Ontology Mapping for Interagent Communication

Ioan Alfred LETIA, Mihai COSTIN
Technical University of Cluj-Napoca, Department of Computer Science
Baritiu 28, RO-400391 Cluj-Napoca, Romania

Abstract—**The problem of ontology mapping is tackled in the context of MAS and real life ontologies, context that requires an increased degree of dynamism and the ability to handle high amounts of data. Our solution follows the guidelines of lazy mapping, striving to map only what is necessary and only when it is necessary by using lexical based mapping methods combined with proxy ontologies. However, we are not omitting from this endeavor the accuracy of mapping since our two layer process has FCA as the second mapping step.**

Index Terms—**ontologies, mismatch, MAS, lazy mapping, FCA.**

I. CONTEXT. PROBLEMS

A. Heterogeneity in MAS

Heterogeneity in MAS can be caused by multiple reasons, both technical and economical and removing this heterogeneity by aligning all the available knowledge to a standard might prove more than difficult, if not impossible. Moreover, even if communication is hampered by this issue, the heterogeneity is one of the strong points in a multiagent system since each part of the system is usually specialized on a given direction, allowing the MAS to handle problems in an efficient way. In this case specialization means specific data structures, tools and reasoning systems, all contributing to the heterogeneity element of the system.

Beside the mentioned aspects so far, communication and information sharing between two entities makes sense only if the two have differences in knowledge, otherwise there would be no possible information for them to exchange.

Under these circumstances, the solution to communication in MAS that are using heterogeneous ontologies must come not from imposing a standard to all the available knowledge but from a form of translation and different mapping techniques. These processes should solve the problems of agent communication and at the same time keep the MAS functionality strong points unaltered.

Another important issue regarding the mentioned process is the fact that since we are dealing with a MAS, we can not adopt a whole-ontology or a priori-mapping. The mapping process should be, as much as possible, done by agents only when they need it and it should include only those concepts that are required at a certain moment. This idea is sustained also by the real-life ontologies, as we will see in the described scenario.

B. Syntactic Heterogeneity

Two or more ontologies can present differences at the level of syntax because of the multiple representation languages for ontologies.

Even though in the recent period OWL[1] seems to have been more or less chosen as a standard language candidate, there is no guaranty that an actual standardization will occur. Moreover, even if a standard would be adopted, it would prove almost impossible to translate all the existing ontologies, related tools and technology, including the agents using those ontologies to the new standard. Ontologies using other languages, even if less expressive than the one previously mentioned, were constructed and used with success for more than a couple of years now. Obviously, the effort and costs to switch to another technology, when this action can only be backed up by the ideal of a standard, would not easily be accepted by anyone and so, assuming that all the ontologies the agents will have to deal with are written using the same language might prove to be not such a good idea.

However, even if syntactic heterogeneity is still an issue, the problem in this case is not as difficult as the one raised by the semantic mismatches. Different tools and frameworks have already been developed in order to permit transparent access for an agent to the information contained in an ontology (e.g. Jena[2]).

The problem that still needs addressing in this case is the scenario when an agent using a much more expressive language wants to transmit some information to another agent, information that is beyond the syntactic capacities of the receiver's ontology language. A simple example in this matter are two agents that interact, one being backed-up by an ontology using OWL as representing language, while the other is using RDFS[3] . Obviously, the agent using RDFS won't be able to cope with some of the constructions made by OWL.

C. Semantic heterogeneity

The semantic mismatches at ontology level can be caused by a variety of reasons. The most obvious one is the different domain case - two or more ontologies that describe different domains, possibly with no overlapping at all. In the extreme case, sharing information between agents might prove impossible without involving the process of ontology merging. Mapping would not be of very much help in this case since the two domains most likely have only a few similar concepts and without any common ground there would be no starting point for the process.

The next situation is the one with different domains that have an overlapping area. In this case, the overlapping concepts, besides being the subject of the mapping process

---

1 http://www.w3.org/OWL
2 http://jena.sourceforge.net/
3 http://www.w3.org/TR/rdf-schema/

would also provide a connection bridge between the two ontologies, making possible further mapping and concept translation even if a higher computational cost would be involved.

