ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-871

Attuning Ontology Alignments to Semantically Heterogeneous Multi-Agent Interactions

Paula Chocron[1,2] and **Marco Schorlemmer**[1]

Abstract.
In this paper we tackle the problem of semantic heterogeneity in multi-agent communication, i.e., when agents in a multi-agent system use different vocabularies for message passing, or might interpret shared vocabulary in varying ways. The problem of achieving meaningful communication in such semantically heterogeneous multi-agent interactions has been mainly tackled either by using ontology alignments to translate vocabularies, or by using methods that learn an alignment by observing how the utterance of particular terms affects the unfolding of an interaction. We propose solutions that combine these approaches and study how agents can use external alignments with possibly incomplete or erroneous mappings when communicating with each other in the context of a multi-agent interaction. We further show experimentally that with the experience gained through repeated interactions and by using simple learning techniques agents can find and repair those mappings of an ontology alignment that lead to unsuccessful interactions, thus improving the success rate of their future interactions.

1 INTRODUCTION

An important problem in the design and implementation of distributed systems is to guarantee a sufficiently good level of interoperability between separately engineered software components as for the whole system to function adequately with respect to the expected functionality it is supposed to deliver. A particularly critical problem is that of semantic heterogeneity, i.e., when the terms used in the exchange of information between system components are interpreted differently by each of them [14]. This problem has triggered a significant amount of research in the fields of databases, the semantic web, or multi-agent systems [9, 11, 8]. In this paper we will focus on the problem of semantic heterogeneity in multi-agent communication, having in mind open multi-agent systems for which an interaction model or protocol is specified, but whose agents might have differences in the vocabulary they use for message passing, or might interpret shared vocabulary in varying ways.

When faced with multi-agent systems whose agents use different vocabularies in their communicative acts, the immediate and most common solution is to resort to some semantic alignment technique, maybe supplemented with some process of vocabulary or alignment negotiation, so as for agents to determine how foreign vocabulary needs to be interpreted using the local one [11, 16, 22, 17]. The success of this solution is obviously very dependent on the quality of the alignments that can be computed, which in turn is very much conditioned by the detail in which vocabularies are specified and to the external semantic resources that might be available. For ontologies with a rich taxonomic structure and with detailed axiomatic specifications of the intended meaning of entities and relations, one can take advantage of state-of-the-art ontology matching tools that exploit many different kind of techniques, going from simple syntactic matching all the way to formal logical reasoning [11]. But when faced with underspecified vocabularies, the alignments obtained may prove to be insufficient for achieving the semantic interoperability required for a multi-agent interaction to be successful.

An alternative approach that does not depend on any ontology, semantic alignment tool or external semantic resource, was described in [3], in which agents gradually learn from the experience of repeated interactions those mappings of their vocabularies that lead to successful multi-agent communication. In that approach, meaning is assumed to be only determined by the interaction context, and it is never explicitly communicated. Unfortunately, the convergence to a common vocabulary using this approach can be very slow, i.e. many repetitions of the same interaction need to be enacted to get reasonable expectations of success, since no other source of meaning besides the interaction is taken into account.

Consequently, it seems reasonable to attempt to improve upon these two complementary approaches by combining them, developing novel semantic alignment techniques for multi-agent communication that can take advantage from the strengths of both approaches. Such techniques would exploit the availability of the ontological knowledge associated to a vocabulary and the powerful ontology matching techniques that make use of it, but would also take into account the experience that agents accumulated of the actual use of their vocabulary in the concrete contexts of particular interactions and the outcomes of these.

In this paper we set out to show how agents that are to perform a task together as specified in an interaction model or protocol can learn dynamically a translation that is useful for their interaction, with the help of semantic alignments computed by some external ontology matcher. While relying on these semantic alignments can provide valuable information that eases the convergence to a meaningful translation, external alignments can also contain errors or be inadequate for the particular interaction context in which the agents are using them, something that could hinder successful communication.

We first propose a straightforward way of combining a previously computed, external ontology alignment between separate vocabularies with the alignment technique that learns the semantic relationship between vocabularies from the experience gained by repeated multi-agent interaction. We than move on to describe a more powerful technique that uses reinforcement learning techniques to han-

[1] Artificial Intelligence Research Institute (IIIA-CSIC), Bellaterra (Barcelona), Catalonia, Spain. e-mail: pchocron, marco@iiia.csic.es
[2] Universitat Autònoma de Barcelona, Bellaterra (Barcelona), Catalonia, Spain

dle low quality mappings of the external ontology alignment in the context of the interaction, and thus to improve the ratio of success in repeated interactions, by taking into account previous experience with these low quality mappings.

We compare experimentally the proposed techniques, showing that they both solve the drawbacks of using only one source of meaning (only ontology alignment or only interaction experience) and improve the success rate of multi-agent interactions. Our methods are independent of the how vocabularies are ontologically structured and of the internal structure of each agent. In addition, each agent computes its own alignment based on its own interaction experience, so that no shared framework for meaning negotiation is required.

2 ALIGNMENTS VS. INTERACTION EXPERIENCE

In this paper we focus on interactions between two agents a_1 and a_2, who might use words from different vocabularies to communicate. A vocabulary is the finite set of words an agent is allowed to use in its messages, and we will write V_1 and V_2 for the vocabularies of our two agents. Each agent can organise its vocabulary in its own way, with additional structure that makes it a taxonomy of words, or even a fully fledged ontology specifying the intended meaning of the words of the vocabulary. In our work, however, we only need to assume that a vocabulary V is provided with some *similarity measure* $sim : V \times V \to [0, 1]$ between its words.[3]

We have mentioned two main approaches to tackling semantic heterogeneity in agent communication: those that rely on external alignments and those that learn from repeated interaction. In this section we will explain both of these techniques, formulating them as solutions to the problem of choosing how to interpret a foreign word in a received message. We focus on the situation in which agent a_1 receives a message with a word v_2 sent by agent a_2 when it was actually waiting for one from some known set U of expected words to receive. The agent therefore needs to choose a word $v_1 \in U$ that matches with the received one v_2, in such a way that the interaction proceeds correctly. We will explain each of the two approaches as a technique to compute a probability distribution for each $v_1 \in U$ according to which agent a_1 can choose a possible match for v_2.

2.1 Using External Vocabulary Alignments

One approach to achieving mutual understanding between agents that use different vocabularies is to use an external alignment, taking advantage of the multiple matching techniques that have been developed in the last decades. These techniques vary from sophisticated OWL ontologies matchers to syntactic similarity measures, and the choice between these possibilities will depend on the additional structure in the vocabularies of agents, the availability of the matching tools, and the access to the vocabulary and structure of the agents's interlocutors.

