Approximate Kalman Filters for Embedding Author-Word Co-occurrence Data over Time

Purnamrita Sarkar[1], Sajid M. Siddiqi[2], and Geoffrey J. Gordon[1]

[1] Machine Learning Department,
Carnegie Mellon University, Pittsburgh, PA 15213
{psarkar,ggordon}@cs.cmu.edu
[2] Robotics Institute,
Carnegie Mellon University, Pittsburgh, PA 15213
siddiqi@cs.cmu.edu

Abstract. We address the problem of embedding entities into Euclidean space over time based on co-occurrence data. We extend the CODE model of [1] to a dynamic setting. This leads to a non-standard factored state space model with real-valued hidden parent nodes and discrete observation nodes. We investigate the use of variational approximations applied to the observation model that allow us to formulate the entire dynamic model as a Kalman filter. Applying this model to temporal co-occurrence data yields posterior distributions of entity coordinates in Euclidean space that are updated over time. Initial results on per-year co-occurrences of authors and words in the NIPS corpus and on synthetic data, including videos of dynamic embeddings, seem to indicate that the model results in embeddings of co-occurrence data that are meaningful both temporally and contextually.

1 Introduction

Embedding discrete entities into Euclidean space is an important area of research for obtaining interpretable representations of relationships between objects. This is very useful for visualization, clustering and exploratory data analysis. Recent work [1] proposes a novel technique for embedding heterogeneous entities such as author-names and paper keywords into a single Euclidean space based on their co-occurrence counts. When applied to the NIPS corpus, the resulting clusters of keywords and authors reflect real-life relationships between different research areas and researchers in those respective areas. However, it would be interesting to see how these relationships evolve over time, an aspect which these techniques do not address. Recent work has examined the dynamic behavior of social networks [2], but only with homogeneous entities, and with point estimates of the embedding coordinates. The problem we are interested in differs in two ways: first, embedding time-series co-occurrence data from two kinds of entities (essentially weighted link data from a bipartite graph) in a dynamic model could be useful for temporal data visualization, link prediction and group detection in such networks. Examples of such bipartite data are author-word co-occurrences

E.M. Airoldi et al. (Eds.): ICML 2006 Ws, LNCS 4503, pp. 126–139, 2007.

in conference proceedings over time, actor-director collaborations throughout their careers, and so on. Second, modelling a *distribution* over the coordinates of these embeddings instead of point estimates (as in [2]) would tell us about the correlation and uncertainty in the entities' coordinates. In this paper, we explore one possible approach to achieve both these goals.

The layout of the rest of this paper is as follows. We discuss some related work, in particular the model of [1] which we utilize. We then extend this model to the dynamic case, describing how our dynamic model can be used for posterior estimation using a Kalman filter after some approximations. The resulting model keeps track of the belief state over all author and word coordinates in the latent space based on the approximated co-occurrence observation model and a zero-mean Gaussian transition model. We give derivations and intuition for the operation of this dynamic model, as well as results on the NIPS corpus of author-word co-occurrence data and on synthetic data.

2 Related Work

The problem of embedding discrete entities into euclidean space is well-studied. Principal Components Analysis (PCA) is a standard technique based on eigen-decomposition of the counts matrix [3]. Multi-Dimensional Scaling (MDS) [4] is another technique. However, these techniques are not suitable for temporal data if one wishes to enforce smoothness constraints on embeddings over time.

[5] introduced a model similar to MDS in which entities are associated with locations in p-dimensional space, and links are more likely if the entities are close in latent space. However their work does not take the sequential aspect of the data into account. Also, the distribution over latent positions are obtained by sampling, which becomes intractable for large networks. Their work also assumes binary link data.

The most closely related work is the CODE model of [1], which gives a technique for embedding heterogenous entities (such as authors and keywords) based on co-occurence data for the static case. We briefly introduce their model here, and our notation is similar to theirs.

The basic model of CODE is a conditional model $p(w|a)$, where w denotes the words and a denotes the authors. Let ϕ_i and ψ_j denote the hidden variables representing the coordinates of author a_i and word w_j in the latent space respectively. By $\Phi_t(A)$, $\Psi_t(W)$ we represent the states related to all author and word positions at timestep t. The conditional probability of seeing word w_j given an author a_i is related (inversely) to the distance $d_{ij} = |\phi_i - \psi_j|$ of author i and word j in the latent space, as well as the marginal counts of each individual entity, $\bar{p}(a_i)$ and $\bar{p}(w_j)$. For latent coordinates in a d dimensional space,

$$\begin{aligned} p(w_j|a_i) &= \tfrac{\bar{p}(w_j)}{Z(a_i)} e^{-|\phi_i - \psi_j|^2} \\ Z(a_i) &= \textstyle\sum_{w_j} \bar{p}(w_j) e^{-|\phi_i - \psi_j|^2} \\ |\phi_i - \psi_j|^2 &= \textstyle\sum_{k=1}^{d} (\phi_i^k - \psi_j^k)^2 \end{aligned} \tag{1}$$

