Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)

Vocabulary Alignment for Collaborative Agents: a Study with Real-World Multilingual How-to Instructions

Paula Chocron,[1,2] **Paolo Pareti**[3]
[1] IIIA, CSIC, Barcelona, Spain
[2] Universitat Autònoma de Barcelona, Barcelona, Spain
[3] University of Southampton, Southampton, United Kingdom
pchocron@iiia.csic.es, p.pareti@soton.ac.uk

Abstract

Collaboration between heterogeneous agents typically requires the ability to communicate meaningfully. This can be challenging in open environments where participants may use different languages. Previous work proposed a technique to infer alignments between different vocabularies that uses only information about the tasks being executed, without any external resource. Until now, this approach has only been evaluated with artificially created data. We adapt this technique to protocols written by humans in natural language, which we extract from instructional webpages. In doing so, we show how to take into account challenges that arise when working with natural language labels. The quality of the alignments obtained with our technique is evaluated in terms of their effectiveness in enabling successful collaborations, using a translation dictionary as a baseline. We show how our technique outperforms the dictionary when used to interact.

1 Introduction

Enabling collaboration between agents with different backgrounds is one of the objectives of open multi-agent systems. For agents that need to perform complex sequential tasks, collaborating with others may be necessary when they lack the resources to perform all steps by themselves, or convenient for efficiency reasons. To coordinate the collaboration, the ability to communicate meaningfully is essential. However, in truly open environments it is difficult to ensure that all participants speak the same language. Even when the language is shared, names for specific tools or activities are notably diverse between communities of speakers. Most techniques to tackle vocabulary heterogeneity in multi-agent interactions rely on external resources, such as dictionaries or data corpora. These resources are not always available, and even when they are, they are often not contextualized and may therefore not be useful for the specific interactions agents need to perform. An alternative approach are *interaction-based alignment techniques* [Chocron and Schorlemmer, 2016; Atencia and Schorlemmer, 2012; Chocron and Schorlemmer, 2017], in which agents use (only) the information in the procedural specification of the tasks to find an alignment between their vocabularies. For example, consider two agents that collaborate to make tea. They follow the same steps, but one uses a specification in English and the other one in Spanish. The idea is that, if agents make tea together many times, they can learn the procedure in the foreign language, by observing the outcomes of different utterances. Instead of using external resources, agents leverage shared information, obtaining a cheap and context-specific alignment.

Until now, interaction-based alignment techniques had only been applied to artificial, randomly generated protocols. Our work considers a concrete scenario in which agents collaborate to complete real-world step-by-step instructions. This scenario is based on previous research which showed how step-by-step instructions described in natural language can be automatically formalised into machine understandable data [Pareti *et al.*, 2014]. Unlike related work in processing instructional knowledge [Addis and Borrajo, 2011; Kiddon *et al.*, 2015; Tenorth *et al.*, 2010; Schumacher *et al.*, 2012; Malmaud *et al.*, 2014], the work in [Pareti *et al.*, 2014] provides a formalisation that allows agents to automatically understand and execute instructions, provided they have the necessary abilities [Pareti *et al.*, 2016; Pareti, 2016]. We consider a publicly available dataset that follows this formalisation and that has been extracted from instructional websites.[1]

The contribution of this work is twofold. First, we test interaction-based alignment techniques in a real-world instructional dataset, to uncover potential challenges that may arise in the process, and we provide solutions to solve these challenges. At the same time, we provide a novel automatic tool to allow artificial agents that speak different languages to collaborate when following human-crafted protocols. Our alignment technique does not rely on external resources and it aims to be language independent. We are considering a scenario where approaches such as Machine Translation are unavailable or unfeasible to use. However, linguistic resources are used in the preparation of the data used in the experiments.

The following is the general outline of this paper. In Section 2 we describe in formal terms the protocols we use and

[1] https://github.com/paolo7/KnowHowDataset

the collaboration setting. Section 3 presents the adaptation of interaction-based alignment techniques to make them suitable for our protocols.[2] In doing so, we show how to deal with the challenges that arise from using natural language. Section 4 provides details about the data extraction of bilingual protocols (in English and Spanish) that we use for the evaluation. Section 5 describes the evaluation of the techniques when used by agents following protocols in our corpus, measuring the percentage of successful interactions achieved by the agents. We used the Oxford Dictionary [Stevenson, 2010] as a benchmark. In Section 6 we discuss the obtained results.

