A Latent Space Approach to Dynamic Embedding of Co-occurrence Data

Purnamrita Sarkar
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213

Sajid M. Siddiqi
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Geoffrey J. Gordon
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

We consider dynamic co-occurrence data, such as author-word links in papers published in successive years of the same conference. For static co-occurrence data, researchers often seek an embedding of the entities (authors and words) into a low-dimensional Euclidean space. We generalize a recent static co-occurrence model, the CODE model of Globerson et al. (2004), to the dynamic setting: we seek coordinates for each entity at each time step. The coordinates can change with time to explain new observations, but since large changes are improbable, we can exploit data at previous and subsequent steps to find a better explanation for current observations. To make inference tractable, we show how to approximate our observation model with a Gaussian distribution, allowing the use of a Kalman filter for tractable inference. The result is the first algorithm for dynamic embedding of co-occurrence data which provides distributional information for its coordinate estimates. We demonstrate our model both on synthetic data and on author-word data from the NIPS corpus, showing that it produces intuitively reasonable embeddings. We also provide evidence for the usefulness of our model by its performance on an author-prediction task.

1 Introduction

Suppose we have a graph whose nodes represent entities and whose links represent associations. It is common to ask how we can embed such a graph in a low-dimensional Euclidean space so that nodes which share links tend to be close to one another. Graphs like this often arise when analyzing social networks, distributions of document topics, co-authorship patterns, or recommender systems; the resulting embeddings are useful for tasks like clustering, visualization, information retrieval, and exploratory data analysis.

Well-known algorithms for the embedding problem include MDS (Borg & Groenen, 1997), IsoMap (Tenenbaum et al., 2000), Locally Linear Embedding (LLE) (Roweis & Saul, 2000) and spectral clustering (Ng et al., 2001). (Raftery et al., 2002) 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. Recent work (Globerson et al., 2004) proposes a novel technique for embedding heterogeneous entities such as author-names and paper keywords into a single space based on co-occurrence counts.

Now suppose that, instead of a single graph, we observe a series of graphs which evolve over time, such as co-authorship links for successive years of the same conference. We could embed each year's graph separately and attempt to align the embeddings from successive years to analyze trends. However, there is no reason to suppose that embeddings from different years would be consistent with one another. Furthermore, we might have limited data for each year, in which case we would expect to find a better embedding for each year by taking into account information from neighboring years. In the limit we might have only a few data points at each time step (for example, imagine co-authorship data from a journal which publishes only a few articles in an issue), and embedding just a single time step's graph would yield very poor results.

In this paper, we extend Globerson et al.'s CODE algorithm to handle time series data. The result is D-CODE, a model for dynamic co-occurrence data which provides full distributional information about the embedding coordinates. We show how to approximate the observation model in D-CODE with a Gaussian distribution, so that we can use a Kalman filter to infer coordinates efficiently. The validity of our approxima-

Table 1: Notation

Symbol	Definition
$\overline{p}_a(i),\ \overline{p}_i$	Marginal empirical prob. of author i
$\overline{p}_w(j),\ \overline{p}_j$	Marginal empirical prob. of word j
$\overline{p}(a_i, w_j), \overline{p}_{ij}$	Joint emp. prob. of author i, word j
A, W, T	Number of authors,words,timesteps
$\Phi_t(A)$	All author coordinates at t
$\Psi_t(W)$	All word coordinates at t
C_t	Co-occurrence counts matrix at t
ξ_i	Author coordinate of Taylor expansion
ζ_j	Word coordinate of Taylor expansion
$\eta,\ \Lambda$	Canonical parameters of a Gaussian
$\mu,\ \Sigma$	Moment parameters of a Gaussian
$\eta_{t\|t-1}$	η_t conditioned on $C_1 \dots C_{t-1}$

tion is demonstrated experimentally by showing that the resulting embeddings are qualitatively sensible and that they outperform sensible baseline models on a prediction task.

While some authors, e.g. Sarkar and Moore (2005), have previously considered time-series co-occurrence data, D-CODE is to our knowledge the first dynamic embedding algorithm which provides principled uncertainty estimates for its coordinates. We present experiments which demonstrate that D-CODE finds high-quality embeddings as well as that D-CODE's uncertainty estimates allow more accurate answers to questions such as the most likely author for a given paper. We also compare our algorithm's performance with the most natural alternative statistical algorithm, namely PCA over overlapping windows of data, described in more detail in the Experiments section.

2 Preliminaries

Our aim is to model the cross-interactions of two sets of entities over time, denoted by $\{a_i\}_{i=1}^M$ and $\{w_j\}_{j=1}^N$ respectively. Data is given to us as a sequence of *co-occurrence-count matrices* $\{C_t\}_{t=1}^T$, where T denotes the number of discrete timesteps. Thus $C_t(i,j)$ denotes the number of times entity a_i interacted with entity w_j at time t. In much of this paper, our entity sets are assumed to consist of authors $\{a_i\}$ and words $\{w_j\}$, and $C_t(i,j)$ denotes the number of times word w_j was used by author a_i in papers published in year t. For any particular timestep t, we can normalize the counts matrix to obtain joint and marginal empirical probabilities, denoted by $\overline{p}_t(a_i, w_j)$ and $\overline{p}_t(a_i), \overline{p}_t(w_j)$. We will drop the subscript t from these probabilities since it will be clear from the context.