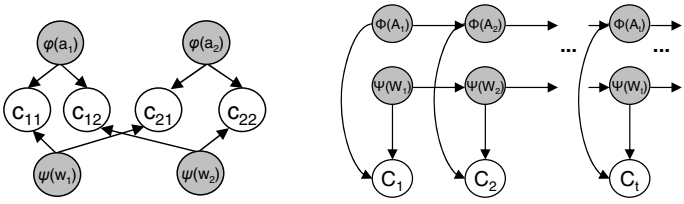


Fig. 1. Shaded nodes indicate hidden random variables. (A) The graphical model relating author/keyword positions to co-occurrence counts at a single timestep. (B) The corresponding factored state-space model for temporal inference.

The hidden coordinates $\Phi_t(A)$, $\Psi_t(W)$ are learned by maximizing the likelihood objective function using conjugate gradient or other such techniques.

3 The Single-Timestep Model

The original conditional model was chosen by considering $\frac{p(w|a)}{\bar{p}(w)}$ to be inversely proportional to the exponentiated squared distance between the latent embeddings $\phi(a)$ and $\psi(w)$. Similarly, our model of the joint is motivated by considering the initial ratio to be $\frac{p(w,a)}{\bar{p}(w)\bar{p}(a)}$ instead, and deriving the resultant $p(w, a)$. The reason for dividing by the empirical marginals is to normalize the joint by the overall frequencies of the individual entities in the joint. This represents the single timestep graphical model shown in Figure 1(A). The resultant $p(w, a)$ is as follows:

$$\begin{aligned} p(a_i, w_j|\phi_i, \psi_j) &= \tfrac{1}{Z}\bar{p}(a_i)\bar{p}(w_j)e^{-|\phi_i-\psi_j|^2} \\ Z &= \textstyle\sum_{a_i}\sum_{w_j}\bar{p}(a_i)\bar{p}(w_j)e^{-|\phi_i-\psi_j|^2} \end{aligned} \tag{2}$$

4 Dynamic Embedding of Co-occurrence Data Through Time

We consider the unknown coordinates of authors and words to be hidden variables in a latent space. Our goal is now to estimate these continuous hidden variables given discrete co-occurrence observations. As shown above, we model the joint posterior probability of author and word coordinates (given the observations) based on the distances between those coordinates. To make the problem tractable, we aim to derive a Gaussian distribution that is somehow close to our observation model, which would allow us to use Kalman Filters, which are described below. The natural approach which we follow is to minimize the KL-divergence of a Gaussian distribution (as an approximation to the observation model) and the normalized likelihood of our model. However, this turns out to be difficult since the KL-divergence has no closed-form solution, mainly due to the non-standard $\log(Z)$ term (where Z is defined in equation (2). We investigate two methods for making this expression tractable and obtaining a

Gaussian that approximates the observation model. We will see how the approximated model, together with a Gaussian transition model for the coordinates, can be formulated as a standard dynamic model.

4.1 The State-Space Model

For our state-space model in the dynamic setting, we choose a factored state space model as shown in Figure 1(B), similar to a factorial HMM [6] or switching state space model [7]. It is a natural choice over the full joint model because we consider the hidden coordinates of authors and words to be decoupled Markov chains conditionally coupled given their co-occurrence. This model closely resembles the factorial HMM model yet is distinct because of the hidden variables being real-valued. Exact filtering and smoothing are very difficult in this model because the prior belief state is not conjugate to the discrete observation density for typical belief distribution choices like the Normal distribution. Instead, we would like to approximate this exact model in order to formulate it as a Kalman Filter.

4.2 Kalman Filters

A Kalman filter [8] is a linear chain graphical model with a backbone of hidden real-valued states emitting a real-valued observation at every timestep. Both the observation and transition models are assumed to be Gaussian. It is commonly used in tracking the states of complex systems or locations of moving objects such as robots or missiles. Filtering and smoothing are tractable in this model because of the conjugacy of the Gaussian distribution to itself, which enables the belief state to remain Normally distributed at each timestep after the three standard steps of *conditioning* (factoring in a new observation to the current belief state), *prediction* (propogating the belief through the transition model) and *rollup* (marginalizing to obtain the new belief state). These steps are described in more detail below.