2 Performing Tasks Collaboratively

The question of what should be expressed by an interaction protocol has been largely discussed in the multiagent systems community [Singh, 2000; Alberti *et al.*, 2007]. In this paper we will use a simple formalism that captures well our data. Intuitively, a procedural protocol is a set of instructions that must be performed to reach some goal, together with a *dependency relation* that specifies an order between these instructions. Formally, let V be a vocabulary and V^* be the set of sentences in V, that is, the set of sequences of elements in V. A *protocol* over V is a tuple $\langle T, \prec \rangle$ where $T \subseteq V^*$ is a set of sentences representing *tasks*; and $\prec$ is a strict partial order $\prec$: $T \times T$ that represents the *dependencies* between tasks. More specifically, $t_1 \prec t_2$ means that task t_1 must always precede task t_2. As an example, consider the following protocol Tea that specifies how to achieve the goal of making tea. We rename the tasks for simplicity.

- T = { "Get a tea bag" ($\mathtt{t}_1$), "Get hot water" ($\mathtt{t}_2$), "Add the tea bag in the cup" ($\mathtt{t}_3$) }
- $\mathtt{t}_2 \prec \mathtt{t}_3, \quad \mathtt{t}_1 \prec \mathtt{t}_3$

In this work we make the assumption that tasks are executed sequentially. An *execution* is a sequence of tasks. A *successful* execution E of a protocol $\langle T, \prec \rangle$ is a sequence of tasks that satisfies two criteria:

- The sequence E contains all and only the tasks in T.
- The order of tasks in E does not violate the dependency relation $\prec$. Let $E' \,.\, t_2$ be the sequence obtained by appending element t_2 to sequence E', then $t_1 \prec t_2$ and $E = E' \,.\, t_2 \Rightarrow t_1 \in E'$.

Both $[\mathtt{t}_1, \mathtt{t}_2, \mathtt{t}_3]$ and $[\mathtt{t}_2, \mathtt{t}_1, \mathtt{t}_3]$ are successful executions of the Tea protocol. In the first one, the tea bag is obtained before getting water, and in the second one before adding it to the cup. An unsuccessful execution is $[\mathtt{t}_1, \mathtt{t}_3, \mathtt{t}_2]$, which puts the tea bag in a cup of water before getting the water, violating the dependency between $\mathtt{t}_2$ and $\mathtt{t}_3$.

In this paper we consider a message-based collaboration scenario. Let a and b be two agents that need to work together to achieve some goal. Agents perform tasks individually and have their own local representation of the execution. When one of them completes an action, it communicates this to its partner by sending a message with the label of the completed task. We assume messages are not lost and are received immediately. In the general case, when an agent receives a message from its partner, it adds the corresponding task to its local execution. We tackle the situation in which this step is hindered because agents speak different languages.

[2]The code for the techniques is available in https://github.com/paulachocron/WikiHow-alignment

2.1 Protocol Compatibility

We consider agents that share the dependency structure of the protocol, but not the vocabulary of tasks. To characterize this situation we will use the notion of *compatibility* between protocols, which has been defined for different types of interaction protocols (see for example [Chopra and Singh, 2008]). In this section we will define the compatibility of the protocols we presented previously. To formalize it, we first need to define *alignments*. Let $P = \langle T, \prec \rangle$ and $P' = \langle T', \prec' \rangle$ be two protocols with vocabularies V and V' respectively. An alignment between P and P' is a function $\alpha : T \to T'$ such that, for each execution E that is successful for P, the execution that results from translating each task with α, called $\alpha(E)$, is successful in P'. The two protocols are compatible if there exists a bijective alignment α from P to P' such that its inverse function α^{-1} is an alignment from P' to P. Intuitively, this implies that they have the same number of tasks and that the structure of their dependencies is the same. To illustrate these ideas, consider another protocol Tea' with a different procedure to make tea:

- T' = "Get a tea bag" ($\mathtt{t}'_1$),"Get hot water" ($\mathtt{t}'_2$), "Put the water in a cup" ($\mathtt{t}'_3$), "Add the tea bag in the cup" ($\mathtt{t}'_4$) }
- $\mathtt{t}'_1 \prec' \mathtt{t}'_4, \quad \mathtt{t}'_2 \prec' \mathtt{t}'_3, \quad \mathtt{t}'_3 \prec' \mathtt{t}'_4$

The protocol Tea' is not compatible with Tea, since all its successful executions have four tasks instead of three. Instead, consider a third protocol Tea'':

- T'' = { "Get tea" ($\mathtt{t}''_1$), "Microwave cup of water for 3 minutes" ($\mathtt{t}''_2$), "Add the tea bag in the cup" ($\mathtt{t}''_3$) }
- $\mathtt{t}''_1 \prec'' \mathtt{t}''_3, \quad \mathtt{t}''_2 \prec'' \mathtt{t}''_3$

Tea'' is clearly compatible with Tea, under alignment $\alpha(t_i) = t''_i$ for $1 \leq i \leq 3$. Notice that the mapped labels are not a literal translation: "Microwave cup of water for 3 minutes" is not the same as "Get hot water". Moreover, Tea'' is also compatible with Tea under another alignment α', in which $\alpha'(t_1) = t''_2$, $\alpha'(t_2) = t''_1$, and $\alpha'(t_3) = t''_3$.