Definition 1 *An* alignment $\mathcal{A}$ *between two vocabularies* V_1 *and* V_2 *is a finite set of* mappings *between words in* V_1 *and* V_2*. A mapping is defined as a quadruple* $\langle v_1, v_2, n, r\rangle$*, where* $v_1 \in V_1$*,* $v_2 \in V_2$*,* $n \in (0, 1]$ *is the degree of confidence on the mapping, and* r *is the kind of relation that holds between words. An alignment contains at most one tuple for each pair of words* $\langle v_1, v_2\rangle$*. [5]*

[3] Adequate similarity measures will depend on the additional structure given to the vocabulary. For taxonomies, for instance, a choice could be the Wu-Palmer measure [24]. Even if no similarity measure is provided, we can always resort to the trivial one that assigns 1 to the identity and 0 otherwise.

In this work, we will consider alignments with only equivalence ($\equiv$) as the relation holding between words in the mappings. Given an alignment $\mathcal{A}$, if a mapping $\langle v_1, v_2, n, \equiv\rangle$ belongs to $\mathcal{A}$ we will write $v_1 \equiv v_2$ and denote its confidence with $conf(v_1 \equiv v_2)$.

The quality of vocabulary alignments is typically measured in comparison with a *reference alignment*, for which values of *precision* and *recall* are computed. As it is commonly done, we do not take into account the confidence degrees in these measures.

Definition 2 *Given an alignment* $\mathcal{A}$*, let* $\mathcal{A}'$ *denote the set of mappings of* $\mathcal{A}$ *for which we have removed the confidence degree, i.e.,* $\mathcal{A}' = \{\langle v_1, v_2, r\rangle \mid \langle v_1, v_2, n, r\rangle \in \mathcal{A} \text{ for some } n\}$*. The* precision *of an alignment* $\mathcal{A}$ *with respect to a reference alignment* $\mathcal{B}$ *is the fraction of the mappings in* $\mathcal{A}'$ *that are also in* $\mathcal{B}'$*:*

$$precision(\mathcal{A}, \mathcal{B}) = \frac{\mid \mathcal{A}' \cap \mathcal{B}' \mid}{\mid \mathcal{A}' \mid}$$

while its recall *is the fraction of the mappings in* $\mathcal{B}$ *that were found by* $\mathcal{A}$*:*

$$recall(\mathcal{A}, \mathcal{B}) = \frac{\mid \mathcal{A}' \cap \mathcal{B}' \mid}{\mid \mathcal{B}' \mid}$$

The most straightforward approach to use an alignment to tackle the problem of semantic heterogeneity in agent communication is to use it directly for translating words in messages. If an agent receives the word v_2 while waiting for words in U, it will choose the word that matches with v_2 with highest confidence in the alignment; if there is no such word, it chooses one randomly. This approach will work well if the alignment is adequate for the task the agents are performing; however, low recall will mean more random choices, which can cause unsuccessful interaction, while low precision implies a higher probability of choosing incorrect matches, which can also cause an interaction to fail.

One way of mitigating the effect of low recall is to consider not only the mappings that are explicitly present in the alignment, but to take into account the additional structure vocabularies may have. This can be achieved by using the similarity measure between words of one vocabulary in order to choose a word that is close to a match. Consider a similarity threshold $\theta \in [0, 1]$. For each $v_1 \in U$, let $V_{\equiv v_2}(v_1)$ be the set of words that match with v_2 in the alignment and that are closer than θ to v_1:

$$V_{\equiv v_2}(v_1) = \{v_1' \in V_1 \mid v_1' \equiv v_2 \text{ and } sim(v_1, v_1') \geq \theta\}$$

To compute a probability distribution over the interpretations, we first assign a value to each possibility. In this case, the value of interpreting v_2 as v_1 is given by

$$\mathcal{V}_{alg}(v_1, v_2) = \begin{cases} \max_{v_1'} conf(v_1' \equiv v_2) sim(v_1, v_1') & \text{if } V_{\equiv v_2}(v_1) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

where $v_1' \in V_{\equiv v_2}(v_1)$.

To reduce the effects of low precision, a solution is to not trust the alignment completely, including an exploration parameter ξ_1 in the definition of a probability distribution over words in U.

Alignment Criterion (`alg`).
Let $\hat{\mathcal{V}}_{alg}(v_1, v_2)$ be the normalised value of $\mathcal{V}_{alg}(v_1, v_2)$ for each $v_1 \in U$. Choose $v_1 \in U$ with probability:

$$p_{alg}(v_1) = \xi_1 \times \hat{\mathcal{V}}_{alg}(v_1, v_2) + (1 - \xi_1) \frac{1}{\mid U \mid}$$

The exploration parameter ξ_1 introduces a well known dilemma. Setting it to large values result in a very weak method when there are wrong mappings in the alignment, while with low values we can be losing useful information. Ideally, the parameter should depend on the precision of the alignment $\mathcal{A}$, but this is something agents are not expected to know in advance.

2.1.1 The Alignment Criterion in Action: A Running Example

In what follows we introduce an illustrative example using a simple travel agency scenario adapted from [2]. The complete vocabularies and specifications of the ontologies used by the agents can be found in [2]; we do not need them explicitly here.

Consider a Travel Agent (TA) offering two services: to book a flight for a given date and destination, or to provide information about the available hotels in a city. TA uses its own vocabulary, which is not necessarily shared by its clients; we consider a particular Customer (C) who uses a different language. To be able to interact with C, agent TA may use an ontology alignment provided by some external source. Table 1 shows a relevant fragment of the alignment provided by the matcher Falcon-AO [15] as reported in [2]. Consider a situation in which TA is waiting for agent C to specify if it wants a return flight or not, so $U = \{OneWay, RoundTrip\}$ at this state. Using the alignment criteria and the simple $0-1$ similarity measure, if C sends $\{Single\}$, the travel agent will interpret it as $\{RoundTrip\}$.