4.3 Kalman Filter Formulation for Dynamic Embedding

In a standard Kalman Filter, all three steps mentioned above have closed form solutions, i.e.:

$$\begin{aligned} &\text{Conditioning: } P(\Phi_t, \Psi_t | C_{1:t-1}, C_t = c_t) \\ &\propto P(C_t = c_t | \Phi_t, \Psi_t) P(\Phi_t, \Psi_t | C_{1:t-1}) \\ &\text{Prediction and Rollup: } P(\Phi_{t+1}, \Psi_{t+1} | C_{1:t}) \\ &= \int_{\Phi_t} \int_{\Psi_t} P(\Phi_{t+1}, \Psi_{t+1} | \Phi_t, \Psi_t) P(\Phi_t, \Psi_t | C_{1:t}) \partial\Phi_t \partial\Psi_t \end{aligned} \tag{3}$$

These are the Kalman filter updates in our model. Lets see what happens for our model in the conditioning step. The observation model is:

$$\begin{aligned} &\log p(C_t | \Phi_t, \Psi_t) \\ &= -\sum_{a_i} \sum_{w_j} \bar{p}(a_i, w_j) |\phi_{t,i} - \psi_{t,j}|^2 - \log Z \end{aligned} \tag{4}$$

However, this is not a Gaussian kernel, so we do not have a closed form update equation available. Now we look at approximations to project this family of density functions to a Gaussian, in order to overcome this problem.

4.4 Approximate Conditioning Step

A simple approach: Jensen's Inequality. One natural approach is to apply Jensen's inequality to approximate the difficult portion of the likelihood (i.e. the $\log Z$ term), which happens to be concave. However as we shall see, this approximation causes us to lose much of the information encoded in the normalization constant, and will not be used in our final model. The log normalizing function of our joint model is

$$logZ = log(\sum_{a_i}\sum_{w_j} \bar{p}(a_i)\bar{p}(w_j)e^{-||\phi_{t,i}-\psi_{t,j}||^2}) \tag{5}$$

Using Jensen's inequality,

$$logZ \geq -\sum_{a_i}\sum_{w_j} \bar{p}(a_i)\bar{p}(w_j)||\phi_{t,i}-\psi_{t,j}||^2 \tag{6}$$

This gives us a lower bound on the KL divergence between an approximate Gaussian distribution p and our distribution q. We denote $p(a_i)$ by p_i and $p(w_j)$ by p_j. We also denote by χ the random variables $< \Phi, \Psi >$. Maximizing the KL divergence (details in the Appendix) gives us the parameters for the closest Gaussian approximation to our observation model with mean zero and covariance Σ given by the following equation.

$$\Sigma^{-1} = 2\hat{\Lambda} \tag{7}$$

Where $\hat{\Lambda}$ is defined as follows:

$$\hat{\Lambda}_{ij} = \begin{cases} \sum_j \tilde{c}_{ij} I_{2\times 2} & j = i,\ 1 \leq i \leq 2A-1 \\ \sum_i \tilde{c}_{ij} I_{2\times 2} & i = j,\ 2A+1 \leq j \leq 2(A+W)-1 \\ -2\tilde{c}_{ij} I_{2\times 2} & i \neq j,\ 1 \leq i \leq 2A-1, \\ & \quad 2A+1 \leq j \leq 2(A+W)-1 \\ 0_{2\times 2} & \text{otherwise} \end{cases} \tag{8}$$

In the above equation $\tilde{c}_{ij} = \bar{p}_{ij} - \bar{p}_i\bar{p}_j$. Note that there is no correlation between the x and y coordinates in this model. It is clear that the numerator of our observation model doesn't give rise to any such correlation.

However the log-normalization constant gives rise to such correlation, which is clear from figure 2. Unfortunately this approximation removes the correlations between the x, y coordinates as we can see from equation 8. Having uncorrelated x and y coordinates implies that higher-dimensional embeddings are not beneficial, and that we may as well be embedding to a line. In practice, this model often leaves us with such an embedding even when the space is two-dimensional,

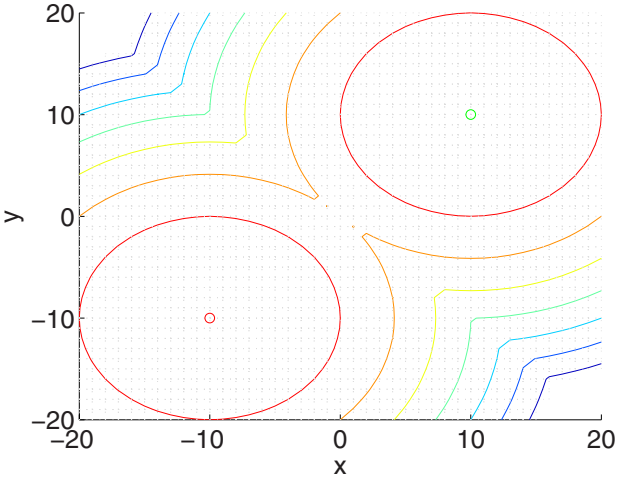


Fig. 2. A plot of the log normalizing constant $log(e^{(-(x-a)^2-(y-b)^2)}+e^{(-(x-c)^2-(y-d)^2)})$ for two given coordinates a, b and c, d. Two things are apparent: the correlation of x and y coordinates , and the presence of multiple optima in this function. We desire an approximation that preserves the $x-y$ correlation.

since we are optimizing over the two dimensions independently. Also the mean of the observation model is zero. Also this method is effectively minimizing a lower bound on the KL divergence, which is not necessarily beneficial. We therefore look for a better model.