When the subject ontologies are referring to the same domain and the concepts involved in the communication act are in a mismatch situation, the mapping process becomes the most suitable. In this case, as well as for the overlapping area from the previous one, the cause of heterogeneity is represented by the concepts and relations between them and that is the main problem that must be faced.

Concepts of the same domain can be in a mismatch situation mainly for three reasons: same notation for different concepts, different notations for the same concept, and different relations or properties. Also, we can have the case of partially defined concepts in both ontologies - e.g. when a common ontology concept with same notation and same underlying meaning has part of its properties defined in one ontology, and other properties in the second one. Likewise, a concept may be viewed from different perspectives in different ontologies, thus giving birth to different properties that actually refer to the same elements. Another aspect, that is making the mapping process even more cumbersome, is the relations between concepts that may differ from ontology to ontology.

Another source of mismatch that may go even beyond the boundaries of the ontology level is the use of different logics and reasoning systems. Agents that have different ways of using the available knowledge will most likely get into conflictual concept-related situations when trying to communicate. This aspect is not easily tackled and for the time being will not make the point of this paper although we admit its importance and we do have it in plan for future work as noted also in.

D. Hidden Ontologies

A very important aspect of the ontologies being used in MAS is the fact that most of them are hidden, or not intended to be public available. By this we mean that access is restricted to the agents that rely on them, those agents representing the only entry point to the knowledge stored in the ontology. This approach may have various reasons at its roots, as it is for example security, but we also consider that it is a best practice to view the agent and the ontology it relies on as a black box since an agent may also use a composite ontology. Such an ontology could have parts that are internal to the agent, created and used on the fly, and so not accessible to other agents.

We assume thus that the ontologies are not public available and this presumption excludes the possibility of an a priori whole ontology mapping, in order for the agents to use a common ontology for communication. The whole ontology mapping is also not feasible for performance reasons since it would be impossible to predict all the interactions between MAS actors in order to decide what ontologies to map. Moreover, when two or more agents interact, it may be that they need to use, for the communication process, only a very small part of the available underlying concepts and so mapping all the concepts from all the involved ontologies would be more than a waste of time and effort.

II. SCENARIO

For the proposed scenario we decided to use two ontologies from the medical domain, namely the Foundational Model of Anatomy ontology or FMA[4] and Generalized Architecture for Languages, Encyclopedias and Nomenclatures in medicine or GALEN[5] . The choice was made towards these ontologies because we needed a real case scenario, with elements coming from the real world environment, so that our proposed solution, if proven to be an efficient one, could successfully step into practice. The chosen ontologies are, as one can easily see, large knowledge repository with more that 75000 classes that are being used with success for more than a couple of years now even if they still have some flaws.

They are also very well suited for our mapping purpose since both refer to the medical domain but have different approaches for representing knowledge as well as different views. While the FMA represents a body of knowledge focusing mainly on human anatomy and trying to provide a solid machine-based platform for bioinformatics, GALEN has its declared domain as medical concepts, thus including a wider area from the medical domain and taking interest also in clinical notions other than anatomical ones.

The proposed scenario domain is the obvious one in the given context: How could two agents, one using FMA (or a part of it) as its ontology, the other GALEN, exchange information with ease? We can assume that the agent relying on FMA is the anatomy specialist, called *AA*, while the one based on Galen could be a generalist doctor called *CA*. In these conditions, one possible situation is for example the case when agent *CA* has to make a diagnostic and requires additional, precise, anatomical information regarding the involved area and organs. In order to get the required information he will initiate a dialog with the *AA* agent, inquiring about some anatomical concepts that in his ontology is only superficially described or has missing properties. The two agents will have to correctly identify the concept and start exchanging information related to it.

The mentioned information exchange is taking place in a heterogeneous environment and unless a mapping process of some sort is involved the two agents will not understand each other at all.

The misunderstanding is obviously at concept level since we assume from the start that the two agents have a common language for communicating, similar to, for example, English for two scientists in different domains of research. This common language permits the agents in MAS to interact but keeps the interaction at a simple level, unlike, meaning that it won't allow complex semantics beyond a language with predicates like *ask*(c_1), *explain*(c_2,*tokens*), *acknowledge* and so on. The data wrapped in these predicates is processed by each agent and it receives semantics at that time, thus becoming useful information. The vocabulary can of course be expanded but the actual mapping won't take place at language level.

