Logic Tensor Networks for Semantic Image Interpretation*

Ivan Donadello
Fondazione Bruno Kessler and
University of Trento
Trento, Italy

Luciano Serafini
Fondazione Bruno Kessler
Trento, Italy
ciao

Artur d’Avila Garces
City University,
London, UK

Abstract

Semantic Image Interpretation (SII) is the task of extracting structured semantic descriptions from images. It is widely agreed that the combined use of visual data and background knowledge is of great importance for SII. Recently, statistical relational learning (SRL) approaches have been developed to deal with reasoning and learning in the presence of rich data and knowledge under uncertainty. Logic Tensor Networks (LTNs) is an SRL framework which integrates neural networks with first-order (fuzzy) logic in an attempt to allow (i) efficient learning from noisy data in the presence of logical constraints and (ii) reasoning with logical formulas describing properties of the data. In this paper, we develop and apply LTNs to two of the main tasks of SII, namely, classification of bounding boxes and the detection of part-of relations between bounding boxes. To the best of our knowledge, this is the first application of SRL to such SII tasks. The proposed approach is evaluated on a standard image processing benchmark. Experiments show that the use of background knowledge in the form of logical axioms improves the performance of the state-of-the-art data-driven approaches in both tasks. Moreover, we show that the use of background logical knowledge adds robustness to the learning system in the presence of noisy and erroneous training data.

1 Introduction

Semantic Image Interpretation (SII) is the task of generating a structured semantic description of the content of an image. This structured semantic description can be represented as a labelled directed graph, where each vertex corresponds to a bounding box around an object in the image and edges represent semantic relations between pairs of objects. Vertices are labelled with a set of object types, whereas edges are labelled with a set of relational labels. Such a graph is also called scene graph in [Krishna et al., 2016].

The major obstacle that need to be overcome in SII is the so-called semantic gap [Neumann and Möller, 2008], that is the lack of a direct correlation between low-level features of the image and high-level semantic concepts. To solve this problem a SII system should learn the latent correlations between numerical features that can be observed in an image and semantic properties of objects. In this task the availability of background logical knowledge about semantic relations is in general of great help. Thus, recent SII systems have tried to combine, or even integrate, visual features obtained from data with symbolic knowledge in the form of logical axioms [Zhu et al., 2014; Chen et al., 2012; Donadello and Serafini, 2016].

The area of Statistical Relational Learning (SRL), or statistical AI (StarAI), seeks to combine learning from data with symbolic knowledge in the presence of uncertainty [Wang and Domingos, 2008; Bröcheler et al., 2012; Gutmann et al., 2010; Diligenti et al., 2015; Rocktaschel et al., 2015; Ravkic et al., 2015]. However, only very few SRL systems have been applied to SII (c.f. Section 2) due to the high computational complexity associated with such image learning tasks. Most systems for solving SII tasks have been based, instead, on deep learning and neural network models. These, on the other hand, do not in general offer a well-founded way of learning from data in the presence of relational logical constraints, requiring the neural models to be highly engineered. In this paper, we develop and apply, for the first time, the SRL framework called Logic Tensor Networks (LTNs) to computationally challenging SII tasks. LTNs combine learning in deep networks with relational logical constraints [Serafini and d’Avila Garcez, 2016]. It uses a First Order Logic (FOL) syntax interpreted in the real numbers, which is implemented as a deep tensor network. Logical terms are interpreted as feature vectors in a real n-dimensional space; function symbols are interpreted as real functions, and predicate symbols as fuzzy logic relations. This syntax and semantics, called real semantics, allow LTNs to learn efficiently in hybrid domains, where elements are composed of both numerical and relational information. We argue, therefore, that LTNs are a good candidate for learning SII because they can express relational knowledge in FOL which serves as constraints on the data-driven learning within tensor networks. Being LTN a logic, it naturally provides a notion of logical consequence, which forms the basis for learning in LTNs, which is defined