A more sophisticated approach: Taylor approximation of a variational upper bound. Now we try and come up with a model which preserves the correlations between the axes. We look at a variational upper bound on the log normalizing constant [9].

$$\log Z \leq \lambda \sum_{ij} \overline{p}_i \overline{p}_j e^{-(\phi_i-\psi_j)^T(\phi_i-\psi_j)} - 1 - \log \lambda$$

Minimizing this upper bound effectively minimizes an upper bound on the KL-divergence. However, direct minimization of this bound is difficult because of the term inside the expectation, and because the expression is not convex. Instead, we take a second order Taylor approximation of the $e^{-(\phi_i-\psi_j)^T(\phi_i-\psi_j)}$ values around ξ_i, ξ_j. A Taylor approximation of a function $g(x)$ is given by,

$$g(x) = g(0) + x^T[\frac{\partial g}{\partial x_1}, \frac{\partial g}{\partial x_2}]_{\xi_i,\xi_j} + \frac{1}{2}x^T H(\xi_i, \xi_j)x$$

Where $H(\xi_i, \xi_j)$ is the Hessian of the function evaluated at ξ_i, ξ_j.

Now we have a Gaussian approximation to our observation model, which has canonical parameters Λ, η. These parameters , as derived in the appendix, are functions of the Jacobian and Hessian matrix of the taylor approximation, evaluated at ξ_i, ξ_j. We shall describe how we choose these parameters later in this section.

In (3), we multiply two Gaussians i.e. prior $p(\Phi_t, \Psi_t | C_{1:t-1})$ with canonical parameters $(\eta_{t|t-1}, \Lambda_{t|t-1})$ and the approximate observation distribution with η, Λ. The notation $\eta_{t|t-1}$ denotes the value of a parameter at time t conditioned on observations from timesteps $1 \ldots t-1$. The resulting Gaussian $p(\Phi_t, \Psi_t | C_{1:t})$ is distributed with $\eta_{t|t}, \Lambda_{t|t}$, where

$$\eta_{t|t} = \eta_{t|t-1} + \eta$$
$$\Lambda_{t|t} = \Lambda_{t|t-1} + \Lambda$$

We compute the moment parameters $\mu_{t|t}$, $\Sigma_{t|t}$ from the canonical parameters. And we get the $\eta_{t|t-1}, \Lambda_{t|t-1}$ from the previous time-step of the Kalman Filter.

When applying the Taylor expansion, we set the ξ values to the $\mu_{t|t-1}$ learnt from the previous timestep. We found this to be most effective, and this also makes sense since given the former time-steps' data we are most likely to be around the conditional means predicted from the former time-steps. Because of the nonconvex structure of the log-normalizer, which is due to the presence of saddle points (Figure 2), the resulting Λ can become non-positive definite and have negative eigenvalues. To project to the closest possible positive definite matrix, we set the negative eigenvalues to zero (plus a small positive constant). Together these approximations succeed in giving us a tractable expression while not losing the highly informative inter-coordinate interactions (e.g. x-y correlation in two dimensions) that the simple Jensen's inequality approach would discard.

4.5 Prediction and Rollup Step

Our transition model is very simple, just a zero-mean symmetric increase in uncertainty:

$$(\Phi_{t+1}, \Psi_{t+1}) = (\Phi_t, \Psi_t) + N(0, \Sigma_{transition})$$

Here $\Sigma_{transition}$ is a diagonal noise term denoting the spread of uncertainty along both axes, which must be fixed beforehand. The prediction and rollup steps give the following result:

$$(\Phi_{t+1}, \Psi_{t+1}) \sim N(\mu_{t+1|t}, \Sigma_{t+1|t})$$

where $\mu_{t+1|t} = \mu_{t|t}$ and $\Sigma_{t+1|t} = \Sigma_{t|t} + \Sigma_{transition}$.

4.6 Computational Issues

Note that we model all author-word interactions with a *single* large Kalman filter, where the authors and words relate through the covariance matrix. This introduces complexity issues since the size of the covariance matrix is proportional to the number of authors and words. However some sparseness properties of the covariance matrix can be exploited for faster computation.