4 http://sig.biostr.washington.edu/projects/fm/
5 http://www.openclinical.org

III. PROPOSED SOLUTION

One of the main goals is to *map only what is needed and only when is needed*, that is, to follow the guidelines of a lazy mapping approach. In this situation, the agents will communicate without concerning about mapping issues until a mismatch occurs, mismatch that will have to be solved as easily and quickly possible by the two actors, in order for the process to go on. The mapping will take place only when required and for that reason it will not be seen as the main purpose of the communication. As such, the process can go on without the need of involving every concept from the two ontologies, unlike, or creating a global ontology like it is done in.

The mentioned mapping process will be conducted in a divide-et-impera way because of the ample search space we are dealing with. This means that when searching a concept the agent will start top-down first identifying the super-classes of that concept. This search will be supported at the other end by the pair agent that provides mapping information constructed in a bottom-up manner (base concept plus information of its super class plus the super class of the super class and so on). For example, when the *AA* agent sends mapping information for the concept *LeftLung* it will also send information connected to *LobularOrgan*, a super class of *Lung* concept. This way, the agent *CA* will be able to use the combined information from the three concepts to search for the best-match base class. Also, previous mappings that already took place between the two agents will be taken into consideration in the sense that if, for example, *LobularOrgan* has already been mapped, the search will start with it's peer as the base class, and find the best matching concept from there on searching "downwards" through its subconcepts (applying the same rule of pruning for already mapped subconcepts).

An important assumption has been made here, and that is the following: if concept S_{C1} from ontology O_1 is mapped to concept S_{C2} from ontology O_2, and C_1 is a subconcept of S_{C1}, then the peer of C_1 in O_2 will also be a subconcept of S_{C2}. While this is obviously arguable, we think it to be a safe assumption for most ontologies that are referring to the same domain. More formally, this could be expressed as:

$$S_{C1} \equiv S_{C2},\ C_1 < S_{C1} \Rightarrow map(C_1, O_2) < S_{C2} \qquad (1)$$

where $A \equiv B$ means that concept A is mapped to concept B and $A<C$ means that concept A is a subconcept of concept C.

Described until now was only the first level of the mapping process and, as we will see, it will be supported by string metrics. The second mapping stage (used if necessary) will have its roots in the Formal Concept Analysis theory and it will be described in the next sections.

Another main component of our solution is the use of proxy ontologies (already mentioned through the previous example as the concepts already mapped being taken into account). The proxy ontology is constructed on the fly during the communication process and is mainly composed of the already mapped concepts in a conversation with a given agent. Viewed in a simple way it can be compared to a map of concepts, a map that has as keys concepts from one agent's ontology and as values the equivalents from the other's ontology. However, since it is an ontology, it gives us the opportunity to tailor out our own values in this map. That is, concepts resembling the already known ones, but have a slightly different meaning given by the added properties and relations taken from the other agent. This way an agent can, in a way, adapt his concepts to different points of view taken by other agents.

It can be imagined for the previously given example the addition of a new agent that is only interested in possible malformations and diseases of certain organs. The interaction with this agent will probably generate entities that are not part of the standard ontology for *AA* or *CA* from the properties point of view. Nevertheless, the two agents will be able to keep the entities in the proxy ontologies as derived concepts from the base ones as long as the interaction requires this.

A. Similarity Algorithms

A number of similarity algorithms may be used in order to find a match between ontology concepts, varying from probabilistic approaches to linguistic techniques of combining and using the available information in the ontologies.

Similarity algorithms for textual information can vary from the simple Hamming Distance (the number of bits that are different from $string_1$ to $string_2$) or Levenshtein Distance (basic edit distance, given by the minimum number of operation need to transform $string_1$ into $string_2$) to the more complex methods like Jaro Winkler or Soundex that take into account also the position of the character in the string or the probability of occurrence. Most of these metrics can be found in open source libraries like simmetrics[6], and thus easily accessible. Furthermore, since these metrics are widely spread and used with success in different fields there has been also an interest in comparing and analysing them so the path to improved performance should be much easier.

We will choose here only one of this metrics, namely the Jaro-Winkler method but it is obvious that this metric can be exchanged by another with better results at any time. The actual mapping mechanism may use various methods, chosen depending on the context and performance. Further we can use more that one metric by combining results given by different metrics applied on the same textual input but that would cost more time and processing power so an optimal balance should be found. We can even imagine an agent that freely exchanges these metrics depending on the context and on live performance measurements.