This last point is important. Two protocols being compatible under α does not imply that every label t is semantically equivalent to $\alpha(t)$. The notion of compatibility is structural and not semantic. As we will explain later, we work under the implicit assumption that there exists one α that makes protocols compatible which is also meaningful semantically, but the alignment technique is only defined in structural terms.

2.2 Collaboration Dynamics

We assume agents a and b need to collaborate to achieve different goals, because they are unable to perform all the tasks by themselves. For each of these goals, a and b have protocols $P = \langle T, \prec \rangle$ and $P' = \langle T', \prec' \rangle$ respectively, and these protocols are compatible, in particular under an alignment α. For each pair of protocols, let $K_a \subset T$ and $K_b \subset T'$ be the set of tasks that a and b can perform respectively. The only restriction to be able to work together is that, by acting jointly, they

can perform the complete protocol. That is, $T \subseteq K_a \cup \alpha(K_b)$ for some α.

Agents maintain their own execution E^a and E^b respectively, and perform actions individually. After execution E^a, agent a can perform any *possible* task t. The set of possible tasks for agent ag, denoted $Poss^{ag}$, includes all tasks that 1. $t \in K_a$, 2. $t \notin E^a$, and 3. for all t' such that $t' \prec t$, $t' \in E^a$. The sequence of real performed tasks is a sequence $E^{a,b} \in T \cup T'$ that we call a *joint execution*. As we mentioned, agents send messages to each other to communicate the tasks that they perform. These messages are in the language of the sender, so the receiver has to interpret them in its own set of tasks to be able to continue the execution. To this aim receivers use a local alignment α^{ag}, for $ag \in \{a, b\}$. When an agent receives a foreign message m, it finds $\alpha^{ag}(m)$ and chooses it as an interpretation, adding it to the execution. Agents finish the interaction when either their local executions are successful, or none of them has possible messages, that is, $Poss^a = Poss^b = \emptyset$.

Two points should be noted. First, there is no guarantee that the protocols are compatible under the local alignments α^{ag}. When they are not, agents will finish the interaction before obtaining successful local executions, since they misinterpreted some message. Second, even if agents finish with successful local executions, the joint execution may not be semantically correct. For example, if a uses Tea and b uses Tea'', they could perform the execution $[t_1, t_1'', t_3]$. This translates to successful interactions under the previously described α', but it results in agents obtaining two tea bags and no water, which intuitively violates the semantic interpretation of the instruction. In next section we present a technique to learn an α^{ag} that optimises successful local execution, making agents obtain successful interactions more often. The notion of semantically correct execution is not used here; moreover, we do not assume there is a way of deciding if an execution is semantically correct or not in the collaboration scenario. We will show experimentally how alignments that optimise successful local executions also lead to semantically correct executions.

3 Alignment Learning Technique

To optimise local successes, agents need to find alignments under which their protocols are compatible, that we propose to obtain from the experience of interacting, based on the approach in [Chocron and Schorlemmer, 2017; 2016]. In these papers, agents maintain a confidence distribution for possible mappings between foreign and local messages. When an agent receives a new message, it uses the confidence distribution to decide how to interpret it locally. In addition, it updates the confidence distribution taking into account which local messages are expected and which ones are not; that is, which messages are in the set $Poss$. With repeated interactions agents improve their confidence distribution, making better interpretation choices. By using a probabilistic technique, agents can use information even when it is uncertain, since possible messages depend on previous interpretations.

Having a confidence distribution for mappings between messages is useful when messages are indivisible units that are frequently repeated, but not in our case, where tasks are sets of words. For example, consider an agent that learns, from repeated interactions, that the sentence "Put the tea bag in the water" maps to the Spanish "Poner la bolsita de té en el agua". Even being correct, this translation might not be useful for future interactions. While words such as *water* or *tea* are likely to be encountered again, the agent could never receive the exact same message in a different protocol.

The method we propose computes two confidence distributions: one between words and one between sentences. Consider again agents a and b with vocabularies V^a and V^b that interact repeatedly using, for each interaction, different but compatible protocols in their own languages. We will explain the technique from the point of view of agent a who needs to learn α^a, but the method is analogous for b. On one side, a has a confidence distribution $\omega : V^b \times V^a \to \mathbb{N}$. This is a partial function that assigns confidences to mappings between foreign and local words, and it is is updated when new evidence is obtained. A second function, $\delta : V^{b^*} \times V^{a*} \to \mathbb{N}$, assigns confidences to mappings between foreign and local sentences, and it is computed using ω. When it receives a foreign message t^b, agent a performs two actions. First, it computes its expected messages in $Poss^a$ and updates ω for the words in t^b with this information. Second, it computes δ to choose a local interpretation for the foreign sentence. In this way, the similarity of two sentences is computed using the mapping degree between their words. At the same time, the update of the word-level mapping confidences takes into account the whole sentences in which words appear.