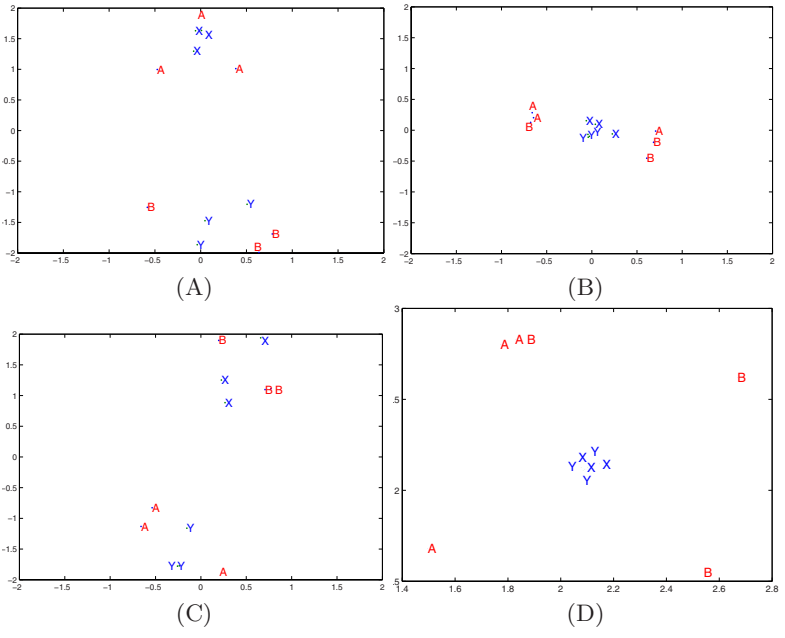


Fig. 3. Dynamic embedding of synthetic data vs. static embedding. A, B are two groups of authors and X, Y are two groups of words. The 140-timestep data smoothly varies from strong A-X and B-Y links to strong A-Y and B-X links. The entities are initialized randomly (not shown). A. $t = 20$, strong A-X and B-Y links. B. $t = 70$, Intermediate configuration, noisy uniform links. C. Strong A-Y and B-X links. D. A static embedding of the aggregate co-occurrence matrix, which is effectively a noisy uniform matrix, resulting in entities mixing with each other.

5 Experiments

We divide the results section in three parts. We present some snapshots from our algorithm on embeddings of a synthetic datasets with pre-specified dynamic structure. We then present snapshots and closeups of embeddings of author-word co-occurrence data from the NIPS corpus over thirteen years. We also show how the distance in our embedding between author-word pairs in the corpus evolve over time. In all cases, $\Sigma_{transition}$ is currently set heuristically to give a smoothly varying embedding that is still responsive to new data. We finish our experimental section with a comparison with PCA [3], a well-studied static embedding technique.

5.1 Modeling Trends over Time

We wish to inspect the performance of dynamic embedding in cases where the underlying model is known. To do this, we generate noisy co-occurrence matrices of 3 words and 3 authors over 140 timesteps. The matrices have some amount

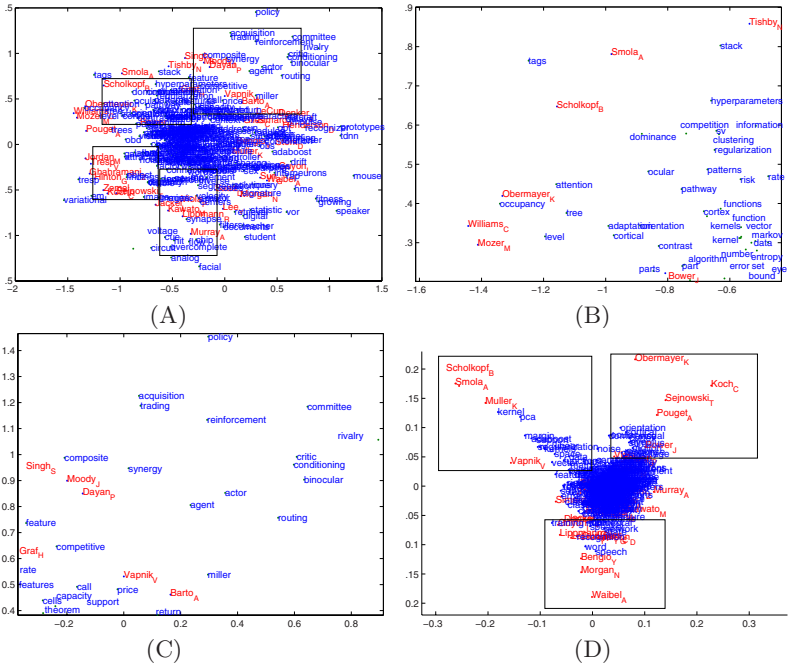


Fig. 4. (A). $t = 13$ Dynamic embedding of NIPS data (final timestep, 1999). (B),(C). Close-ups of (roughly) the top two rectangles in (A). The first Both contain authors and keyword groups that are interrelated (e.g. (B) contains entities related to kernels, (C) contains reinforcement-learning-related terms and authots. (D). PCA embedding of aggregate counts matrix of NIPS data, that averages out any sequential patterns.

of random sparseness in every timestep, to be more realistic. We divide the authors in two groups, namely A, B and the words in two groups X, Y. We vary the co-occurrences between these groups smoothly such that in the first 20 steps, authors A have high co-occurrence counts with X, and B with Y, whereas the A-Y and B-X counts are very low. After $t = 20$, this pattern starts becoming less sharp, blending to a completely uniform matrix with noise at $t = 70$. From then until $t = 120$, the authors and words "switch" i.e. A-Y and B-X counts begin to dominate. From $t = 120$ to 140, the data continues to reflect strong A-Y and B-X co-occurrences. A movie with this and other dynamic embeddings is available at http://www.cs.cmu.edu/∼psarkar/icml06/. Figure 3(A,B,C) shows three snapshots from a dynamic embedding of this data sequence, which clearly reflect the underlying dynamic structure at different timesteps. In contrast, Figure 3(D) shows a static embedding of the aggregate summed counts matrix, which happens to be approximately uniform and thus not indicative of any interesting structure in the data.