*To appear in Proc. of IJCAI 2017
as best satisfiability, c.f. Section 3. Solving the best satisfiability problem amounts to finding the latent correlations existing between a relational background knowledge and continuous data attributes. This formulation enables the specification of learning as reasoning, a unique characteristic of LTNs, which is nevertheless seen as highly relevant for SII. This paper specifies SII within LTNs, evaluating it on two important tasks of SII: (i) the classification of bounding boxes, and (ii) the detection of the part-of relation between any two bounding boxes. Both tasks are tested in the context of the PASCAL-PART dataset [Chen et al., 2014]. It is shown that LTNs outperform the state-of-the-art object classifier Fast R-CNN [Girshick, 2015] on the bounding box classification task. LTNs also outperform a rule-based heuristic for the detection of part-of relations between bounding boxes, which uses the inclusion ratio of two bounding boxes. Finally, LTNs are evaluated on their ability to handle uncertainty, in the form of errors (i.e. noise) in the dataset, including errors in the training data classification labels. Very large visual recognition datasets now exist which are noisy [Reed et al., 2014]. Thus, it becomes important to have a learning system that is robust to noise. LTNs were trained systematically on noisy datasets, with results, on both SII tasks, showing that the LTN’s logical constraints are capable of adding robustness to the system, dealing well with the presence of noisy training data. The paper is organized as follows: in Section 2 we contrast LTNs with the systems used currently for SII which integrate visual features and background knowledge. Section 3 specifies LTNs in the context of SII. Section 4 defines the best satisfiability problem in this context, which enables the use of LTNs. Section 5 describes in detail the comparative evaluation of LTNs on the SII tasks. Section 6 concludes the paper and discusses directions for future work.

2 Related Work

The idea of exploiting logical background knowledge to improve SII tasks dates back to the early days of AI. In what follows, we review the most recent results in the area in comparison with LTNs.

Logic-based approaches have used expressive Description Logics (DL) to use a bottom-up approach where the basic components of the scene are all already discovered (e.g. simple object types or spatial relations). Then, with logical reasoning they derive new facts in the scene from these basic components [Neumann and Möller, 2008; Peraldi et al., 2009]. Other logic-based approaches have used fuzzy DL to tackle uncertainty in the basic components [Hudelot et al., 2008; Dasiopoulou et al., 2009; Atif et al., 2014]. These methods have limited themselves to spatial relations or to refining the labels of the detected objects. LTNs can handle many types of semantic relations. In [Donadello and Serafini, 2016], the scene interpretation is derived by combining image features with constraints described using DL, but the method is tailored to the part-of relation and cannot be extended easily to account for other relations. In [Marszalek and Schmid, 2007; Forestier et al., 2013], a symbolic Knowledge-base is used to improve object detection, but only the subsumption relation between labels is explored, and it is not possible to inject complex knowledge expressed using logical axioms.

A second group of approaches seek to encode background knowledge and visual features within probabilistic graphical models. In [Zhu et al., 2014; Nyga et al., 2014], visual features are integrated with knowledge (gathered from datasets, Web resources or annotators) about object labels, properties (e.g. shape, colour, size) and affordances (what an object is useful for) using Markov Logic Network (MLN) [Richardson and Domingos, 2006]. MLNs, are used here to predict facts from new unseen images, but the inference performed in this application is limited in expressiveness: formulas are Horn clauses with a single literal in the body. As before, it is not easy to see how this approach would be extended to more complex axioms. Other related work [Chen et al., 2012; Kulkarni et al., 2011] encode background knowledge in a generic Conditional Random Field (CRF), where the nodes represent detected objects and the edges represent logical relationships between them. The task is to find a correct labelling for this graph. In [Chen et al., 2012], the edges encode logical constraints on a DL knowledge base. Although this work is very close in spirit to the approach presented in this paper, it is not formalised as LTNs are (also, LTNs do not use CRFs in principle). As a result, the theory behind the potential functions defined in the CRF is unclear. In [Kulkarni et al., 2011], potential functions are defined as text priors such as co-occurrence of terms found in the image descriptions of Flickr (e.g. object labels with property labels or relationship labels).

