Probabilistic Language Modeling with Hidden Stochastic Automata

Mark Andrews
Division of Psychology
Nottingham-Trent University
and
Division of Psychology and Language Sciences
University College London

Abstract

In this paper, we introduce a novel dynamical Bayesian network model for probabilistic language modeling. We refer to this as the *Hidden Stochastic Automaton*. This model, while based on a generalization of the Hidden Markov model, has qualitatively greater generative power than either the Hidden Markov model itself or any of its existing variants and generalizations. This allows the Hidden Stochastic Automaton to be used as a probabilistic model of natural languages in a way that is not possible with existing dynamical Bayesian networks. Its relevance to Cognitive Science is primarily as a computational — in the Marr (1982) sense of the term — model of cognition, but potentially also as a model of resource bounded cognitive processing, and as a model of the implementation of computation in physical dynamical systems.

A probabilistic language model is a hypothetical generative model of a language, where a language is defined most generally as a set of strings concatenated out of a finite set of symbols. By far the most widely used formalisms for specifying probabilistic language models are stochastic grammars, which are symbol rewriting rules with accompanying probabilities. The use of grammars is motivated by the fact that human languages are structurally complex, with properties that place them between the so-called context-free and context-sensitive formal languages (see, e.g., Chomsky, 1956, 1963; Shieber, 1985), and formal grammars are computationally universal in the sense that they can generate any recursively enumerable set (see, e.g., Hopcroft, Motwani, & Ullman, 2001).

By contrast to the case of language modeling, in probabilistic modeling more generally, the most widely used formalism for specifying probabilistic models is the *graphical model* (see, e.g., Koller & Friedman, 2009; Jordan, 2004). Graphical models are directed or undirected graphs whose vertices are identified with random variables and whose edges indicate conditional dependencies. The appeal of graphical models is their flexibility to represent complex relationships between large numbers of variables, and their graph-theoretic properties that afford general and computationally efficient algorithms for probabilistic inference, whether exactly or approximately by, for example, Monte Carlo methods. As a result, graphical models have effectively become a graph-based modeling language for developing and extending probabilistic models. They have had widespread application in fields such as bioinformatics (e.g., Friedman, 2004), computer vision (e.g., Oliver, Rosario, & Pentland, 2000), machine learning (e.g., Bishop, 2006, 2013), expert systems (e.g., Lauritzen & Spiegelhalter, 1988; Pearl, 1988), information retrieval (e.g., Salakhutdinov & Hinton, 2009), and in cognitive science (see, e.g., Chater & Oaksford, 2008; Griffiths, Chater, Kemp, Perfors, & Tenenbaum, 2010, for overviews).

Despite their breadth of appeal, graphical models have had a rather limited role as language models, if by language models we specifically mean generative models of language. There are at least two important reasons for this. On the one hand, stochastic grammars can not, in general, be represented as graphical models. (In some cases, notably stochastic regular grammars, the terminal and nonterminal variables of the grammar can be identified with vertices of a directed Markovian graph. For the super-regular grammars, however, this is not the case and the variables of the grammar can not be identified with the vertices of any fixed graph). On the other hand, the most widely used graphical models for sequential probabilisitic modeling, including the Hidden Markov model and its extensions, are limited in their generative power to the regular languages (i.e. the Type-3 languages in the Chomsky hierarchy). In other words, graphical models have had a relatively limited role as language models because the most widely used probabilistic models that have sufficient generative power to model human languages can not be represented as graphical models, and the most widely used graphical models for sequential structures do not have sufficient generative power to model natural languages.

There is, however, no inherent limitation to the generative power of graphical models. In this paper, we introduce a graphical model, specifically a dynamical Bayesian network, whose generative power is equivalent to that of an arbitrary stochastic grammar. This model, that we will refer to as the Hidden Stochastic Automaton, is based on a novel generalization of the widely used Hidden Markov model. As such, it retains many of the appealing characteristics of the Hidden Markov model while extending its generative power.

