Probabilistic relational learning of event models from video

Krishna S. R. Dubba[1], Paulo Santos[2], Anthony G. Cohn[1], and David C. Hogg[1]

[1] School of Computing, Leeds University, Leeds, UK
[2] Department of Electrical Engineering
Centro Universitário da FEI, São Paulo, Brazil

Abstract. This paper investigates the application of an inductive logic programming system, allied with Markov Logic Networks (MLNs), to the task of learning event models from large video datasets. A learning from interpretations setting is used to learn event models efficiently, these models define the structure of a MLN. The network parameters are obtained from discriminative learning and probabilistic inference is used to query the MLN for event recognition.

1 Introduction

Much work in computer vision in the 70's and 80's had as a goal the development of high-level vision, whereby the numerical processes feed a symbolic level of knowledge with which an agent is capable of interpreting the world. These early attempts were frustrated by the non-existence at the time of efficient algorithms for dealing with uncertainty, of tractable knowledge representation formalisms and also by the rudimentary stage of image-processing algorithms. Since then, important advances in Artificial Intelligence (AI) suggest that we may be at the stage of bridging the gap between AI and Computer Vision. The present paper stems from two of these advances: *Inductive Logic Programming* (ILP) [7], which provides computational tools for learning first-order formulae from examples and *Markov Logic Networks* (MLNs) [9], which is a relational graphical framework that provides efficient probabilistic inference methods.

The purpose of this paper is to investigate the application of an ILP system, based on a *learning from interpretations* setting [2], to learn event definitions from large video datasets (∼2.5 million frames). These event definitions have the form of first-order rules that are relational descriptions of the structure of Markov logic networks. The resulting MLNs are used to encode the domain uncertainty by means of parameters (weights) that are obtained from discriminative learning. Probabilistic inference is used to query the MLN about the events observed in test videos. The main novelty in this paper is the presentation of a complete supervised learning framework to learn probabilistic relational models from complex videos and the investigation of its application on real world data. To the best of our knowledge, there is no such framework that can be used for the problem we investigate here. A very recent paper [6] only partially addresses this problem by using hand modelled rules and arbitrary weights.

2 Related Work

The large majority of state-of-the-art event recognition systems are based on probabilistic frameworks such as Bayesian Networks or Hidden Markov Models (as surveyed in [5]). The reason for this preference is that probabilistic methods provide a robust treatment of noise and uncertainty. However, these methods are propositional, i.e., they can only be applied to fixed-length domains and cannot explicitly represent domain knowledge. To the best of our knowledge, the investigation reported in [11, 6] was the first to use a first-order probabilistic language to the task of event modelling from video sequences. The work presented in [11, 6], however, only uses probabilistic inference for querying the scenes, leaving aside the problem of learning event models from video. In the present paper we investigate how rules can be learnt from complex real world video sequences using a first-order probabilistic logic.

The present work is concerned with a particular subset of first-order probabilistic models, that relate to statistical relational learning (or probabilistic inductive logic programming), such as Bayesian logic programs (BLP), Markov logic networks (MLNs), stochastic logic programs (SLP), probabilistic relational models (PRM), and statistical relational models (SRM), among others as overviewed in [10]. In the work reported in [8] some of these approaches are compared in terms of their relative expressive power, from which translations between them were obtained. Briefly, the conclusion of this comparison is that BLPs are equivalent to an extended version of SLPs, as they represent equivalent probability distributions. It was also concluded that BLPs (SLPs) can express PRMs and SRMs and, thus, BLP (SLP) is a more general language. It is informally argued in [3] that Markov logic networks are capable of representing (and, in some cases, to generalise) most of the statistical relational learning approaches cited above. The extent to which it can be applied to the task of event recognition from video is an issue that we address in the present work.

Markov logic networks [3] are probabilistic relational graphical models defined by a set of weighted first-order formulae. The formulae define the topology of a Markov network that has in its vertices each possible groundings of the formulae predicates and its edges represent the logical connectives in the formulae. Thus, each formula defines a clique in the graph where the formula weight defines the clique's potential.

More formally, a Markov logic network L is a set of formulae F_i in first-order logic with a weight w_i (a real number) attached to each formula. This can be viewed as a template for constructing Markov networks $M_{L,C}$, where C is a set of constants and the probability distribution over possible worlds x is given by: $P(X = x) = \frac{1}{Z} exp\left(\sum_i w_i f_i(x)\right) = \frac{1}{Z} \prod_i \phi_i(x_{\{i\}})^{n_i(x)}$, where $n_i(x)$ is the number of true groundings of F_i in x, $x_{\{i\}}$ is the state of the i-th clique which has a corresponding feature $f_i(x) \in \{0, 1\}$ and an associated weight $w_i = log\phi_i(x_{\{i\}})$ (for a potential function ϕ_i defined on the cliques $x_{\{i\}}$). Z is the partition function.