The Jaro-Winkler method is an extension of the Jaro metric, given by the following formula:

$$Jaro(s_1, s_2) = \frac{1}{3} * \left(\frac{|s'_1|}{|s_1|} + \frac{|s'_2|}{|s_2|} + \frac{|s'_1| - T_{s'_1, s'_2}}{2 * |s'_1|} \right) \qquad (2)$$

where s_1 and s_2 are the two strings to be compared, s'_1 the common characters in s_1 for s_2, and s'_2 the common characters in s_2 for s_1. Using this Jaro metric, the Jaro-Winkler extension modifies the resulting metric values of poorly matching pairs s_1, s_2 that share a common prefix. So the Jaro-Winkler metric, $JaroWinkler(s_1, s_2)$ is given by:

$$Jaro(s_1, s_2) + (pfxLen(s_1, s_2) * w * (1 - Jaro(s_1, s_2))) \qquad (3)$$

where $pfxLen(s_1, s_2)$ is the length of the common prefix and w is a constant that controls the importance of the

[6] http://sourceforge.net/projects/simmetrics/

common prefix (weight of the prefix).

This string metric used in combination with a similarity threshold will provide the concepts that are candidates for mapping. We can then choose the highest similarity value concept as the result or decide that no mapping was found if none has risen above the threshold. As mentioned in the previous paragraphs, the mapping process will be conducted in a divide-et-impera way by identifying the concept in a top-down manner. Using the information given by the pair-agent, the agent will gradually find the best matching superconcept/concept until the match of the inquired concept has been established. We discriminate between found concepts and superconcepts by comparing the degree of similarity between the pairs (full description, found concept description) and (simple description, found concept description).

This first step in the method proposed here develops on the grounds of textual descriptions found in ontologies. These descriptions attached to the concepts are used in conjunction with string metrics based techniques in order to find related concepts and allow information exchange between agents using heterogeneous ontologies. However, not all existing ontologies have a rich textual description attached to their concepts. In order to surpass this problem, a dictionary can be used to retrieve meaningful text for a given concept and one such dictionary is WordNet[7] . WordNet can be very helpful in this situation since it regards the words as concepts and defines more types of links between them (e.g. synonyms, meronyms, hypernyms etc). These various types of links can be of great help to a linguistic based mapping process although in this paper we will not exploit this advantage to its full extent, using it more like an additional, secondary resource.

Even so, with the available textual information (be it directly from the ontology or with the help of an external dictionary) the mapping process won't capture more semantics than the superficial textual one and so, relations and properties will probably elude the process. However it will be quite fast (at least comparing it to an elaborate method) and will probably cover most of the agent's needs. But since we target also reliability, we will need a component with stronger semantics as well. This is the main reason why our solution is a two step mapping. The first step, just described, will try to map the concepts most of the time. If this first level fails or if more mapping information is required, the second phase mapping will be used: FCA-Mapping.

B. Formal Concept Analysis Mapping

The FCA mapping method proposed has two main parts and the first one is generating the concept lattice from the ontology concept/attribute pairs. This will be done in a straightforward approach as done in most of the FCA applications, meaning that we will take the ontology concepts to be the "formal objects" and their attributes to be the "formal attributes".

However, we will continue to take into account the intial proposed goal, meaning, to map only what is necessary and only when it's necessary. In the FCA context, this means that we will not construct the whole lattice from the given ontology but only a small fraction that will allow us to solve our problem.

As also mentioned previously, the FCA Mapping will only be used if the first mapping step didn't fulfill the agent's needs. This can happen in two major cases: either the concept mapped requires more mapping information (like relations or properties), either the concept's equivalent was not found in the other ontology by the first step of our process. For both cases, the agents will step in the second phase of the mapping by constructing the FCA lattice. If the concept was not found in the other's agent ontology, we will take as starting point the last mapped superconcept and if it's the case of more mapping information we will start with the superconcept of the inquired concept. The starting concept with all its subconcepts defines the boundaries of our formal context since that will be the input for the lattice construction process.

