



EPiC Series in Health Sciences

Volume 1, 2017, Pages 269–275

CAOS 2017. 17th Annual Meeting of the International  
Society for Computer Assisted Orthopaedic Surgery



# Towards a Surgical Phase Detection Using Markov Logic Networks

Patrick Philipp<sup>1</sup>, Johannes Bleier<sup>1</sup>, Yvonne Fischer<sup>2</sup> and Jürgen Beyerer<sup>1,2</sup>

<sup>1</sup>Karlsruhe Institute of Technology KIT, Vision and Fusion Laboratory IES, Karlsruhe, Germany

<sup>2</sup>Fraunhofer Institute of Optronics, System Technologies and Image Exploitation IOSB,  
Karlsruhe, Germany,

p.philipp@kit.edu, johannes.bleier@student.kit.edu,

yvonne.fischer@iosb.fraunhofer.de, juergen.beyerer@iosb.fraunhofer.de

## Abstract

For mastering challenges of contemporary surgical interventions, the use of computer-implemented assistance functions has become an emerging trend. These assistance functions create opportunities to prevent malpractices and preserve a high level of satisfaction for patients as well as employees. To enable such functions in context of a computer-assisted orthopaedic surgery (CAOS), we elaborate the use of Markov Logic Networks (MLNs) for modelling surgical phases. In contrast to commonly researched systems for surgical process modelling, MLNs combine rule-based as well as probabilistic approaches. This allows us to integrate soft and hard constraints into our network – which greatly expands the scenery of currently researched models for phase detection in surgical interventions. In our contribution, we present the necessary fundamentals of MLNs and show the application to a comprehensible test case. The results are promising concerning the use of MLNs for surgical phase detection. In particular, MLNs have shown two advantages: Firstly, due to their template characteristics, few logic rules allow to model numerous interdependencies between the different surgical phases. Secondly, the combination of probabilistic and logic approaches allows to handle sensor inaccuracies and misclassifications of features directly. E.g., the inaccuracy of a sensor can be expressed by reducing the weight of corresponding formulas, allowing for a softening of constraints.

## 1 Introduction

The use of assistance functions for diagnosis and surgical interventions has become an evolving area for mastering challenges of contemporary medicine. Inter alia, these assistance functions can help

to prevent malpractices, optimize treatment and preserve a high level of satisfaction for patients as well as employees (Philipp, Fischer, Hempel, & Beyerer, 2016), (Lalys & Jannin, 2014).

To enable such assistance functions in context of a computer-assisted orthopaedic surgery (CAOS), we elaborate the use of Markov Logic Networks (MLNs). These Networks are utilized to model the progress of a surgical process by a sequence of surgical phases. In contrast to commonly researched models for surgical process modelling (Lalys & Jannin, 2014), MLNs combine rule-based as well as probabilistic approaches (Richardson & Domingos, 2006). This allows us to integrate soft and hard constraints into our models – which greatly expands the scenery of currently researched models for phase detection in surgical interventions.

In this contribution we investigate to which extend MLNs are suitable for the estimation of the progress of an ongoing surgery. This is the basis for establishing an assistance function, which provides on the fly medical practitioners with tailored (i.e. context sensitive) information, such as x-ray images, medical records or an exemplary representation of options for actions during a surgery.

In the following sections, we present the required fundamentals of MLNs and show their capability of estimating the presence of two standard surgery phases as well as an emergency phase. These three phases serve as a comprehensible test case and can be seen as representatives for specific phases, e.g. in context of a knee- or hip replacement.

## 2 Materials and Methods

A Markov logic network (MLN) consists of a knowledge base, which contains weighted formulas of first-order logic. Thereby, the weighting softens the hard constraints on the set of possible worlds provided by the first-order logic. That means by assigning a weight  $\neq \infty$  to a formula, a world  $v$  violating this formula is not impossible anymore – it is less probable. The higher the weight of a violated formula, the lower the probability of the resulting world. Formally, a MLN  $L$  is defined as a set of pairs (Richardson & Domingos, 2006):

$$L = \{(F_1, w_1), \dots, (F_N, w_N)\}, \quad (1)$$

whereby each pair  $(F_i, w_i)$  represents a first-order logic formula  $F_i$ , as well as the corresponding weighting  $w_i \in \mathbb{R}$ . With a finite set of constants  $C = \{c_j: j = 1, \dots, |C|\}$ , a MLN  $L$  can be used as a template for a Markov network  $M_{L,C}$  (also known as Markov Random Field). In order to generate a Markov network  $M_{L,C}$  from a MLN  $L$ , all variables of the formulas in  $L$  are replaced with constants  $C$  (the replacement is also known as “grounding”), further leading to so called “ground atoms”. The template characteristic of a MLN is reflected by two facts: Firstly, choosing different sets of constants will result in different Markov networks  $M_{L,C}$  (i.a. that is why  $C$  is used as an index of the Markov network). Secondly, all groundings of a formula are assigned with the same weights as specified in  $L$ . These weights can be set by hand or learned from training data.