In a final group of approaches, here called language-priors, background knowledge is taken from linguistic models [Ramanathan et al., 2015; Lu et al., 2016]. In [Ramanathan et al., 2015], a neural network integrating visual features and a linguistic model is built to predict semantic relationships between bounding boxes. The linguistic model is a set of rules derived from WORDNET [Fellbaum, 1998], stating which type of semantic relationship occurs between a subject and an object. In [Lu et al., 2016] a similar neural network is proposed for the very same task but with a more sophisticated language model. The model embeds in the same vector space triples of subject-relation-object terms, such that semantically similar triples are mapped closely together in the embedding space. In this manner, even if there are no examples of some triples, they can be inferred from similarity to more frequent triples. The drawback here is the possibility of extracting inconsistent triples (such as man-eat-chair) due to the embeddings. LTNs avoid these inconsistencies with its logic-based approach, so that an axiom to the effect that chairs are not normally edible would suffice. Notice how LTNs, at the same time, can account for exceptions as it offers a system capable of dealing with crisp axioms and real-valued data, as specified in what follows.

3 Logic Tensor Networks

Let $\mathcal{L}$ be a first order language, whose signature is composed of three disjoint sets $C$, $F$, and $P$ for constant, functional, and predicate symbols, respectively. For any functional/predicate symbol $s$, $\alpha(s)$ is its arity. $\mathcal{L}$-formulas allow us to specify
reational knowledge. E.g. the atomic formula partOf(o₁, o₂) states that object o₁ is part of object o₂: ∀x∀y(partOf(x, y) → ¬partOf(y, x)) states that the relation partOf is asymmetric; ∀x∀y(partOf(x, y) ∧ Tail(y)) states that every cat has a part which is a tail. As mentioned, exceptions can be dealt with by interpreting the above rule as normally, every cat has a tail in presence of an example of a tailless cat, as exemplified later.

As for the semantics of L, we define the interpretation domain as a subset of ℝⁿ, i.e. every object in the domain is associated with an n-dimensional vector of real numbers. Intuitively, this n-tuple represents n numeric features of the object, e.g. in the case of a person: his name in ASCII, height, weight, social security number, etc. Functions are interpreted as real functions, and predicates are interpreted as fuzzy relations on real vectors. To emphasise the fact that we interpret symbols as real numbers, we prefer to use the term grounding in place of interpretation. This leads to the following definition of semantics.

**Definition 1** Let n ∈ ℕ. An n-grounding, or simply grounding, G for a FOL L is a function defined on the signature of L satisfying the following conditions:

1. G(c) ∈ ℝⁿ for every constant symbol c ∈ C;
2. G(f) ∈ ℝⁿ×n(f) → ℝⁿ for every f ∈ F;
3. G(P) ∈ ℝⁿ×n(P) → [0, 1] for every P ∈ P.

Given a grounding G, the semantics of closed terms and atomic formulas is defined as follows:

\[ G(f(t₁, \ldots, tₘ)) = G(f)(G(t₁), \ldots, G(tₘ)) \]
\[ G(P(t₁, \ldots, tₘ)) = G(P)(G(t₁), \ldots, G(tₘ)) \]

The semantics for connectives is defined according to fuzzy logic semantics. For instance if µ is the Lukasiewicz t-norm

\[ G(\neg φ) = 1 - G(φ) \]
\[ G(φ ∧ ψ) = \max(0, G(φ) + G(ψ) - 1) \]
\[ G(φ ∨ ψ) = \min(1, G(φ) + G(ψ)) \]
\[ G(φ → ψ) = \min(1, 1 - G(φ) + G(ψ)) \]

Fuzzy semantics for ∀ is defined in [Serafini and d’Avila Garcez, 2016] using the max operator, i.e. G(∀xφ(x)) = \min_{t∈term(L)} G(φ(t)) where term(L) is the set of ground terms of L. This, however, is inadequate for our purpose, as it barely tolerates exceptions. Indeed, the presence of a single exception to the universally quantified rule (e.g. a cat without a tail) would falsify a rule such as ∀x(partOf(x) → Φ). Therefore, our intention is that

\[ G(∀xφ(x)) = \lim_{T→term(L)} mean_p(G(φ(t)) \mid t ∈ T) \]

where \( mean_p(x₁, \ldots, xₖ) = \left( \frac{1}{p} \sum_{i=1}^{p} x_i^p \right)^{\frac{1}{p}} \) for some p ∈ ℤ⁺.

