JMLR: Workshop and Conference Proceedings vol 52, 50-61, 2016 PGM 2016

Learning Complex Uncertain States Changes via Asymmetric Hidden Markov Models: an Industrial Case

Marcos L.P. Bueno[1] MBUENO@CS.RU.NL
Arjen Hommersom[1,2] ARJENH@CS.RU.NL
Peter J.F. Lucas[1,3] PETERL@CS.RU.NL
Sicco Verwer[4] S.E.VERWER@TUDELFT.NL
Alexis Linard[1] A.LINARD@CS.RU.NL

[1] *Institute for Computing and Information Sciences, Radboud University (The Netherlands)*
[2] *Faculty of Management, Science and Technology, Open University (The Netherlands)*
[3] *Leiden Institute of Advanced Computer Science, Leiden University (The Netherlands)*
[4] *Department of Intelligent Systems, Delft University of Technology (The Netherlands)*

Abstract

In many problems involving multivariate time series, Hidden Markov Models (HMMs) are often employed to model complex behavior over time. HMMs can, however, require large number of states, that can lead to overfitting issues especially when limited data is available. In this work, we propose a family of models called Asymmetric Hidden Markov Models (HMM-As), that generalize the emission distributions to arbitrary Bayesian-network distributions. The new model allows for state-specific graphical structures defined over the space of observable features, what renders more compact state spaces and hence a better handling of the complexity-overfitting trade-off. We first define asymmetric HMMs, followed by the definition of a learning procedure inspired on the structural expectation-maximization framework allowing for decomposing learning per state. Then, we relate representation aspects of HMM-As to standard and independent HMMs. The last contribution of the paper is a set of experiments that elucidate the behavior of asymmetric HMMs on practical scenarios, including simulations and industry-based scenarios. The empirical results indicate that HMMs are limited when learning structured distributions, what is prevented by the more parsimonious representation of HMM-As. Furthermore, HMM-As showed to be promising in uncovering multiple graphical structures and providing better model fit in a case study from the domain of large-scale printers, thus providing additional problem insight.

Keywords: Probabilistic graphical models; multivariate time series; hidden Markov models; asymmetric independence; industrial processes.

1. Introduction

Modern industrial artifacts are becoming more and more complex machines by embracing progress in computing, control, and sensor technology. Whereas engineers understand the workings of the individual components in considerable depth and detail, as a consequence of their design, they find it much more difficult to understand the behavior of the artifact at a certain level of abstraction. Yet, such abstraction is needed to overcome the complexity of the artifacts. Understanding the sequential and temporal behavior of such systems is inherently difficult, and there is a need to identify different states in which this dynamic system can be, further leading to valuable information for engineers in charge of maintenance scheduling, or component designing for instance.

Hidden Markov Models (HMMs) have been extensively used to model sequential and temporal behavior in many real-life domains, such as speech recognition (Rabiner, 1989; Bilmes, 2000), in-

formation retrieval (Freitag and McCallum, 2000), biological sequence analysis (Won et al., 2006), and process mining (Rozinat et al., 2008) with promising results. However, it has been also recognized that HMMs have limitations to properly capture distributions from limited amounts of data (Bilmes, 2006; Ghahramani, 2001; Markov et al., 2006). The usage of HMMs in practice often resorts to a single chain of states and impose a naive structure over the feature space. While this alleviates learning and inference costs, it gives rise to larger state spaces that lead to learning issues and unsatisfactory problem insight.

Research has been dedicated to extend HMMs for representing more structural information, aiming to render more useful and accurate models, as for example factorial HMMs (Ghahramani and Jordan, 1997), HMM/BN (Markov et al., 2006), autoregressive HMMs (Poritz, 1982). These extensions, however, capture the (in)dependence information by means of a single graphical model, which prevents capturing more specialized (in)dependences, sometimes referred to as asymmetric (in)dependences (Geiger and Heckerman, 1996), i.e. (in)dependences valid only for strict subsets of variables. The representation of asymmetries in graphical models has been recognized important for achieving better probabilistic inference (Vlasselaer et al., 2016; Boutilier et al., 1996), learning (Pensar et al., 2016; Friedman and Goldszmidt, 1996), and for improving problem insight (Kirshner et al., 2004). In the context of HMMs, however, research has been much narrower, mainly confined to representing asymmetries using Chow-Liu trees (Kirshner et al., 2004) and representations based on variations of autoregressive HMMs (Motzek and Möller, 2015; Bilmes, 2000). Hence, there is still a need to better understand the effect of representing more general asymmetries over the feature space, thus overcoming limitations of tree-based representations (Kirshner et al., 2004), lack of direct interaction between asymmetry-controlling ("activator") variables (Motzek and Möller, 2015), as well as the lack of the so-called instantaneous interactions in models tailored solely for classification (Bilmes, 2000).

In this paper, we propose a family of probabilistic graphical models, called *Asymmetric Hidden Markov Models* (HMM-As), that generalize HMMs by capturing general asymmetries in the space of observable features. Observations in asymmetric HMMs are emitted according to state-specific Bayesian-network distributions, thus representing independencies that are not captured in HMMs. We also show that HMM-As can be used successfully in understanding the behavior of modern, complex engineered artifacts, for which we use a modern industrial printing machine as a case study. It is shown that HMM-As offer a sufficiently rich representation formalism to capture abstract state behavior from measurement data.

