CRIME EVENT EMBEDDING WITH UNSUPERVISED FEATURE SELECTION

Shixiang Zhu, Yao Xie

Georgia Institute of Technology

ABSTRACT

We present a novel event embedding algorithm for crime data that can jointly capture time, location, and the complex free-text component of each event. The embedding is achieved by regularized Restricted Boltzmann Machines (RBMs), and we introduce a new way to regularize by imposing a ℓ_1 penalty on the conditional distributions of the observed variables of RBMs. This choice of regularization performs feature selection and it also leads to efficient computation since the gradient can be computed in a closed form. The feature selection forces embedding to be based on the most important keywords, which captures the common modus operandi (M. O.) in crime series. Using numerical experiments on a large-scale crime dataset, we show that our regularized RBMs can achieve better event embedding and the selected features are highly interpretable from human understanding.

***Index Terms*—** RBM, ℓ_1 regularization, feature selection

1. INTRODUCTION

Event data such as crime incidents, medical reports, and social media data [1, 2] is increasing rapidly in the era of big data. Each event data sample usually consists of time, location, and other description of the event such as free text. Learning correlation between events has become a pressing need for event data. Embedding is an efficient solution to represent unstructured events data by mapping the discrete events into a continuous Euclidean space, and the distance between two data points is proportional to their similarity or correlation [3]. Recently, research on embedding has received a lot of attention. The central problem is to learn the mapping, which has been achieved via various approaches including neural networks such as Restricted Boltzmann Machine.

However, a major challenge remains in how to handle free text of the event. For crime data, the free-text part may actually contain the most important information. Current embedding approaches based on tokenizing text part as simple "marks" of events [4] may not be sufficient to capture these useful information since they are buried in the complex and unstructured free-text narratives of the police event report. To achieve the goal, we need to develop models and tools that combines embedding and natural language processing.

In this paper, we present a novel event embedding approach that jointly captures the free-text part of the event data together with other structured data (such as time and location). We focus on crime event data, where each event is a police report that contains time, location, and narrative of the crime incident entered by the investigator. We leverage the Restricted Boltzmann Machines' (RBMs) structure to perform embedding and introduces a novel way to regularize RBM to automatically select the most important observed (visible) variables in order to improve the quality of the embeddings.

The motivation for the new regularization for variable selection in our work is two-fold. First, as shown in Appendix A[1], it is motivated because we observe in crime data, very few of certain keywords in the free-text (which corresponds to the observed variables in RBMs) are the most important in defining the correlation between events. Thus, it is crucial to identify these keywords while eliminating the irrelevant ones. This can be viewed as performing "variable selection" for RBM (as analogous to variable selection using lasso for linear regression model). We found that by focusing on the most important observed variables (implicitly through regularization), the performance of embedding will be drastically improved. Second, this regularizes the over-parameterization of RBM (as typically there is a large number of observed and hidden variables) in the presence of limited data. Since we do not know which keywords are the most important *a priori* in determining the correlation of events, we extracted as much as possible from the free-text and this may lead to over-representation. To a certain extent, the regularization automatically get rid of redundant information and only select the most crucial variables (keywords).

Our method is *unsupervised*: training RBM does not require any "label" of events since RBM essentially captures the co-occurrence pattern of the observed variables. Thus, our work can be viewed as an unsupervised variable selection for embedding purposes. This can be quite important because, in crime data analysis, it is hard for police officers to provide a large number of crime incidences that are known to be related to each other. In our later example, we are only able to identify 6 groups of labeled data for verification purposes only but

[1] See https://arxiv.org/abs/1806.06095 for the appendix.

not for training purposes. To the best of our knowledge, our work is the first to demonstrate the value of embedding based on RBM for finding correlated cases in crime data. Although in the paper, we focus on crime event analysis, this framework can be generalized to other event embeddings as well.