As in other embedding scenarios, we assume that entity-pair interactions in the data can be explained by real-valued latent variables residing in a low-dimensional space $\mathcal{R}^k$. Let ϕ_i and ψ_j denote the latent random variables corresponding to author a_i and word w_j respectively. By Φ_t and Ψ_t we represent all author and word coordinates at time t. We would like to be able to say that pairs of entities closer together in $\mathcal{R}^k$ will interact more often. In particular, we will model the count $C_t(i,j)$ as being inversely proportional to the exponentiated squared distance $d_{ij} = ||\phi_i - \psi_j||^2$ between the latent variables of author a_i and word w_j. We would like our embeddings to exhibit temporal consistency as well, i.e. coordinates at time t should reflect evidence $\{C_1, \dots, C_t\}$ accumulated until that time, and not just the data seen at time t.

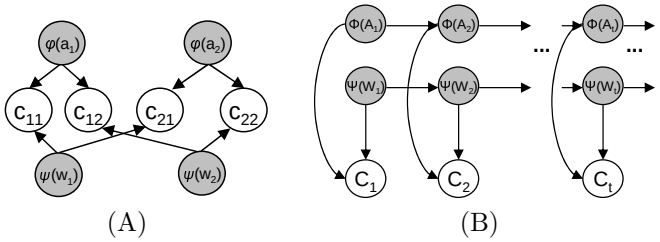


Figure 1: Shaded nodes indicate hidden random variables. (A) The graphical model relating author/keyword positions to co-occurrence counts at a single timestep. (B) The corresponding factored state-space model for temporal inference.

When talking about multivariate Gaussians, we use both the well-known moment parameters (μ, Σ), i.e. the mean and covariance matrix, and the canonical parameters (η, Λ), where $\Lambda = \Sigma^{-1}$ is the precision matrix and $\eta = \Sigma^{-1}\mu$. Additionally, the notation $\eta_{t|t-1}$ denotes the value of a parameter η at time t conditioned on observations from timesteps 1 to $t-1$. Table 1 contains the most commonly used notation.

3 The Algorithm

3.1 The Single-Timestep Model

As in the CODE model, our basic building block is the probability of a single pairwise author-word interaction given the respective coordinates ϕ_i, ψ_j:

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

This represents the single-timestep graphical model shown in Figure 1(A). For an entire counts matrix at timestep t, we get the following likelihood (up to a constant depending on the total number of interactions):

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

CODE obtains point estimates for ϕ and ψ via gradient descent. Since we treat these latent variables probabilistically but would also like to be computationally

efficient, we choose to model the distribution over entity coordinates at time t, i.e. $P(\Phi_t, \Psi_t \mid C_{1:t-1})$, as a Gaussian distribution.

However, even if we instantiate this joint distribution and initialize it with reasonable values, we cannot obtain a closed form update for $P(\Phi_{t+1}, \Psi_{t+1} \mid C_{1:t})$ at the next timestep since the observation likelihood (2) is not Gaussian. So, we will approximate the true observation model by a Gaussian as well, in Section 3.2.1.

3.2 Extension to Dynamic Embedding

The natural choice for our dynamic model is a Kalman Filter (Kalman, 1960), as shown in Figure 1(B). The three standard steps of *conditioning* (factoring in a new observation to the current belief state), *prediction* (propagating the belief through the transition model) and *rollup* (marginalizing to obtain the new belief state) are all closed-form updates assuming that the observation and transition models are Gaussian:

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

The conditioning step decreases uncertainty in the system, and the prediction step increases uncertainty. The conditioning step corresponds a simple addition of corresponding canonical parameters. The resulting Gaussian is characterized by:

$$\begin{aligned}\Phi_t, \Psi_t | C_{1:t} &\sim N(\eta_{t|t}, \Lambda_{t|t}) \\ \eta_{t|t} &= \eta_{t|t-1} + \eta_{obs} \\ \Lambda_{t|t} &= \Lambda_{t|t-1} + \Lambda_{obs}\end{aligned} \tag{4}$$

where $(\eta_{obs}, \Lambda_{obs})$ are canonical parameters of the observation model. Let us now examine the prediction step. Our transition model is a zero-mean symmetric increase in uncertainty by adding a diagonal noise term, in order to inject uncertainty without biasing the coordinates in any particular direction:

$$\begin{aligned}\Phi_{t+1}, \Psi_{t+1} | C_{1:t} &\sim N(\mu_{t+1|t}, \Sigma_{t+1|t}) \\ \mu_{t+1|t} &= \mu_{t|t} \\ \Sigma_{t+1|t} &= \Sigma_{t|t} + \Sigma_{transition}\end{aligned} \tag{5}$$

This step controls the degree of diffusion in author and word positions between consecutive timesteps. The $\Sigma_{transition}$ parameter balances the tradeoff between temporal consistency and the effect of new evidence.