Hidden Stochastic Automata

To introduce the Hidden Stochastic Automaton (HSA), it is necessary to first briefly describe the Hidden Markov model (HMM). Given a set of J independent discrete

valued sequences $\mathbf{w}_1, \mathbf{w}_2 \ldots \mathbf{w}_j \ldots \mathbf{w}_J$, where the jth sequence is $\mathbf{w}_j = w_{j1}, w_{j2} \ldots w_{ji} \ldots w_{jn_j}$, the generative model assumed by the HMM treats each w_{ji} as drawn from one of K discrete probability distributions $\phi_1, \phi_2 \ldots \phi_k \ldots \phi_K$ over a finite vocabulary of length V. Which distribution is chosen for w_{ji} is determined by the value of the unobserved variable $x_{ji} \in \{1 \ldots K\}$ that corresponds to w_{ji}. For all j, each $x_{j1}, x_{j2} \ldots x_{ji} \ldots x_{jn_j}$ is a first-order Markov chain, with initial distribution π and a $K \times K$ transition matrix θ. More formally, the HMM assumes that for all j,

$$\begin{aligned} w_{ji}|x_{ji}, \phi &\sim \text{Categorical}(w_{ji}|\phi_{x_{ji}}) & 1 \leq i \leq n_j, \\ x_{ji}|\pi &\sim \text{Categorical}(x_{ji}|\pi) & i = 1, \\ x_{ji}|x_{ji-1}, \theta &\sim \text{Categorical}(x_{ji}|\theta_{x_{ji-1}}) & 1 < i \leq n_j. \end{aligned}$$

The graphical model for the HMM is shown below.

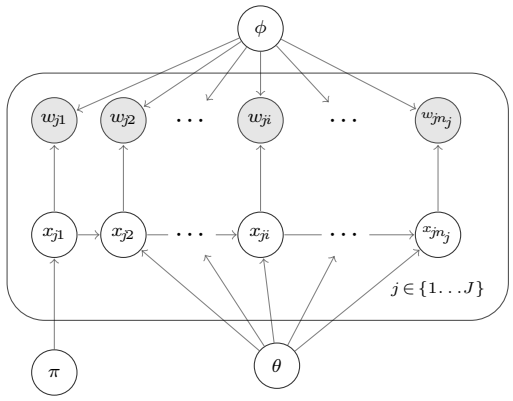

Figure 1: The graphical model or dynamical Bayesian network for the Hidden Markov model. The shaded nodes indicate the observed variables. For simplicity, we have omitted the priors on ϕ, π and θ.

Precisely because graphical models naturally afford generalizations and extensions, the HMM has lead to many variants. Most notably, these include the mixed memory HMM (Saul & Jordan, 1999), the coupled HMM (Brand, Oliver, & Pentland, 1997), the factorial HMM (Ghahramani & Jordan, 1997), and the hierarchical HMM (Fine, Singer, & Tishby, 1998). These extensions are often based on introducing additional chains of latent variables with varying degrees of conditional independence between them. Despite the evident value of these models, they do not qualitatively alter the formal generative complexity of the underlying model. In all of these extensions, the sequences generated are equivalent to regular or Type-3 formal languages

From HMM's to Hidden Stochastic Automata

It is possible, however, to generalize the HMM in such a way that its generative complexity is increased. This can be done by replacing the single valued x_{ji} in the HMM by a *variable sized array* or *vector*. In other words, while in the HMM, each state variable is $x_{ji} \in \{1 \ldots K\}$, this may be generalized to $x_{ji} \in \{1 \ldots K\}^*$. Here * indicates *Kleene star*, or the union of all concatenations of the elements from $\{1 \ldots K\}$ and $\{\emptyset\}$. This change clearly increases the cardinality of the state space to a countably infinite set. Importantly, however, as we will elaborate, if the set of operations that can increase or decrease the state-vector are limited to a finite set, and if the the conditional dependencies on this state-vector are limited to a finite range of elements, then inference in this generalized model is almost identical in kind to inference in the standard HMM.

For reasons that will be made clear, we will collectively refer to generalizations of the HMM using a state-vector as *Hidden Stochastic Automata* (HSA). For the purposes of this paper, however, we will mostly concentrate on one specific form of the HSA. For simplicity, we will also refer to this particular case of the model as the HSA, with the understanding that it is but one of many variants based on the same principles.

Just as with the HMM, the HSA is a generative model of discrete valued sequences. It assumes that each variable w_{ji} in the sequence of observations $\mathbf{w}_j = w_{j1}, w_{j2} \ldots w_{ji} \ldots w_{jn_j}$ is drawn from one of $(H+1) \times K$ discrete probability distributions $\phi_{01}, \phi_{02} \ldots \phi_{hk} \ldots \phi_{HK}$ over a length V vocabulary. Which of these $(H+1) \times K$ distributions is chosen is determined by the values of two latent or unobserved state variables that correspond to w_{ji}. On the one hand, there is a standard *finite state* variable $x_{ji} \in \{1 \ldots K\}$. On the other hand, there is an additional *state-vector* variable $z_{ji} \in \{1 \ldots H\}^*$, with w_{ji} being conditional on only the first element of z_{ji}, if $z_{ji} \neq \emptyset$. In other words, w_{ji} is sampled from $\phi_{[z^1_{ji}, x_{ji}]}$, where z^1_{ji} indicates the value of the first element of the state-vector z_{ji}, or else 0 when $z_{ji} = \emptyset$.