Thus, the contributions of this paper are as follows. We first introduce the novel formalism of HMM-As, compare it to other HMM families (including the well-known independent and standard HMMs), and provide a learning algorithm, taking as inspiration the structural expectation-maximization framework (Friedman, 1997). Finally, in order to illustrate the empirical behavior of HMM-As, we used extensive simulations, as well as an industry-based case aiming at understanding the sequential behavior that governs the health of multiple components of large-scale printers. The experiments resulted in insight of several practical aspects of asymmetric and state-of-the-art HMMs, including the dimension of state spaces, goodness of fit and overfitting issues.

2. Preliminaries

Problems involving time series and sequences can be modeled by several families of probabilistic graphical models, where HMMs are amongst the most used ones (Murphy, 2002; Rabiner, 1989).

HMMs represent stochastic processes as a two-part dynamic system, accounting for an observable dynamics and a hidden (or latent) dynamics that governs the observable part. In these systems, the following set of assumptions is usually considered: the Markovian property (i.e. first-order chain), and the state blockage (i.e. inter-temporal interactions between features are indirect). These allow for modularizing the specification of HMMs via the state at the time points in three distributions, being the initial, transition and emission.

The emission distribution has been recognized as an important component for allowing HMMs to properly capture complex distributions (Motzek and Möller, 2015; Kirshner et al., 2004; Bilmes, 2000), since it governs how observations are emitted to the external world. In the case of discrete HMMs involving multinomial features, the emissions normally take the form of a Bayesian network (BN) over the feature space, conditioned on the state variable. In these cases, a single graphical structure is shared among all the states, meaning that distinct states might parameterize the conditional probability table (CPT) of each feature differently, although the parent set remains fixed. This is illustrated by the following example.

Example 1 *On a regular basis, measurements of print quality, media type and room temperature are taken for an industrial printer, denoted by A, B and C respectively. An HMM for this problem has three hidden states that dictate the underlying dynamics, accounting for 'no failure', 'acceptable', and 'unacceptable', denoted by* $\mathrm{dom}(S) = \{u, v, w\}$. *Each state induces a set of emission parameters, hence, assuming that feature space takes the form of the* independent HMM *(i.e. the features are conditionally independent given the state), this means that, e.g., states u and w produce distinct distributions, as in $P(a, \neg b, \neg c \mid u) = \theta_{a|u}\theta_{\neg b|u}\theta_{\neg c|u}$ and $P(a, \neg b, \neg c \mid w) = \theta_{a|w}\theta_{\neg b|w}\theta_{\neg c|w}$, although the factorization is always the same. Here, $\theta_{X_i|s}$ is a parameter of the CPT of X_i with parent S instantiated to s, where the parent set for $X_i \in \{A, B, C\}$ is $\{S\}$.*

We considered an industry case originated from the domain of large-scale printers, to illustrate the application of HMMs and HMM-As to real-life problems. In particular, we focus on one specific component in this case study – the nozzle – which is designed to jet the ink on paper. In order to measure the print quality, and also the proper functioning of the nozzles themselves, test pages are printed regularly. For each test page, features that describe the health of components are extracted, e.g. describing the deviation of jetted ink of different colors. Figure 1 shows a schematic printhead, with its nozzles at the top, and ink jetted on paper at the bottom. In Section 6, we consider data related to some of the most critical and costly components of these large-scale industrial printers – the nozzle health and maintenance actions – in order to construct state-based models that support a thorough understanding of the behavior of these complex machines.

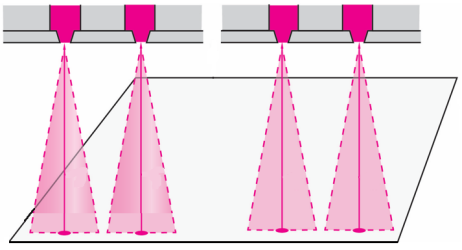


Figure 1: Simplified view of printheads and nozzles from an industrial printer.

3. Asymmetric Hidden Markov Models

Asymmetric Hidden Markov Models generalize HMMs by enriching emission distributions to represent additional qualitative independence per state explicitly. In the following we define HMM-As by first defining the association between states and arbitrary Bayesian-network distributions, followed by a discussion on model parameterization.

3.1 Model specification

Definition 1 *Let S be a random variable and $\mathbf{B}$ be the set of all Bayesian networks over a set of random variables $\mathbf{X}$. Consider a function $\phi : \mathrm{dom}(S) \to \mathbf{B}$ that associates each element of $\mathrm{dom}(S)$ to an element in $\mathbf{B}$, where the elements in $\mathrm{dom}(S)$ are called* states*. A Bayesian network associated to a state s via ϕ is called an* asymmetric Bayesian network for s*, or simply an asymmetric Bayesian network when s is evident.*

For brevity, we denote the asymmetric BN associated to state s by $\mathcal{B}_s = (G_s, P_s)$, where G_s and P_s are the graphical structure and the set of CPTs of the BN $\mathcal{B}_s$ respectively. Note that each state s induces a *state factorization* of $\mathbf{X}$ as dictated by $\mathcal{B}_s$, indicated by:

$$P_s(\mathbf{X}) = \prod_{i=1}^{n} P_s(X_i \mid \psi_s(X_i)) \tag{1}$$

where $\psi_s(X_i)$ denotes the set of parents of X_i in G_s. In the context of dynamic systems, S corresponds to the (unobserved) state variable. For convenience, an asymmetric BN is defined by mapping hidden states to BNs, which in turn implies that each feature in $\mathbf{X}$ has multiple sets of parents, one for each asymmetric BN.