For the first timestep, we initialize all coordinates around the origin with some perturbation. We set the initial estimate of the covariance matrix to reflect a high degree of uncertainty, in order to allow the embedding to adapt to the initial observations.

3.2.1 Approximate Conditioning Step

In order to obtain a tractable Kalman Filter, we approximate the observation model (2) by a joint Gaussian over all entity positions. However, we are unable to obtain a closed-form solution due to the log-normalization constant $\log Z$. We address this problem by approximating $\log Z$ by Taylor expansions around suitably chosen points. This leads to closed-form Gaussian parameters as a solution, which is our desired approximate observation model. Though this is an approximation without guarantees, we will show that the resulting models (a) sensibly represent uncertainty in entity coordinates, and (b) outperform alternative models in an author prediction task, indicating the validity of the approximation. Some details are in the Appendix.

First-order Taylor approximation of $\log Z$ around z gives

$$\log Z \approx \lambda Z - 1 - \log \lambda \tag{6}$$

where $\lambda = 1/z$, and z is the value of Z at $\phi_i = \xi_i, \psi_j = \zeta_j \quad \forall\ i, j$. However, direct maximization of the log-likelihood is still difficult since the normalization constant is a sum of exponentiated terms. Therefore, we do a second order Taylor expansion of the exponentiated distance term $g([\phi_i \ \ \psi_j]) = e^{-(\phi_i - \psi_j)^T(\phi_i - \psi_j)}$ around $\xi_i, \zeta_j \ \forall\ i, j$. We set the ξ, ζ values to $\mu_{t|t-1}$, the predicted conditional means based on the previous timesteps. We found this to be most effective, and this also makes sense since $\mu_{t|t-1}$ is the most likely value in the absence of any information.

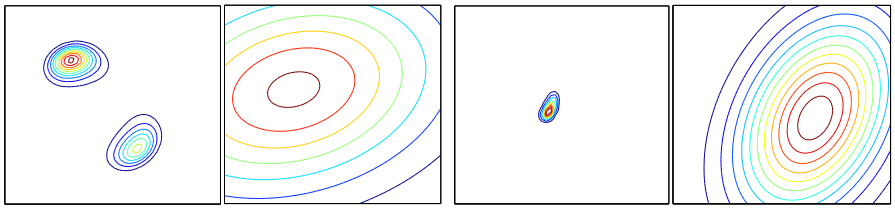
The second-order Taylor approximation of $g(x)$ is:

$$g(x) = g(0) + x^T \nabla(\xi_i, \zeta_j) + \frac{1}{2} x^T H(\xi_i, \zeta_j) x \tag{7}$$

where $\nabla(\xi_i, \zeta_j)$ and $H(\xi_i, \zeta_j)$ are the gradient and Hessian of $g(x)$ respectively, evaluated at ξ_i, ζ_j. They are defined as follows:

$$\begin{aligned}\nabla_1(\xi_i, \zeta_j) &= (\tfrac{\partial g}{\partial \phi_i})_{\xi_i, \zeta_j} = -2e^{-(\xi_i - \zeta_j)^T(\xi_i - \zeta_j)}(\phi_i - \psi_j) \\ \nabla_2(\xi_i, \zeta_j) &= (\tfrac{\partial g}{\partial \psi_j})_{\xi_i, \zeta_j} = -\nabla_1(\xi_i, \zeta_j)\end{aligned} \tag{8}$$

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



(A) (B)

Figure 2: Two pairs of contour plots of an author's true posterior conditional (left panel) in a 3-author, 5-word embedding and the corresponding approximate Gaussian posterior conditional obtained (right panel). B is a difficult-to-approximate bimodal case, C is an easier unimodal case.

$$H_{11} = 2e^{-(\xi_i-\zeta_j)^T(\xi_i-\zeta_j)}(2(\xi_i-\zeta_j)(\xi_i-\zeta_j)^T - I)$$
$$H_{12} = -H_{11},\ H_{21} = -H_{11}^T,\ H_{22} = H_{22} \tag{9}$$

A few more terms are required for our main result. Define $\tilde{\Lambda}$ to be a symmetric matrix of size $k(A+W) \times k(A+W)$, as in (10). I is the $k \times k$ identity matrix, where k is the embedding dimension. In our experiments we use $k = 2$. Let $\tilde{\Lambda}(a, b)$ denote the 2×2 block $\tilde{\Lambda}([2a-1, 2a], [2b-1, 2b])$.

