding vectors. However, once these

representations are available, similarity becomes an implicit notion and does not require manual or (semi-)automatic alignments.

6 Conclusions

In order to facilitate ontology integration we proposed a hybrid representation approach based on a combination of FOL-based ontologies and multiple concept representations in individual CS. Representing concepts following the CS theory enables representation of instances as vectors in a respective CS and consequently, the automatic computation of similarities by means of spatial distances between distinct vectors. The CS-based representation is supported through a dedicated CS formalisation, i.e. a CS ontology which enables the instantiation of a corresponding CS (member) for each individual concept (instance) as described in Section 3. Following our two-fold representational approach supports implicit representation of similarities between instances across heterogeneous ontologies, and consequently, provides a means to facilitate ontology interoperability. Moreover, by maintaining the knowledge represented within FOL-based ontologies but additionally applying the CS approach to individual concepts of each ontology, our approach overcomes the individual issues posed by each of the two approaches (Sections 2 and 2.3). For instance, it allows representing arbitrary relationships between distinct concepts and instances, and consequently, between distinct CS while still taking advantage of the implicit similarity information inherent in a CS representation. In order to facilitate our approach, we furthermore proposed a formal method on how to derive CS representations for individual concepts (Section 3). Within proof-of-concept prototype applications [5][6], an OCML [26] representation of the proposed hybrid representational model was utilized to validate the applicability of the approach.

As shown in Section 5 and in previous applications of this approach, applying our proposed representational approach significantly reduces the effort required to align distinct heterogeneous ontologies and the extent to which two distinct parties have to share their conceptualisations. Whereas traditional ontology mapping methodologies rely on mechanisms to semi-automatically detect similarities at the concept and the instance level, our approach just requires a common agreement at the concept level since similarity information at the instance level is implicitly defined.

However, the authors are aware that our approach requires a considerable amount of additional effort to establish CS-based representations. Future work has to investigate this effort in order to further evaluate the potential contribution of our approach proposed here. Moreover, while overcoming issues *I1 – I6* (Sections 2 and 2.3), further issues related with CS-based knowledge representation still remain. For instance, whereas defining instances, i.e. vectors, within a given CS appears to be a straightforward process of assigning specific quantitative values to quality dimensions, the definition of the CS itself is not trivial at all and dependent on individual perspectives and subjective appraisals. Whereas semantics of instances are grounded to metrics within a CS, the quality dimensions themselves are subject to ones interpretation what might lead to ambiguity issues. With regard to this, CS do not fully solve the symbol grounding issue but to shift it from the process of describing instances to the definition of a CS. Furthermore, whereas the size and resolution of a CS is indefinite, defining a reasonable CS may become a challenging

task. Nevertheless, distance calculation not only relies on the fact that quantitative metrics are established but also that resources are described in equivalent geometrical spaces. However, particularly with respect to the latter, traditional ontology and schema matching methods could be applied to align heterogeneous spaces. Moreover, we would like to point out that the increasing usage of upper level ontologies, such as DOLCE or SUMO, and the progressive reuse of ontologies, particularly in loosely coupled organisational environments, leads to an increased sharing of ontologies at the concept level. As a result, our proposed hybrid representational model becomes increasingly applicable by further enabling similarity-computation at the instance-level towards the vision of interoperable ontologies.

7 References

[1] Bisson, G. (1995). Why and how to define a similarity measure for object based representation systems. Towards Very Large Knowledge Bases, pages 236–246, 1995.

[2] Choi, N., Song, I., and Han, H. (2006), A survey on ontology mapping, SIGMOD Rec., Vol. 35, No. 3. (September 2006), pp. 34-41.

[3] Cregan, A. (2007), Symbol Grounding for the Semantic Web. 4th European Semantic Web Conference 2007, Innsbruck, Austria.

[4] Devore, J., and Peck, R. (2001), Statistics - The Exploration and Analysis of Data, 4th ed. Pacific Grove, CA: Duxbury, 2001.

[5] Dietze, S., Gugliotta, A., Domingue, J., (2008) Conceptual Situation Spaces for Situation-Driven Processes. 5th European Semantic Web Conference (ESWC), Tenerife, Spain.

[6] Dietze, S., Gugliotta, A., Domingue, J., (2008) Mobile Situation Spaces. Workshop: International Workshop on the Role of Services, Ontologies, and Context in Mobile Environments (RoSOC-M '08), The 9th International Conference on Mobile Data Management (MDM'08), Beijing, China.

[7] Doan, A., Madhavan, J., Domingos, P., and Halevy, A. (2002), Learning to map between ontologies on the semantic web. In Proceedings of the Eleventh International World Wide Web Conference, Honolulu, Hawaii, USA, May 2002.

[8] Ehrig, M., and Staab, S. QOM - Quick Ontology Mapping. International Semantic Web Conference 2004, 683-697

[9] Ehrig, M, Sure, Y. (2004), Ontology Mapping - An Integrated Approach, in Proceedings of ESWS, 2004.