2. RELATED WORK

Most existing events embedding methods focus on modeling spatial-temporal dependencies between events [2] for the location and time information of the events, by building (hierarchical) probabilistic models. Some recent work also takes into account the actual content of the events, for example, the type (or category) of events [4]. This is done by using Recurrent Neural Networks (RNN) treating category as marks of events, and thus far usually only a small number of possible (discrete) marks are considered in the model. However, this can be restrictive for some real data, as it cannot handle the case when there is a much richer textual description for each event, and it is not easy to extend existing work based on "marks" to handle rich textual event data.

To perform event embedding in the presence of textual data, a mainstream approach is the so-called Skip-gram method [5, 6, 7, 1]. Skip-gram was originally introduced to model the semantic structure of the language [3]. Generally, it can capture the co-occurrence patterns of words in each document. However there is no co-occurrence concept in crime analysis since the manpower of police is too limited and expensive to define enough context in order to calculate the co-occurrence of the crime events.

By and large, this is an extensive work based on our paper [8]. In addition to previous work, this paper extensively considers the regularization for RBM in order to select key features from the observed variables. There is also some other good work take into account regularization for RBM for different purposes. The close work [9, 10] propose various of regularization on the hidden variables of RBM for yielding sparsity in hidden variables. Another work [11, 12, 13] came up with the similar idea of sparse feature learning by imposing regularization on RBM structure. However, the proposed deep architecture is different from our method and it mainly focuses on producing good representations with deep feature hierarchies, which is completely different from the motivation of feature selection.

3. PROBLEM SETUP

A single event data point consists of a set of observed variables $\mathcal{X} = \{x_s, x_t, \} \cup \{x_i\}_{i\in\mathbb{Z}^+, i\leq V}$, where $x_s \in \mathbb{R}^D$ (D is the dimension of the space), $x_t \in \mathbb{R}$ are temporal and spatial variables respectively, which are explicitly retained in the model (meaning that they will not be eliminated by the variable selection). And $\{x_i\}_{i\in\mathbb{Z}^+, i\leq V}$ is a set of observed variables represent the tf-idf value [14] of the keywords in the vocabulary that appeared at least once in the corpus where $x_i \in \mathbb{R}$, i indicate the index of the keyword and V is the total number of the keywords. In our model, $\mathbf{x} = \{x_i\}_{i\in\mathbb{Z}^+, i\leq V}$ will be regularized in order to select key variables. Each event in the dataset is denoted as a triplet $(x_s, x_t, \mathbf{x})^{(k)}$ where k is the index of the event.

Given an event point $(x_s, x_t, \mathbf{x})$, we define its embedding as $\mathbf{h} \in \{0, 1\}^H$ with the dimension of H (in our later examples we set $H = 1,000$). The similarities between two embedding vector can be evaluated by their cosine distance $\mathbf{h}\cdot\mathbf{h}'/\|\mathbf{h}\|\cdot\|\mathbf{h}'\|$, where $\|\mathbf{h}\|$ denotes the ℓ_2 norm of vector $\mathbf{h}$. The goal is to take the event dataset $\{(x_s, x_t, \mathbf{x})^{(k)}\}$ as input and produce their embeddings $\{\mathbf{h}^{(k)}\}$ accordingly.

4. REGULARIZED RBM

In this section, we present our new regularized RBM with feature selection. A basic introduction to the vanilla RBM and its related notations are delegated to the Appendix B[2].

4.1. Observed variables selection in RBM

We introduce a ℓ_1-regularizer to the log-likelihood of RBM (2) to mitigate the impact of those noisy variables specifically. As discussed in Section 1, directly learning the statistical dependencies between all observed variables (the Bag-of-Words in the corpus) will bring noisy information from irrelevant variables into the model. To achieve the selection, we impose an ℓ_1 penalty on the probability $1-P(x_i < t|\mathbf{x})$[3] weighted by λ, which penalizes the reconstructed observed variables that are sensitive to large values. This penalty introduces a natural way to select the most important features (correspond to observed variables in RBM). A nice feature of this penalty is that the corresponding gradient can be computed easily.

