From Learning Through Signal Processing to Argumentation on Ontological Representations

Kemo Adrian

IIIA, Artificial Intelligence Research Institute
CSIC, Spanish Council for Scientific Research,
Campus UAB, 08193 Bellaterra, Catalonia (Spain) kemo.adrian@iiia.csic.es

Abstract. Communication systems associate symbols to meaning by a process referred to as the symbol grounding. The symbolic nature of classical artificial intelligence guarantees the grounding to be explicit, unlike the connectivist approach of deep learning. This is a problem when two systems without explicit grounding try to discuss their respective meanings, as human do when they use the reflexive function of language[1]. This publication presents the baseline of a two-layer model that aims to explicit semantic knowledge obtained through deep learning and allow argumentation over it.

1 Introduction

Communication systems need to associate the symbols of their vocabulary to the inputs they have of reality. Achieving such association is known as the symbol grounding problem in the theory of meaning developed by Harnard [4]. Research has been done to show that two systems that want to communicate must share their groundings [6][2]. If this sharing is not originally present, the systems should exchange information about their groundings. Existing models show that this exchange can be an argumentation that leads to building a consistent common vocabulary [7].

In previous work, we developed a model of argumentation that allows systems to reach a state of mutual intelligibility in simple scenarios [1]. The groundings are explicit relations between inputs that are examples from data-sets using the Ψ-term formalism [3], and outputs that are labels and take the role of symbols. Ψ-terms, also called feature-terms or feature-structures, are a generalization of first-order terms and have a similar expressive power as description logic. The groundings are also represented by Ψ-terms, providing an ontological structure to the knowledge representations that a system has of its groundings. This allows the systems to hold argumentation over the meaning.

[1] The metalingual or reflexive function of language is one of the six functions of language listed by Jakobson [5]. It is the ability of language to discuss and describe itself.

[2] This does not mean that the grounding should be identical in both systems; however, the systems should be aware of the dissimilarities in their groundings, which means that they share this information.

The work presented in this paper aims to extend this initial model to scenarios where the grounding is implicit. Where the argumentation model was using inductive learning to ground symbols, the two-layer system presented in this paper aims to accelerate the grounding process by using artificial neural networks. Using hyper planes to classify over a vocabulary's symbols gives directly a set of grounded symbols as an output of the learning. We call this set a contrast set (see section 5). The next step will be to test this architecture, before replacing the single layer neural networks by deep-learning techniques.

2 Scope and Related Work

The fundamental aspect of the presented system is its ability to generate an ontological representation[3] from the result of an opaque learning, by a process of reification. It is important to note that we do not understand "features" as the sensory primitives learned by certain types of neural networks (convolutional neural network for example).

There exists a wide range of publications on rule extraction algorithms for neural networks [2]. A candidate to replace the first level of our system seemed to be the CRED algorithm [8], since it already has an extension for deep learning, DeepRED [9]. However, this algorithm does not directly provide information on the semantic relations between the different outputs of the network: holonyms, meronyms, hyponyms, hypernyms etc. This feature is a key feature of our approach.

3 A Two Layer Model

The first layer of our model is a parallelized set of linear classifiers that share the same inputs represented as a vector $I = i_1, ..., i_m$. Each one of the k classifiers regroups a vector of outputs $O_k = o_1, ..., o_n$ and its associated matrix of weights $W_k = w_{1,1}, ...w_{m,n}$ that links inputs to outputs. This first layer will be used for feature extraction after performing a supervised learning, as we explain later in Section 4.

The second layer contains Ψ-terms divided into two classes: the examples and the rules. An example e is a Ψ-term; e possess a feature *label* that has a value l. A rule r_l is a classification rule (as is common in inductive concept learning) represented by a pair $< l, \Psi >$ containing a Ψ-term and a label l. This level is used to hold the argumentation over the vocabulary.

In each layer, the categories can be defined in two different ways. The first definition is extensional: it is the assignment of the same expected output $o \in O_l$ to a set of inputs at the first level, and the assignment of the same value l to the feature *label* of a set of examples at the second level. The second is intensional: it lies in the weights that associate a set of inputs to an output at the first level, and in the rules that associate a set of examples to a label at the second level.

[3] Understood as an explicit and organized model of knowledge

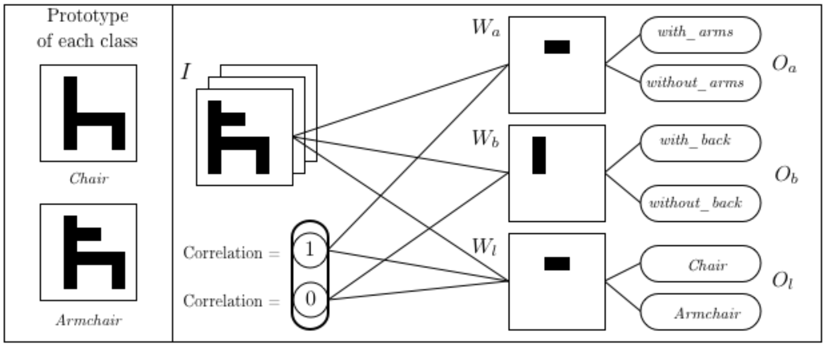

Fig. 1. First level of the system allowing the detection of correlating weights for Ψ-term creation. The input prototypes are listed on the left. The first layer's connections are represented on the right.