[10] Euzenat, J., Guegan, P., and Valtchev, P. OLA in the OAEI 2005 Alignment Contest. K-Cap 2005 Workshop on Integrating Ontologies 2005, 97-102.

[11] Felder, R. M., Silverman, L. K. (1988). Learning and Teaching Styles in Engineering Education, Engineering Education, 78, 1988, pp. 674–681. Preceded by a preface in 2002: http://www.ncsu.edu/felderpublic/ Papers/LS-1988.pdf.

[12] Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J. (2006): Enabling Semantic Web Services – The Web service Modelling Ontology, Springer 2006.

[13] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L. (2002). Sweetening Ontologies with DOLCE. In: A. Gómez-Pérez , V. Richard Benjamins (Eds.) Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web: 13th International Conference, EKAW 2002, Siguenza, Spain, October 1-4, 2002.

[14] Gärdenfors, P. (2000), Conceptual Spaces - The Geometry of Thought. MIT Press, 2000.

[15] Gärdenfors, P. (2004), How to make the semantic web more semantic. In A.C. Vieu and L. Varzi, editors, Formal Ontology in Information Systems, pages 19–36. IOS Press, 2004.

[16] Giunchiglia, F., Shvaiko, P., and Yatskevich, M. S-Match: An Algorithm and an Implementation of Semantic Matching. ESWS 2004, 61-75

[17] Gruber, T. R., Toward principles for the design of ontologies used for knowledge sharing. International Journal of Human-Computer Studies, Vol. 43, Issues 4-5, November 1995, pp. 907-928.

[18] Klein, M., Ding, Y., Fensel, D., and Omelayenko, B. (2002), Ontology management - Storing, aligning and maintaining ontologies. In J. Davis et al. (eds.), Towards the Semantic Web: Ontology-Driven Knowledge Mangement, Wiley, 2002.

[19] Krause, E. F. (1987). Taxicab Geometry. Dover.

[20] Le, B.T., Dieng-Kuntz, R., and Gandon, F. On Ontology Matching Problems - for Building a Corporate Semantic Web in a Multi-Communities Organization. ICEIS (4) 2004, 236-243.

[21] Li, J. (2004), LOM: A Lexicon-based Ontology Mapping Tool, Proceedings of the Performance Metrics for Intelligent Systems (PerMIS. '04), 2004.

[22] Madhavan, J., Bernstein, P.A., and Rahm, E. Generic Schema Matching with Cupid. VLDB 2001, 49-58

[23] Melnik, S., Garcia-Molina, H., and Rahm, E. Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. ICDE 2002, 117-128

[24] Mitra, P., Noy, F. N., Jaiswals, A. (2005), OMEN: A Probabilistic Ontology Mapping Tool, International Semantic Web Conference 2005.

[25] Moncan, A., Cimpian, E., Kerrigan, M., Formal Model for Ontology Mapping Creation, in I. Cruz et al. (Eds.): ISWC 2006, LNCS 4273, pp. 459–472, 2006. Springer-Verlag Berlin Heidelberg 2006.

[26] Motta, E. (1998). An Overview of the OCML Modelling Language, the 8th Workshop on Methods and Languages, 1998.

[27] Nosofsky, R. (1992), Similarity, scaling and cognitive process models, Annual Review of Psychology 43, pp. 25- 53, (1992).

[28] Noy, N.F., and Musen, M.A. The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping. International Journal of Human-Computer Studies 59 (2003), 983-1024.

[29] Qu, Y., Hu, W., Cheng, G. (2006) Constructing Virtual Documents for Ontology Matching, WWW 2006, May 23–26, 2006, Edinburgh, Scotland. ACM 1595933239/06/0005.

[30] Pease, A., Niles, I., Li, J.(2002), The suggested upper merged ontology: A large ontology for the semanticweb and its applications. In: AAAI-2002Workshop on Ontologies and the Semantic Web. Working Notes (2002)

[31] Pinto, S., H., Gomez-Perez, A., Martins, J. P. (1999), Some Issues on Ontology Integration, In Proc. of IJCAI99's Workshop on Ontologies and Problem Solving Methods: Lessons Learned and Future Trends, 1999.

[32] Raubal, M. (2004) Formalizing Conceptual Spaces. in: A. Varzi and L. Vieu (Eds.), Formal Ontology in Information Systems, Proceedings of the Third International Conference (FOIS 2004). Frontiers in Artificial Intelligence and Applications 114, pp. 153-164, IOS Press, Amsterdam, NL.

[33] Schwering, A. (2005). Hybrid Model for Semantic Similarity Measurement, in R. Meersman and Z. Tari (Eds.): CoopIS/DOA/ODBASE 2005, LNCS 3761, pp. 1449 – 1465, 2005..

[34] Suppes, P., D. M. Krantz, et al. (1989). Foundations of Measurement - Geometrical, Threshold, and Probabilistic Representations. San Diego, California, USA, Academic Press, Inc.

[35] Xiaomeng Su. (2002). A text categorization perspective for ontology mapping, Technical report, Department of Computer and Information Science, Norwegian University of Science and Technology, Norway, 2002.