Thus, given one training data $\mathbf{x}$, we need to solve the following optimization problem. *This leads to our new formulation which performs the selection of observed variables for RBM:*

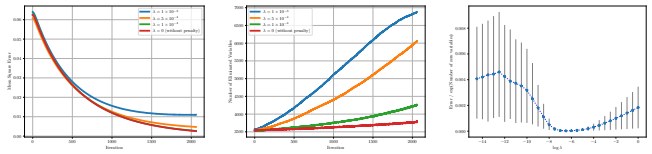
$$\max_{\mathbf{w},\mathbf{b},\mathbf{c}} \Big\{ \log\ \mathcal{L}(\theta|(x_s, x_t, \mathbf{x})) - \lambda \sum_{i\leq V} |1 - P(x_i < t|\mathbf{x})| \Big\} \tag{1}$$

By tuning the weight λ of the penalty, we can achieve various levels of sparsity in terms of a subset of observed variables $\{x_i\}, i \leq V$.

We solve this optimization problem by gradient descent (note that this is a non-convex problem and gradient descent is a default approach to solve it). Because, in crime analysis, the observed variables are real values (tf-idf), we take the Gaussian-Bernoulli RBM (GBRBM) as an example. $1 - P(x_i < t|\mathbf{x})$ can be rewritten into the following expression

[2]See https://arxiv.org/abs/1806.06095 for the appendix.

[3]Here t is a very small constant, we preset $t = 10^{-2}$ here since the impact of a tf-idf value lower than 10^{-2} can usually be ignored.



(a) (b) (c)

Fig. 1: Fitting RBM with and without designed penalty term over 2,056 crime events (including 7,038 observed variables). Under same experiment settings (learning rate $\alpha = 10^{-3}$, threshold value $t = 10^{-2}$): (1a): training errors over iterations; (1b): numbers of eliminated (disabled) variables over iterations, and (1c): result of cross-validation over λ.

by substituting Eq.(7) (other types of RBM can be processed in a similar fashion):

$$|1 - P(x_i < t|\mathbf{x})| = 1 - \int_{-\infty}^{t} \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2\sigma^2}(x_i - b_i - \sigma\sum_{j=1}^{H} h_j w_{ij})^2} dx_i.$$

We can directly compute its gradients with respect to w_{ij} and b_i in a closed-form due to the exponential structure in the second term of it log-likelihood function. By introducing this penalty term, the gradients ∇w_{ij} and ∇b_i in Eq.(3) and Eq.(4) can be rewritten as follows:

$$\begin{aligned}\nabla w_{ij} &= \langle x_i h_j \rangle_{p(\mathbf{h}|\mathbf{x})} - \langle x_i h_j \rangle_{p(\mathbf{x},\mathbf{h})} \\ &\quad - \lambda \frac{h_j}{\sqrt{2\pi}} \sum_{i \le V} \exp(-\frac{1}{2\sigma^2}(t - b_i - \sigma \sum_{j=1}^{H} h_j w_{ij})), \\ \nabla b_i &= x_i - \langle x_i \rangle_{p(\mathbf{x})} \\ &\quad - \lambda \frac{1}{\sigma\sqrt{2\pi}} \sum_{i \le V} \exp(-\frac{1}{2\sigma^2}(t - b_i - \sigma \sum_{j=1}^{H} h_j w_{ij})).\end{aligned}$$

4.2. Training and interpretation

The standard training approach for RBM is still applied in the regularized RBM thanks to the explicit closed-form of w_{ij} and b_i. The training objective of the regularized RBM is to maximize a likelihood function defined via the energy function defined in (1). We adopt the k-step contrastive divergence (CD-k) approach for the training, which is an approach to approximate the gradient in training RBM via gradient descent [15], where the gradients have been defined above. The Gibbs chain is initialized with a training example of the training set and yields the sample after k steps. The iterations are repeated until certain empirical convergence has achieved.