5.2 The NIPS Corpus

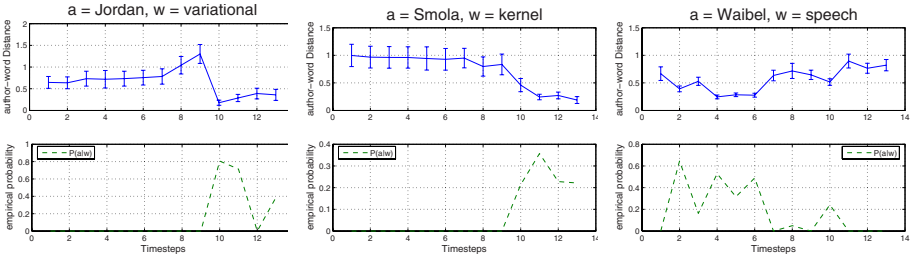
In this section we shall look at word-author co-occurrence data over thirteen years from the NIPS proceedings of 1986-1999. We implemented the dynamic Kalman filter models on a subset of the NIPS dataset. The NIPS data corpus[1] contains co-occurrence count data for 13,649 words and 2,037 authors appearing together in papers from 1986 to 1999. We partitioned this data into yearly raw count matrices using additional information in the dataset, and picked a set of well-known authors and meaningful keywords. The experiments shown here are carried out on small subsets of authors and words in order to get easily interpretable 2-D plots for this paper, however the algorithm scales well to larger sets.

Qualitative Analysis. The resulting embedding has some very interesting properties. The words on different parts of it define different areas of machine learning. We also find the corresponding authors in those areas. For example in figure 4(A) we have presented the embedding of 40 authors and 428 words. These are the overall most popular authors, and the words they tend to use.

We can divide the area in the figure in four clear areas, within the rectangles. The top right region magnified in Figure 4(C) has words like `reinforcement`, `agent`, `actor`, `policy` which clearly are words from the field of reinforcement learning. We also have authors such as `Singh`, `Dayan` and `Barto` in the same area. `Dayan` is known to have worked on `acquisition` and `trading` which are also words in this region. However the very neighboring region on the left belongs to words like `kernel`, `regularization`, `error` and `bound`. We see some overlap with that region via the entities `support` and `Vapnik`. Also one of the other two interesting regions consists of authors `Jordan`, `Hinton`, `Gharamani Zemel`, `Tresp`. The lowest rectangular region is filled with words and authors like `image`, `segmentation`, `motion`, `movement`. Notably we find that author `Viola` is placed very close to these words and words like `document`, `retrieval`,`facial`. Also we have author `Murray` co-placed with words `voltage`, `circuit`, `chip`, `analog`, `synapse`. These are strongly supported by the co-occurrence data and anecdotal evidence.

Quantitative Analysis. A single embedding does not tell us whether our algorithm models dynamic structure. To investigate this aspect, in Figure 5 we plot the average distance per timestep between three word-author pairs of interest, along with the empirical probability of that pair per timestep, to see whether the distances correlate to the probabilities. As we can see in the bottom panels of Figures 5, (`Jordan`,`variational`) and (`Smola`,`kernel`) have high empirical probabilities in the later timesteps, corresponding to drops in the distance between these entities' coordinates. In contrast, (`Waibel`,`speech`) co-occurs mostly in the first half of the data set, and so we see the distance between the author-word embeddings shrinking initially then gradually increasing over time.

[1] http://www.cs.toronto.edu/ ∼ roweis/data.html



(A) (B) (C)

Fig. 5. Average distance between author-word pairs over time (above), along with corresponding empirical probabilities (below). A. Jordan and variational. B. Smola and kernel C. Waibel and speech. The graphs on the bottom reflect empirical $\overline{p}(author \mid word)$ from the NIPS data which varies inversely over time with the average author-word distance in the embedding shown in the top row, demonstrating the responsiveness of the embeddings to the underlying data.

5.3 Comparison with PCA

An embedding of the aggregate data with PCA is shown in Figure 4(D). The embedding reflects relationships in the overall data very well, as seen in the three rectangles highlighted. For example, one of them has entities like `Scholkopf`, `Smola`, `kernel` and `pca`, and the others also have consistent sets of authors and the keywords they are known to use. However the data fails to capture dynamic trends in the data that our model successfully reflects. For example, `Waibel` and `speech` do not co-occur at all in the latter timesteps of the dataset, as is clear from the lower panel of Figure 5(C). However, since the aggregate counts matrix embedded by static PCA averages out all sequential structure, `Waibel` and `speech` are still relatively close in the PCA embedding.