$v_1 \in V_1$	$v_2 \in V_2$	Confidence
Return	Package	0.41
Single	RoundTrip	0.19
UnregCustomer	OneWay	0.03
Flight	Customer	0.01
destination	airlineCompany	0.99
carrier	to	0.99
departing	leavingDate	0.99
origin	from	0.99
returning	returnDate	0.76
hotelBookingsIn	city	0.30

Table 1: Extract of the alignment provided by Falcon-AO as reported by Atencia in [2]

2.2 Learning from Interaction Experience

A second approach to communicating with linguistically heterogeneous partners does not use any external resource, but instead considers meaning to be determined by the specification of interactions. Agents that perform the same task repeatedly interacting with the same partners can learn the meaning of words by simply observing the outcomes for different possibilities and choosing again the ones that gave good results. This technique was developed in [3], and in this section we reformulate it as a solution to a learning problem, which will be useful to introduce, in Section 3, the novel methods we propose.

In this approach, each agent has its own specification of the interaction it takes part in. In the last decades, the multi-agent community has discussed thoroughly the question of how to model agent interactions and communication languages, and many different techniques have been proposed [23]. In this work we will abstract these techniques to consider only a very simple message exchange mechanism, modelled as a finite-state automaton in which state transitions are triggered by messages. To represent different outcomes of the interaction that all agents can recognise, we define a set of predicates called *state properties* to characterise final states.

Definition 3 *Given two agents a_1 and a_2, a vocabulary V and a set of state properties SP, an* interaction model *IM is defined as a tuple $\langle Q, q_0, \delta, F, \rho, speaks\rangle$ where Q is a finite set of states, $q_0 \in Q$ is the initial state, $F \subseteq Q$ is the set of final states, $\rho : F \to \mathcal{P}(SP)$ is a function assigning a subset of state properties to each final state, and $speaks : Q \to \{a_1, a_2\}$ is a function assigning to each state its sender agent, and $\delta : Q \times V \to Q$ is a partial function called the* transition function.

Note that while we do not specify any particular turn-taking pattern, we do require that, for each state, all messages labelling transitions from this state share the same sender agent, who is determined with the *speaks* function.

While we assume that the language to specify state properties in SP is shared, agents may use different vocabularies for the messages they send to each other. Consequently we will denote with IM_1 the interaction model followed by agent a_1 using vocabulary V_1, and with IM_2 the one followed by agent a_2 using vocabulary V_2.

When the interaction is in a state $q \in Q$ for which $speaks(q) = a_1$, agent a_1 chooses a word from V_1 to utter according to IM_1. If instead $speaks(q) = a_2$, a_1 will wait to receive a message from a_2. Since the received word is from V_2, it will need to interpret it in the context of that particular interaction state, following a transition according to IM_1. That is, it will choose a word from the set of expected words for state q, given by $U(q) = \{v \in V_1 \mid \delta(q, v) \text{ is defined}\}$. Since a_2 does the same, an interaction between two agents can be defined as a sequence of uttered messages along with how they were interpreted. A successful interaction is one that leads both agents to final states with the same state properties.

For the following definitions it will be useful to restrict interaction models to deterministic ones, and to recall that in this case an accepted string can be associated with only one sequence of states that are visited to produce it.

Definition 4 *A* successful interaction *between interaction models IM_1, IM_2 is a finite sequence of pairs of words $\langle v_1, v_2\rangle$ with $v_1 \in V_1$ and $v_2 \in V_2$ such that the projection of its first coordinates is a string accepted by IM_1, the projection of its second coordinates is a string accepted by IM_2, and both projections visit, in their respective interaction model, sequences of states with the same senders, reaching final states with the same state properties.*

Successful interactions lead to an intuitive notion of alignment between two interaction models, which is composed of all tuples $\langle v_1, v_2\rangle$ that belong to successful interactions. However, this alignment could have different interpretations for the same word, corresponding to different states. Agents will be interested, more specifically, in finding which interpretation they need to choose according to each state in order to interact successfully:

Definition 5 *Let int be a successful interaction between IM_1 and IM_2, and let $states$ be the sequence of states visited in IM_1 determined by the projection of the first coordinates of int. Then all tuples $\langle q, v_1, v_2\rangle$ obtained by adding the ordered items of $states$ to the pairs in int belong to the* pragmatic alignment *from interaction model IM_1 to interaction model IM_2.*

Notice, first, that pragmatic alignments are defined from one interaction model to another one, and second, that unlike those in Defini-

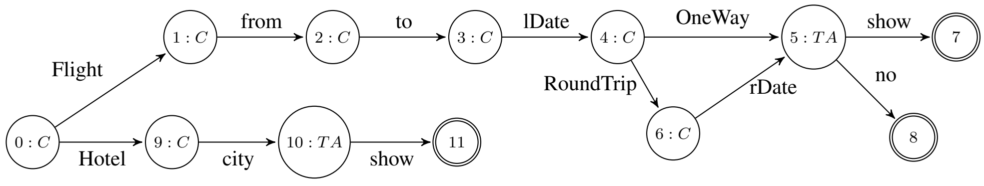


Figure 1: Interaction Model IM_{TA} for the Travel Agent

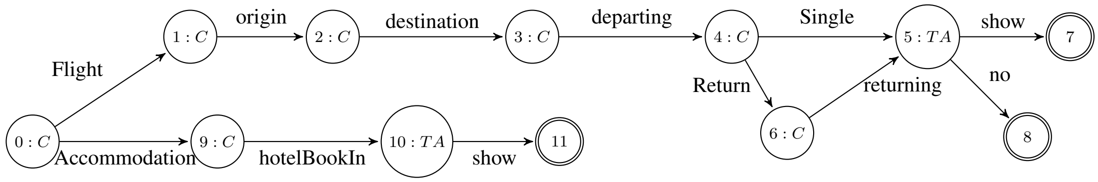


Figure 2: Interaction Model IM_C for the Customer

tion 1, mappings are parametrised by states. We will write $v_1 \simeq_q v_2$ if $\langle q, v_1, v_2 \rangle$ belongs to the pragmatic alignment from IM_1 to IM_2.

Definition 6 *Two interaction models IM_1 and IM_2 are* structurally equivalent *if all strings accepted by them separately are projections of a successful interaction between them.*

Interaction models are structurally equivalent if they are equivalent modulo interpretation of words in messages, or in other words, interactions can always finish successfully if the correct mappings are chosen. In this work we will assume that IM_1 and IM_2 are structurally equivalent. While the methods we propose can be easily adapted to interaction models that have minor differences, they will not perform well when such differences are substantial; how to adapt heterogeneous protocols is a difficult problem that is out of the scope of this paper.