For all j, the sequence $(x_{j1}, z_{ji}), (x_{j2}, z_{ji}) \ldots (x_{ji}, z_{ji}) \ldots (x_{jn_j}, z_{ji})$ is a first-order Markov chain of coupled state variables. The distribution over x_{j1} is given by the K valued distribution π, and the value of z_{j1} is deterministically set to $z_{j1} = \emptyset$. For $1 < i \leq n_j$, both x_{ji} and z_{ji} are conditional on x_{ji-1} and, if $z_{ji} \neq \emptyset$, the first element of z_{ji}. The value of x_{ji} is determined by sampling from the K dimensional probability distribution specified by $\theta_{[z^1_{ji-1}, x_{ji-1}]}$, where θ is a $(H+1) \times K \times K$ stochastic transition matrix, and z^1_{ji-1} is as above. The value of z_{ji} is determined by applying one of $H+1$ different operations to z_{ji-1}, specifically prepending z_{ji-1} by one symbol from $\{1 \ldots H\}$ or removing the first element from z_{ji-1}. For example, if $\sigma_1\sigma_2\sigma_3$ (with each $\sigma_l \in \{1 \ldots H\}$) is the value of the state-vector z_{ji-1}, a possible sequence of

operations and their effect on the state-vector could be

$$z_{ji\text{–}1} = \sigma_1\sigma_2\sigma_3 \xrightarrow{\text{remove}} z_{ji} = \sigma_2\sigma_3,$$
$$z_{ji} = \sigma_2\sigma_3 \xrightarrow{\text{prepend 3}} z_{ji\text{+}1} = 3\sigma_2\sigma_3,$$
$$z_{ji\text{+}1} = 3\sigma_2\sigma_3 \xrightarrow{\text{prepend 2}} z_{ji\text{+}2} = 23\sigma_2\sigma_3.$$

Which of these $H + 1$ operations is applied is determined by sampling from the $H + 1$ dimensional probability distribution specified by $\Omega[z^1_{ji\text{–}1}, x_{ji\text{–}1}]$, where Ω is a $(H+1) \times K \times (H+1)$ stochastic transition matrix.

More formally, the probabilistic generative model defined by this HSA is, for $i \leq i \leq n_j$,

$$w_{ji}|x_{ji}, z_{ji}, \phi \sim \text{Categorical}(w_{ji}|\phi_{[z^1_{ji}, x_{ji}]}),$$

and for $i = 1$,

$$x_{ji}|\pi \sim \text{Categorical}(x_{ji}|\pi), \quad z_{ji} = \emptyset,$$

and for $1 < i \leq n_j$,

$$x_{ji}|x_{ji\text{–}1}, z_{ji\text{–}1}, \theta \sim \text{Categorical}(x_{ji}|\theta_{[z^1_{ji\text{–}1}, x_{ji\text{–}1}]}),$$
$$z_{ji}|u_{ji\text{–}1}, z_{ji\text{–}1} = O_{[u_{ji\text{–}1}]}(z_{ji\text{–}1}),$$
$$u_{ji\text{–}1}|x_{ji\text{–}1}, z_{ji\text{–}1}, \Omega \sim \text{Categorical}(u_{ji\text{–}1}|\Omega_{[z^1_{ji\text{–}1}, x_{ji\text{–}1}]}).$$

Here, we use the auxilary variable u_{ji} to refer to the operation applied to z_{ji}, and O is the set of $(H + 1)$ functions that map z_{ji} to $z_{ji\text{+}1}$ when these operations are applied. In other words, this makes clear that the value of $z_{ji\text{+}1}$ is a deterministic function of z_{ji} when the value of u_{ji} is known, but this value is stochastically conditional on x_{ji} and z_{ji}. In terms of the original variables, the graphical model for the HSA is as follows:

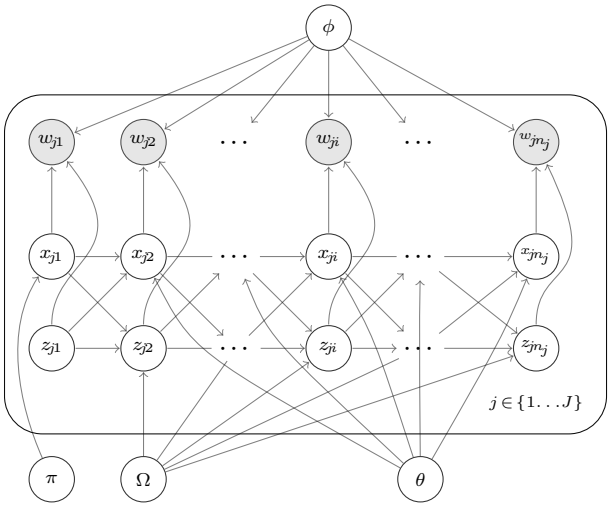

Figure 2: The graphical model or dynamical Bayesian network for the Hidden Stochastic Automaton. As with Figure 1, shaded nodes indicate observed variables and we have omitted the priors on ϕ, π, θ and Ω.

Generative Power of Hidden Stochastic Automata

The generative power of the HSA model (as shown in Figure 2) relative to that of the standard HMM (as shown in Figure 1) arises from the fact that the state-space of the state-vector z_{ji}, namely $\{1 \dots H\}^*$, is a countably infinite set yet the conditional relationships to and from z_{ji} are finitely specifiable. The consequences of this can be better appreciated by reference to discrete automata of the kind that form the foundations of theoretical computer science (see, e.g., Hopcroft et al., 2001).

As we have described it, the state-vector z_{ji} is identical to a pushdown stack with a symbol set $\{1 \dots H\}$. Prepending an element to the state-vector is equivalent to a push operation, while removing the first element is a pop operation. Assuming known values for Ω, which operation is applied to z_{ji} is dependent only on the value of the finite state variable $x_{ji\text{–}1}$ and the first element or *head* of $z_{ji\text{–}1}$. Likewise, assuming known values for θ, the value taken by x_{ji} is also dependent only on $x_{ji\text{–}1}$ and the head of $z_{ji\text{–}1}$. In other words, the HSA model described above is equivalent to a stochastic generative version of a pushdown stack automaton.

If we allow a greater variety of operations on the state-vector than just prepending or removing symbols from the left, the computational power of the HSA can be beyond that of a generative pushdown stack automaton. For example, if

$$\sigma_1\sigma_2\dot{\sigma}_3\sigma_4\sigma_5\sigma_6$$

is the value of the state-vector, we may treat an arbitrary element — in this cases σ_3 — as its *head*. If we allow for the appending of new elements to the left or the right of the head, or for the deleting of the element at the head, followed by the moving of the head pointer to the left or right, then this state-vector is equivalent to a two-way memory tape. As before, assuming known values for Ω, which of the operations is applied to the state-vector z_{ji} is again dependent only on the value of the finite state variable $x_{ji\text{–}1}$ and the head of $z_{ji\text{–}1}$. Likewise, as before, assuming known values for θ, the value taken by x_{ji} is also dependent only on $x_{ji\text{–}1}$ and the head element of $z_{ji\text{–}1}$. As such, with these changes the HSA is now equivalent to a stochastic generative version of the Turing machine.

Inference

As is clear from Figure 2, only the variables $\mathbf{w} = \{w_{j1} \dots w_{ji} \dots w_{jn_j}\}_{j=1}^{J}$ are observed. In general, therefore, the problem of inference in the HSA is the problem of inferring the joint posterior

$$\text{P}(\theta, \phi, \Omega, \pi, \mathbf{x}, \mathbf{z}|\mathbf{w}, \alpha, \beta, \gamma, \nu),$$

where $\mathbf{x}$ and $\mathbf{z}$ are the set of finite state and state-vectors variables, and α, β, γ, ν are the Dirichlet priors for θ, ϕ, Ω, π, respectively.

The procedure for inference that we will follow is to use a collapsed Gibbs sampler to draw samples from the posterior

$$\mathrm{P}(\mathbf{x}, \mathbf{z}|\mathbf{w}, \alpha, \beta, \gamma, \nu),$$

that integrates over the values of θ, ϕ, Ω, π. This Gibbs sampler is identical in nature to the collapsed sampler used in Andrews and Vigliocco (2010) for the case of a hierarchical mixture of Hidden Markov models.

For all $j \in \{1 \dots J\}$ and $i \in \{1 \dots n_j\}$, the Gibbs sampler iteratively draws samples from the posterior over x_{ji} and over z_{ji}, conditioned on sampled values for all remaining variables.