Definition 2 *An* Asymmetric Hidden Markov Model *over the random variables $(\mathbf{X}, S)$ is a dynamic system $(M_{\rightarrow}, M_{\downarrow}, M_0)$, where M_0 is an* initial distribution $P(S^{(0)})$*, $M_{\rightarrow}$ is a* transition distribution $P(S^{(t+1)} \mid S^{(t)})$*, and $M_{\downarrow}$ is an* emission distribution *given by*

$$P(\mathbf{X}^{(t)} \mid S^{(t)}) = P_{S^{(t)}}(\mathbf{X}^{(t)}) \tag{2}$$

From the definitions shown above, HMM-As are able to capture more qualitative independencies in their topology than HMMs. Yet, HMM-As will share a few assumptions with HMMs, namely: the Markovian property and time-invariance. A third assumption that will also hold in HMM-As establishes that the inter-temporal interaction between features must occur via state variables. Hence, given these assumptions, an unrolled HMM-A over the time horizon $[0, T]$ has the following joint distribution:

$$P(S^{(0:T)}, \mathbf{X}^{(0:T)}) = P(S^{(0)}) \prod_{t=0}^{T-1} P(S^{(t+1)} \mid S^{(t)}) \prod_{t=0}^{T} \prod_{i=1}^{n} P_{S^{(t)}}(X_i \mid \psi_{S^{(t)}}(X_i)) \tag{3}$$

Example 2 *An asymmetric HMM $\mathcal{M}_1$ based on the printer problem of Example 1 is shown in Figure 2 as an automaton. In $\mathcal{M}_1$, there are three states as well, denoted by s_1, s_2 and s_3 respectively. This automaton runs by alternating taking probabilistic transitions and emitting multivariate observations $(A^{(t)}, B^{(t)}, C^{(t)})$ according to the states which it traverses.*

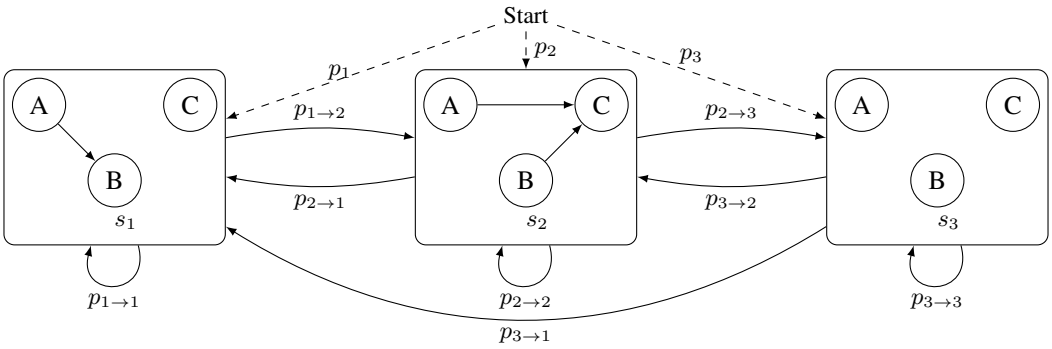


Figure 2: Probabilistic automaton representation for HMM-A $\mathcal{M}_1$ (dashed lines show initial transitions, zeros are not shown). The asymmetric BNs for states are shown in rectangles.

3.2 Parameterization

Parameterizing HMM-As raises the question of whether HMMs can represent HMM-A distributions. It is possible to show that this is true in general in the case of standard HMMs, i.e. HMMs with a fixed structure for all the states (however, not necessarily a naive one), such that given an HMM-A with k states, a standard HMM with k states can be constructed to simulate it. Such simulation can be achieved by constructing a fully connected network on the feature space of the HMM, which allows each state for capturing any distribution. Although more efficient ways can be employed to construct such HMM seeking for a minimal simulating HMM, intuitively the dependences that exist in each state of the HMM-A must be preserved in the HMM.

On the other hand, HMM-As cannot be simulated by independent HMMs with the same number of states in the general case. It turns out, however, that this process become possible at the cost of expanding the state space of HMM-Is. One way to construct such HMM-I is as follows. Given an arbitrary HMM-A, we associate to each joint distribution parameter from each of its states a state in the HMM-I with fully deterministic distributions (according to the assignment of variables for the parameter). To produce the intended results, we construct the initial distribution of the HMM-I by scaling the initial distribution of the HMM-A by the original emission parameters. The transition distribution is constructed analogously. We note that this construction lead to an exponential number of states in the number of observed features, precisely an HMM-I with $k2^n$ states. This is an upper bound for setting the correspondence between HMM-As and HMM-Is that can be tightened constructing simulating HMM-Is that do not necessarily rely on deterministic distributions. The key point is that additional states might be needed when simulating HMM-As by means of HMM-Is, what intuitively depends to a large degree on the amount of structure and the numerical parameterization in the original HMM-A.