Finally, the classical semantics of ∃ is uniquely determined by the semantics of ∀, by making ∃ equivalent to ¬∀¬. This approach, however, has a drawback too when it comes to SII: if we adopt, for instance, the arithmetic mean for the semantic of ∃ then

\[ G(∃xφ(x)) = G(∀xφ(x)) \]

Therefore, we shall interpret existential quantification via Skolemization: every formula of the form ∀x₁, …, xₙ(∀yφ(x₁, …, xₙ, y)) is rewritten as ∀x₁, …, xₙ(φ(x₁, …, xₙ, f(x₁, …, xₙ))), by introducing a new m-ary function symbol, called Skolem function. Existential quantifiers can be eliminated in the usual way from the language by introducing Skolem functions.

**Formalizing SII in LTNs**

To specify the SII problem, as defined in the introduction, we consider a signature Σ_{SII} = ⟨C, F, P⟩, where C = ∪_{p ∈ P_{cats} b(p)} is the set of ids for all the bounding boxes of all the pictures, F = ∅ and P contain a set P₁ of unary predicates, one for every object class, e.g. P₁ = {Dog, Cat, Tail, Muzzle, Train, Coach, …}, and a set P₂ of binary predicates representing relations between objects. Since in our experiments we focus on the part-of relation, P₂ = {partOf}. Formulas of the FOL L-based language on this signature can specify simple facts, e.g. the fact that bounding box b contains a cat, written Cat(b), the fact that b contains either a cat or a dog, written Cat(b) ∨ Dog(b), and general rules such as ∀x(partOf(x) → Φ).

A grounding for Σ_{SII} can be defined as follows: for each constant b, denoting a bounding box, can be associated with a set of geometric features, and a set of semantic features obtained from the output of a bounding box detector. In more detail, each bounding box is associated with geometric features describing the position and the dimension of the bounding box, and semantic features, one for each class, describing the classification value returned by the bounding box detector for each class. For example, for each bounding box b ∈ C, G(b) is the ℝ¹⁺∥P₁ real vector:

\( (class(C₁, b), \ldots, class(C₁P₁), b, x₀(b), y₀(b), x₁(b), y₁(b)) \)

where the last four features are the coordinates of the top-left and bottom-right corner of b and class(C₁, b) ∈ [0, 1] is the classification value obtained from the bounding box detector for b, C₁ ∈ P₁.

An example of groundings for predicates can be defined by taking a one-vs-all multi-classifier perspective, one can define the following grounding for each class C₁ ∈ P₁ (below, the more elements of the domain satisfy φ(x), the higher the truth-value of ∀xφ(x). To capture this, we base the semantics of ∀ on a mean-operator, as follows:

\[ G(∀xφ(x)) = \lim_{T→term(L)} mean_p(G(φ(t)) \mid t ∈ T) \]

where \( mean_p(x₁, \ldots, xₖ) = \left( \frac{1}{p} \sum_{i=1}^{p} x_i^p \right)^{\frac{1}{p}} \) for some p ∈ ℤ⁺.