The posterior distribution over x_{ji}, conditioned on known values for all the other variables is[1]

$$\begin{aligned}
&\mathrm{P}(x_{ji}|w_{ji}, z_{ji}, x_{\neg ji}, w_{\neg ji}, z_{\neg ji}, \alpha, \beta, \gamma) \propto \\
&\int \mathrm{P}(w_{ji}|x_{ji}, z_{ji}, \phi)\mathrm{P}(\phi|w_{\neg ji}, x_{\neg ji}, z_{\neg ji}, \beta)d\phi \times \\
&\int \mathrm{P}(z_{ji+1}|x_{ji}, z_{ji}, \Omega)\mathrm{P}(\Omega|x_{\neg ji}, z_{\neg ji}, \gamma)d\Omega \times \\
&\int \mathrm{P}(x_{ji+1}|x_{ji}, z_{ji}, \theta)\mathrm{P}(x_{ji}|x_{ji-1}, z_{ji-1}, \theta)\mathrm{P}(\theta|x_{\neg ji}, z_{\neg ji}, \alpha)d\theta.
\end{aligned}$$

This leads to the following closed form:

$$\begin{aligned}
\mathrm{P}(x_{ji} = k|w_{ji}, z_{ji}, x_{\neg ji}, w_{\neg ji}, z_{\neg ji}, \alpha, \beta, \gamma) \\
\propto \frac{S^{\neg ji}_{hkv} + \beta_v}{S^{\neg ji}_{hk\cdot} + b} \times \frac{Q^{\neg ji}_{hkq} + \gamma_q}{Q^{\neg ji}_{hk\cdot} + c} \\
\times \frac{(R^{\neg ji}_{hkk_+} + \delta_{k_-,k,k_+} + \alpha_{k_+})(R^{\neg ji}_{h_-k_-k} + \alpha_k)}{R^{\neg ji}_{hk\cdot} + \delta_{k_-,k} + a}.
\end{aligned}$$

Here, we are assuming that the value of the observed variable at ji is v, the value of head of the state-vector at ji is h, its value at $ji-1$ is h_-, the value of the finite state variable at $ji-1$ is k_- and its value at $ji+1$ is k_+. The S, Q and R are rank-3 arrays of frequencies, with the superscript of $\neg ji$ indicating that they are based on excluding variables at ji. As such, $S^{\neg ji}_{hkv}$ is the number of times the observed variable has a value of v when the finite state variables has the value k and the head (e.g., the first) element of state-vector takes the value of $h \in \{0 \dots H\}$, $Q^{\neg ji}_{hkq}$ is the number of times that state-vector operation q occurs whenever the head element of the state-vector takes the value of k and the finite state variable takes the value of k, and $R^{\neg ji}_{hkk_+}$ gives the number of times the finite state variable takes the value k_+ whenever its value at the previous index is k and the value of the head of the state-vector at the previous index is h. The dot in place of the third index, e.g., $S^{\neg ji}_{hk\cdot}$, indicates a sum over the index. The term δ_{k_-,k,k_+} takes the value of 1 is $k_- = k = k_+$ and takes the value of zero otherwise. Likewise, $\delta_{k_-,k}$ takes the value of 1 when $k_- = k$, and takes the value of 0 otherwise. The terms a, b and c are the sums of α, β, γ, respectively.

For the case of the posterior distribution of the state-vector, it is sufficient to infer the distribution over operations applied to it. As mentioned, the value of the state-vector z_{ji} is deterministic function of z_{ji-1} when the operation u_{ji-1} is known. The posterior distribution over u_{ji} is given by

$$\begin{aligned}
&\mathrm{P}(u_{ji}|w_{ji}, z_{ji}, x_{\neg ji}, w_{\neg ji}, z_{\neg ji}, \alpha, \beta, \gamma) \propto \\
&\times \left[\int \mathrm{P}(w_{ji+1} \dots w_{jn_j}|x_{ji+1} \dots x_{jn_j}, z_{ji+1} \dots z_{jn_j}, \phi) \right. \\
&\qquad \left. \mathrm{P}(\phi|w_{\neg \vec{ji}}, x_{\neg \vec{ji}}, z_{\neg \vec{ji}}, \beta)d\phi\right] \\
&\times \left[\int \mathrm{P}(x_{ji+1} \dots x_{jn_j}|x_{ji} \dots x_{jn_j-1}, z_{ji} \dots z_{jn_j-1}, \theta) \right. \\
&\qquad \left. \mathrm{P}(\theta|x_{\neg \vec{ji}}, z_{\neg \vec{ji}}, \beta)d\theta\right] \\
&\times \mathrm{P}(z_{ji+1} \dots z_{jn_j}|u_{ji}, z_{ji}) \\
&\times \int \mathrm{P}(u_{ji}|x_{ji}, z_{ji}, \Omega)\mathrm{P}(\Omega|x_{\neg \vec{ji}}, z_{\neg \vec{ji}}, \gamma)d\Omega,
\end{aligned}$$