Using the starting formal context and the inquired concept, C_i, the agents will try to construct a lattice that will comprise information about C_i merged into the starting formal context. We can of course, for improved results, to consider not only C_i, but also some other concepts related to it (like the direct superconcept, property and relation concepts as well as sibling concepts). Concept C_i will be merged into the considered context by mapping its properties/attributes to the formal attributes taken as input for the context. The resulting lattice will give us as result either a perfect match for the concept, in the sense that C_i will overlap an existing concept in the other ontology, either an ontological structure containing C_i that will then be stored into the proxy ontology of the agent.

During this context merge process we can also use the information stored in the proxy structures, information generated by previous interaction between the two agents. Concepts found in a relation (super/sub-concept, sibling concept etc.) with C_i that have already been mapped to a concept, part of the starting context, will help integrate C_i in the new lattice structure. These already mapped concepts will drag a set of C_i's ontology attributes in the formal context and thus will have more available information related to C_i.

Let's assume we have concept *SC* from ontology O_2 as starting point and concept C_i, from ontology O_1 that we want to map to something in ontology O_2. The communication between the two agents based on O_1 and O_2 was already started for some time and so we assume that we have some information in the proxy structures. The first step will be to define our initial formal context starting from SC by including all its subconcepts. This will give us two well defined sets: one of the concepts, Cset, and one of the attributes related to them, Aset. The second step of the process will be to populate the two sets with information from Ci's side, ontology O1. This would translate to adding the attributes of Ci to the Aset and of course Ci to Cset. The addition of the new attributes to the Aset involves the attribute mapping process, meaning that we try to find for each new added attribute its equivalent in the already existing set (please note that equivalent may vary from synonym to meronym, depending on how strict we are about the mapping and the desired accuracy). This attribute-mapping can very well make use of WordNet or any other

7 http://wordnet.princeton.edu/

tool, including the previously described similarity algorithms. As mentioned before, if we also take advantage of the existing mapping information, we will include in the Aset also attributes of the already mapped concepts in relation to C_i (and the concepts to C_{set}). With the resulting formal context, we can construct the concept lattice, thus obtaining either an overlapping O_2 concept for C_i, either a structure that would "explain" concept (for example, $C_i < C_1 < SC$ and C_i has sibling C_2 where $C_2 < SC$).

C. Implementation

An interagent communication framework that takes advantage of the proposed mapping methods is under implementation using Java based technologies in order to enable information sharing with ease in a MAS. The building bricks for this application were OAA[8] as a platform for agent intercommunication and Jena for ontology handling. We also used, as mentioned before, the simmetrics library for similarity algorithms and WordNet for lexical processing. The ontologies we worked upon were simple test ontologies created as input for our application as well as the real-life ontologies given as example throughout the paper, namely FMA and GALEN, even if we used only small fractions of them.

Each ontology agent relies on a "hidden" ontology, meaning that the only way to access the information in that ontology from the MAS is through the agent itself. We belief that this approach resembles very much to the OO concept of data encapsulation and the resulting structure will benefit from all its pros. The ontology data is handled with the help of Jena, be it the one from the main ontology or the proxy ontologies. The implemented MAS contains also other types of agents (beside the ontology ones) in order to achieve the desired goal like the dictionary agent (the interface to WordNet) and the query agent (used as an interface to the innerworking of the MAS).

The MAS allows the agents to exchange information using the OAA interagent communication language (namely ICL). As described in this paper, the communication language is used only for sending information packages (and not for semantic mapping) and ICL suits the description even if it allows for something more that this (e.g. service request, triggering events and so on).

A set of simple actions will be provided to the user by the interface agent, actions that would lead to agent interaction, like *diagnosis*(*symptoms*) that can lead to the provided example in this paper. We will also provide a simple way for the MAS user to add its own actions and queries to the system with the help of a simple description language.

IV. RELATED WORK

Ontology mapping is a very hot topic at the moment and multiple lines of work have been followed by different researchers in order to find the best solution to the problem. There are multiple aspects to be taken into account and so, multiple points of view over what is to be considered an adequate solution. Approaches vary from what we called the "whole-ontology" mapping where the only goal is the mapping itself no matter how big the ontologies are or how consuming the process is, to more usability-driven approaches like the ones focused on performance or communication. The classification itself can be driven over different axis depending on the main goal, and without doubt each approach has its own benefits and limitations but we belief that for the real case scenarios, a dynamic method has the upper hand. The "lazy" approach we proposed in this paper is not a new one - definitely not a new one within the computer science field - other mapping solutions adhering to it as well, and for large amounts of data and ample comunities of agents it may be one of the best dynamic methods.