The most popular mean operators, such as arithmetic, geometric and harmonic means can be obtained by setting p = 1, p = 2 and p = −1, respectively.
\( \mathbf{x} = \langle x_1, \ldots, x_{|P|} \rangle \) is the real vector corresponding to the grounding of a bounding box):

\[
G(C_i)(\mathbf{x}) = \begin{cases} 
1 & \text{if } i = \text{argmax}_{1 \leq l \leq |P|} x_l \\
0 & \text{otherwise}
\end{cases}
\] (1)

A simple rule-based approach for defining the grounding for partOf relations, is based on the naïve assumption that the more that a bounding box \( b \) is contained within a bounding box \( b' \), the higher the probability that \( b \) is a part of \( b' \). Accordingly, one could define \( G(\text{partOf}(b, b')) \) as the inclusion ratio \( ir(b, b') = \frac{\text{area}(b)}{\text{area}(b')} \). A slightly more sophisticated rule-based grounding for partOf (used as baseline in the experiments to follow) takes into account also type compatibilities by multiplying the inclusion ratio by a factor \( w_{ij} \). Hence, we define \( G(\text{partOf}(b, b')) \) as follows:

\[
\begin{cases} 
1 & \text{if } ir(b, b') \cdot \max_{j=1}^{|P|} (w_{ij} \cdot x_i \cdot x'_j) \geq th_{ir} \\
0 & \text{otherwise}
\end{cases}
\] (2)

for some threshold \( th_{ir} \) (we use \( th_{ir} > 0.5 \)) and with \( w_{ij} = 1 \) if \( C_i \) is a part of \( C_j \), and 0 otherwise. Given the above definition, we can compute the grounding of any atomic formula, e.g. \( \text{cat}(b_1), \text{dog}(b_2), \text{leg}(b_3), \text{partOf}(b_1, b_2), \text{partOf}(b_3, b_2) \), expressing the degree of truth of the formula.

The rule-based groundings (1) and (2) might not satisfy some of the constraints that one would like to impose; for instance, the classifier might be wrong, or it could be that a bounding box included in another is not in part-of relation. Furthermore, in many situations, it is not possible to define the grounding a priori, which, instead should be automatically learned, from examples, by optimizing the truth value of the formulas of a background knowledge about the specific domain.

4 Learning as Best Satisfiability

A partial grounding, denoted by \( \hat{G} \), is a grounding that is defined on a subset of the signature of \( L \). A grounding \( G \) is said to be a completion of \( \hat{G} \), if \( G \) is a grounding for \( L \) and coincides with \( \hat{G} \) on the symbols where \( \hat{G} \) is defined.

**Definition 2** A grounded theory \( \langle K, \hat{G} \rangle \) is a pair of a set \( K \) of closed formulas and a partial grounding \( \hat{G} \).

**Definition 3** A grounding \( G \) satisfies a GT \( \langle K, \hat{G} \rangle \) if \( G \) completes \( \hat{G} \) and \( G(\phi) = 1 \) for all \( \phi \in K \). A GT \( \langle K, \hat{G} \rangle \) is satisfiable if there exists a grounding \( G \) that satisfies \( \langle K, \hat{G} \rangle \).

According to the previous definition, deciding the satisfiability of \( \langle K, \hat{G} \rangle \) amounts at searching for an extension of \( \hat{G} \) such that all the formulas of \( K \) have value 1. Differently from the classical (crisp) satisfiability, when a GT is not satisfiable, we are interested in the best level of satisfaction that we can reach with a grounding. This is defined as follows.

**Definition 4** Let \( \langle K, \hat{G} \rangle \) be a grounded theory. We define the best satisfiability problem as the problem of finding an extensions \( G^* \) of \( \hat{G} \) that maximizes the truth values of the conjunction of all clauses \( c_l \in K \), i.e.,

\[
G^* = \text{argmax}_{G \subseteq \hat{G}} G(\land_{c_l \in K} c_l)
\]

Groundings capture the latent correlation between the quantitative attribute of objects and their categorical and relational properties. Not all the functions are suited for a grounding; they should present some form of regularity. If \( G(\text{Cat})(\mathbf{x}) \approx 1 \) (the bounding box with feature vector \( \mathbf{x} \) contains a cat) then for every \( \mathbf{x}' \) close enough to \( \mathbf{x} \) (for every bounding box with features similar to \( \mathbf{x} \)) \( G(\text{Cat})(\mathbf{x}') \approx 1 \). In this paper we consider groundings of a specific form, but the approach could be applied to any family of functions which can be characterised by a set \( \Omega \) of parameters.