The structure of a Markov network  $M_{L,C}$  is given by an undirected graph  $G = (V, E)$ . Thereby, the vertices  $V$  represent ground atoms. Two vertices are connected by an edge  $e \in E$  iff the corresponding ground atoms share at least one grounding in  $L$ . Furthermore, the probability of a world  $v$  is given by:

$$P(v) = Z^{-1} \exp\left(\sum_{i=1}^{|F|} w_i n_i(v)\right) = Z^{-1} \prod_{i=1}^{|F|} \phi_i(v_{F_i})^{n_i(v)}, \quad (2)$$

Predicate	Description
$Previous(phase)$	$phase$ is the previous phase of the surgery
$Current(phase)$	$phase$ is the current phase of the surgery
$Presence(feature)$	$feature$ is present in the operation theatre

**Table 1:** The listed predicates are used to define appropriate first order logic formulas for the proposed Markov Logic Network (1). The used variables are typed, that means the corresponding variables can only be replaced (or: “grounded”) by constants of the same type. E.g. the variable  $phase$  can only be replaced by a constant of type  $Phase$ , which may be given by the following subset of constants:  $\{Phase1, Phase2, \dots, PhaseN\}$ .

whereby  $Z$  is a normalization constant (so that  $P \in [0,1]$ ),  $w_i$  is the weight and  $n_i(v)$  is the number of true groundings of a formula  $F_i$  in world  $v$ . Furthermore,  $\phi_i(v_{F_i}) = e^{w_i}$ . In other words, the probability of a world  $v$ , i.e.  $P(v)$ , is proportional to  $exp(\sum \text{weights of grounded formulas satisfied})$ .

To utilize MLNs for a surgical phase detection, formulas and weights have to be specified. For the former, we define predicates as given in Table 1. We use the predicate  $Previous(phase)$  to model that the previous phase of the surgery is given by  $phase$  (cf. Table 1). Similarly, we define  $Current(phase)$  to represent that  $phase$  is the current phase of the surgery. Furthermore, predicate  $Presence(feature)$  is used to model that a feature (e.g. a surgical instrument) is present in the operation theatre. Please note, that the used variables  $phase$  and  $feature$  are typed, which means they can only be replaced by constants of the same type. I.e.,  $phase$  can only be replaced by a specific set of constants, e.g.  $\{Phase1, Phase2, \dots, PhaseN\}$ . A formula  $F_i$  is constructed (e.g.) by combining these predicates with logical operations and quantifiers. For example, “the previous phase  $Phase1$  causes the current phase  $Phase2$ ” is formally given by:

$$Previous(Phase1) \Rightarrow Current(Phase2).$$

Furthermore, features like surgical instruments used in a phase, are represented by (e.g.):

$$Presence(Instrument1) \Rightarrow Current(Phase1).$$

To soften the hard constraints given by the knowledge base of first-order logic formulas, a weighting is specified (cf. Equation (1) for the definition of a MLN). By adjusting the corresponding weights, we are able to model (e.g.) that the presence of  $Instrument1$  in  $Phase1$  is more likely than in  $Phase2$ . Furthermore, softening a formula by a weight  $\neq \infty$  means, that a world  $v$  violating this formula is not impossible anymore – it is less probable. These weights can be learned from training data and/or set by an expert. E.g., a medical domain expert like a surgeon can set the information about reasonable sequences of phases.

### 3 Results

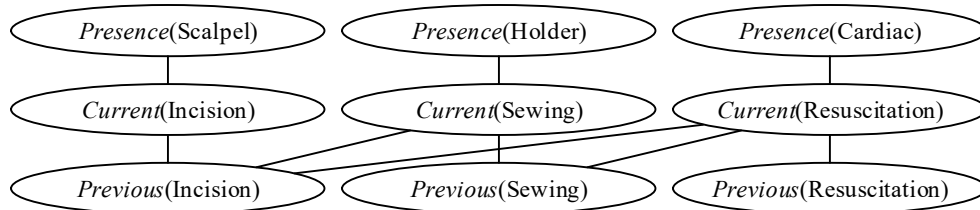
We illustrate our approach by a simplified example of a surgery comprising three phases: “Incision” (performing a skin incision) and “Sewing” (sewing up a skin incision with sutures).