3 The Airport logistics domain

This paper investigates event learning from videos obtained from an airport ramp. The airport apron under consideration has been installed with *six* cameras that record logistical activities from different angles. Altogether, 20 data sets each reflecting a turn-around[3] have been used. Each video is on average $50,000$ frames long (15 frames per sec). We have been supplied with 3D tracking data which fuses tracking on videos from six cameras to provide a ground plane interpretation. The tracking data is noisy because of the low quality, inadequate illumination and low contrast of CCTV videos.

[3] A turn-around is the time an aircraft spends in the apron area.

The zones of the apron area are defined in accordance to the International Air Transport Association (IATA) regulations and are shown in Fig. 2.

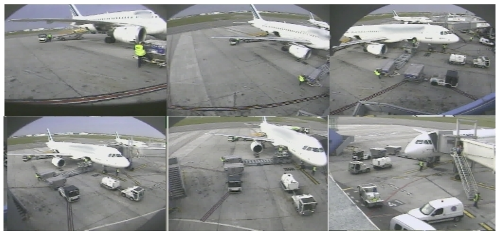

Fig. 1. Videos are captured from six cameras and tracking is fused to get ground plane data.

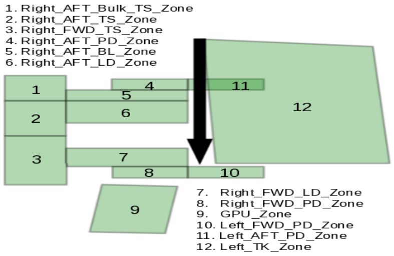

Fig. 2. Zones on the airport apron. The arrow shows the position of the aircraft on the apron.

The events of which we aim to learn models from the airport domain are the following: **Aircraft_Arrival**: Aircraft comes into the apron; **Aircraft_Departure**: Aircraft moves away from the apron from its position; **PBB_Positioning**: Passenger Boarding Bridge attaches itself to the aircraft; **FWD_CN_LoadUnload_Operation**: Container Loading/Unloading at the front end of the aircraft; **AFT_CN_LoadUnload_Operation**: Container Loading/Unloading at the rear end of the aircraft; **AFT_Bulk_LoadUnload_Operation**: Baggage Loading/Unloading at the rear end of the aircraft.

4 Learning probabilistic models of events

Events are considered to be the interactions of objects within a time interval. In this work, these interactions are encoded as spatio-temporal relational facts using spatial and temporal primitive relations between objects and intervals respectively. To represent the interactions of objects in video, they are first tracked using computer vision algorithms and their bounding boxes are obtained. This tracked data is used to compute the temporally extended qualitative spatial relations among the bounding boxes of objects. This spatio-temporal data represents the objects' interactions that define an event. We use the spatial relations *Surrounds* (sur), *Touches* (tch) and *Disconnected* (dis) and Allen's temporal relations [1] to represent the relational data. Fig. 3 shows how an interaction between two objects (A and B), represented by their bounding boxes over a time period (T = ⟨ T1, T2, T3⟩), is converted to relational data.

We use the Typed-ILP introduced in [4] for learning the MLN structure, exploiting the type hierarchy of objects in real world domains. Unlike many domains, generating positive and negative examples is not straightforward in event learning from real world videos. The presence of concurrent events, the lengthy duration of the video when compared to the actual events, the occurrence of unrelated objects in the scene and the vagueness of the start and end of the events makes the annotation task challenging. In order to cope with these issues, we generate positive and negative examples required for supervised learning delineated by a space-time cuboid. In this kind of supervision, the