Regarding the actual matching algorithm employed here, the two components have been previously used in this field under various forms. Lexical and vocabulary based methods are quite popular amongst mapping technologies and string similarity methods have been used with success over the years when it comes to aligning and measuring data. However, our work was not targeted at developing a new metric of this kind, nor did it had the lexical component as main part unlike. We used the vocabulary only as mean of transferring raw information, information to be processed by the agents and the string metrics as an abstract, replaceable component capable of doing the right task in the right environment.

The other main component, Formal Concept Analysis, has been now for a while mixed in the computer science field and for that matter, it has strong supporters for its benefits, but not until recently researchers in the ontology mapping sub-domain became interested in its potential. At the moment there are applications both theoretical and practical[9] that are using FCA and this shows without doubt that FCA has some leverage in the knowledge related fields. However, we belief that FCA has much more potential in this area as an ontology tool and FCA similarity and mapping methods will be followed by other uses as well, like validation and enhancement.

V. CONCLUSION AND FUTURE WORK

As a final note on this work we will like to emphasize the importance of finding a solution for the mapping problems in MAS using real ontologies.

The FMA and Galen ontologies were suited for this task because only when using such enormous ontologies one can appreciate the great advantages provided by a lazy mapping method as the one described in our paper. The whole-ontology mapping methods are sure to fail in a dynamic environment that uses more than one large ontology and where agent (and thus ontology) interactions grow at an exponential rate. Our method, besides following the lazy approach guide, tries to improve performance even more by using the proxy ontologies constructs, which allow the agents to store the results of a mapping in order to use it in a future interaction with the same agent. At the same time, the proxy constructs allow the agent also to keep its main ontology immutable and permit communication with more than one peer agent simultaneous since the communication threads are independent.

The first level of mapping described in our solution is also adding a plus of quickness to the process by using

8 http://www.ai.sri.com/oaa/

9 http://sourceforge.net/projects/conexp

string metrics combined with a divide-et-impera driven approach. The string metric component was used as an abstract entity, without relying on the benefits of one or another metric, and this translates into an interchangeable metric. One future line of work from this point of view will be to adapt the used metric to the context, allowing the agent to use the mapping tool that best suits the processed ontology.

We also aim at providing a reliable mapping and so, for this goal, the proposed solution has the FCA component that might prove slower that the first level of mapping - textual one - but it brings more semantics into the process. As noted before in this paper, the FCA is a somehow new addition to the ontology mapping field but it is quite powerful when handling knowledge and we belief that it is a precious tool in this context. We intend in our future work to extend the use of FCA also at validating the result of a mapping and the ontology itself.

Another important line of work that we will follow throughout our research is of course adding more semantics to the mapping process. The FCA is a step ahead in this direction but we would like to enhance also the first level of mapping, the lexical one, with more power in this matter. WordNet or other similar repositories and dictionaries are without doubt of great help for this task and we will try to further explore their use not only for concept description but also for attributes and relations.

REFERENCES

[1] M. Costin and I. A. Letia, “Dynamic ontology mapping using proxy based channels,” in IEEE 2nd International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, vol. Poster, pp. 11 – 16, September 2006.

[2] M. Donnelly, T. Bittner, and C. Rosse, “A formal theory for spatial representation and reasoning in biomedical ontologies.” Artif Intell Med, vol. 36, no. 1, pp. 1–27, January 2006.

[3] J. van Diggelen, R. J. Beun, F. Dignum, R. M. van Eijk, and J.-J. Meyer, “Optimal communications vocabularies and heterogeneous ontologies,” in Agent Communication. International Workshop on Agent Communication. Revised Selected and Invited papers Vol. 3396.1em plus 0.5em minus 0.4emBerlin: Springer Verlag, pp. 76–90, 2005

[4] J. van Diggelen, R. J. Beun, F. Dignum, R. van Eijk, and J.-J. Meyer, “Anemone: An effective minimal ontology negotiation environment,” in Proceedings of the 5th International Conference on Autonomous Agents and Multi-Agent Systems, Hakodate Japan, 2006.