$$\begin{aligned}\tilde{\Lambda}(i,i) &= \textstyle\sum_j \overline{p}_{ij} I_{2\times 2} \\ \tilde{\Lambda}(A+j, A+j) &= \textstyle\sum_i \overline{p}_{ij} I_{2\times 2} \\ \tilde{\Lambda}(i, A+j) &= -2\overline{p}_{ij} I_{2\times 2}\end{aligned} \tag{10}$$

where $i \in 1:A,\ j \in 1:W$. All other entries are zero. Define $\overline{\eta}$ to be a vector of length $k(A+W)$ and $\overline{\Lambda}$ to be a symmetric matrix of size $k(A+W) \times k(A+W)$, as in (11) and (12). Let $\overline{\eta}(a)$ denote the 2×1 sub-vector $\overline{\eta}([2a-1, 2a])$.

$$\begin{aligned}\overline{\eta}(i) &= \overline{p}_i \textstyle\sum_j \overline{p}_j \nabla_1(\xi_i, \zeta_j) \\ \overline{\eta}(A+j) &= \overline{p}_j \textstyle\sum_i \overline{p}_i \nabla_2(\xi_i, \zeta_j)\end{aligned} \tag{11}$$

Using the same notation as $\tilde{\Lambda}$ above, we define

$$\begin{aligned}\overline{\Lambda}(i,i) &= \overline{p}_i \textstyle\sum_j \overline{p}_j H_{11}(\xi_i, \zeta_j) \\ \overline{\Lambda}(A+j, A+j) &= \overline{p}_j \textstyle\sum_i \overline{p}_i H_{22}(\xi_i, \zeta_j) \\ \overline{\Lambda}(i, A+j) &= \overline{p}_i \overline{p}_j H_{12}(\xi_i, \zeta_j)\end{aligned} \tag{12}$$

Let Θ_t denote the stacked vector $[\Phi_t^T \Psi_t^T]^T$ of all author and word coordinates at time t. The resultant approximate log-likelihood has the following form:

$$\log p(C_t|\Phi_t, \Psi_t) \propto -C + (-\lambda \overline{\eta}^T \Theta) - \frac{1}{2}(\Theta^T(2\tilde{\Lambda} + \lambda\overline{\Lambda})\Theta)$$

Note that it corresponds to a Gaussian in canonical form. The final set of observation model parameters thus obtained are:

$$\begin{aligned}\eta_{approx} &= -\lambda\overline{\eta} \\ \Lambda_{approx} &= (2\tilde{\Lambda} + \lambda\overline{\Lambda})\end{aligned} \tag{13}$$

Thus the Gaussian approximation to our observation model has canonical parameters $(\eta_{approx}, \Lambda_{approx})$. In the conditioning step we use (4) to get $(\eta_{t|t}, \Lambda_{t|t})$. From these, we compute the moment parameters $(\mu_{t|t}, \Sigma_{t|t})$. Now in the prediction and roll-up step we use these parameters to obtain estimates of $(\mu_{t+1|t}, \Sigma_{t+1|t})$ using (5).

The resulting Λ may have negative eigenvalues. To project to the closest possible symmetric-positive-definite matrix, we set the negative eigenvalues to a small positive number. Together these approximations give us a tractable expression while retaining the highly informative inter-coordinate interactions (e.g. $x - y$ correlation in two dimensions).

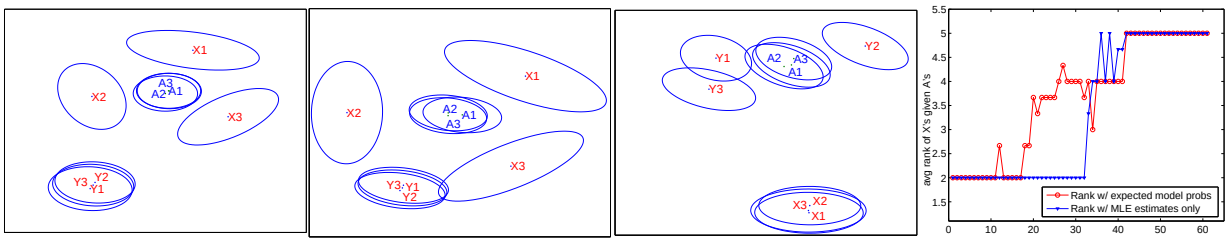
In Figure 2 we compare contour-plots of the true posterior conditional to the one obtained by our method. The true posterior may be multimodal, as in the left panel of Figure 2(A), when it is difficult to approximate with any unimodal distribution. Even then, the corresponding Gaussian is centered reasonably between the two peaks (Figure 2(A) right panel). In most cases we observed, however, the true posterior is unimodal and the approximation is a good fit, though with higher variance (Figure 2(B)).

4 Experiments

We evaluate D-CODE based on the quality of visualizations produced, their temporal consistency and correspondence to the data. We empirically evaluate the usefulness of distributions provided by D-CODE, and see whether useful properties of the distribution are preserved. We also quantitatively test performance on an author-prediction task.