4. Learning Asymmetric Hidden Markov Models

In order to learn asymmetric HMMs, we assume that state variables are not observed and the graphical structure for emission distributions are usually unknown. In this case, i.e. leaning under missing data and unknown structure, the score function is non-decomposable by the graphical structure, which makes analytical methods impossible. The structural expectation-maximization (SEM)

(Friedman, 1997) is often employed in these settings. In the following we develop a learning procedure for asymmetric HMMs based on SEM.

The learning setting is as follows. Given a dataset $\mathbf{D}$ of sequences with arbitrary length complete for a set of features $\mathbf{X}$, and an integer k, we aim at learning an HMM-A $\mathcal{M}$ with k states that best fits $\mathbf{D}$. In this score-based formulation, the score function is defined as $score(\mathcal{M}) = \log P(\mathbf{D} \mid \mathcal{M}) - \text{pen}(K)$, where $P(\mathbf{D} \mid \mathcal{M}) = L(\mathcal{M})$ is the likelihood, K is the number of parameters for $\mathcal{M}$, and pen is some penalty function. Common score functions are, for example, the AIC and BIC scores. In order to make the SEM process feasible, the time-invariance property (see Section 3) makes the number of variables in learning independent of time-series length. This is further aided by not representing autoregressions in HMM-As, as these typically double the number of features in emissions, leading to an exponential increase in structure learning. Due to model modularity, the score can be decomposed *per state*, and as the penalty function is additive for AIC and BIC, the score can be written as:

$$\begin{aligned} \text{score}(\mathcal{M}) &= \log L(\mathcal{M}) - \text{pen}(K) \\ &= \log L(M_0) + \log L(M_{\rightarrow}) + \textstyle\sum_S \text{score}(P_S) + c \end{aligned}$$

where c is the penalty for the transition matrix, which is constant given a fixed number of states. The main advantage now is that the structure of each asymmetric Bayesian network can be learned locally, which significantly reduces computational costs. The resulting algorithm is outlined in Algorithm 1. In Line 3, expected statistics are computed via the forward-backward algorithm (Rabiner, 1989). At each iteration, in Line 5, a new asymmetric Bayesian network is constructed for each state, i.e. *this can be learned independently from other asymmetric Bayesian networks*. Here, the expected score is $E[\log(P(S, \mathbf{D} \mid \mathcal{M}))] = \sum_S \sum_{i=1}^{m} \log P(S \mid D_i) P(D_i \mid \mathcal{M})$. After finishing this, initial and transition probabilities are updated via maximum likelihood over expected counts.

The computational complexity of reasoning within HMM-A is favorable due to its independence assumptions. The maximization part (i.e. the Bayesian-network structure learning) is done for each iteration of SEM, however, as long as $|\text{dom}(\mathbf{X})|$ is moderate, this is very feasible using existing search-and-score methods.

Algorithm 1 Structural expectation-maximization for asymmetric HMM

Input: $\mathbf{D} = \{D_1, \ldots, D_m\}$, a dataset of m sequences (each D_i is an instantiation of $\mathbf{X}^{(0:T)}$).
Output: An asymmetric HMM $\mathcal{M}$ that maximizes the expected score for $\mathbf{D}$.

1: Choose an initial model $\mathcal{M}$ randomly
2: **while** stopping criterion is not reached **do**
3: **E step**. Compute expected statistics for the each sequence D_i, where $i = 1, \ldots, m$
4: **for** each state $s \in \text{dom}(S)$ **do**
5: **M step** for state s. Learn a new model P_s that maximizes the expected score
6: set $M_{\downarrow}$ to distributions according to P_s
7: **M step** for M_0 and $M_{\rightarrow}$. Estimate new M_0 and $M_{\rightarrow}$ which maximize the expected likelihood
8: $\mathcal{M} := (M_0, M_{\rightarrow}, M_{\downarrow})$

5. Learning assessment via simulations

In Section 3, we provided upper bounds for representing HMM-A distributions via simulating HMM-Is and standard HMMs. To obtain practical insight on this process, we simulated data from HMM-A distributions, thus assessing overfitting issues, state space dimension, and the HMM-As learning algorithm. We describe the model section procedure and the reference models next.

5.1 Experiments settings

Given a sequence dataset, models are learned incrementally by increasing the number of states until observing overfitting, where a model score is based on a 10-fold cross-validation setting. Once the number of states has been determined, the final model is learned using the entire dataset, and is evaluated by means of 20 independent datasets (i.e. datasets that were not used during cross-validation). Given a learned model L and a reference model R, we define the relative likelihood **RLL** to assess the fit quality of L as $\text{RLL} = \text{LL}_L/\text{LL}_R - 1$, where LL_L and LL_R are the log-likelihoods of L and R respectively. Learned models with relative likelihood closer to 0 indicate that they fit to the data as well as the reference models do, hence these are preferred.