Function symbols are grounded to linear transformations. If \( f \) is a \( m \)-ary function symbol, then \( G(f) \) is of the form:

\[
G(f)(\mathbf{v}) = M_f \mathbf{v} + N_f
\]

for some \( n \times mn \) real matrix \( M_f \) and \( n \)-vector \( N_f \), where \( \mathbf{v} = [v_1, \ldots, v_n]^\top \) is the \( mn \)-ary vector obtained by concatenating each \( v_i \).

The grounding of an \( m \)-ary predicate \( P \), namely \( G(P) \), is defined as a generalization of the neural tensor network (which has been shown effective at knowledge completion in the presence of simple logical constraints [Socher et al., 2013]), as a function from \( \mathbb{R}^{mn} \) to \([0, 1]\), as follows:

\[
G(P)(\mathbf{v}) = \sigma (u^\top P^{[1:k]} \mathbf{v} + V_P \mathbf{v} + b_P)
\]

where \( P^{[1:k]} \) is a 3-D tensor in \( \mathbb{R}^{mn \times mn \times k} \), \( V_P \) is a matrix in \( \mathbb{R}^{k \times mn} \), \( b_P \) is a vector in \( \mathbb{R}^k \) and \( \sigma \) is the sigmoid function. With this encoding, the grounding (i.e. truth-value) of a clause can be determined by a neural network which first computes the grounding of the literals contained in the clause, and then combines them using the specific t-norm.

In the following we describe how we build a suitable GT for SII. Let \( \text{Pics}^\text{expl} \subseteq \text{Pics} \) be a set of bounding boxes of pictures that are correctly labelled with the classes they belong to and each pair of bounding boxes is correctly labelled with the part-of relation. In machine learning terminology \( \text{Pics}^\text{expl} \) is a training set without noise. In real logic, a training set can be represented with a grounded theory \( T_{\text{expl}} = (\mathcal{K}_{\text{expl}}, \hat{G}_{\text{expl}}) \), where: \( \mathcal{K}_{\text{expl}} \) contains the set of closed literals \( C_i(b) \), (resp. \( \neg C_i(b) \)) and \( \text{partOf}(b, b') \) (resp. \( \neg \text{partOf}(b, b') \)), for every bounding box \( b \) labelled (resp. not labelled) with \( C_i \) and for every pair of bounding boxes \( (b, b') \) connected (resp. not connected) by the \( \text{partOf} \) relation. The partial grounding \( \hat{G}_{\text{expl}} \) is defined on all bounding boxes of all the pictures in \( \text{Pics} \) where both the semantic features \( \text{class}(C_i, b) \) and the bounding box coordinates are computed with the Fast R-CNN object detector [Girshick, 2015]. \( \hat{G}_{\text{expl}} \) is not defined for the predicate symbols in \( P \) and needs to be learned. \( T_{\text{expl}} \) contains only assertional information about specific bounding boxes. This is the classical setting of machine learning where classifiers (i.e., the grounding of predicates) are inductively learned from positive and negative examples of a classification. In this learning setting mereological constraints like “cats have no wheels” or “the tail is part of cats” are not
We evaluate the performance of our approach to SII on two tasks, namely: the classification of bounding boxes and the detection of relationships between pairs of bounding boxes. In particular, for this evaluation, we concentrate on the part-of relation. We choose part-of relation because state-of-the-art provides both datasets (i.e., PASCAL-P ART [Chen et al., 2014]), and ontologies (WORDNET) on part-of relations. In addition, the part-of relation can be used to represent, via reification, a large class of relations [Guarino and Giuzzardi, 2016] (E.g., the relation “a plant is lying on the table” can be reified in an object of type “lying event” whose parts are the plant and the table).