4.1 Algorithms and Tasks

D-CODE The filtering distribution over entity coordinates learned from dynamic co-occurrence data per timestep can be used to calculate expected probabilities for prediction. These can be estimated in closed form using our approximation for $\log Z$, by marginaliz-



(A) (B) (C) (D)

Figure 3: Dynamic embedding of a noisy synthetic data sequence with authors $X_1 \ldots X_3$,$Y_1 \ldots Y_3$ and words $A_1 \ldots A_3$, with 0.95 confidence ellipses. A. Initially the $X - A$ pairs have high co-occurrence counts. B. over time this trend shifts to equal counts, C. and eventually shifts to high co-occurrence counts between $Y - A$ pairs. D. For the same sequence, average predicted rank of authors X_i given words A over time as ranked by Naive Bayes with and without the distributions. Notice the gradual change in D-CODE's prediction compared to the sharp change when not modeling uncertainty.

ing over the latent positions. This is the most powerful model among the alternatives considered, capturing both uncertainty in entity coordinates as well as temporal dynamics. To evaluate the usefulness of dynamic modeling, **Static D-CODE** is a variant that learns an embedding based on C_{T-1} to predict coordinates for year T. The alternative of prediction based on embedding the aggregate data $\sum_{t=1}^{T-1} C_t$ fares worse.

D-CODE MLE To evaluate the usefulness of modeling uncertainty in entity coordinates, we can evaluate model probabilities using (1) at $\mu_{t|t}$, which is the posterior mean of Φ and Ψ. **Static D-CODE MLE** is a variant analogous to Static D-CODE described above.

Dynamic PCA We compare D-CODE against a dynamic embedding algorithm based on PCA (M.W. Berry & Letsche, 1995) of overlapping windows of the data. We create a new data set by averaging our co-occurrence counts over fixed-size windows of consecutive timesteps to maintain temporal consistency. On the NIPS data we use a window of size 4, heuristically chosen for good performance.

For an aggregated counts matrix C over a window, we compute D, which is a diagonal matrix with $D(i) = \sum_j C(i,j)$. Let $M = \epsilon I + (1-\epsilon)D^{-1}C$. This defines the transition probabilities of a random walk on a graph with words and authors as nodes, and links with respective co-occurrence counts as weights. ϵ is the probability with which at any step the random walk stays at the same node. M^p then denotes the transition probabilities of a p-step random walk. Since M is un-symmetric, the standard practice is to work with a symmetric matrix, i.e. $N = D^{1/2}MD^{-1/2} = \epsilon I + (1-\epsilon)D^{-1/2}CD^{-1/2}$. Note that $N^p = D^{1/2}M^pD^{-1/2}$. The top eigenvector of N is a constant vector. The top k (excluding the first) eigenvectors of N, scaled by eigenvalues raised to powers of $k/2$, give us the PCA projection of the counts matrix. We manually picked the ϵ and p for which the algorithm seems to perform best. The projection in the current timestep is transformed via the Procrustes transform (Sibson, 1979) to best align with the previous timestep's configuration.

LLE We embed co-occurrence data using Locally Linear Embedding (Roweis & Saul, 2000). Like the static D-CODE variants above, we embed data for year $T-1$ and predict for year T. Since LLE cannot meaningfully embed heterogenous sets of entities based on pairwise counts alone, we define author-author distances based on the words they use, as in Mei and Shelton (2006). This allows us to compare with LLE, though the resulting algorithm sometimes returns degenerate embeddings. We report results from cases with non-degenerate embeddings.

Any of the above embedding techniques can then be used to get point estimates of the coordinates, or distribution over the coordinates, using which we perform the following prediction task:

Naive Bayes Author Prediction We use the distributions over entity locations at each timestep to perform Naive Bayes ranking of authors given a subset of words from a paper in the next timestep.

4.2 Data sets

NIPS We looked at word-author co-occurrence data over 13 years from the NIPS proceedings of 1986-1999[1]. We implemented D-CODE on a subset of the data with the 40 most prolific authors and 428 most common words appearing in their papers.

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

Synthetic We generate a synthetic data set to closely examine D-CODE's ability to model temporal patterns and represent correlations in its posterior distributions. The data consists of a sequence of co-occurrence counts matrices involving two groups of authors $X_1 \dots X_3$ and $Y_1 \dots Y_3$, and a single group of words $A_1 \dots A_3$. The data exhibits three distinct epochs. $X - A$ co-occurrences are high and $Y - A$ are low in the first few timesteps. Afterwards, these co-occurrences start changing slowly until $X - A$ counts are low and $Y - A$ counts are high.