Random HMM-As were generated taking into account that many real-life networks have an average degree between 2 and 4 per node (Scutari and Denis, 2014), thus we set the maximum degree of each node on each network to 3. In order to build a random HMM-A with k states, k graphical structures are sampled from a uniform distribution (Melançon and Philippe, 2004); afterwards, random parameters for the nodes, the initial and transition distributions are generated as well. We considered scenarios with varied number of features $n \in \{3, 6, 10, 14, 18\}$, and state space dimension of reference models $k \in \{2, 6\}$.

5.2 Results

Table 1 shows the dimension of state spaces associated to learned models, while Figure 3 shows the corresponding relative likelihoods. In general, approximating distributions of HMM-As required HMM-Is with state spaces substantially larger than those of the former. This occurred notoriously in the situation of larger datasets (right part of Table 1), as the model dimension could increase more until overfitting was noticed. On the other hand, the learned standard HMMs had considerably small state spaces, what is explained by the rather dense graphical structures that needed to be learned to capture different state-specific independencies underlying the data. Such graphical structures were often a fully connected graph (or almost), what prevented learning models with more states. Regarding running time, learning HMM-Is was interestingly more costly in most cases: although learning HMM-As is done via structural EM, its combination with search heuristics and smaller space state were in practice more efficient than the EM used to learn HMM-Is. A similar conclusion can be drawn for the comparison with standard HMMs.

A second relevant point, drawn from Figure 3, is that learning HMM-Is was in fact limited: as HMM-Is tended to need more states to learn HMM-A distributions, their number of parameters was, on the other hand, limited due to overfitting avoidance. This was notoriously evident when learning from scarcer datasets. This learning difficulty could be in some degree mitigated (but not completely) by increasing the amount of data available to learn. Thus, although in theory the HMM-Is's limited representation can be compensated by expanding their state space, in practice this can limit to a large degree learning decent HMM-Is. On the other hand, the learned standard HMMs

also suffered from overfitting prominently, however not because of large transition matrices, but due to their graphical structure instead. Such structures made standard HMMs approach overfitting much faster than HMM-As and HMM-Is. The results indicate that HMM-As suffered substantially less from the issues faced by other HMMs, thanks to its more compact representation

n	k	r	t	k	r	t	k	r	t	n	k	r	t	k	r	t	k	r	t
	HMM-I			HMM-S			HMM-A				HMM-I			HMM-S			HMM-A		
Data = 300 seqs (l = 10)										**Data = 1500 seqs (l = 20)**									
3	3	17	8.7	3	26	16.2	2	10	13.1	**3**	4	27	92.2	2	15	79.2	2	12	93.1
6	9	134	44.9	2	73	26.3	2	33	12.0	**6**	13	246	580.6	2	129	190.9	2	39	122.3
10	10	199	81.1	2	141	48.8	2	57	27.0	**10**	23	758	1792.9	2	481	353.3	2	66	318.6
14	12	311	164.3	2	193	18.7	2	81	16.9	**14**	36	1799	4001.0	2	961	543.8	2	97	163.6
18	12	359	228.8	2	413	121.4	2	96	20.1	**18**	41	2418	5934.9	2	1167	831.1	2	103	155.1

(a) Number of states in reference models = 2.

n	k	r	t	k	r	t	k	r	t	n	k	r	t	k	r	t	k	r	t
	HMM-I			HMM-S			HMM-A				HMM-I			HMM-S			HMM-A		
Data = 300 seqs (l = 10)										**Data = 1500 seqs (l = 20)**									
3	4	27	14.2	4	43	27.6	3	19	25.0	**3**	7	69	187.4	6	77	203.6	7	80	259.9
6	6	71	37.2	4	119	56.4	5	86	85.0	**6**	17	390	812.9	6	407	437.1	7	196	647.0
10	13	298	134.8	3	320	80.6	6	167	181.2	**10**	29	1130	2309.8	5	2244	762.4	9	342	1493.0
14	17	526	241.2	2	399	105.1	6	240	281.4	**14**	39	2066	>4001.0	2	1909	718.2	8	364	1971.2
18	12	359	241.4	3	548	195.1	6	283	132.2	**18**	44	2727	>5939.9	3	4034	1196.6	6	347	820.1

(b) Number of states in reference models = 6.