To interact successfully with each other, agents need to discover the pragmatic alignment between their interaction models. The method proposed in [3] that we explain in this section, as well as the novel ones presented in the next one, are techniques to let agents learn these pragmatic mappings automatically from repeated interaction. We will formulate these solutions using standard concepts and notation from Reinforcement Learning (see, e.g., [20]). As usual, we start by defining the learning model.

Since a_1 needs to learn which interpretation is good for a received word in a specific state, the states of the learning model will be pairs $\langle q, v_2 \rangle$, where $q \in Q$, $speaks(q) = a_2$, and v_2 is a word received. In that situation, a_1 can choose how to interpret v_2 from a the set of expected messages, therefore the set of actions for a state $\{q, v_2\}$ of the learning model are the words in $U(q)$. Let us make two remarks. First, we are abstracting the interaction states in which a_1 speaks, because it does not need to learn any interpretation in those, and the pragmatic alignment will be independent of the messages it utters. Second, the agents do not know the learning model a priori, since they ignore which messages their interlocutor can utter. We will use methods that do not require agents to know the model. Our objective is to estimate action values $\mathcal{V}(v_1 \simeq_q v_2)$, which represent the confidence in that v_2 should be interpreted as v_1 in q. In this section we present a simple solution: values for all mappings in an interaction are updated when the interaction ends, adding 1 if it succeeded, or 0 if it failed.

In each interaction they take part in, agents will keep a record of the mapped pairs as a sequence $(q_0, v_{10}, v_{20}), \ldots, (q_n, v_{1n}, v_{2n})$. When the interaction ends, the values of mapped pairs are updated in the following way.

$$\mathcal{V}_{exp}(v_1 \simeq_q v_2) = \begin{cases} \mathcal{V}_{exp}(v_1 \simeq_q v_2) + 1 & \text{if the interaction succeeded} \\ \mathcal{V}_{exp}(v_1 \simeq_q v_2) & \text{if the interaction failed} \end{cases}$$

Let $\#exp(q, v_2) = \sum_{v_1{}' \in U(q)} \mathcal{V}_{exp}(v_1' \simeq_q v_2)$, and consider an exploration parameter ξ_2 close to 1. The interpretation can be chosen according to the following criterion:

Experience Criterion (`exp`).

Let $e(q, v_1, v_2) = \dfrac{\mathcal{V}_{exp}(v_1 \simeq_q v_2)}{\#exp(q, v_2)}$. Choose $v \in U(q)$ with probability:

$$p_{exp}(v_1 \simeq_q v_2) = \begin{cases} \xi_2\, e(q, v_1, v_2) + (1 - \xi_2)\frac{1}{|U(q)|} & \text{if } \#exp(q, v_2) > 0 \\ \dfrac{1}{\mid U(q) \mid} & \text{if } \#exp(q, v_2) = 0 \end{cases}$$

This method divides the matching decisions in two phases. While there is not enough information from the experience, the agent maps randomly; once interactions start to be successful, it repeats good choices.

Let us make two remarks. First, the exploration parameter ξ_2 is included to consider situations in which the configuration of the interaction protocols can make two mappings be correct in one state. Second, since the outcome of the interaction is only known once it finished, this mechanism is affected by the *credit assignment problem*: learning from unsuccessful interactions is difficult, because it is not known which of the mappings was wrong.

2.2.1 The Experience Criterion in Action

Consider again the Travel Agency example introduced in Section 2.1.1, but now suppose Travel Agent TA has specified the tasks it can perform with the interaction model IM_{TA} in Figure 1. The letter in each state represent its speaker agent, while transitions are only labeled with the content of messages. Consider $SP = \{success, failure, book, info\}$ and the following state property function: $\rho(7) = \{success, book\}$, $\rho(8) = \{failure, book\}$, $\rho(11) = \{success, info\}$.

The Customer Agent C, from its side, follows interaction model IM_C which is structurally equivalent to IM_{TA}. An example of a successful interaction between these interaction models is given by the sequence $\langle Hotel, Accommodation\rangle, \langle hotelBookIn, city\rangle, \langle show, show\rangle$. This implies, for example, that in the pragmatic alignment from IM_{TA} to IM_C, $city \simeq_9 hotelBookIn$.

Using criterion `exp`, the TA will first go through a learning phase, in which the interactions will be mostly unsuccessful since it is choosing mappings randomly. However, since the interaction has few interpretations choices and each of them with few options, it should not take long to find correct mappings.

3 COMBINING ALIGNMENTS WITH INTERACTION EXPERIENCE

In this section we propose methods than combine an external source of meaning with the interaction context. We consider again a_1 interacting repeatedly with a_2; now, in addition, a_1 has access to an external alignment $\mathcal{A}$ between V_1 and V_2.

A central concern when using $\mathcal{A}$ is that, since it was produced by an external resource, it does not necessarily agree with the pragmatic alignment between IM_1 and IM_2. This raises the question of how vocabulary alignments relate to pragmatic ones; taking into account that the last ones are parametrised by states. A straightforward definition of the precision and recall measures when compared to two interaction models considers as reference alignment all pairs in all successful interactions, considering correct all the mappings that are useful when interacting.

There is a situation in which a mapping in an external alignment results particularly harmful for the interaction. The problem arises when a mapping $v_1 \equiv v_2$ belongs to $\mathcal{A}$ and $v_1 \in U(q)$, but $v_1' \simeq_q v_2$ does not belong to the pragmatic alignment between the interaction models of both agents. In the travel agency example, this happens in state 5, because $RoundTrip \equiv Single$ and $RoundTrip \in U(5)$, but $RoundTrip \not\simeq_5 Single$ because it does not lead to any successful interaction. When the alignment is followed, most of the times $RoundTrip$ will be chosen as an interpretation for $Single$, causing the interaction to fail. We will refer to this kind of mappings as *misleading*. A misleading mapping can be *repaired* by making its value lower than other possibilities, so that it is not chosen anymore.

The following is a very straightforward combination of the alignment with the interaction experience:

Alignment and Experience Criterion (`alg-exp`).

Choose $v_1 \in U(q)$ with probability:

$$p_{alg-exp}(v_1 \simeq_q v_2) = \begin{cases} p_{exp}(v_1 \simeq_q v_2) & \text{if } \#exp(q, v_2) > 0 \\ p_{alg}(v_1 \simeq_q v_2) & \text{if } \#exp(q, v_2) = 0 \end{cases}$$ [4]

This method affects only the exploratory part of the learning in the `exp` criterion; successful interactions are taken into account in the same way. This is because mappings that lead to successful experiences belong to the pragmatic alignment by definition. This straightforward combination has two drawbacks. First, it still considers only the successful matches and discards all the information in the ones that failed. Second, the dilemma of when to choose randomly instead of following the alignment that we explained in the Alignment Criterion `alg` is not solved.