4.3 Visualizing trends and uncertainty in synthetic data

To investigate whether distributions over entity coordinates give us any advantage, we ran D-CODE on the synthetic data set described earlier. Figure 3 illustrates the D-CODE embedding of this data in timesteps from these three distinct periods, along with 95% confidence ellipses of the conditional posterior for each entity, fixing every one else's locations fixed at their posterior means. The orientation of ellipses around entities is informative. For example, figure 3(A) indicates that uncertainty in X_i locations is most acceptable in directions orthogonal to the $A - X$ axis. This indicates that our variational approximation of the observation model manages to represent uncertainty consistently with the data.

We calculated the *average rank* of authors X_i given the word list A_1, A_2, A_3, A_4 over all timesteps using the Naive Bayes prediction. We expect this rank to be close to 2 in the beginning (mean of $1, 2, 3$), and drop gradually to 5 to reflect the dynamic trend in the data. The change happens very smoothly over time-steps $20 - 60$. Figure 3(D) shows that the ranks induced by D-CODE fulfill this expectation, since ranks change smoothly from low to high over this period. This is because of the increase in uncertainly in author positions, which is also reflected in the enlarged confidence intervals of the X's in figure 3(B). The MLE estimate, on the other hand is overconfident and switches too abruptly.

4.4 Visualizing the NIPS data

We embedded the NIPS data using D-CODE. The words in 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,acquisition` authors such as `Singh, Dayan` and `Barto` which clearly are words from the field of reinforcement learning. In the top-left region are words like `kernel,regularization,error,bound`. The other two regions also have noticeable patterns.

4.5 Predicting authors of NIPS papers

We define a prediction task by attempting to rank authors given a set of words, say from a paper taken from a subsequent timestep. D-CODE and D-CODE MLE can calculate the required marginal probabilities $p(w \mid a)$ for Naive Bayes as described earlier. We can also compute $p(w \mid a)$ from Dynamic PCA or LLE embeddings by using entity locations as Φ and Ψ in the model probability equation (1). We first describe a few specific author-keyword pairs and their conditional probability-based rank predictions over time.

In Figure 5 we plot the rank of particular authors given particular keywords over time according to Naive Bayes prediction with uniform priors, in comparison to the empirical conditional probabilities. In the bottom panels of Figure 5, `(Jordan,variational)` and `(Smola,kernel)` have high empirical probabilities in the later timesteps, corresponding to ranks closer to 1 in the top panel according to D-CODE. The prediction according to Dynamic PCA is less consistent and does not correspond to the data nearly as well.

In table 2, we show *median predicted rank* of *true authors* of papers using embeddings of different sized sets of authors and words, according to Naive Bayes prediction. Note that this is a harsh metric since a paper may have multiple authors and the metric expects each of them to be ranked as first-author, which is impossible. Here our aim is just to compare with alternative models, not to compare with the state of the art.

For each size of data set (a, w), random subsets are obtained from the 100 most prolific authors and their 500 most common words. We perform filtering up to $t = 12$ on the NIPS data, then predict author ranks for all papers in $t = 13$ with an author included in the embedding. Average predicted rank is calculated for each true-author by ranking all possible authors given words in the paper, noting the rank of the true author, then averaging this measure over all (true-author,paper) pairs. This process is repeated for several embeddings, to counter randomness. We see in the table that D-CODE-based predicted ranks are better in most cases. This can be attributed to D-CODE's usage of distributions. LLE-based embeddings, as well as the static counterparts of D-CODE and D-CODE MLE, perform poorly in most cases. These algorithms, Static D-CODE and Static D-CODE MLE, embed the counts matrix for $t = 12$ and use it to predict authors for $t = 13$ with and without using distributions, re-

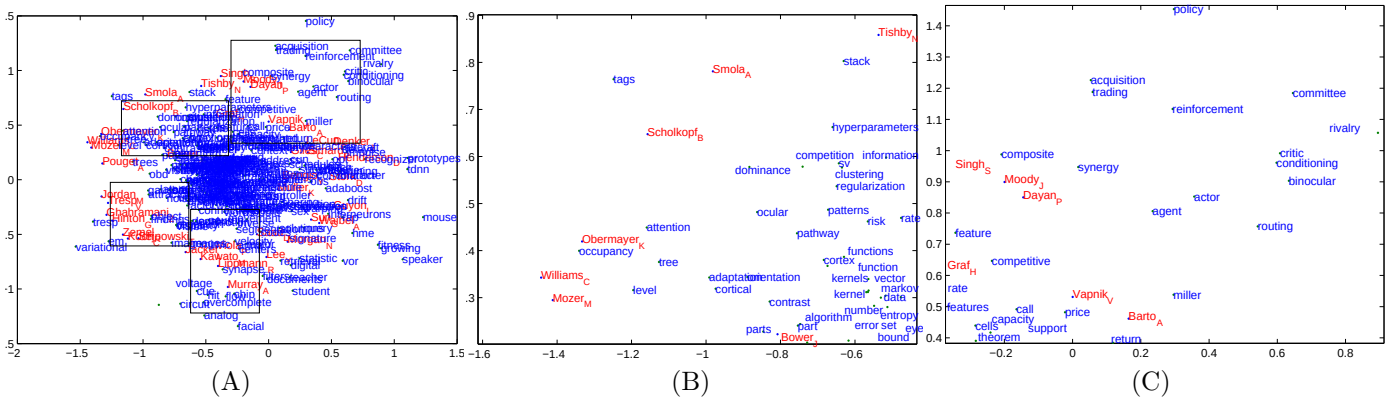


Figure 4: (A). $t = 13$ Dynamic embedding of NIPS data (1999). (B),(C). Close-ups of roughly the top two boxes in (A), showing regions dominated by distinct sub-fields.