6 Conclusion and Future Work

We have proposed and demonstrated a model for Euclidean embedding of co-occurrence data over time by formulating the problem as a factored state space model, and used an approximation to yield a tractable Kalman filter formulation. The resulting model gives us an estimate of the posterior distribution over the coordinates of the entities in latent space. The previous work we are extending addresses this problem only for the single-timestep case, giving only point estimates for the coordinates. Experimental results show that our model yields interpretable visual results and reflects dynamic trends in the data. For future work we will implement smoothing in the dynamic model to see if it offers improved results over filtering. We will also obtain quantitative results for the model on problems such as link prediction in social networks and classification in word-document embedding.

Acknowledgements

We warmly thank Carlos Guestrin for his guidance. This work was funded in part by DARPA's CS2P program under grant number HR0011-006-1-0023. The opinions and conclusions expressed are the authors'.

References

1. Globerson, A., Chechik, G., Pereira, F., Tishby, N.: Euclidean embedding of co-occurrence data. In: Proc. Eighteenth Annual Conf. on Neural Info. Proc. Systems (NIPS). (2004)
2. Sarkar, P., Moore, A.: Dynamic social network analysis using latent space models. In: Proc. Nineteenth Annual Conf. on Neural Info. Proc. Systems (NIPS). (2005)
3. Berry, M., Dumais, S., Letsche, T.: Computational methods for intelligent information access. In: Proceedings of Supercomputing. (1995)
4. Breiger, R.L., Boorman, S.A., Arabie, P.: An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling. J. of Math. Psych. **12** (1975) 328–383
5. Raftery, A.E., Handcock, M.S., Hoff, P.D.: Latent space approaches to social network analysis. J. Amer. Stat. Assoc. **15** (2002) 460
6. Ghahramani, Z., Jordan, M.I.: Factorial hidden Markov models. In Touretzky, D.S., Mozer, M.C., Hasselmo, M.E., eds.: Proc. Conf. Advances in Neural Information Processing Systems, NIPS. Volume 8., MIT Press (1995) 472–478
7. Ghahramani, Z., Hinton, G.E.: Switching state-space models. Technical report, 6 King's College Road, Toronto M5S 3H5, Canada (1998)
8. Kalman, R.: A new approach to linear filtering and prediction problems. (1960)
9. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An Introduction to Variational Methods for Graphical Methods. Machine Learning (1998)

Appendix

In this section we give a detailed description of the derivations.

Derivation of Section 4.4

We compute the KL projection of our observation model (p) to the closest Gaussian family (q).

$$\begin{aligned} &D(p,q) = \int p \ln p - \int p \ln q \\ &= -H(p) + \int (\textstyle\sum_{ij} \overline{p}_{ij} (\phi_i - \psi_j)^T (\phi_i - \psi_j)) dp + E_p(\ln Z) \\ &= -(A+W) - \frac{\ln((2\pi)^{2(A+W)}|\Sigma|)}{2} \\ &+ E_p(\textstyle\sum_{ij} \overline{p}_{ij} (\phi_i - \psi_j)^T (\phi_i - \psi_j)) + E_p(\ln Z) \end{aligned} \tag{9}$$

Using equations 5 and 6 we get a lower bound on equation 9.

$$\begin{aligned} &D(p,q) \geq -(A+W) - \frac{\ln((2\pi)^{2(A+W)}|\Sigma|)}{2} \\ &+ E_p(\textstyle\sum_{ij} (\overline{p}_{ij} - \overline{p}_i \overline{p}_j)(\phi_i - \psi_j)^T (\phi_i - \psi_j)) \\ &\geq -(A+W) - \frac{\ln((2\pi)^{2(A+W)}|\Sigma|)}{2} + E_p(\chi^T \hat{A} \chi) \end{aligned}$$

We get the expression in equation 8 by parameter matching. Differentiating the above equation w.r.t Σ gives us the parameters for the closest Gaussian we project our distribution into.