3.1 Learning from Unsuccessful Experiences

We now present a more elaborate method that is able to repair misleading mappings and to find missing ones more efficiently. This is achieved by updating the original confidences in the alignment with the experience of unsuccessful interactions, combining the following two ideas to deal with low quality alignments. First, to mitigate low precision, mappings involved in unsuccessful interactions are punished. Second, to mitigate low recall, the confidence in a mapping is updated taking into account the quality of the alignment possibilities that were found subsequenty. The intuition behind this second idea is that, if good mappings were found after a particular choice of interpretation, it is likely that it was correct. Consider a simple analogy with human conversations: if someone is not sure of having understood a message, but the dialogue continues as expected, she will assume her understanding was correct, whereas if stranger messages arrive, her confidence will decrease.

Our method, again, divides the learning in two phases. The difference with the already presented criteria is that, in the first phase, agents also compute a distribution over the possible interpretations, representing their confidence in a mapping belonging to the pragmatic alignment. This value depends depend on the original confidence (given by the external alignment) and on the observations of what happened when the mapping was chosen.

In this first phase, we use a method that resembles classical temporal difference reinforcement learning techniques, but instead of computing an expected reward, we will update a value representing the confidence in that a mapping belongs to the pragmatic alignment. This confidence starts being the value given in the alignment, and it evolves with the experience. Agents will try to minimize the punishment, which is assigned to the last mapping when an interaction fails; in this way they explore different possibilities.

Let $\alpha \in (0, 1]$ be a *forgetting parameter*, and $C \in (0, 1]$ a *punishment*. As initial values, we use the confidences in the alignment:

$$\mathcal{V}_{ev}(v_1 \simeq_q v_2) = conf(v_1 \equiv v_2)$$

When an interaction finishes in failure, agents will have a sequence of the states in which they made mapping decisions like in the `exp` method. For each of these states, each agent will update $\mathcal{V}_{ev}(v_1 \simeq_q v_2)$ as follows:

- If $i = n$, a punishment of $-C$ is assigned for having failed:

$$\mathcal{V}_{ev}(v_{1n} \simeq_{q_n} v_{2n}) = (1 - \alpha)\mathcal{V}_{ev}(v_{1n} \simeq_{q_n} v_{2n}) + \alpha(-C)$$

- For $i < n$, each agent takes into account the mapping possibilities in future states:

$$\begin{aligned} \mathcal{V}_{ev}(v_{1i} \simeq_{q_i} v_{2i}) =& (1 - \alpha)\mathcal{V}_{ev}(v_{1i} \simeq_{q_i} v_{2i}) + \\ & \alpha \max_v \mathcal{V}_{ev}(v_{1i+1} \simeq_{q_{i+1}} v_{2i+1}) \end{aligned}$$

  where $v \in U(q_{i+1})$.

We do not need to allow explicitly for exploration, since the backpropagation of the punishment already has that effect.

Evolving Alignment and Experience Criterion (`ev-alg-exp`).

[4] It would be reasonable to use $p_{alg}(v_1 \simeq_q v_2)$ in the exploration of `exp` instead of choosing randomly, we do not add it for clarity. The same holds for the next criterion.

Let $max = \text{argmax}_{v_1' \in U(q)}(\mathcal{V}_{ev}(v_1' \simeq_q v_2))$. Choose $v_1 \in U(q)$ with probability:

$$p_{ev}(v_1 \simeq_q v_2) = \begin{cases} p_{exp}(v_1 \simeq_q v_2) & \text{if } \#exp(q, v_2) > 0 \\ \frac{1}{|max|} & \text{if } \#exp(q, v_2) = 0,\ v_1 \in max \\ 0 & \text{if } n = 0,\ v_1 \notin max \end{cases}$$

Note that, since values are updated when the interaction is over, the new maximum value for future states can be used. This will back-propagate the punishment to all mappings already in the first unsuccessful interaction, repairing misleading mappings in less interactions, although it is less stable. This is the approach we use in the experimentation.

Analysis of `ev-alg-exp`: repairing misleading mappings. As we will show, the `ev-alg-exp` technique succeeds to find misleading mappings for most configurations; however, there is one particular case in which it does not work well. The pathological case arises when, for an interaction model IM and an alignment $\mathcal{A}$, the following conditions hold: 1. Two strings s and s' accepted by IM have the same word v in the j-th position, 2. $\mathcal{A}$ has a misleading mapping for a word in s before j, and 3. If q is the state for v in s', $\mathcal{A}$ has a correct mapping $v \simeq_q w$, but it also has a misleading mapping $v' \simeq_q w$ [5]. In this case, the technique the $v \simeq_q w$ mapping the first time s' is successful, thus always choosing it from there on, and not being able to decrease the value of $v' \simeq_q w$. There are possible fixes to this problem, but since it is a rare case, we choose to resort in the exploration from `exp` to repair it.

If the case above does not hold, and all messages in the protocol have some probability of being uttered, `ev-alg-exp` always repairs misleading mappings. This is simple to see if we consider a misleading mapping $v_1 \equiv v_2$ in q and all mappings made after that one in an interaction. If there are no positive mappings, the value of $\mathcal{V}_{ev}(v_1 \simeq_q v_2)$ will decrease. This may not be enough to make it lower than other options, but since the values of subsequent mappings will never increase, $\mathcal{V}_{ev}(v_1 \simeq_q v_2)$ will continue to decrease, and by a greater factor in future interactions. If, on the other hand, there are positive mappings, they need to be misleading, so they will also be repaired eventually, getting to the first situation. It can be the case that this mappings are correct for other strings, but since correct mappings do not modify the values they will not damage the process, unless the case above occurs, preventing one mapping of being chosen. Since this is true for any experience including $v_1 \simeq_q v_2$, it will eventually be repaired.

3.2 The Combination Methods in Action

Let us analyse the performance of the two criteria presented in this section in the travel agency scenario. As we already mentioned, there is a misleading mapping between $Single$ and $RoundTrip$; as consequence, the `exp-alg` method will fail many times, until the agent chooses to explore. This is solved in few interactions when using the `ev-alg-exp` criterion. Since the interaction fails right after $Single$ is mapped with $RoundTrip$, the fourth criterion will find this error in just one unsuccessful interaction. Also the first time after choosing it, the agent can confirm the $\langle Flight, Flight \rangle$ mapping, since its confidence will be increased with the $\langle leavingDate, departing \rangle$ and $\langle from, origin \rangle$ mappings.