where we see that because a change to the operation u_{ji} deterministically changes the values of $z_{ji+1} \dots z_{jn_j}$, the likelihood terms for the u_{ji} variable include the variables $w_{ji+1} \dots w_{ji+1}$ and $x_{ji+1} \dots x_{ji+1}$[2]. In the above, the notation $\neg \vec{ji}$, e.g., in $x_{\neg \vec{ji}}$, indicates the exclusion of variables $ji \dots jn_j$. This distribution leads to the closed form

$$\begin{aligned}
&\mathrm{P}(u_{ji} = q|w_{ji}, z_{ji}, x_{\neg ji}, w_{\neg ji}, z_{\neg ji}, \alpha, \beta, \gamma) \propto \\
&\frac{\prod_{\{hkv:\, S^q_{hkv}>0\}} \prod_{s=0}^{S^q_{hkv}-1} S^{\neg \vec{ji}}_{hkv} + \beta_v + s}{\prod_{\{hk:\, S^q_{hk\cdot}>0\}} \prod_{s=0}^{S^q_{hk\cdot}-1} S^{\neg \vec{ji}}_{hk\cdot} + b + s} \times \\
&\frac{\prod_{\{hkq:\, Q^q_{hkq}>0\}} \prod_{s=0}^{Q^q_{hkq}-1} Q^{\neg \vec{ji}}_{hkq} + \gamma_q + s}{\prod_{\{hk:\, Q^q_{hk\cdot}>0\}} \prod_{s=0}^{Q^q_{hk\cdot}-1} Q^{\neg \vec{ji}}_{hk\cdot} + c + s} \times \\
&\frac{\prod_{\{hkl:\, R^q_{hkl}>0\}} \prod_{s=0}^{R^q_{hkl}-1} R^{\neg \vec{ji}}_{hkl} + \alpha_l + s}{\prod_{\{hk:\, R^q_{hk\cdot}>0\}} \prod_{s=0}^{R^q_{hk\cdot}-1} R^{\neg \vec{ji}}_{hk\cdot} + a + s}.
\end{aligned}$$

Here, $S^{\neg \vec{ji}}_{hkv}$, $Q^{\neg \vec{ji}}_{hkq}$ and $R^{\neg \vec{ji}}_{hkl}$ have the same meaning as $S^{\neg \vec{ji}}_{hkv}$, $Q^{\neg \vec{ji}}_{hkq}$ and $R^{\neg \vec{ji}}_{hkl}$ with the difference being that the frequencies are calculated excluding variables at the indices $ij \dots jn_j$. By contrast, the arrays S^q_{hkv}, Q^q_{hkq} and R^q_{hkl} are the frequencies of the co-occurrences the values

[1] We will provide the conditional distributions for values of x_{ji} and z_{ji} where $1 < i < n_j$. The distributions for the cases where $i = 1$ and $i = n_j$ require minor modifications, which we will omit here in the interests in space.

[2] In graphical model terms, the variables $w_{ji+1} \dots w_{ji+1}$, $x_{ji+1} \dots x_{ji+1}$ and $z_{ji+1} \dots z_{jn_j}$ are all *children* of u_{ji}.