We also evaluate the robustness of our approach to SII with respect to, noisy and erroneous training data. Indeed, with the growing size and complexity of training sets for visual recognition [Krishna et al., 2016], some data annotations may be affected by noise, such as missing labels, not localized objects or disagreement among human annotators (E.g., “part-of” is often mistaken with “have”) [Reed et al., 2014].

We use the PASCAL-P ART-dataset that contains 10103 images with bounding boxes annotated with object types and the part-of relation between pairs of bounding boxes. Labels are divided into three main groups: animals, vehicles, indoor objects with their corresponding parts along with the “part-of” label. Whole objects inside the same group can share parts, whole objects of different groups do not share any part. Labels for parts are very specific, e.g., “left lower leg”, and, without loosing generality, we merge the bounding boxes that refer to the same part in a unique bounding box. E.g., bounding boxes labelled with “left lower leg” and “left upper leg” are merged in a single bounding box of type “leg”. In this way we have 20 labels for whole objects and 39 for parts. In addition, the dataset is cleaned from the bounding boxes with height or weight smaller than 6 pixels. The resulting pictures are split into a training set and a test set with a proportion of 80% and 20%, respectively. We maintain the same proportion on the number of bounding boxes for each label.

5 Experimental Evaluation

We evaluate the performance of our approach to SII on two tasks, namely: the classification of bounding boxes and the detection of relationships between pairs of bounding boxes. In particular, for this evaluation, we concentrate on the part-of relation. We choose part-of relation because state-of-the-art provides both datasets (i.e., PASCAL-P ART-dataset [Chen et al., 2014]), and ontologies (WORDNET) on part-of relations. In addition, the part-of relation can be used to represent, via reification, a large class of relations [Guarino and Giuzzardi, 2016] (E.g., the relation “a plant is lying on the table” can be reified in an object of type “lying event” whose parts are the plant and the table).

We also evaluate the robustness of our approach to SII with respect to, noisy and erroneous training data. Indeed, with the growing size and complexity of training sets for visual recognition [Krishna et al., 2016], some data annotations may be affected by noise, such as missing labels, not localized objects or disagreement among human annotators (E.g., “part-of” is often mistaken with “have”) [Reed et al., 2014].

We use the PASCAL-P ART-dataset that contains 10103 images with bounding boxes annotated with object types and the part-of relation between pairs of bounding boxes. Labels are divided into three main groups: animals, vehicles, indoor objects with their corresponding parts along with the “part-of” label. Whole objects inside the same group can share parts, whole objects of different groups do not share any part. Labels for parts are very specific, e.g., “left lower leg”, and, without loosing generality, we merge the bounding boxes that refer to the same part in a unique bounding box. E.g., bounding boxes labelled with “left lower leg” and “left upper leg” are merged in a single bounding box of type “leg”. In this way we have 20 labels for whole objects and 39 for parts. In addition, the dataset is cleaned from the bounding boxes with height or weight smaller than 6 pixels. The resulting pictures are split into a training set and a test set with a proportion of 80% and 20%, respectively. We maintain the same proportion on the number of bounding boxes for each label.

Object Classification and Part-Of Detection The task of object classification is defined as follows, given a set of bounding boxes detected by an object detector (we use FastRCNN [Girshick, 2015]) classify each bounding box according to a class of objects. The task of Part-Of detection is defined as follows: given two bounding boxes, decide if the object contained in the first is part of the object contained in the second. We use LTN to simultaneously resolve both tasks. This is important because bounding box types and part-of relations are not independent. Their dependencies are specified in the background knowledge by logical axioms.