This approach makes two assumptions. First, to compute sentence similarities from word similarities it is necessary that the meaning of a sentence is related with the meanings of the words that appear in it. Second, using full sentences to determine word similarities is only useful if similar words tend to appear surrounded by words that are also similar. These assumptions are similar to the hypothesis of *distributional semantics* [Turney and Pantel, 2010; Partee *et al.*, 1992] known as *principle of compositionality* and *distributional hypothesis* respectively. Instead of assuming the existence of a large corpora to extract information about word similarity, our approach harnesses shared structure, which in this case is a protocol. In the rest of this section we explain how confidence distributions are updated.

3.1 Choosing a Mapping

Consider agent a receives a message t^b. Finding an interpretation for message t^b involves two steps: 1. Computing δ from ω. 2. Computing α^a from δ.

The second step is straightforward. The local interpretation of message t^b is chosen randomly between the possible local tasks that map with t^b with maximal confidence:

$$\alpha^a(t^b) \in \underset{t^a \in Poss^a}{\operatorname{argmax}}(\delta(t^b, t^a))$$

Obtaining the sentence mapping degree from ω requires more work. Intuitively, the confidence for the mapping between two sentences is computed from the confidences of the mappings between their words. The challenge is how to combine the words, considering that word ordering is different for each language. For example, adjectives precede

nouns in English, but not in Spanish. Existent measures of sentence similarity [Li *et al.*, 2006; Agirre *et al.*, 2016; Mihalcea *et al.*, 2006] require either external resources such as semantic databases, large corpora of data, or information about the grammar of particular languages, which we do not assume to be available. We present an approach to combine the information about individual word mappings that, in spite of its simplicity, is effective in dealing with simple imperative sentences such as the ones commonly found in how-to instructions. Concretely, we consider all possible combinations of mappings.

Let $t^a \in V^{a*}$ and $t^b \in V^{b*}$, that is, $t^a = w_1, \ldots, w_n$ and $t^b = v_1, \ldots, v_m$, with $w_i \in V^a, v_j \in V^b$ for $1 \leq i \leq n$ and $1 \leq j \leq m$. Suppose, in this case, that $m \leq n$ (otherwise everything is analogous). The value $\delta(t^b, t^a)$ is computed as follows. Consider all the permutations of length m of the sentence t^b, that we will call $Perm_m(t^b)$. For each $c \in Perm_m(t^b)$, we compute its *partial confidence degree* δ_p with t^a by considering the mappings between words in the same indexes:

$$\delta_p(c, t^a) = \sum_{1 \leq i \leq m} \omega(c[i], t^a[i])$$

Then, the confidence degree between the original words is the maximal partial value:

$$\delta(t^b, t^a) = \max_{c \in Perm_m(t^b)} \delta_p(c, t^a)$$

This does not consider the difference in length between the sentences. Although it should not be considered too strongly, this information can be helpful, particularly in the first interactions when there are many options. To take it into account, we subtract to $\delta(t^b, t^a)$ a value that is computed in relation to the difference in their length. Given a constant parameter ρ, and if abs is the absolute value of a number:

$$\delta(t^b, t^a) = \delta(t^b, t^a) - \rho * abs(n - m)$$

Taking Word Frequencies into Account

The technique we proposed still does not take into account an inherent property of natural languages: some words are more frequent than others. Frequent words will be updated more often, and therefore their mapping with any other word will have higher value. This can be avoided by including the information about word frequency when computing δ. Coherently with our assumption that agents have no information a priori, we propose to track the frequency of foreign words dynamically. Concretely, agent a maintains a partial frequency function $Freq : V^a \cup V^b \to \mathbb{N}$. The function is partial because it only has values for those words that have already been used at least once. For local words $v \in V_a$, $Freq(v)$ is updated each time the agent starts using a new protocol, counting how many times they appear. For foreign words, it is updated each time the agent receives a new sentence. When computing the partial mapping degree between two sentences, the value of each mapping is divided by these frequencies.

$$\delta_p(c, t^a) = \frac{\sum_{0 \leq i \leq m} \omega(c[i], t^a[i])}{Freq(c[i]) + Freq(t^a[i])}$$