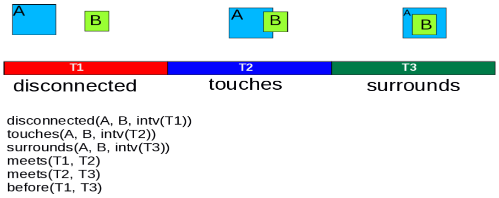

Fig. 3. Interactions as spatio-temporal data

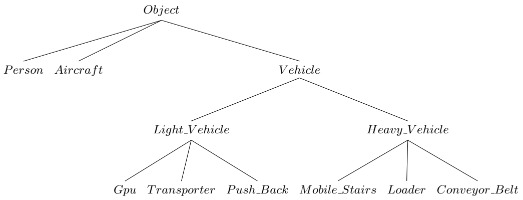

Fig. 4. Type hierarchy in the airport domain

user roughly annotates the spatio-temporal extent of the events using the video instead of doing a detailed annotation. All the objects and zones that are in the deictic spatial region in the given temporal interval for an event instance are considered as relevant objects for that event. A spatio-temporal fact is part of the example if its corresponding interaction (object-object or object-zone) of the relevant objects occurs in the given deictic temporal interval. This set of spatio-temporal facts forms the positive example and the rest of the data in the video is considered as a negative example for that event.

As the examples are independent of each other, Typed-ILP uses the learning from interpretation setting [2], since it makes an efficient test of hypotheses by evaluating the dataset locally in the interpretation, rather than the global coverage test usually done in ILP. The Typed-ILP is based on searching hypotheses in a lattice (the possible hypothesis space) bounded by the *empty clause* and the *most-specific clause* [7], that is computed using an example and mode declarations. This lattice is traversed with the standard Progol [7] refinement operator that specializes the hypothesis by adding predicates from the *most-specific clause* and a type-refinement operator that generalizes the type of one of the objects, until a hypothesis is found that maximises a score function.

After applying the Typed-ILP and learning the rules for events (which form the MLN structure), weights are learnt using a discriminative weight learning algorithm. In order to do that we give a unique tag for each example. This tag is present as an additional argument in all the facts related to that example and represents that the examples are independent from each other (for instance, any fact or constant in an example is independent from the facts and constants of the others). All the predicates in the learned rules also retain this tag during the discriminative weight learning algorithm. Inference is accomplished by using the grounded head of the rules as queries. Weight learning and probabilistic inference are done with Alchemy [3], which is a standard tool for MLNs.

5 Experimental Results and Evaluation

In this section we present the learnt rules from the *airport apron events* video datasets and also precision-recall curves from event recognition using MLN probabilistic inference. The *Leave-one-out* testing strategy was used in this experimental evaluation.

Each turn-around is separately processed to get relational data that consists of a set of spatial relations among vehicles and zones. The spatio-temporal data for each video has on average 350 facts and this depends on the number of objects and also the interactions between different activities. The temporal relations *before* and *after* were not used, since the increase in data size related to these relations makes inference computationally infeasible.

In order to cope with a type hierarchy in MLNs, we introduce a predicate for each node in the tree representing the type hierarchy. If an object belongs to a particular type, then all the predicates on the path from root to the object's node are grounded with the object. Fig. 4 shows a type hierarchy for the domain used in this work.

Rules for two events learnt from the data (with their learnt weights) are given in the formulae below, where `tag` is a variable representing unique tag for the event instances. Formulae for the remaining events are ommitted here for brevity.

```
aircraft_arrival(tag) <=   obj_aircraft(obj1,tag) ∧ dis(obj1,Left_tk_zone,tag,time1) ∧
                           tch(obj1,Left_tk_zone,tag,time2) ∧ meets(time1,time2,tag)        :4.05

aircraft_departure(tag) <= obj_aircraft(obj1,tag) ∧ tch(obj1,Right_aft_ld_zone,tag,time1) ∧
                           dis(obj1,Right_aft_ld_zone,tag,time2) ∧ meets(time1,time2,tag)   :3.27

fwd_cn_loadunload_op(tag) <=obj_veh_heavyveh_loader(obj1,tag) ∧ dis(obj1,Right_fwd_ld_zone,tag,time1)
                           ∧ tch(obj1,Right_fwd_ld_zone,tag,time2) ∧ meets(time1,time2,tag) :2.42
```

Due to the data complexity, MCMC (Gibbs sampling) is used instead of exact inference. Fig. 5 shows the *precision-recall* results for the inference performance of Typed-ILP and MLN in the airport domain using the learnt rules. The diamond in the graphs is the Typed-ILP result and the curves represent the results obtained with MLN when varying the discrimination threshold. It was not possible to obtain curves for the ILP results, as there is no discrimination threshold for deterministic rules.

The results obtained with the MLN (whose structure is defined by these rules) in general outperforms the performance of the Typed-ILP rules alone. There is a region of discrimination threshold values on the MLN curves (Fig. 5) that presents higher precision and recall values than those obtained with the Typed-ILP rules. There is roughly a 100% increase (increased from 0.285 to 0.5875) in the *F-measure* when averaged across all events. This suggests that the weights learned were capable of capturing relevant patterns in the data that were overlooked by the ILP system.