A Ψ-term can be associated to an ontological representation. They can be visualized as tree structures where the edges are description logic features, and the nodes are terms that can point to symbols. The tree originates from a term known as the root. If a person had to be represented as a Ψ-term, the root's symbol would be a social security number and have the features *name*, *surname*, and *mother*. The last feature would point to a term of which the symbol is a social security number, having the four same features as the root.

4 Learning Ontological Representations Through Classifiers

The second level of our model needs to create the Ψ-terms of examples and rules used in communication from the input, weights and outputs of the first level. Thus, the first level has to learn its classifications through supervised learning before the creation process. The supervised learning is achieved by presenting a sequence of inputs along with a sequence of sets of k expected outputs (k being the number of classifiers), and using a learning rule to modify the weights of the matrices $W_1 ... W_k$ until the difference between the expected and obtained outputs is almost null. This rule can be the Hebb's rule, the Widrow-Hoff rule, or any other rule for a single layer neural network.

An example of this, illustrated in Figure 1, is a set of images of seats presented as input to three different classifiers. A first classifier k_a learns to recognize the features of the seats: if a seat has arms, a second one k_b if a seat has a back, and the last one k_l which label should be used for this seat. The first classifier has its possible outputs $O_a = \{\mathit{with\_\ arms}, \mathit{without\_\ arm}\}$, the second has $O_b = \{\mathit{with\_\ back}, \mathit{without\_\ back}\}$ and the third has $O_l = \{\mathit{chair}, \mathit{armchair}\}$.

We associate each classifier to a name that refers to the feature they represent. In our example, k_a is named *arm*, k_b is named *back* and k_l is named *label.* In order to create an example which corresponds to a presented input, we create a unique identifier for this input. This identifier will be the root's value of the example's Ψ-term. A feature is added to the root for each classifier, and their observed outputs are the symbols of the classifier's terms.

In the case of a rule, we select a label $l \in O_l$ and take all the matrices of weights in the classifiers other than *label*. Then, for each classifier k, we separate the matrices by output: having the set of matrices $w_{1,o}, ..., w_{n,o}$ for each output o, where n is the number of outputs in k. The matrices have the same dimension as I. A Ψ-term ψ is created with no symbol associated to its root, meaning that the terms can take any value. Then, for each $k \in \{arm, back\}$, the matrix $w_{i,o} \in W_k$ that correlates the most with w_l is selected, and o is set as the value of k's term. We create the rule $< l, \psi >$. This rule generalizes all the examples that have l for label, and only them.

5 Argumentation over the Grounding Using Classifiers

The second layer of our model corresponds to the argumentation model presented in a separate paper [1]. It defines concepts as a triadic relation between the symbol, the intensional definition (a set of rules) and the extensional definition (a set of examples). The aim of the agents is to have concepts that partition their whole sets of examples. Such a partition is called a contrast set.

In order to reach the state of mutual intelligibility mentioned in Section 1, systems can modify their partition. They do so by exchanging their intensional definitions and arguing on them. In the current system, the first layer does not have an explicit ontological representation corresponding to the intensional definition that can be exchanged. However, Section 4 shows how to create these explicit representations. If an explicit relation is obtained from another system, it can be used to recreate a new partition of the examples and their corresponding inputs. A new classification is learned over this partition, and then converted into an ontological representation.

6 Conclusion and future work

We presented the structure of a model that will be tested in the near future. This model will use an argumentation model that we improved since its publication. However, these improvements do not impact the integration of the model in the two-layer system. Currently, only the root term has features. We are investigating methods to extend this property to other Ψ-terms. A solution might be to use the matrices of weights as a tool to segment the inputs and extract features. In the case of seats, the first layer would be able to segment the backs of seats. Coupled with a classifier that recognizes colors, the weights that link the inputs to a color correlating with the weights that link the inputs to the back segment would associates a *color* feature to the term of the *back* feature. Finally, we plan to investigate the possibility to use our model with artificial neural networks that have one or more hidden layers.

References

1. Adrian, K., Plaza, E.: Agent-based agreement over concept meaning using contrast sets. In: CCIA. pp. 19–28 (2015)
2. Andrews, R., Diederich, J., Tickle, A.B.: Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-based systems 8(6), 373–389 (1995)
3. Carpenter, B.: The logic of typed feature structures: with applications to unification grammars, logic programs and constraint resolution, vol. 32. Cambridge University Press (2005)
4. Harnad, S.: The symbol grounding problem. Physica D: Nonlinear Phenomena 42(1-3), 335–346 (1990)
5. Jakobson, R.: Linguistics and poetics. In: Style in language, pp. 350–377. MA: MIT Press (1960)
6. Manzano, S., Ontanón, S., Plaza, E.: A case-based approach to mutual adaptation of taxonomic ontologies. In: International Conference on Case-Based Reasoning. pp. 226–240. Springer (2012)
7. Ontanón, S., Plaza, E.: Concept convergence in empirical domains. In: International Conference on Discovery Science. pp. 281–295. Springer (2010)
8. Sato, M., Tsukimoto, H.: Rule extraction from neural networks via decision tree induction. In: Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on Neural Networks. vol. 3, pp. 1870–1875. IEEE (2001)
9. Zilke, J.R., Loza Mencía, E., Janssen, F.: Deepred – rule extraction from deep neural networks. In: Calders, T., Ceci, M., Malerba, D. (eds.) Discovery Science: 19th International Conference, DS 2016, Bari, Italy, October 19–21, 2016, Proceedings. pp. 457–473. Springer, Cham (2016)