<b>i</b>	<b>F<sub>i</sub></b>	<b>w<sub>i</sub></b>
1	$\forall phase: Previous(phase) \Rightarrow Current(phase)$ , with $phase \in \{Incision, Sewing, Resuscitation\}$	1
2	$Previous(Incision) \Rightarrow Current(Sewing)$	0.5
3	$\forall phase: Previous(phase) \Rightarrow Current(Resuscitation)$ , with $phase \in \{Incision, Sewing, Resuscitation\} \setminus \{Resuscitation\}$	0.1
4	$Presence(Scalpel) \Rightarrow Current(Incision)$	1
5	$Presence(Holder) \Rightarrow Current(Sewing)$	1
6	$Presence(Cardiac) \Rightarrow Current(Resuscitation)$	3

**Table 2:** The listed formulas and weights are used to define the proposed Markov Logic Network (1). By using weights, the hard constraints of a common knowledge base of first-order logic formulas can be softened. The probability of world  $v$  is exponentially proportional to the product of satisfied grounded formulas weighted by their corresponding weight (cf. Equation (2) for reference).

Furthermore, there is one emergency phase “Resuscitation” (performing a cardiopulmonary resuscitation). These three phases serve as a comprehensible test case and can be seen as representatives for specific orthopaedic phases, e.g. in context of a knee- or hip replacement. Used features are: “Scalpel” (the used scalpel for skin incision), “Holder” (needle holder used to hold a suturing needle) and “Cardiac” (cardiac arrest during surgery).

The used MLN is given by Table 2, and the corresponding structure is depicted in Figure 1. Formula  $F_1$  is used to model the probabilities of staying in a certain phase during surgery (cf. Table 2). Formulas  $F_2$  and  $F_3$  specify the transition of phases, e.g., if the previous phase is incision, worlds where the current phase is sewing are more probable than worlds where the current phase is resuscitation. Formulas  $F_4$ - $F_6$  are used to model the presence of selected features of specific surgical phases. The different weights are chosen according to the probability of observation and to the different accuracy of the used sensors. The graphical representation of the proposed MLN (cf. Table 2) is depicted in Figure 1.

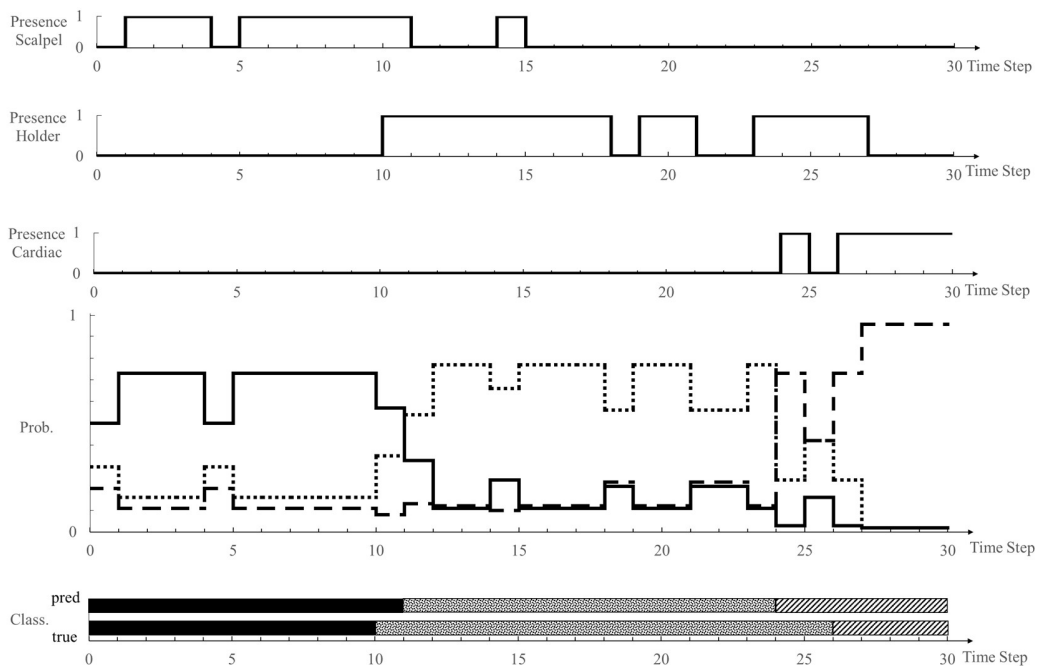


**Figure 1:** Structure of the proposed MLN (cf. Table 1). The vertices  $V$  in the undirected graph  $G = (V, E)$  represent ground atoms. Two vertices are connected by an edge  $e \in E$  iff the corresponding ground atoms share at least one grounding in the MLN.

For the actual phase detection, i.e. the calculation of the probabilities of a phase being present, the concept of inference is needed. Inference answers queries in a Markov network like “What is the probability that formula F1 holds given that formula F2 holds?”. Since inference is NP-complete (Roth, 1996), approximate algorithms are common in practical use. Parag proposes an algorithm called LazySAT (Parag, 2009) which is a memory usage optimized algorithm based on MaxWalkSAT (Kautz, Selman, & Jiang, 1997), a solver for Boolean Satisfiability Problems, which can handle weighted formulas. The results of inference are shown in the next section.