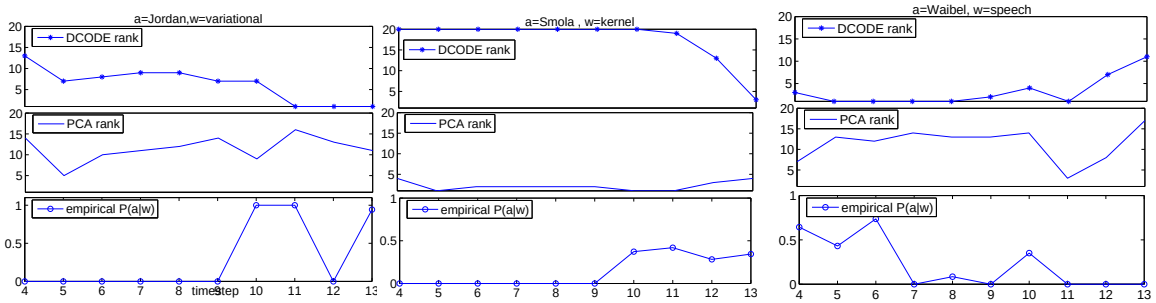


Figure 5: Average author rank given a word, predicted using D-CODE (above) and Dynamic PCA (middle), and the empirical probabilities $\overline{p}(a \mid w)$ on NIPS data (below). $t = 13$ corresponds to 1999. A. Jordan and variational. B. Smola and kernel. C. Waibel and speech. Note that D-CODE's predicted rank is close to 1 when $\overline{p}(a \mid w)$ is high, and larger otherwise. In contrast, Dynamic PCA's predicted rank shows no noticeable correlation.

Table 2: Median predicted rank of true authors of papers in $t = 13$ based on embeddings until $t = 12$. Values statistically indistinguishable from the best in each row are in bold. D-CODE is the best model in most cases, showing the usefulness of having distributions rather than just point estimates. D-CODE and D-CODE MLE also beat their static counterparts, showing the advantage of dynamic modeling.

Data size (authors,words)	D-CODE	D-CODE MLE	Static D-CODE	Static D-CODE MLE	Dynamic PCA	LLE
20a, 188w	**4.1**	7.4	14.4	9.5	7	11.8
30a, 289w	**8**	10.5	**9**	12	**9**	13
40a, 348w	14.2	**9.5**	12.8	19	16.8	21

spectively. This shows the usefulness of modeling dynamics of the data, since information from prior years accumulates in the filtering distribution and aids in making better predictions.

5 Discussion

We have proposed and demonstrated D-CODE, a model for Euclidean embedding of co-occurrence data over time by formulating the problem as a factored state space model. Aside from this novel formulation of dynamic embedding, the resulting model is unique in its probabilistic treatment of the coordinates, modeled as latent variables with posterior distributions rather than the point estimates of previous models.

While the approximation applied to yield a tractable observation model is uncontrolled, the visualizations in Sections 4.3 suggest that the model obtained still preserves important correlations in the posterior, and the NIPS author prediction results(Section 4.5) con-

firm that these correlations in the posterior translate into superior performance in realistic scenarios. These results also indicate benefits of modeling the dynamics in the data for prediction purposes, and not just for obtaining smooth temporal visualizations.

There are several possibilities for future work. There may be a choice of different approximations for the observation model that lead to dynamic models of different kinds, such as particle filters. A Markov Chain Monte Carlo simulation of the exact observation model would allow us to compare the exact shape of the posterior with our Gaussian approximation. There is also room for improvements in the computational aspects. Increasing the numbers of authors and words, or increasing the number of embedding dimensions, both linearly affect the size of the precision matrix that is inverted in the Kalman filter update steps. However, sparseness properties of this matrix could be explored and utilized for faster filtering.

Acknowledgements

We are grateful to Carlos Guestrin for valuable discussions. We also thank the reviewers for their insightful comments.

References

Borg, I., & Groenen, P. (1997). *Modern multidimensional scaling*. Springer-Verlag.

Globerson, A., Chechik, G., Pereira, F., & Tishby, N. (2004). Euclidean embedding of co-occurrence data. *Proc. NIPS*.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*.

Mei, G., & Shelton, C. R. (2006). Visualization of collaborative data. *Proc. UAI*.