Regarding feature selection, the effects of the ℓ_1 regularization can be interpreted intuitively. The ℓ_1 norm yields sparsity within the given subset of observed variables, which means that some variables' norms are set zero, or their activation possibilities tend to produce zero values. During the course of training, only very few of observed variables in the subset will win this competition, and their impact to the embeddings will get enhanced, though meanwhile, the effect of irrelevant variables will vanish eventually. Therefore, for our application, we impose the penalty on the subset of observed variables that associate the input of the Bag-of-Words vectors and successively eliminate keywords due to the regularization. As shown in Fig.1a and Fig.1b, the regularized RBM prominently disable the most of keywords in the vocabulary (only 280 keywords are retained after convergence) when $\lambda = 10^{-3}$ without losing too much accuracy on training errors. And in Fig.1c, we also show a surrogate cross-validation result over λ in which the model achieves the best performance at some point. Differs from feature selection in a linear model (e.g., through lasso), here we have to perform a *reconstruction* (similar to back propagation) step on the observed variables. The benefits of denoising observed variables will be presented in the embeddings due to the symmetrical structure of RBM.

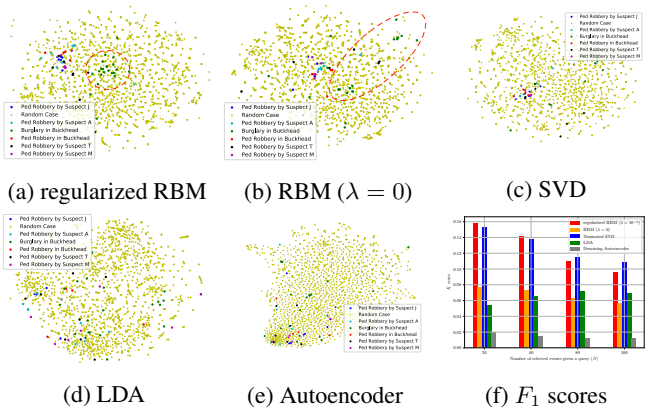
5. EXPERIMENTS ON REAL-DATA

To provide a comprehensive validation, we test our approach on crime events data provided by the Atlanta Police Department over 2016 and 2017 with carefully picked $\lambda = 10^{-3}$ according to the result of cross-validation in Fig.1c. We show the superiority of our method over other competitors under the same parameter settings, including the vanilla **RBM** without regularization, Latent Dirichlet Allocation (**LDA**), Singular Value Decomposition (**SVD**), **denoising Autoencoder**. Specifically, the objective goal of these methods is to generate embeddings for each data point with the length of 1,000. We are ultimately looking for the distributed representations for each of the crime events, and the representations would tend to be closer in Euclidean distance if the crimes follow the same M.O. (committed by the same arrestee, suspect, or criminal gang).

Dataset. The dataset for the experiments contains 2,056 crime events in total happened in Atlanta over past two years, which consists of 56 hand-labeled events that belong to 5 individual crime series committed by different arrestees, and 2,000 randomly selected unknown events. The overall crime events involve 123 crime categories. Each crime event mainly includes time, location (latitude and longitude), crime category and free text part. The most important part is free text, all the text have been preprocessed into a Bag-of-Words, including 7,039 keywords and 2,056 documents.