3.2 Updating ω

The values in ω, which represent the confidence on mappings between individual words, are updated from the experience of interacting. When agent a receives t^b, it first updates the mappings between the words in t^b and the words in possible messages. Let r be a constant parameter and $v^b \in t^b$. We assume the agent initializes $\omega(v^a, v^b) = 0$ for all $v^a \in V^a$ the first time it receives v^b. Then, for all $t^a \in Poss^a$ and $v^a \in t^a$:

$$\omega(v^a, v^b) = \omega(v^a, v^b) + r$$

If v^a does not belong to a possible message, the value of its mappings remains the same. Using only this simple approach, agents would be overlooking useful information that would be easy to take into account. For example, about the other words that appear in the sentence. Suppose an agent following a protocol in Spanish receives the sentence t^{en} = *cup of barley*. The agent may not know the word *barley*, but if it has performed other protocols before, it probably knows that *cup* maps to *taza*. If there is a local task t^{es} = *taza de cebada*, it may infer that t^{es} maps with t^{en} only using *cup*, and learn that *barley* maps with *cebada*. The information about the other words in a sentence is in the already computed confidence values for mappings between sentences. Agents only need to also use these values to update the confidence degree.

Again, suppose a receives t^b. For all $t^a \in Poss^a$, consider all $v^b \in t^b$ and all $v^a \in t^a$. Assuming again that t^a is shorter than t^b, Let $Perm_{v^b, v^a}$ be all the permutations c of t^b of the same length as t^a such that the index of v^b in c is the same as the index of v^a in t^a. Since the v^b only maps with one word, the agent updates ω as follows:

$$\omega(v^a, v^b) = \omega(v^a, v^b) + \max_{c \in Perm_{v^b, v^a}} \delta_p(c, t^a)$$

In our technique, agents perform first the simple update and then, once the mappings are computed they add these values.

4 Data Acquisition

To evaluate our techniques in a concrete scenario we test them against real-world protocols of human activities. We obtain these protocols from the Human Instructions Dataset [Pareti *et al.*, 2014] which contains over 200,000 sets of instructions extracted from instructional websites such as wikiHow.[3] In this work, we focus on the English and Spanish subsets of this dataset.[4]

Sets of instructions in this dataset are formalized as graphs using the PROHOW instructional model [Pareti *et al.*, 2014] and are serialized as RDF [Raimond and Schreiber, 2014] triples. The PROHOW model represents instructions using the concepts of *steps*, *methods* and *requirements*. Intuitively, steps decompose a set of instructions into a set of simpler tasks; methods provide information about different ways to decompose a a set of instruction into steps; and requirements provide a notion of dependency between tasks.

[3] http://www.wikihow.com/

[4] https://www.kaggle.com/paolop/human-instructions-multilingual-wikihow

We focus our experiment on a particular domain of human instructions, namely cooking recipes. To do so we select instructions that belong to 8 wikiHow categories related to cooking, and that are available both in English and Spanish. Sets of instructions are converted into a protocol $\langle T, \prec \rangle$ as follows. Steps and requirements of a set of instructions are interpreted as the elements of the set of tasks T. The ordering of the steps in the set of instructions is included in the dependencies $\prec$ of the derived protocol. Moreover, for each requirement r of a set of instructions, a dependency $r \prec s$ is added, where s is the first step in the set of instructions that requires requirement r.

Each pair of sets of instructions (one in English and one in Spanish) is converted into a pair of protocols. Pairs of protocols which are not compatible with each other are excluded from our experiment. For the sake of brevity, here we give just a summary of the approach we use to verify compatibility. Compatibility between protocols can be verified by defining the *dependency tree* of a protocol $\langle T, \prec \rangle$ as the transpose of the graph which has T as the set of nodes and $\prec$ as the set of edges. The way in which protocols are constructed ensures that this graph is a rooted tree with the task corresponding to the last step of the set of instructions as the root. Two protocols are compatible if their dependency trees are isomorphic. This is a result of the fact that if two rooted trees are not isomorphic, then there is no bijective alignment α between the nodes of the trees that is edge-preserving [Elgot *et al.*, 1978].

After this selection, we obtain a final set of 327 pairs of compatible protocols. The labels of the tasks of these protocols are cleaned using standard natural language approaches using the FreeLing natural language processing tool suite [Padró and Stanilovsky, 2012]. We perform Part of Speech tagging and only keep the lemmatised version of nouns, adjectives, verbs, and adverbs. Other parts of the label, such as numbers, are removed. The software used in this data acquisition phase, along with details of its exact configuration, is available on GitHub.[5]

5 Evaluation

To evaluate the performance of the techniques that we propose we analysed how well agents can interact using the alignments that they learn with our technique. Here, we used the notion of *semantic success*. A joint execution is semantically successful if it is successful in both protocols when translated with a *semantically correct alignment*. Using the previous example, the execution in which two tea bags and no water are obtained would be semantically unsuccessful for the protocol Tea. The semantically correct alignment is one that maps semantically equivalent tasks. In our case, this alignment was computed using a combination between extra information in the protocols, external tools, and human supervision.