To evaluate our predictions, we studied experimentally the performance of the four criteria in the travel agency scenario, letting agents interact for 60 times. We measured the proportion of successful experiences, as well as after how many interactions they converged, i.e., always understood each other. The results, which are as expected, are shown in Table 2.

Criterion	`alg`	`exp`	`alg-exp`	`ev-alg-exp`
successes (%)	30	92	81	96
convergence	-	7.1	20.1	3.9

Table 2: Results for the travel agency scenario

4 EXPERIMENTAL EVALUATION

To evaluate the methods we propose, we studied how they perform experimentally when used by agents with different vocabularies. In this section we present the results, after discussing the generation of data for experimentation.

4.1 Data Generation

Designing experimentation with interaction protocols raises the immediate question of how to obtain test cases. While it is simple to build random finite state automata, it is not clear that all possible protocols model a realistic interaction, and the literature does not offer a useful characterisation of interaction or conversation protocols. We chose to generate deterministic automata parametrised by the size (given by the number of states) and to only restrict their shape by using a reasonably uniform distribution of the outgoing arrows among the states.

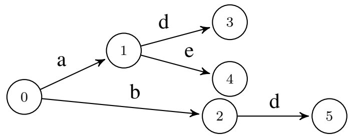
On the other side, we created vocabularies V_1 and V_2 randomly and defined a translation $\mathcal{A}$ between them. Since these are simple sets of words, the trivial similarity measure was used (1 for the same word, 0 for different ones). We labeled an interaction protocol IM_1 with words in V_1 and a structurally equivalent one IM_2 with its translations to V_2. Finally, we explored alignments between V_1 and V_2 of different quality with parametrised values of precision and recall with respect to $\mathcal{A}$. We used confidences of 1 for all the relations in the alignment.

4.2 Experiments

The performance of the methods we propose can be analysed in at least three different dimensions:

1. **The complexity of the interaction models:** we decided not to focus on this dimension for two reasons. First, it is already investigated in [3], and second, preliminary experiments did not show interesting variations of the performance. We used protocols with a fixed size of 90 states for all experiments.
2. **The parameters used:** For `ev-alg-exp`, we experimented with different values of α and C, concluding that low (between 0.2 and 0.4) values of α gave the best results. The results for the punishment were less clear, but values between 0.7 and 0.9 seemed to be

[5] An example is the following IM if the sender is always the same agent and $b \simeq_0 b'$, $a \simeq_0 a'$, but $\equiv a'$ (it is misleading), $d \simeq_1 d'$, $d \equiv d'$ and $e \equiv d'$



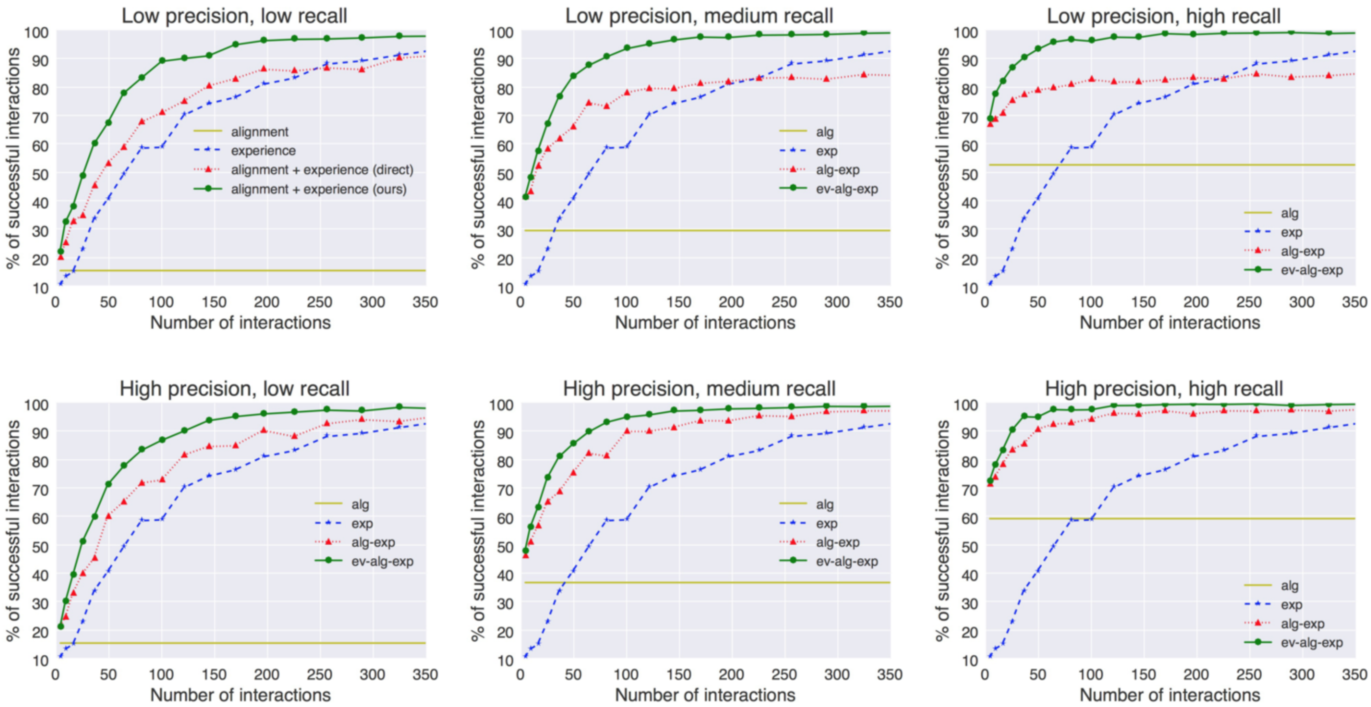


Figure 3: Results for Experiment 1, with size=90 and different alignment qualities

better. An hypothesis that should be confirmed is that this depends on the average of the mapping confidences in the alignment. We used $\alpha = 0.3, C = 1$, and $\xi_{1,2} = 0.1$ for the exploration parameters in the alignment and experience criteria.

3. **The quality of the alignments:** this dimension turned out to be the most interesting one, and we develop it in detail in this section.