To show the effect of logical axioms, we separately train two grounded theories: the former contains only positive and negative training examples for types and part-of relation ($T_{expl}$), the latter extends the former with logical axioms about part-of and types ($T_{prior}$). Corresponding LTNs have

We implemented LTN with Google’s TENSORFLOW$^TM$. Code, part-of ontology and the dataset are available at https://goo.gl/zEiLLx.
been set up with tensor of $k = 6$ layers and a regularization parameter $\lambda = 10^{-10}$. We choose the Łukasiewicz T-norm ($\mu(a, b) = \max(0, a + b - 1)$) and use the harmonic mean as aggregation operator. We run 1,000 training epochs of the RMSProp optimisation algorithm available in TENSORFLOW™. We compare our results with two baselines $G_{FRCNN}$, $G_{RBPOF}$ according to the bounding box classification and part-of detection tasks respectively. The baseline grounding $G_{FRCNN}$ implements Equation (1) where the underlying bounding box classifier is Fast R-CNN. The baseline grounding $G_{RBPOF}$ is based on the inclusion ratio $ir$ between two bounding boxes: if $ir$ is bigger than a given threshold (0.7) then the bounding boxes are in partOf relation. We frame these classification tasks as multiclass and multilabel classification problems: given a grounding $G_x \in \{G_{FRCNN}, G_{RBPOF}, G_{expl}, G_{prior}\}$ and a threshold $th$, every bounding box $b$ can be classified with many labels in $P_1$ if $G_x(C(b)) \geq th$, with $C \in P_1$. This also holds for the partOf binary predicate. Results for indoor objects are shown in Figure 1 (AUC is the area under the precision-recall curve). The results show that, for both object types and part-of predicate, LTN with prior knowledge (given by mereological axioms) has better performance than LTN with only training examples. Moreover, the prior knowledge leads LTN to overcome the performance of the well-known Fast R-CNN object detector on the bounding box classification task. Regarding other object types (animals and vehicles) $LTN_{prior}$ has results comparable with $G_{FRCNN}$.

Robustness to Noisy Training Data In this evaluation we want to show that logical axioms improve the robustness of the system in presence of noise in the labels of training data. We artificially add an increasing amount of noise to the PASCAL-PART-dataset training data, and we measure the degradation of the performance, in presence or in absence of axioms. For $k \in \{10, 20, 30, 40\}$, we randomly select the $k\%$ of bounding boxes (training data) and randomly change their classification labels. In addition, we randomly select the $k\%$ of pairs of bounding boxes and we flip the part-of relation. Thus, for both SII tasks we add noise to both positive and negative training examples. Then, for every $k$ we train $T_{expl}^k$ and $T_{prior}^k$. We finally evaluate the obtained grounding on both SII tasks. We empirically notice that adding too much noise leads to a big drop of performance (we are adding errors to both positive and negative examples) thus the maximum percentage of errors is 40%. Figure 2 shows the degradation of the AUC measure for indoor objects (we use the same evaluation of the previous task) according to an increasing percentage of noisy labels. Every pair of bars represents the results for the grounded theories $T_{prior}^k$, $T_{expl}^k$ for a given $k$ percent of errors. We can see a decrease of performance due to the huge amount of errors in the training data but the prior knowledge in LTN gives a bigger robustness to the system. Indeed we can see a growing gap between the performance of the grounded theories $T_{prior}^k$, $T_{expl}^k$ according to the increasing percentage of errors.

Figure 2: AUC degradation for indoor objects types and part-of predicate according to the percentage of noisy labels in training data. The drop in performance is comparative smaller for LTNs with prior knowledge.

6 Conclusion and future work

SII systems are required to address the semantic gap problem: combining visual low-level features with high-level concepts. We argue that the problem can be addressed by the integration of numerical and logical representations in deep learning. LTNs learn from numerical data and logical constraints, enabling approximate reasoning on unseen data to predict new facts. In this paper, LTNs were shown to outperform state-of-the-art method Fast R-CNN on the bounding box classification task, and to outperform a rule-based method at learning part-of relations in the PASCAL-PART-dataset. Moreover, LTNs were evaluated on how to handle noisy data through the systematic creation of training sets with wrong labels. Results indicate that logical knowledge can add robustness to neural systems in the case of noisy labels. As future work, we plan to apply LTNs to very large datasets such as VISUAL GENOME, and to continue to compare the various instances of LTN with SRL, deep learning and other neural-symbolic approaches on such challenging visual intelligence tasks.
References