In each experiment, we let agents perform a fixed number of training interactions, in which they executed protocols chosen randomly. During these interactions they updated their confidence distribution ω as explained in the previous section. Then, we performed 100 test interactions on the same set of protocols, without updating. We measured how many of the test interactions were successful. We repeated the experiment for 2^n training interactions, with n between 0 and 11. This experiment was repeated 5 times for 5 different sets of training and test protocols.

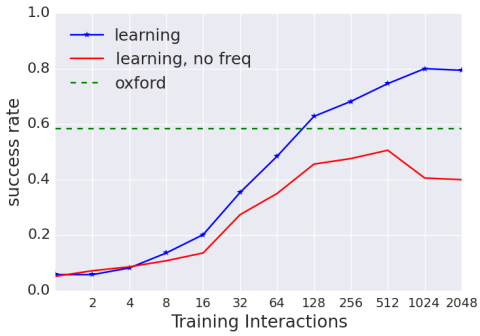

Figure 1: Success rate for different training interactions. *learning* agents use the technique that we propose, *learning no freq* agents do not take into account the frequency of words, and *oxford* agents use the Oxford Dictionary alignment, without learning.

We compared two different strategies to compute δ: (1) the basic agent described in the main part of 3.1 and (2) the basic agent, without taking frequencies into account. We compared these agents with the rate of success obtained when agents use an external alignment, which was extracted from the Oxford Dictionary Spanish-English [6] module. Agents using the Oxford Dictionary choose their interpretations using δ as we described, but instead of learning a distribution over word mappings δ, they use one extracted from the dictionary as follows. For a foreign and local words v^b and v^a respectively,

$$\omega(v^b, v^a) = \begin{cases} 1 & \text{if } v^a \text{ is a translation for } v^b \text{in the dictionary} \\ 0 & \text{otherwise} \end{cases}$$

Figure 1 shows the success rate for the three agents. Agents that take into account word frequencies perform better than those who do not. In particular, the latter become worse after many interactions, when observing some words more often than others starts to affect the values.

Our alignment technique allows agents to collaborate successfully in nearly 80% of the cases, and they outperform the success rate obtained with the dictionary after only around 100 interactions. These results are obtained with very simple updating techniques and no semantic resources at all. Interestingly, an alignment that is semantically correct can be obtained from an update that only takes into account structural properties. Even if a pair of protocols has many compatible alignments, our technique will find the one that is semantically correct. This is because agents learn from many pairs of protocols and not only one, and the correct one is the only alignment that makes all pairs compatible. *Get a tea bag* may be mapped with *microwave water* in one protocol, but the alignment will not work for others. To confirm this, we measured the success rate obtained with the original interaction-based alignment method from [Chocron and Schorlemmer,

[5] https://github.com/paolo7/protocol-generators

[6] https://es.oxforddictionaries.com/english-spanish

2017; 2016], that is, using a confidence distribution between sentences directly. The results were always close to 0, because agents do not choose the alignment that is semantically correct.

The success rate does not reach 100%. This is because some protocols have translations that are difficult to learn. For example, one of the protocols pairs has *cuchara de helado* (ice cream spoon) corresponding to *melon baller*, even when their individual words are not similar. As we discuss later, an approach considering noun phrases would solve this issue. Other protocols had misspelled or strangely written words. For example, a protocol called for a *tea spoon* (instead of a *teaspoon*), which the agent mapped to *té* (tea).

6 Discussion

The main objective of this work is to learn an alignment between words which increases the success rate of the interactions between the agents, without explicitly considering any semantic interpretation of this alignment. However, it is reasonable to expect that a useful alignment would map together words which are the translations of each other. We tested this hypothesis by evaluating the "correctness" of the mappings between words under the interpretation that they represent a multilingual translation. We do so by defining a translation between words in V^a and V^b as a relation $Trans : V^a \times V^b$. If agent a has confidence distribution ω^a between words in its vocabulary and V^b, the translation $Trans$ is the set of all the pairs with highest confidence value. For $v^a \in V^a$, $v^b \in V^b$,

$$(v^a, v^b) \in Trans \Leftrightarrow v^a \in \underset{v \in V^b}{\operatorname{argmax}}\, \omega(v, v^b)$$

First we automatically evaluated the translations that we obtained using the Oxford Dictionary as a reference. Concretely, we considered a pair of words in $Trans$ as correct if and only if the Oxford Dictionary lists one of the two words as a translation of the other. This precision starts at a value of approximately 17% when the experiment is initialised. As expected, precision increases as agents interact and reaches a plateau of 60% after 1000 interactions. One problem with this evaluation is that not all correct translations are present in the dictionary. Therefore, we performed a manual evaluation of 400 translations randomly selected from the set generated after 1500 interactions. This manual evaluation measured a precision of 64.2%.