4.2.1 *Experiment 1: General Performance*

The first experiment we performed provides a general comparison of the four methods. One test in this experiment is composed of two aligner agents that use the same matching criterion, one following IM_1 and the other IM_2, each of them with an alignment with given values of precision and recall. For each learning criterion, we let agents go through a learning phase in which they interacted n times, performing the experiment for $n = i^2$ and $i \in [2, 20]$. After this training phase, we let agents interact again 100 times, without knowledge update, and measured the proportion of successful interactions. We performed 50 repetitions of each experiment, each time with a different alignment, but maintaining the same values of precision and recall.

We considered three quality classes for the precision and recall values: `low`: 0.2, `medium`: 0.5, `high`: 0.8, and evaluated the criteria that uses alignment with the nine combinations. Figure 3 presents the ontained results, showing the proportion of successful interactions for different lengths of the training phase. For space reasons, we only show six of the nine cases, but the remaining ones follow the same trend. Repeated interactions have no effect on the `alg` alignment; we plot the result of one experiment as a constant. The same happens for `exp` with different alignment qualities.

The two methods that combine the alignment and the learning from the experience perform better in the general case. Between them, `ev-alg-exp` is always the best one, performing better than all other methods. Results are in general very good, achieving 90% of correct matches after only $\sim$ 60 interactions. The method that only uses the learning also reaches values of success close to 1, but more slowly. Using only the alignment is the worst option, except when very short training periods are allowed. A more detailed analysis provides interesting observations about precision and recall:

- Recall affects performance more drastically than precision. This becomes clear when comparing the success rate for `alg`; while it increases significantly with higher values of recall, there is much less variation with different precision values. The two combining methods are also much better with high recall. This shows that errors in a contextualised environment are less dramatic, because it is more rare to find one in the expected messages.
- With low levels of precision, `ev-alg-exp` is significantly better than `alg-exp` after longer learning phases. This can be seen in the plots for low precision, particularly for high or medium recall, where the evolutionary technique reaches values close to 1 while `alg-exp` does not, being even worse than the technique without the alignment. This is explained because low precision implies higher possibility of misleading matches, which are only solved by making the alignment evolve.
- With low levels of recall, `ev-alg-exp` grows faster after short training periods. This is because it takes into account future good mappings, using the available information more efficiently.

4.2.2 *Experiment 2: Focus on Precision and Recall*

In Experiment 1, the effects of using alignments of different qualities are only suggested. To analyse in depth how the performance of our techniques changes with different values of precision and recall, we developed a second experiment. In Experiment 2, we let agents

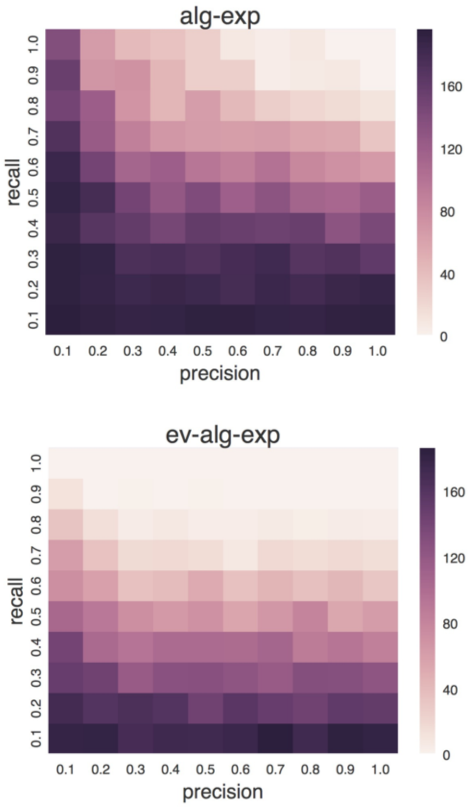


Figure 4: Results for Experiment 2

interact a large number of times (fixed in 200) and measured after how many interactions they converged when using the techniques `alg-exp` and `ev-alg-exp`. In this case, we considered convergence to be having 90% of successful interactions. The results are shown in Figure 4. The colour degrade represents the number of interactions before convergence, which increases with darkness. For the `alg-exp` technique, both low precision and low recall affect the performance, only converging fast when both values are high. As we already pointed out, low recall is more harmful than low precision. In the results for `ev-alg-exp` it can be seen that the precision has less influence; with high levels of recall, low values of convergence are achieved even with very low precision. This shows again how making the alignment evolve repairs misleading mappings, solving low quality in this dimension.

5 RELATED WORK

Although it is considered one of the main applications of ontology matching, the integration of vocabulary alignments in multi-agent interactions, and particularly the problem of how to use and repair them online, has not been deeply studied yet. Most of the work on using alignments in multi-agent systems tackles the problem of how communities of agents can achieve one common alignment from a set of herogeneous ones. This is the approach followed by Laera et al., where argumentation techniques are used to decide between different alignments [16], and by Silva et al., who propose a method for agents to negotiate semantic bridges based on their confidence on each mapping rule. To the best of our knowledge, all existent methods consider an offline negotiation, that results in a common alignment that agents can use to communicate [18].

A well known formulation of the problem of learning meaning automatically from communication is the work of Steels [19]. In addition to the work we presented in Section 2, there exist other approaches, for example the one by Goldman et al. [12], in which the authors investigate how agents can learn to communicate in a way that maximises rewards in an environment that can be modelled as a Markov Decision Process. Our approach differs from this work in the inclusion of alignments and the modelling of an interaction context. In [4], the authors study a version of the multiagent, multiarmed bandit problem in which agents can communicate between each other with a common language, but message interpretations are not know.

A related problem not yet tackled by our approach is the one of aligning structural aspects of interaction protocols. Chopra and Singh have worked extensively on developing alignment techniques for protocols based on commitments [6, 7]. A different approach consists in developing dynamic protocols, that can be modified by agents while interacting according to the situation they are in [1, 13]. Similar approaches have been developed in the Web Services community, with the objective of making dynamic discovery and coordination of services possible [21].

6 CONCLUSIONS AND FUTURE WORK

We proposed methods that combine ontology alignments and language learning techniques, showing that they improve significantly the understanding between agents that interact in a given context. Our second method, in particular, shows how with simple techniques low quality of alignments can be mitigated.

Interesting conclusions about the quality of the alignments can be drawn from the experimentation. First, the level of recall seems to have more impact than the precision when the alignment is used for agent communication. This is worth exploring further, particularly given the current trend of favouring precision over recall in ontology alignment techniques [10].