M.W. Berry, S. D., & Letsche, T. (1995). Computational methods for intelligent information access. *Proceedings of Supercomputing*.

Ng, A., Jordan, M., & Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm.

Raftery, A. E., Handcock, M. S., & Hoff, P. D. (2002). Latent space approaches to social network analysis. *J. Amer. Stat. Assoc.*, *15*, 460.

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*.

Sarkar, P., & Moore, A. (2005). Dynamic social network analysis using latent space models. *Proc. Nineteenth Annual Conf. on Neural Info. Proc. Systems (NIPS)*.

Sibson, R. (1979). Studies in the robustness of multidimensional scaling : Perturbational analysis of classical scaling. *J. Royal Stat. Soc. B*.

Tenenbaum, J., de Silva, V., & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*.

Appendix

In this section we give some details of the derivations. We approximate the log-normalizer of the distribution in order to get a Gaussian observation model. We will now look at the log-likelihood in (2). Let us examine the first part. We ignore t here for simplicity.

$$\begin{aligned} & -\textstyle\sum_{a_i}\sum_{w_j}\bar{p}(a_i,w_j)||\phi_i-\psi_j||^2 \\ & = -\textstyle\sum_{a_i}\sum_{w_j}\bar{p}(a_i,w_j)(\phi_i-\psi_j)^T(\phi_i-\psi_j) \\ & = -\Theta^T\tilde{\Lambda}\Theta \end{aligned} \tag{14}$$

By moment matching, we get $\tilde{\Lambda}$ as in (10). We now linearize the second part of (2), i.e. $\log Z$ using a Taylor approximation around Φ and Ψ. We do a first-order Taylor expansion around Z leading to Equation (6). This is followed by a second-order Taylor expansion of the exponentiated distance term $e^{-(\phi_i-\psi_j)^T(\phi_i-\psi_j)}$ around ξ_i, ζ_j, resulting in (7).

The second-order approximation $g([\phi_i^T \;\; \psi_j^T])$, equation (7), becomes

$$\begin{aligned} & = 1 + \phi_i^T\nabla_1 + \psi_j^T\nabla_2 + \tfrac{1}{2}[\phi_i^T \;\; \psi_j^T]H(\xi_i,\zeta_j)[\phi_i^T \;\; \psi_j^T]^T \\ & = 1 + \phi_i^T\nabla_1 + \psi_j^T\nabla_2 \\ & \quad + \tfrac{1}{2}[\phi_i^T H_{11}\phi_i + \psi_j^T H_{21}\phi_i + \phi_i^T H_{12}\psi_j + \psi_j^T H_{22}\psi_j] \end{aligned} \tag{15}$$

where $H(\xi_i,\zeta_j)$ is the Hessian evaluated at ξ_i,ζ_j, (9). Also $\nabla_1(\xi_i,\zeta_j)$ and $\nabla_2(\xi_i,\zeta_j)$ are the gradients w.r.t. ϕ_i and ψ_j respectively, also evaluated at ξ_i,ζ_j, (8). For convenience, define $\overline{\eta}$ to be a vector of length $2(A+W)$ and $\overline{\Lambda}$ to be a symmetric matrix of size $2(A+W)\times 2(A+W)$, as in (11) and (12). By i we denote author i and by j we index word j.

Now using (11), (12) and (15), $\log Z$ becomes:

$$\begin{aligned} & \log Z = \log\textstyle\sum_{ij}\overline{p}_i\overline{p}_j e^{-(\phi_i-\psi_j)^T(\phi_i-\psi_j)} \\ & \approx C + \lambda[\textstyle\sum_i \phi_i^T\overline{\eta}_i + \sum_j \psi_j^T\overline{\eta}_j + \frac{1}{2}(\sum_i \phi_i^T\overline{\Lambda}_{ii}\phi_i + \\ & 2\textstyle\sum_{ij}\phi_i^T\overline{\Lambda}_{ij}\psi_j + \sum_j \psi_j^T\overline{\Lambda}_{jj}\psi_j)] \\ & = C + \lambda(\overline{\eta}^T\Theta) + \tfrac{1}{2}\lambda(\Theta^T\overline{\Lambda}\Theta) \end{aligned} \tag{16}$$

All terms independent of μ, Σ were combined in the constant term C. Using (14) and (16) we obtain the approximate log-likelihood

$$\begin{aligned} \log p(C_t|\Phi_t,\Psi_t) &= -\Theta^T\tilde{\Lambda}\Theta - C - \lambda\overline{\eta}^T\Theta - \frac{1}{2}\lambda(\Theta^T\overline{\Lambda}\Theta) \\ &= -C + (-\lambda\overline{\eta}^T\Theta) - \frac{1}{2}(\Theta^T(2\tilde{\Lambda}+\lambda\overline{\Lambda})\Theta) \end{aligned}$$

which gives us a Gaussian distribution with canonical parameters as in (13).