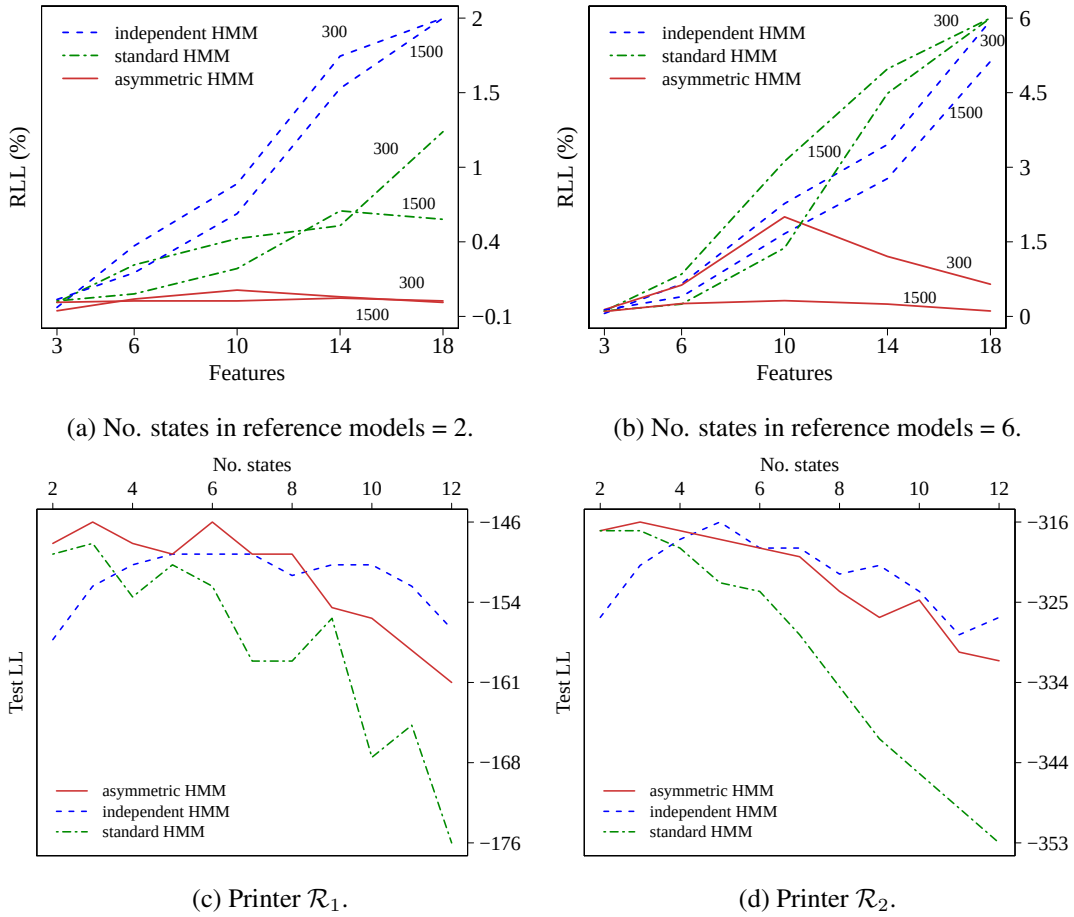
Table 1: Simulation results. A single model was used to sample different data on each line. Notation: k = no. states, r = no. parameters, **l** = seq. length, t = elapsed time (s), HMM-S: standard HMM.

6. Experiments with large-scale printers data

In this section, we discuss experiments on learning HMM-As on a real-world scenario originated from industrial printers (see Section 2). The model selection procedure of Section 5 was used to determine state spaces for these experiments. After discussing the fit of models, we delve into new problem insight that can be gained with HMM-As.

Data was gathered from two printers of the same printer family, denoted by $\mathcal{R}_1$ and $\mathcal{R}_2$. Each one receives different print jobs, e.g. in terms of ink used, time since last maintenance and several environmental parameters. The event-logs that we have to our disposal consist of a 1-year record of nozzle-related factors continuously monitored. We considered a key maintenance action that is performed regularly, and gathered data on nozzle-related components between each maintenance occurrence, such that each multivariate observation accounts for the following features: interval duration (i.e. the length of time since the previous maintenance action, denoted by L), total workload (W), other maintenance-related frequency (M), and color-related features (C_1, C_2, C_3 and C_4)[1]. The amounts of data available are 27 and 52 sequences for printers $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively, where each sequence lasts for 15 time points. Lastly, we considered the normalized log-likelihood to assess learned models, since there is no ground-truth model available. The normalized log-likelihood **NLL** of a model is defined as $\text{NLL} = -\text{LL}/(mTn)$, where m is the number of sequences, T the length

1. The complete description of the feature set is anonymized due to intellectual property restrictions.



(a) No. states in reference models = 2.

(b) No. states in reference models = 6.

(c) Printer $\mathcal{R}_1$.

(d) Printer $\mathcal{R}_2$.