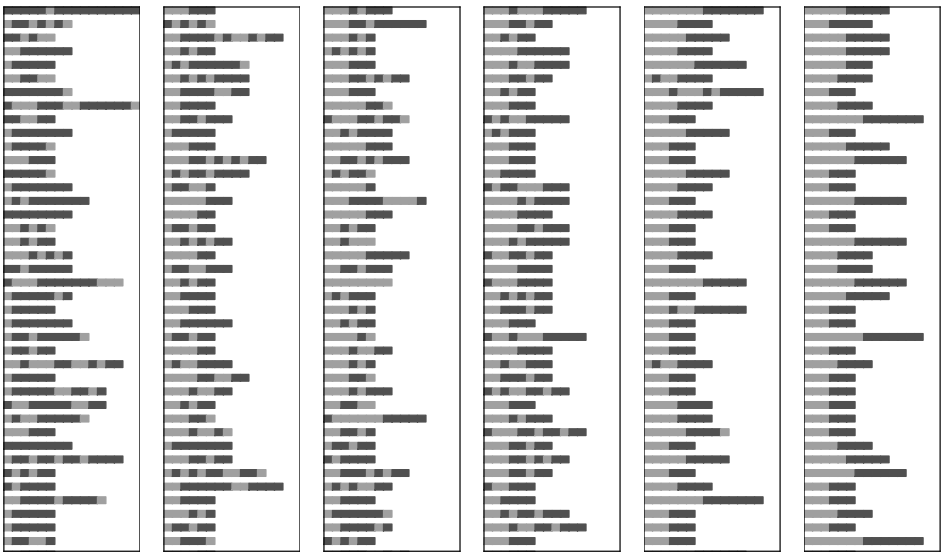

Figure 3: Strings generated by the probabilistic context-free grammar $S \to 0S1$ ($p = 0.66$), $S \to 01$ ($p = 0.34$) were used as observed data in a HSA. Shown above are samples of the binary strings generated by the HSA model on the basis of estimates of the parameters ϕ, θ, Ω and π after 3, 5, 10, 20, 50 and 100 iterations of the Gibbs sampler. The dark shade codes the value of 1. It is evident that by over 50 iterations, the HSA has inferred the correct generative model of the probabilistic language. By 100 iterations, it is only generating strings from the language $L = \{0^n 1^n : n \geq 0\}$.

of the variables *after* operation q is applied to the state-vector z_{ji+1} and the changes to the subsequent state-vectors are deterministically applied.

Demonstration

We demonstrate inference of a language from data by using the textbook example of a simple context-free language, namely $L = \{0^n 1^n : n \geq 0\}$. We can generate strings from a probabilistic version of this language using the probabilistic context-free grammar

$$\begin{aligned} S &\to 0S1, \quad p = 0.66, \\ &\to 01, \quad p = 0.34. \end{aligned}$$

We sample $J = 25$ strings from this language and use them as the data $\mathbf{w} = \mathbf{w}_1, \mathbf{w}_2 \ldots \mathbf{w}_j \ldots \mathbf{w}_J$ for a HSA model of the kind described.

Using the collapsed Gibbs sampler, we can sample from the posterior over the finite state and state-vector trajectories conditional on $\mathbf{w}$. From these, we may then draw sample estimates of ϕ, θ, Ω and π. Shown in Figure 3 are strings generated by the HSA model with parameters ϕ, θ, Ω and π as estimated after, from left to right, 3, 5, 10, 20, 50 and 100 iterations of the Gibbs sampler.

Relevance for Cognitive Science

Our initial motivation for the HSA model was put in terms of the computational advantages of graphical models as formalisms for probabilistic modeling. Graphical models, we have argued, have effectively become a graph-based modeling language for developing and extending probabilistic models. They have had a remarkable influence on the progress of probabilistic modeling in a wide variety of fields, including cognitive science. It is notable, therefore, that graphical models have had a relatively limited role in the probabilistic modeling of natural language. The obvious reason for this is due to the structurally complex nature of natural languages. While this structure is modeled well by probabilistic grammars, grammars can not, in general, be represented by graphical models. By contrast, the graphical models most widely used for modeling sequential data do not have the structural complexity necessary for modeling natural language.

We have introduced the HSA as a dynamical Bayesian network model that is capable of modeling structurally complex sequences. Its principal relevance to cognitive science is therefore as a computational model of cognition, where by *computational model* we specifically mean the Marr (1982) sense of the term: a model of the abstract nature of problem being faced and of its rational solution. However, the HSA model is potentially as relevant as a model of the resource limited practice, or possibly even the physical implementation, of cognition. For example, the HSA is an incremental state-space model, where inference is naturally modeled by the kind of sequential Monte Carlo methods, particularly particle filters, that have been advocated by, for

example, Griffiths, Vul, and Sanborn (2012); Sanborn, Griffiths, and Navarro (2010); Levy, Reali, and Griffiths (2009) as models of memory and time constrained approximations to rational computational models. On the other hand, from the point of view of physical implementation, the state-vector of the HSA can be represented naturally by a real-valued variable. If the state-vector is $\sigma_1\sigma_2\ldots\sigma_i\ldots\sigma_n$, this can be represented exactly by the real number $\sum_{i=1}^{n}\sigma_i(H+1)^{-i}$ and the operations applied to the state vector correspond to real-valued functions. For example, if the state-vector is binary, prepending a $\sigma\in\{0,1\}$ to $\sigma_1\sigma_2\ldots\sigma_i\ldots\sigma_n$ is identical to multiplying $\sum_{i=1}^{n}\sigma_i 2^{-i}$ by $\frac{1}{2}$ and adding $\frac{\sigma}{2}$. By treating the finite state variable as another real number, this allows us to represent the HSA exactly as a stochastic nonlinear dynamical system that is directly comparable to a recurrent neural network (see, e.g., Tabor, 2000, for related discussion).