5.1. Visualization via t-SNE

In order to intuitively inspect how the high-dimensional embeddings distribute in a 2D space, we first use two-dimensional t-distributed stochastic neighbor embeddings (t-SNE) [16] to project the embeddings with 1,000 binary units into a 2D space. t-SNE is capable of capturing local



(a) regularized RBM (b) RBM ($\lambda = 0$) (c) SVD

(d) LDA (e) Autoencoder (f) F_1 scores

Fig. 2: Embeddings of 2,056 crime events projected into a 2D space by t-SNE. (2a), (2b), (2c), (2d), (2e) are generated by regularized RBM ($\lambda = 10^{-3}$), vanilla RBM, truncated SVD, single layer denoising autoencoder and LDA repectively.

structure of the high-dimensional data, while also revealing global structures such as the presence of clusters at several scales [16]. Since the embeddings capture events' similarities by vicinity in the Euclidean space, the performance of events embeddings can be visually evaluated by the aggregation degree of labeled events (colorful dots) in such 2D space. Ideally, events that belong to the same crime series should aggregate highly in a local region of the embedding space. In Fig.2 we compare **LDA**, **truncated SVD**, single layer **denoising Autoencoder**, vanilla **RBM**, and **regularized RBM** ($\lambda = 10^{-3}$) on the crime dataset by observing their distributions visually in t-SNE 2D space. Notice that the each larger and colored dot means individual labeled crime events. **regularized RBM** outperforms **LDA**, **RBM**, **denoising Autoencoder** with **truncated SVD** delivering competitive performance. Compare to the vanilla RBM, the regularization helps improve the quality of the embeddings remarkablely. For instance, the subregion within the red dashes circle over 95% the green dots, the subregion in Fig.2a is obviously tighter than Fig.2b.

5.2. Retrieval performance (F_1 score)

In order to further evaluate the quality of the embeddings accurately, we test them as a retrieval problem, and adopt a basic quality measure, F_1 score, widely used in the text mining literature for the purpose of document clustering [17]. The F_1 score combines the *Precision* and *Recall* ideas from the information retrieval literature. The precision P and recall R given a query (crime event) with respect to its true class (relevant crime events or crime series) are defined as:

$$P = \frac{|\text{relevant crime events} \cap \text{retrieved crime events}|}{|\text{retrieved crime events}|},$$

$$R = \frac{|\text{relevant crime events} \cap \text{retrieved crime events}|}{|\text{relevant crime events}|},$$

And the F_1 score of a query q is defined as $F_1 = 2 \cdot \frac{P \cdot R}{P+R}$. To perform a fair test all the methods, we consider the method with the highest average F_1 score to be the best model that maps crime pattern to the embeddings space.

In the experiment, we first generate embeddings and calculates the pairwise similarities between the crime events according to their cosine distances. And then retrieve the top N keywords given a labeled crime event as a query, as shown in Fig.2f, the RBM with regularization is overwhelmingly better than the others in terms of their F_1 score.

5.3. Keywords selection

As we mentioned in the previous section, we cannot obtain the determinate selected features directly from the model. But we do have some indirect approaches to evaluate the quality of the selected variables intuitively. As shown in Fig.4a, Appendix C[4], we calculate the standard deviations for each of the keywords in the vocabulary over the 2,056 crime events. Most of the keywords in the dataset vary considerably. However, in Fig.4b, after reconstructing the same crime events from the RBM with regularization, a large number of keywords have been disabled and are always zero values without variation.

The remaining activated keywords are very intriguing. We compare top 15 keywords with the highest intensity for both raw dataset and the reconstructed dataset in Fig.4. Obviously, most of the keywords in Fig.4a are irrelevant to the crime behaviors or even meaningless, but the keywords in Fig.4b is the opposite, some of them are even highly consistent with the results we present in Fig.3, Appendix A[5]. The selected words such as *"toyota corolla"*, *"drivers door"*, *"black leather"*, *"silver vehicle"*, *"one ounce"*, *"outside apartment"* or *"hotel"* are the strong indicators to the crime patterns from the perspective of police understanding.

6. CONCLUSION

We have presented a novel approach for learning embeddings for crime events with unsupervised feature selection. By imposing a well-designed ℓ_1 penalty on observed variables' activation probabilities that leads to simple gradient descent based algorithm, our regularized RBMs are able to produce high-quality embeddings as well as eliminate irrelevant and noisy features in observed variables. Additionally, regularized RBMs can select key features without supervision. The selected features are not only highly sparse but also interpretable to human. The techniques introduced in this paper can be also used for learning some other high-dimensional dataset with complex interdependencies between their features. Using real-data, we show promising results on a large-scale real crime dataset comparing to conventional methods for text embedding.