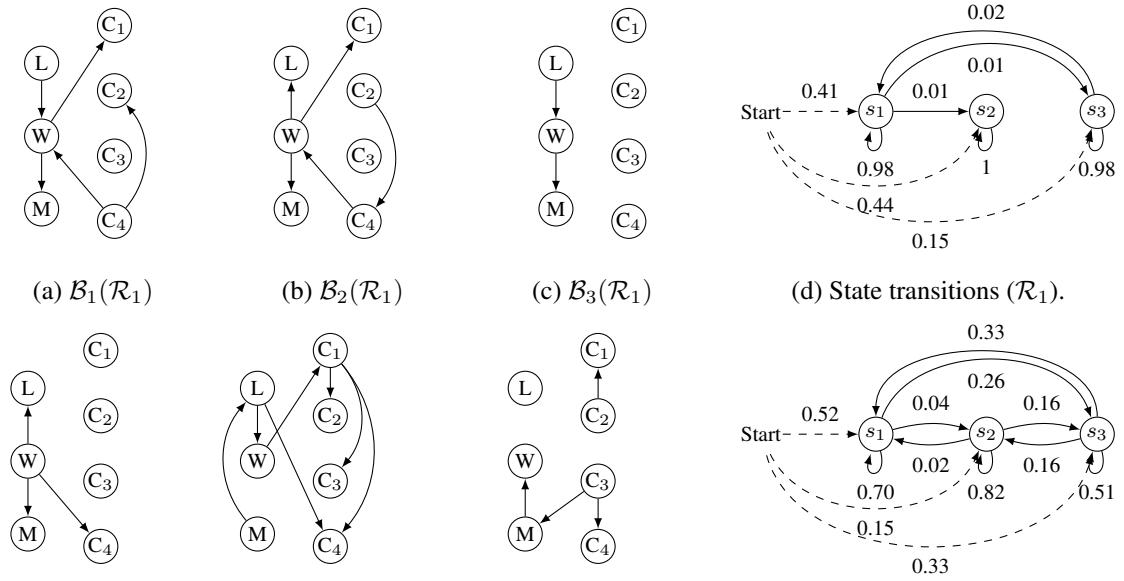
Figure 3: *Top*: Model fitting from simulated data (dataset sizes surround the curves). *Bottom*: Model fitting from printers data. (Y axes have different ranges.)

of each sequence, and n the number of features in the dataset. Hence, NLL is a real number in $[0, 1]$, and the smaller its value the better.

6.1 Results

The summary of fit assessment for independent, standard and asymmetric HMMs from printer datasets is shown in Figure 3 (bottom), indicating that the state space of learned HMM-Is had to be extended in order to improve the fit for capturing the feature set distribution. Although including more states could lead to better HMM-Is up to a certain point – both in $\mathcal{R}_1$ and $\mathcal{R}_2$ cases – overfitting took place after that, preventing from learning models of this family that fitted and generalized as well as asymmetric HMMs. Standard HMMs, in turn, could offer well-fitted solutions in the $\mathcal{R}_2$ case, however not in $\mathcal{R}_1$. On the other hand, HMM-As needed fewer states that allowed achieving better models prior to overfitting, especially in the scarcer $\mathcal{R}_1$ dataset. Here, it is noticeable that asymmetric models generalized better (see Figure 3c). Moreover, underlying non-empty structure could be identified in the printers problem by learning HMM-As, which presented multiple graphical structures in which there are diverse probabilistic influence within the feature set, as indicated in

Figure 4. Finally, we also learned dynamic Bayesian networks (DBNs) from printer datasets, which achieved mean test log-likelihoods equal to -254.8 for $\mathcal{R}_1$, and -359.8 for $\mathcal{R}_2$, indicating a worse performance compared to the HMMs. These results suggest that modeling hidden states provides a better description of the underlying process that explains the observed data.



(a) $\mathcal{B}_1(\mathcal{R}_1)$ (b) $\mathcal{B}_2(\mathcal{R}_1)$ (c) $\mathcal{B}_3(\mathcal{R}_1)$ (d) State transitions ($\mathcal{R}_1$).

(e) $\mathcal{B}_1(\mathcal{R}_2)$ (f) $\mathcal{B}_2(\mathcal{R}_2)$ (g) $\mathcal{B}_3(\mathcal{R}_2)$ (h) State transitions ($\mathcal{R}_2$).

Figure 4: Asymmetric Bayesian networks associated to the HMM-As learned from $\mathcal{R}_1$ and $\mathcal{R}_2$ printer datasets (parameters are not shown) and their probabilistic automata representation.

6.2 Problem insight

The state-transition dynamics of the HMM-As learned for the printers $\mathcal{R}_1$ and $\mathcal{R}_2$ are presented as simplified automata in Figures 4d and 4h. The diagrams and the associated asymmetric BNs in Figure 4 uncover different sequential behavior of the printers, not only quantitatively (i.e. transition probabilities) but also qualitatively (i.e. transitions between states with different structure). In particular, the model $\mathcal{R}_1$ shows that it is very unlikely to change between any of the states over time. On the other hand, while in model $\mathcal{R}_2$ it is still most likely that self-transitions will occur, it is substantially more likely that transitions between more structured states (e.g. states s_2 and s_3) and the less structured state (state s_1) compared to the transitions in $\mathcal{R}_1$. The amount of structure in each state reveals, for example, to which extent the failure rates of different ink colors (features C_1, C_2, C_3, and C_4) influence and are influenced by other features, such as the time since the last maintenance action and total workload on the period (features L and W respectively). Surprisingly, in some states of $\mathcal{R}_2$ (e.g. state s_1) the workload and duration of the interval do not interfere with the failure rate of colors C_1, C_2, and C_3, while this is not the case in state s_2, where all the 7 features are somehow connected. These points represent new problem insight that can be readily identified from HMM-As, as opposed to HMMs, which do not represent such qualitative information. As the goal of identifying HMM-As is to learn what features correlate to nozzle failures, future work

on comparing HMM-As with non-homogeneous dynamic Bayesian networks could show how each type of model capture different process stages and circumstances where feature dependences occur (Dondelinger et al., 2013; Bueno et al., 2016).