Derivation of Section 4.4

Now we derive the approximate observation model using Taylor expansion of the exponentiated distance term of the normalization constant, i.e. $e^{-(\phi_i-\psi_j)^T(\phi_i-\psi_j)}$ around parameters ξ_i, ξ_j. We define the gradient (∇) and Hessian (H) for our function. The gradient is defined as follows:

$$\nabla_1(\xi_i,\xi_j) = (\frac{\partial g}{\partial \phi_i})_{\xi_i,\xi_j} = -2e^{-(\xi_i-\xi_j)^T(\xi_i-\xi_j)}(\phi_i-\psi_j)$$
$$\nabla_2(\xi_i,\xi_j) = (\frac{\partial g}{\partial \psi_j})_{\xi_i,\xi_j} = -\nabla_1(\xi_i,\xi_j)$$

$$H = \begin{pmatrix} \frac{\partial^2 g}{\partial \Phi_t^T \partial \Phi_t^T} & \frac{\partial^2 g}{\partial \Psi_t^T \partial \Phi_t} \\ \frac{\partial^2 g}{\partial \Phi_t^T \partial \Psi_t} & \frac{\partial^2 g}{\partial \Psi_t^T \partial \Psi_t^T} \end{pmatrix}_{\xi_i,\xi_j} = \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix}$$

The second order approximation of $e^{-(\phi_i-\psi_j)^T(\phi_i-\psi_j)}$ gives

$$\begin{aligned} &1 + \phi_i^T\nabla_1 + \psi_j^T\nabla_2 + \tfrac{1}{2}[\Phi_t^T\Psi_t^T]H(\xi_i,\xi_j)[\Phi_t\Psi_t] \\ &= 1 + \tfrac{1}{2}[\phi_i^T H_{11}\phi_i + \psi_j^T H_{21}\phi_i + \phi_i^T H_{12}\psi_j + \psi_j^T H_{22}\psi_j] \end{aligned} \tag{10}$$

Where $H(\xi_i,\xi_j)$ is H evaluated at ξ_i, ξ_j. For our purpose these values evaluate to the following:

$$\begin{aligned} H_{11} &= 2e^{-(\xi_i-\xi_j)^T(\xi_i-\xi_j)}(2(\xi_i-\xi_j)(\xi_i-\xi_j)^T - I) \\ H_{12} &= -H_{11} \\ H_{21} &= -H_{11}^T \\ H_{22} &= H_{22} \end{aligned} \tag{11}$$

We also define the following symmetric matrix $\overline{\eta}$ and $\overline{\Lambda}$ for making the derivations simple. Also here $\overline{\eta}$ is $2(A+W)$ a dimensional vector and $\overline{\Lambda}$ is a $2(A+W), 2(A+W)$ dimensional symmetric matrix. By i we denote author i and by j we index word j.

$$\begin{aligned} \overline{\eta}_i &= \overline{p}_i \textstyle\sum_j \overline{p}_j \nabla_1(\xi_i,\xi_j) \\ \overline{\eta}_j &= \overline{p}_j \textstyle\sum_i \overline{p}_i \nabla_2(\xi_i,\xi_j) \end{aligned} \tag{12}$$

$$\begin{aligned} \overline{\Lambda}_{ii} &= \overline{p}_i \textstyle\sum_j \overline{p}_j H_{11}(\xi_i,\xi_j) \\ \overline{\Lambda}_{jj} &= \overline{p}_j \textstyle\sum_i \overline{p}_i H_{22}(\xi_i,\xi_j) \\ \overline{\Lambda}_{ij} &= \overline{p}_i\overline{p}_j H_{12}(\xi_i,\xi_j) \end{aligned} \tag{13}$$

Now using equations (10), (13) and (11) the expectation of the log normalizing constant under the new distribution becomes:

$$\begin{aligned}
&E_p(\textstyle\sum_{ij} \overline{p}_i \overline{p}_j e^{-(\phi_i - \psi_j)^T(\phi_i - \psi_j)}) \\
&= c + E_p[\textstyle\sum_i \phi_i^T \eta_i + \sum_j \psi_j^T \eta_j] + \\
&\tfrac{1}{2} E_p[\textstyle\sum_i \phi_i^T \overline{\Lambda}_{ii} \phi_i + 2 \sum_{ij} \phi_i^T \overline{\Lambda}_{ij} \psi_i + \sum_j \phi_j^T \overline{\Lambda}_{jj} \phi_j] \\
&= c + E_p[\chi^T \overline{\eta}] + \tfrac{1}{2} E_p[\chi^T \overline{\Lambda} \chi] \\
&= c + \mu^T \overline{\eta} + \tfrac{1}{2} Tr((\mu\mu^T + \Sigma)\overline{\Lambda})
\end{aligned}$$

All terms independent of μ, Σ are combined in the constant term c. Hence the approximation of $D(p, q)$ comes out to be,

$$\begin{aligned}
&D(p,q) \approx C - \tfrac{1}{2} ln|\Sigma| + tr((\mu\mu^T + \Sigma))\tilde{\Lambda}) + \lambda \mu^T \overline{\eta} + \\
&\tfrac{\lambda}{2} Tr((\mu\mu^T + \Sigma)\overline{\Lambda})
\end{aligned}$$

A derivative w.r.t Σ and μ yields

$$\begin{aligned}
&\Lambda = \Sigma^{-1} = 2(\tilde{\Lambda} + \tfrac{\lambda}{2}\overline{\Lambda}) \\
&\eta = -\lambda \overline{\eta}
\end{aligned}$$

which are the required parameters for the Gaussian approximation of the observation model used in the Kalman filter.