[4]See https://arxiv.org/abs/1806.06095 for the appendix.
[5]See https://arxiv.org/abs/1806.06095 for the appendix.

7. REFERENCES

[1] Chao Zhang, "Triovecevent: Embedding-based online local event detection in geo-tagged tweet streams," *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.

[2] Shenda Hong, "Event2vec: Learning representations of events on temporal sequences," *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data*, 2017.

[3] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, 2013.

[4] Nan Du and Manuel Gomez-rodriguez, "Recurrent Marked Temporal Point Processes : Embedding Event History to Vector," *Kdd*, pp. 1555–1564, 2016.

[5] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," *ICLR Workshop*, 2013.

[6] Oren Barkan and Noam Koenigstein, "Item2vec: Neural item embedding for collaborative filtering.," *IEEE International Workshop on Machine Learning for Signal Processing*, 2016.

[7] Dawen Liang, Jaan Altosaar, Laurent Charlin, and David M. Blei, "Factorization meets the item embedding: Regularizing matrix factorization with item cooccurrence," *Proceedings of the 10th ACM conference on recommender systems*, 2016.

[8] Shixiang Zhu and Yao Xie, "Crime incidents embedding using restricted boltzmann machines," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

[9] Heng Luo, Ruimin Shen, and Cahngyong Niu, "Sparse Group Restricted Boltzmann Machines," pp. 1–9, 2010.

[10] Xanadu Halkias, Sebastien Paris, and Herve Glotin, "Sparse Penalty in Deep Belief Networks: Using the Mixed Norm Constraint," pp. 1–8, 2013.

[11] Marc Aurelio Ranzato, Christopher Poultney, Sumit Chopra, and Yann Lecun, "Efficient Learning of Sparse Representations with an Energy-Based Model," *Advances In Neural Information Processing Systems*, vol. 19, pp. 1137–1134, 2007.

[12] Marc Aurelio Ranzato, Y-Lan Boureau, and Yann Lecun, "Sparse Feature Learning for Deep Belief Networks," *Advances in neural information processing systems (NIPS)*, , no. Mcmc, pp. 1185–1192, 2008.

[13] Honglak Lee, Chaitanya Ekanadham, and Andrew Y. Ng, "Sparse deep belief net model for visual area v2," in *Advances in Neural Information Processing Systems 20*, J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, Eds., pp. 873–880. Curran Associates, Inc., 2008.

[14] Stanford NLP Group, "Tf-idf weighting," https://nlp.stanford.edu/IR-book/html/htmledition/tf-idf-weighting-1.html.

[15] Geoffrey E. Hinton, "Training products of experts by minimizing constractive divergence," *Neural Computation*, 2002.

[16] Laurens Van Der Maaten and Geoffrey Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[17] Steinbach Michael, Karypis George, and Kumar Vipin, "A comparison of document clustering techniques," *KDD-2000 Workshop TextMining*, 2010.

[18] Tong Wang, Cynthia Rudin, Daniel Wagner, and Rich Sevieri, "Finding Patterns with a Rotten Core: Data Mining for Crime Series with Cores," *Big Data*, vol. 3, no. 1, pp. 3–21, 2015.

[19] Asja Fischer and Christian Igel, "An Introduction to Restricted Boltzmann Machines," *Lecture Notes in Computer Science: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, vol. 7441, pp. 14–36, 2012.

[20] Max Welling, Michal Rosen-zvi, and Geoffrey E Hinton, "Exponential Family Harmoniums with an Application to Information Retrieval," *Advances in Neural Information Processing Systems (NIPS)*, vol. 17, pp. 1481–1488, 2005.