Other directions of future research are found both from the ontologies and the agents side. The ability to estimate alignment quality measures would be useful for agents, because it would provide them with resources to choose between techniques or to fix parameters. On the other hand, while the `ev-alg-exp` criterion helps finding and avoiding possible errors in the alignment, this is only useful for one particular interaction. A method to repair ontology alignments automatically from the interaction experience could be extracted from these ideas. More generally, the development of matching techniques that create alignments to be used in agent interactions is an unexplored area, and needs both theoretical and practical development. To conclude, in our approach all the reasoning about the alignment is done locally; we think the performance of our methods would improve with a framework of negotiation in which interlocutors could discuss the situation of the alignment.

ACKNOWLEDGEMENTS

This research has been funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 567652 /ESSENCE: Evolution of Shared Semantics in Computational Environments/.

References

[1] Alexander Artikis, 'Dynamic protocols for open agent systems', in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems Volume 1*, AAMAS '09, pp. 97–104, Richland, SC, (2009). International Foundation for Autonomous Agents and Multiagent Systems.

[2] Manuel Atencia, *Semantic alignment in the context of agent interaction*, Doctoral dissertation, Universitat Autònoma de Barcelona, 2010.

[3] Manuel Atencia and Marco Schorlemmer, 'An interaction-based approach to semantic alignment', *Journal of Web Semantics*, **12-13**, 131–147, (2012).

[4] Samuel Barrett, Noa Agmon, Noam Hazon, Sarit Kraus, and Peter Stone, 'Communicating with unknown teammates', in *ECAI 2014 — 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic — Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, eds., Torsten Schaub, Gerhard Friedrich, and Barry O'Sullivan, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pp. 45–50. IOS Press, (2014).

[5] Paolo Bouquet, Jérôme Euzenat, Enrico Franconi, Luciano Serafini, Giorgos Stamou, and Sergio Tessaris, 'Specification of a common framework for characterizing alignment', Deliverable D2.2.1, Knowledge Web, (2004).

[6] Amit K. Chopra and Munindar P. Singh, 'Multiagent commitment alignment', in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '09, pp. 937–944, Richland, SC, (2009). International Foundation for Autonomous Agents and Multiagent Systems.

[7] Amit K. Chopra and Munindar P. Singh, 'Generalized commitment alignment', in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pp. 453–461, Richland, SC, (2015). International Foundation for Autonomous Agents and Multiagent Systems.

[8] Maxim Davidovsky, Vadim Ermolayev, and Vyacheslav Tolok, 'A survey on agent-based ontology alignment', in *ICAART 2012 — Proceedings of the 4th International Conference on Agents and Artificial Intelligence, Volume 2 - Agents, Vilamoura, Algarve, Portugal, 6-8 February, 2012*, eds., Joaquim Filipe and Ana L. N. Fred, pp. 355–361. SciTePress, (2012).

[9] Anhai Doan and Alon Y Halevy, 'Semantic-Integration Research in the Database Community - A Brief Survey', *AI Magazine*, **26**(1), 83–94, (2005).

[10] Zlatan Dragisic et al., 'Results of the ontology alignment evaluation initiative 2014', in *Proceedings of the 9th International Workshop on Ontology Matching collocated with the 13th International Semantic Web Conference (ISWC 2014), Riva del Garda, Trentino, Italy, October 20, 2014.*, pp. 61–104, (2014).

[11] Jérôme Euzenat and Pavel Shvaiko, *Ontology matching*, Springer, 2007.

[12] Claudia V. Goldman, Martin Allen, and Shlomo Zilberstein, 'Learning to communicate in a decentralized environment', *Autonomous Agents and Multi-Agent Systems*, **15**(1), 47–90, (August 2007).

[13] Akin Günay, Michael Winikoff, and Pinar Yolum, 'Dynamically generated commitment protocols in open systems', *Autonomous Agents and Multi-Agent Systems*, **29**(2), 192–229, (March 2015).

[14] Sandra Heiler, 'Semantic interoperability', *ACM Computing Surveys*, **27**(2), 271–273, (1995).

[15] Ningsheng Jian, Wei Hu, Gong Cheng, and Yuzhong Qu, 'Falcon-AO: Aligning ontologies with falcon', in *Integrating Ontologies '05, Proceedings of the K-CAP 2005 Workshop on Integrating Ontologies, Banff, Canada, October 2, 2005*, volume 156 of *CEUR Workshop Proceedings*, pp. 85–91, (2005).

[16] Loredana Laera, Ian Blacoe, Valentina A. M. Tamma, Terry R. Payne, Jérôme Euzenat, and Trevor J. M. Bench-Capon, 'Argumentation over ontology correspondences in MAS', in *6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007), Honolulu, Hawaii, USA, May 14-18, 2007*, eds., Edmund H. Durfee, Makoto Yokoo, Michael N. Huhns, and Onn Shehory, p. 228, (2007).

[17] Maxime Morge and Jean-Christophe Routier, 'Debating over heterogeneous ontologies', *Applied Ontology*, **2**, 333–349, (2007).

[18] Nuno Silva, Gecad Isep Ipp, and Paulo Maio, 'An approach to ontology mapping negotiation', in *Proceedings of the Workshop on Integrating Ontologies*, pp. 54–60, (2005).

[19] Luc Steels, 'The Origins of Ontologies and Communication Conventions in Multi-Agent Systems', *Autonomous Agents and MultiAgent Systems*, **1**, 169–194, (1998).

[20] R.S. Sutton and A.G. Barto, *Reinforcement learning: An introduction*, volume 116, Cambridge Univ Press, 1998.

[21] Katia Sycara, Massimo Paolucci, Anupriya Ankolekar, and Naveen Srinivasan, 'Automated discovery, interaction and composition of semantic web services', *Web Semantics: Science, Services and Agents on the World Wide Web*, **1**(1), 27–46, (2003).

[22] Jurriaan van Diggelen, Robbert-Jan Beun, Frank Dignum, Rogier M. van Eijk, and John-Jules Ch. Meyer, 'Ontology negotiation: goals, requirements and implementation', *International Journal of Agent-Oriented Software Engineering*, **1**(1), 63, (2007).

[23] Michael Wooldridge, *An Introduction to MultiAgent Systems*, Wiley Publishing, 2nd edn., 2009.

[24] Zhibiao Wu and Martha Stone Palmer, 'Verb semantics and lexical selection', in *32nd Annual Meeting of the Association for Computational Linguistics, 27-30 June 1994, New Mexico State University, Las Cruces, New Mexico, USA, Proceedings.*, ed., James Pustejovsky, pp. 133–138, (1994).