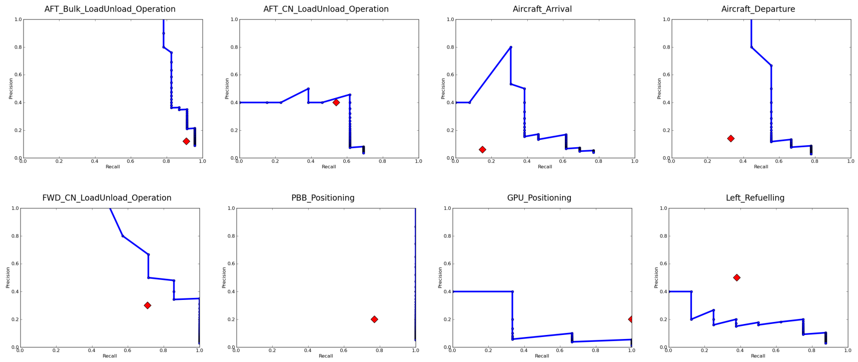

Fig. 5. Precision-Recall curves for MLN comparing the performance against Typed-ILP.

6 Conclusion and future work

This paper reports our investigations on using probabilistic relational techniques for learning event models from complex computer vision data. As far as we are aware, this is the first application of both structure learning (using ILP) and weight learning and probabilistic inference (using an MLN) to event class learning and recognition in videos. First-order event definitions were learnt using an inductive logic programming system, that uses the type hierarchy of domain objects in order to reduce over generalisation. The underlying ILP engine is based on a learning from interpretation setting, which was a key feature for learning appropriate event models, since it avoids the traditional global coverage test in ILP that searches for rules to generalise the entire dataset. The formulae resulting from the Typed-ILP system were used to define the structure of a Markov logic network, whose weights were learnt using a discriminative learning algorithm. Approximate probabilistic inference methods on MLNs were then used to execute event recognition on videos.

These ideas were applied on the *airport apron dataset*, that was obtained from many hours of video surveillance from an airport ramp. We reported results on event recognition using the ILP engine (without probabilities) and using the related MLN. MLN improved the task of event recognition on the videos of the airport apron domain, where the deterministic rules, output directly from the ILP system, presented a poor performance. Results expressed by precision-recall curves show that in the cases investigated there was a region of the decision thresholds (defining the curves related to MLN results) where both precision and recall values were higher than those obtained by the base rules alone.

Future work will consider the application of the ideas described above to learn models of, and further infer, abnormal behaviour from the airport apron area. This is a key, challenging, problem in the development of intelligent vision systems for airport security (and indeed other surveillance applications). Also in the pipeline is learning and inference with soft evidence since video data obtained from realistic scenarios is typically probabilistic in nature.

Acknowledgements

Paulo Santos acknowledges support from FAPESP project LogProb, 2008/03995-5, São Paulo, and also travel support from CAPES and CNPq, Brazil;

References

1. J. F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26:832–843, November 1983.
2. H. Blockeel, L. De Raedt, N. Jacobs, and B. Demoen. Scaling up Inductive Logic Programming by learning from interpretations. *Data Min. Knowl. Discov.*, 3:59–93, 1999.
3. P. Domingos and D. Lowd. *Markov Logic: an interface layer for artificial intelligence*. Morgan & Claypool, 2009.
4. K. S. R. Dubba, A. G. Cohn, and D. C. Hogg. Event model learning from complex videos using ILP. In *Proc. of ECAI*, pages 93–98, 2010.
5. G. Lavee, E. Rivlin, and M. Rudzsky. Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 39(5):489 –504, 2009.
6. V. I. Morariu and L. S. Davis. Multi-agent event recognition in structured scenarios. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.
7. S. Muggleton. Inverse entailment and Progol. *New Gen. Comp.*, 13(3&4):245–286, 1995.

8. S. Muggleton and J. Chen. A behavioral comparison of some probabilistic logic models. In *Probabilistic inductive logic programming*, pages 305–324. 2008.
9. J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.
10. L. De Raedt and K. Kersting. Probabilistic inductive logic programming. In *Probabilistic Inductive Logic Programming*, pages 1–27, 2008.
11. S. D. Tran and L. S. Davis. Event modeling and recognition using Markov Logic Networks. In *Proceedings of the 10th ECCV: Part II*, pages 610–623. Springer-Verlag, 2008.