To illustrate the characteristics of the model, we consider the following test sequence of observations (cf. Figure 2). The first three rows represent the detection of different features. A feature can be present at a specific time step during the surgery (indicated by a value of 1) or not (indicated by a value of 0). In the fourth row, the result of the corresponding inference is shown. E.g. from time step 1 to 3, the probability of phase incision being the current phase is 73%. As the presence of the scalpel temporarily drops (time step 4), this probability decreases to a value of 50%.

In the last row of Figure 2, the phases predicted by our model are shown in comparison to the phases actually present. In particular, the phases incision (solid black bar), sewing (dotted bar) and resuscitation (dashed bar) are depicted. The model predicts a change of phases in time step 11, whereas the actual change of phases occurred in time step 10. This delay is caused by an overlap of the presence of instruments.



**Figure 2:** First 3 rows indicate if a feature is present, which corresponds to the value 1 (value 0 otherwise). In the fourth row, the probabilities of a phase being present are depicted. The solid line represents the probability of phase incision being present; the dotted line represents sewing, whereas the dashed line represents resuscitation. In the last row, the true and predicted phases are depicted. There are wrong predictions (e.g. time step 10, 24, 25) due to noisy sensory data but the model is able to smooth erroneous (non-) detection of features (e.g. time steps 4, 14, 18, 21, 22 and 24).

In practical use, erroneous (non-)detection of features by the sensors and used algorithms can occur. This can lead to a misclassification of surgical phases; however, our model is able to deal with noisy data. In Figure 2, both cases are shown: In time step 24, the model wrongly predicts a change in phases, whereas the actual change occurs in time step 26. This is a consequence of an erroneous detection of a cardiac arrest by the technical system, which causes the model to predict the corresponding phase. Even the temporary drop of the presence of cardiac (time step 25) does not lead to a turnaround in prediction. Nevertheless, the model is able to smooth erroneous (non-)detection of features in many cases. This can be seen in time steps 4, 14 and 18. The inference of other observations delivers comparable results.

## 4 Discussion

The presented work shows promising results concerning the use of MLNs for a surgical phase detection. Three phases serve as a comprehensible test case and are used as representatives for specific orthopaedic phases, e.g. in context of a knee- or hip replacement. A challenge of the proposed model can be seen in the costs of inference. However, efficient algorithmic approaches exist and are subject of past and current research (Kautz, Selman, & Jiang, 1997), (Parag, 2009). Regarding the presented application example, MLNs have shown two advantages: Firstly, the combination of probabilistic and logic approaches allows for a convenient way to model sequences of phases. E.g., only one rule allows to model the transition to an emergency phase like resuscitation for numerous other phases (cf. Table 2,  $F_3$ ). Secondly, sensor inaccuracies and misclassifications of features can be handled directly. For example, the inaccuracy of a sensor can be expressed by reducing the weight of corresponding formulas, allowing for a softening of constraints. The investigation of dynamically adjusted weights is an interesting approach for further research. By this, temporary adaption to varying conditions in the operating theatre leading to varying sensors accuracies (e.g. different illumination levels reducing the accuracy of a specific sensor) could be represented in the phase detection model.

## References

- Kautz, H., Selman, B., & Jiang, Y. (1997). A general stochastic approach to solving problems with hard and soft constraints. In D. J. Gu D, *The Satisfiability Problem: Theory and Applications* (pp. 573-586). New York: American Mathematical Society.
- Lalys, F., & Jannin, P. (2014). Surgical process modelling: a review. *International Journal of Computer Assisted Radiology and Surgery*, 495-511.
- Parag, P. (2009). *Markov Logic: Theory, Algorithms and Applications*. Washington: Dissertation at University of Washington.
- Philipp, P., Fischer, Y., Hempel, D., & Beyerer, J. (2016). Framework for an Interactive Assistance in Diagnostic Processes Based on Probabilistic Modeling of Clinical Practice Guidelines. In H. Arabnia, & Q. Nam Tran, *Emerging Trends in Applications and Infrastructures for Computational Biology, Bioinformatics, and Systems Biology* (pp. 371-390). New York: Elsevier.
- Richardson, M., & Domingos, P. (2006). Markov Logic Networks. *Machine Learning vol. 62(1-2)*, 107-136.
- Roth, D. (1996). On the Hardness of Approximate Reasoning. *Artificial Intelligence vol. 82(1-2)*, 273-302.

## Disclosures

The presented study is part of the project “KonsensOP” which is funded by the Federal Ministry of Education and Research (BMBF), Germany.