[5] R. Agrawal and R. Srikant, “On integrating catalogs,” International WWW Conference, Hong-Kong, 2001.

[6] M. Schorlemmer and Y. Kalfoglou, “Progressive ontology alignment for meaning coordination: An information-theoretic foundation,” in Proceedings of the 4th International Conference on Autonomous Agents and Multi-Agent Systems, Utrecht, Holland, pp. 737–744, July 2005.

[7] W. Cohen, P. Ravikumar, and S. Fienberg, “A comparison of string distance metrics for name-matching tasks,” in Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, 2003.

[8] G. Stumme and A. Maedche, “FCA-MERGE: Bottom-up merging of ontologies,” in Proceedings of IJCAI, pp. 225–234, 2001.

[9] Z. Ding, Y. Peng, R. Pan, and Y. Yu, “A bayesian methodology towards automatic ontology mapping,” in AAAI Workshop on Contexts and Ontologies, July 2005.

[10] Y. Sure and M. Ehrig, “Ontology mapping - an integrated approach,” in Proceedings of the First European Semantic Web Symposium, volume 3053 of Lecture Notes in Computer Science, C. Bussler, J. Davis, D. Fensel, and R. Studer, Eds., pp. 76–91, 2004.

[11] M. Ehrig and S. Staab, “QOM - quick ontology mapping,” The 3rd International Semantic Web Conference, Hiroshima, Japan, 2004.

[12] P. Doherty, W. Lukaszewicz, and A. Szalas, “Approximative query techniques for agents with heterogeneous ontologies and perceptive capabilities,” in Proceedings of 9th International Conference on the Principles of Knowledge Representation and Reasoning, D. Dubois, C. Welty, and M.-A. Williams, Eds.1em plus 0.5em minus 0.4emAAAI Press, 2004, pp. 459–468.

[13] M. Afsharchi, B. H.Far, and J. Denzinger, “Ontology-guided learning to improve communication between groups of agents,” in Proceedings of the 5th International Conference on Autonomous Agents and Multi-Agent Systems, Hakodate Japan, 2006.

[14] G. Stoilos, G. Stamou, and S. Kollias, “A string metric for ontology alignment,” in Proceedings of the 4th International Semantic Web Conference, Galway, 2005.

[15] W. R. van Hage, S. Katrenko, and G. Schreiber, “A method to combine linguistic ontology-mapping techniques,” in Proceedings of the 4th International Semantic Web Conference, Galway, 2005.

[16] U. Priss, “Formal concept analysis in information science,” in Annual Review of Information Science and Technology, pp. 521–543, 2006.

[17] Y. Zhao, X. Wang, and W. Halang, “Ontology mapping based on rough formal concept analysis,” in Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services, 2006.

[18] L. Fan and T. Xiao, “FCA-Mapping: A method for ontology mapping,” in Proceedings of the European Semantic Web Conference, 2007.

[19] U. Priss, “Linguistic applications of formal concept analysis,” in Formal Concept Analysis, pp. 149–160, 2005.

[20] M. Obitko, V. Snel, and J. Smid, “Ontology design with formal concept analysis,” in Proceedings of the CLA 2004 International Workshop on Concept Lattices and their Applications, Czech Republic.

[21] A. Formica, “Concept similarity in formal concept analysis: An information content approach,” in Knowledge Based Systems, 2007.

[22] A. Heß, “An iterative algorithm for ontology mapping capable of using training data,” in Proceedings of the 3rd European Semantic Web Conference, 2006.

[23] H. Alani and C. Brewster, “Metrics for ranking ontologies,” in Proceedings of the 4th Int. EON Workshop, 15th Int. World Wide Web Conf., Edinburgh, 2006.

[24] H. Alani, C. Brewster, and N. Shadbolt, “Ranking ontologies with AKTiveRank,” in Proceedings of the 5th International Semantic Web Conference, Athens, GA, USA, November 5-9, 2006.

[25] K. E. Wolff, “A first course in formal concept analysis - how to understand line diagrams.” in F. Faulbaum, editor, Advances in Statistical Software, Vol. 4, pp. 429–438, 1994.

[26] D. Lin, “An information-theoretic definition of similarity,” in Proceedings of the International Conference on Machine Learning, Morgan Kaufmann, Madison, Wisconsin, USA, 1998.