While the obtained alignments are useful to make agents interact successfully, the values of precision that are obtained are not sufficiently high to consider them reliable translations. At this stage, the alignments can be used to interact, but not for other purposes such as to translate a recipe from scratch. This is due in large part to unresolved issues in the initial cleaning and natural language processing of the protocol labels and would be improved with a more efficient tool, able to identify misspellings and to lemmatize words correctly.

A main issue is not being able to identify noun phrases. An example is *xanthan gum* and its Spanish equivalent *goma xantana*. These words appear always together in the protocol, so the agent has no way of identifying if the correct alingment is {(goma, gum), (xantana,xanthan)} or {(goma, xanthan), (xantana, gum)}. However, both mappings are useful to interact if the words appear together. This explains why the performance of the agents achieves better results than the alignment's precision. Moreover, many concepts (such as *turn off* and *apagar*) can be described with one word in a language but need more in another one, and therefore cannot be translated with naive word-to-word alignments. This issue can be solved in two ways. One option is to use a more powerful processing that identifies noun phrases. The other one is to let agents identify them automatically. This, however, would require more training examples and much more computation time.

With these issues solved, our alignments would be a very useful context-specific resource. On one side, we have already observed that a typical dictionary does not have many of the particular words that are commonly used in a jargon. In addition, general dictionaries provide many possible translations for a word, letting user needs to identify which one is useful for its needs. Our alignments, instead, would directly provide a translation that is suitable for the context.

It is also important to highlight the similarity between this work and the field of Ontology Alignment (OA). In particular, there exist approaches to match different process models with each other [Antunes *et al.*, 2015]. Like typical OA systems, we try to exploit both structural and lexical properties to reach an alignment. However, unlike typical OA problems, we do not assume that agents can share explicit ontological information. Instead, they only share textual messages while following a protocol. While protocols can be thought of as an ontology, our system does not generate alignments between them, but instead finds mappings between words. In addition, the goal of our system is not to reach a semantically correct alignment, but instead an alignment that maximises the success of the collaborations.

7 Conclusions

The methods that we propose in this paper allow agents that follow instructions in two different languages to learn domain-specific alignments between their respective vocabularies in order to perform instructions collaboratively. We start with a structural notion of *compatibility* to find alignments between instructions which does not consider the meaning of the tasks. We then show that, even when there may be different such alignments, agents learn the one that is semantically correct, meaning the one that maps together equivalent actions. This is because we consider agents that engage in multiple different collaborations, and the semantically correct alignments are the most useful to them to achieve the highest rate of compatibility across those protocols. Our techniques allow agents to learn how to collaborate to the completion of protocols without using external resources. In fact, agents that use our technique outperform those that use the translations from a well-known dictionary. This suggests that our learning method can be applied in scenarios where external resources, such as dictionaries or translation services, are not available or expensive to use, or as a complement to these resources to discover additional mappings between words, such as domain specific ones.

Acknowledgments

This research has been funded by the European Commission's Seventh Framework Programme (FP7/2007-2013) ESSENCE (nr. 607062), and by CSIC's Proyectos Intramurales Especiales DIVERSIS (nr. 201750E064) and SMA (nr. 201550E040). We thank Thomas Brochhagen, Dagmar Gromann and Marco Schorlemmer for their useful comments.

References

[Addis and Borrajo, 2011] Andrea Addis and Daniel Borrajo. From Unstructured Web Knowledge to Plan Descriptions. In *Information Retrieval and Mining in Distributed Environments*, volume 324 of *Studies in Computational Intelligence*, pages 41–59. Springer Berlin Heidelberg, 2011.

[Agirre *et al.*, 2016] Eneko Agirre, Carmen Banea, Daniel M Cer, Mona T Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *SemEval@ NAACL-HLT*, pages 497–511, 2016.

[Alberti *et al.*, 2007] Marco Alberti, Federico Chesani, Davide Daolio, Marco Gavanelli, Evelina Lamma, Paola Mello, and Paolo Torroni. Specification and verification of agent interaction protocols in a logic-based system. *Scalable Computing: Practice and Experience*, 8(1), 2007.

[Antunes *et al.*, 2015] Gonçalo Antunes, Marzieh Bakhshandeh, José Luis Borbinha, João Cardoso, Sharam Dadashnia, Chiara Di Francescomarino, et al. The process model matching contest 2015. In *Enterprise Modelling and Information Systems Architectures, Proceedings of the 6th Int. Workshop on Enterprise Modelling and Information Systems Architectures, EMISA 2015, Innsbruck, Austria, September 3-4, 2015.*, pages 127–155, 2015.