7. Conclusions

In this work, we introduced the family of Asymmetric Hidden Markov Models in order to generalize the representation of emission distributions of HMMs for arbitrary Bayesian-network distributions. While representing state-specific independencies explicitly, HMM-As render more compact models, arising a multitude of new possibilities. We empirically evaluated several aspects of HMM-As, concluding that the learning algorithm was effective in retrieving ground truth HMM-A distributions. The experiments also showed that HMM-As can better handle data scarcity than independent and standard HMMs in terms of goodness of fit.

In a case study from industrial large-scale printers, we obtained HMM-As with multiple state-specific graphical structures, allowing for new problem insight. Moreover, the use of this model has been useful to engineers, because it highlighted varied dependencies between features and failure modes, as well as uncovered multiple temporal machine behavior. Furthermore, HMM-As improved over HMMs, as they provided better fit with more compact state spaces than the latter. As further research, extensions to HMM-As can be considered, e.g. by allowing multiple chains of hidden states as in factorial HMMs (Ghahramani and Jordan, 1997). On the other hand, although this paper did not investigate the performance of HMM-As in probabilistic inference deeply, we believe that these can bring concrete benefits over general HMMs.

Acknowledgments

This work has been funded by NWO (Netherlands Organisation for Scientific Research). We thank Patrick Vestjens and Lou Sommers (Océ Technologies) for providing printer datasets.

References

J. Bilmes. Dynamic Bayesian multinets. In *Proc. of the Sixteenth conference on Uncertainty in Artificial Intelligence*, pages 38–45, 2000.

J. Bilmes. What HMMs can do. *IEICE - Trans. Inf. Syst.*, E89-D(3):869–891, Mar. 2006.

C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller. Context-specific independence in Bayesian networks. In *Proc. of the 20th UAI*, pages 115–123, 1996.

M. Bueno, A. Hommersom, P. Lucas, M. Lappenschaar, and J. Janzing. Understanding disease processes by partitioned dynamic Bayesian networks. *J.B. Informatics*, 61:283 – 297, 2016.

F. Dondelinger, S. Lèbre, and D. Husmeier. Non-homogeneous dynamic Bayesian networks with Bayesian regularization for inferring gene regulatory networks with gradually time-varying structure. *Machine Learning*, 90(2):191–230, 2013.

D. Freitag and A. McCallum. Information extraction with HMM structures learned by stochastic optimization. In *Proc. of the 17th AAAI and 20th IAAI*, pages 584–589. AAAI Press, 2000.

N. Friedman. Learning belief networks in the presence of missing values and hidden variables. In *Proc. of the 14th ICML*, ICML '97, pages 125–133, S. Francisco, USA, 1997. M. Kaufmann Pub.

N. Friedman and M. Goldszmidt. Learning Bayesian networks with local structure. In *Proc. of the 20th UAI*, UAI'96, pages 252–262, San Francisco, USA, 1996. M. Kaufmann Publishers.

D. Geiger and D. Heckerman. Knowledge representation and inference in similarity networks and Bayesian multinets. *Artificial Intelligence*, 82(1):45–74, 1996.

Z. Ghahramani. An introduction to hidden Markov models and Bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, pages 9–42, 2001.

Z. Ghahramani and M. Jordan. Factorial hidden Markov models. *Machine Learning*, 29(2):245–273, 1997.

S. Kirshner, P. Smyth, and A. Robertson. Conditional Chow-Liu tree structures for modeling discrete-valued vector time series. In *Proc. of the 20th UAI*, pages 317–324, 2004.

K. Markov, J. Dang, and S. Nakamura. Integration of articulatory and spectrum features based on the hybrid HMM/BN modeling framework. *Speech Communication*, 48(2):161 – 175, 2006.

G. Melançon and F. Philippe. Generating connected acyclic digraphs uniformly at random. *Information Processing Letters*, 90(4):209–213, 2004.

A. Motzek and R. Möller. Indirect causes in dynamic Bayesian networks revisited. In *Proc. of the 24th I.J.C. on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina*, pages 703–709, 2015.

K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, UC Berkeley, Computer Science Division, July 2002.

J. Pensar, H. Nyman, J. Lintusaari, and J. Corander. The role of local partial independence in learning of Bayesian networks. *Int. Journal of Approximate Reasoning*, 69:91 – 105, 2016.

A. Poritz. Linear predictive hidden Markov models and the speech signal. In *Acoustics, Speech, and Signal Processing, IEEE Int. Conf. on ICASSP '82.*, volume 7, pages 1291–1294, May 1982.

L. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

A. Rozinat, M. Veloso, and W. van der Aalst. Evaluating the quality of discovered process models. In *Proc. of Induction of Process Models (ECML PKDD)*, pages 45–52, September 2008.

M. Scutari and J. Denis. *Bayesian Networks with Examples in R*. Chap. and Hall, Boca Raton, 2014.

J. Vlasselaer, W. Meert, G. van den Broeck, and L. Raedt. Exploiting local and repeated structure in dynamic Bayesian networks. *Artificial Intelligence*, 232:43 – 53, 2016.

K. Won, A. Prugel-Bennett, and A. Krogh. Evolving the structure of hidden Markov models. *Trans. Evol. Comp*, 10(1):39–49, Sept. 2006.