References

Andrews, M., & Vigliocco, G. (2010). The Hidden Markov Topics Model: A Probabilistic Model of Semantic Representation. *Topics in Cognitive Science*, *2*, 101-113.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. New York, NY: Springer.

Bishop, C. M. (2013, Feb 13). Model-Based Machine Learning. *Philosophical Transactions of the Royal Society A - Mathematical Physical and Engineering Sciences*, *371*(1984).

Brand, M., Oliver, N., & Pentland, A. (1997). Coupled Hidden Markov Models for Complex Action Recognition. In *1997 IEEE Computer Society Conference On Computer Vision And Pattern Recognition, Proceedings* (p. 994-999).

Chater, N., & Oaksford, M. (2008). *The Probabilistic Mind*. Oxford, UK: Oxford University Press.

Chomsky, N. (1956). Three models for the description of language. *Institute of Radio Engineers Transactions on Information Theory*, *2*, 113-124.

Chomsky, N. (1963). Formal properties of grammars. In R. D. Luce, R. R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology* (Vol. 2, p. 323-418). New York and London: John Wiley and Sons, Inc.

Fine, S., Singer, Y., & Tishby, N. (1998). The Hierarchical Hidden Markov Model: Analysis and Applications. *Machine Learning*, *32*(1), 41–62.

Friedman, N. (2004, Feb 6). Inferring Cellular Networks using Probabilistic Graphical Models. *Science*, *303*(5659), 799-805.

Ghahramani, Z., & Jordan, M. (1997). Factorial Hidden Markov Models. *Machine Learning*, *29*, 245-273.

Griffiths, T. L., Chater, N., Kemp, C., Perfors, A., & Tenenbaum, J. B. (2010, Aug). Probabilistic Models of Cognition: Exploring Representations and Inductive Biases. *Trends in Cognitive Sciences*, *14*(8), 357-364.

Griffiths, T. L., Vul, E., & Sanborn, A. N. (2012). Bridging Levels of Analysis for Probabilistic Models of Cognition. *Current Directions in Psychological Science*, *21*(4), 26-268.

Hopcroft, J., Motwani, R., & Ullman, J. (2001). *Introduction to automata theory, languages and computation* (2nd ed.). Addison Wesley.

Jordan, M. I. (2004). Graphical models. *Statistical Science*, *19*(1), 140-155.

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: Principles and techniques*. Cambridge, MA: MIT Press.

Lauritzen, S., & Spiegelhalter, D. (1988). Local Computations With Probabilities On Graphical Structures And Their Application To Expert Systems. *Journal Of The Royal Statistical Society Series B-Methodological*, *50*(2), 157-224.

Levy, R., Reali, F., & Griffiths, T. L. (2009). Modeling the Effects of Memory on Human Online Sentence Processing with Particle Filters. In *Advances in Neural Information Processing Systems* (Vol. 21, p. 937-944).

Marr, D. (1982). *Vision*. New York, NY: W. H. Freeman & Company.

Oliver, N., Rosario, B., & Pentland, A. (2000, Aug). A Bayesian Computer Vision System for Modeling Human Interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(8), 831-843.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann.

Salakhutdinov, R., & Hinton, G. (2009, Jul). Semantic Hashing. *International Journal of Approximate Reasoning*, *50*(7), 969-978.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2010). Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, *117*, 1144-1167.

Saul, L., & Jordan, M. (1999, Oct). Mixed Memory Markov Models: Decomposing Complex Stochastic Processes as Mixtures of Simpler Ones. *Machine Learning*, *37*(1), 75-87.

Shieber, S. M. (1985). Evidence Against the Context-Freeness of Natural-Language. *Linguistics And Philosophy*, *8*(3), 333-343.

Tabor, W. (2000). Fractal encoding of context-free grammars in connectionist networks. *Expert Systems*, *17*(1), 41-56.