[Atencia and Schorlemmer, 2012] Manuel Atencia and W. Marco Schorlemmer. An interaction-based approach to semantic alignment. *Journal of Web Semantics*, 12:131–147, 2012.

[Chocron and Schorlemmer, 2016] Paula Chocron and Marco Schorlemmer. Attuning ontology alignments to semantically heterogeneous multi-agent interactions. In *ECAI 2016 - 22nd European Conference on Artificial Intelligence, 29 August-2 September 2016, The Hague, The Netherlands - Including Prestigious Applications of Artificial Intelligence (PAIS 2016)*, pages 871–879, 2016.

[Chocron and Schorlemmer, 2017] Paula Chocron and Marco Schorlemmer. Vocabulary alignment in openly specified interactions. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '17, pages 1064–1072, Richland, SC, 2017. International Foundation for Autonomous Agents and Multiagent Systems.

[Chopra and Singh, 2008] Amit K. Chopra and Munindar P. Singh. Constitutive interoperability. In *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 2*, pages 797–804, 2008.

[Elgot *et al.*, 1978] Calvin C. Elgot, Stephen L. Bloom, and Ralph Tindell. On the algebraic structure of rooted trees. *Journal of Computer and System Sciences*, 16(3):362 – 399, 1978.

[Kiddon *et al.*, 2015] Chloé Kiddon, Ganesa Thandavam Ponnuraj, Luke S. Zettlemoyer, and Yejin Choi. *Mise en Place: Unsupervised Interpretation of Instructional Recipes*, pages 982–992. Association for Computational Linguistics (ACL), 2015.

[Li *et al.*, 2006] Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett. Sentence similarity based on semantic nets and corpus statistics. *IEEE Trans. on Knowl. and Data Eng.*, 18(8):1138–1150, August 2006.

[Malmaud *et al.*, 2014] Jon Malmaud, Earl J Wagner, Nancy Chang, and Kevin Murphy. Cooking with Semantics. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, pages 33–38, 2014.

[Mihalcea *et al.*, 2006] Rada Mihalcea, Courtney Corley, Carlo Strapparava, et al. Corpus-based and knowledge-based measures of text semantic similarity. In *AAAI*, volume 6, pages 775–780, 2006.

[Padró and Stanilovsky, 2012] Lluís Padró and Evgeny Stanilovsky. Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey, May 2012. ELRA.

[Pareti *et al.*, 2014] Paolo Pareti, Benoit Testu, Ryutaro Ichise, Ewan Klein, and Adam Barker. Integrating Know-How into the Linked Data Cloud. In *Knowledge Engineering and Knowledge Management*, volume 8876 of *Lecture Notes in Computer Science*, pages 385–396. Springer International Publishing, 2014.

[Pareti *et al.*, 2016] Paolo Pareti, Ewan Klein, and Adam Barker. Linking Data, Services and Human Know-How. In *The Semantic Web. Latest Advances and New Domains*, ESWC 2016, pages 505–520. Springer International Publishing, 2016.

[Pareti, 2016] Paolo Pareti. Distributed Linked Data as a Framework for Human-Machine Collaboration. In Olaf Hartig, Juan Sequeda, and Aidan Hogan, editors, *Proceedings of the 7th International Workshop on Consuming Linked Data (COLD)*, number 1666 in CEUR Workshop Proceedings, Aachen, 2016.

[Partee *et al.*, 1992] Barbara H. Partee, Alice ter Meulen, and Robert E. Wall. Mathematical methods in linguistics. *Journal of Symbolic Logic*, 57(1):271–272, 1992.

[Raimond and Schreiber, 2014] Yves Raimond and Guus Schreiber. RDF 1.1 Primer. W3C Note, W3C, June 2014. http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/.

[Schumacher *et al.*, 2012] Pol Schumacher, Mirjam Minor, Kirstin Walter, and Ralph Bergmann. Extraction of Procedural Knowledge from the Web: a comparison of two workflow extraction approaches. In *Proceedings of the 21st international conference companion on World Wide Web*, WWW '12 Companion, pages 739–747, New York, NY, USA, 2012. ACM.

[Singh, 2000] Munindar P. Singh. A social semantics for agent communication languages. In *Issues in Agent Communication*, pages 31–45, 2000.

[Stevenson, 2010] Angus Stevenson. *Oxford Dictionary of English*. Oxford University Press, 2010.

[Tenorth *et al.*, 2010] M. Tenorth, D. Nyga, and M. Beetz. Understanding and Executing Instructions for Everyday Manipulation Tasks from the World Wide Web. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 1486–1491, 2010.

[Turney and Pantel, 2010] Peter D. Turney and Patrick Pantel. From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188, January 2010.