SUMMARIZED BEHAVIORAL PREDICTION

Shih-Chieh Su
Information Security Risk and Management
Qualcomm
San Diego, CA 92121, USA
shihchie@qualcomm.com

ABSTRACT

In this work, we study the topical behavior in a large scale. Both the temporal and the spatial relationships of the behavior are explored with the deep learning architectures combing the recurrent neural network (RNN) and the convolutional neural network (CNN). To make the behavioral data appropriate for the spatial learning in the CNN, several reduction steps are taken in forming the topical metrics and placing them homogeneously like pixels in the images. The experimental result shows both temporal and spatial gains when compared against a multilayer perceptron (MLP) network. A new learning framework called the spatially connected convolutional networks (SCCN) is introduced to better predict the behavior.

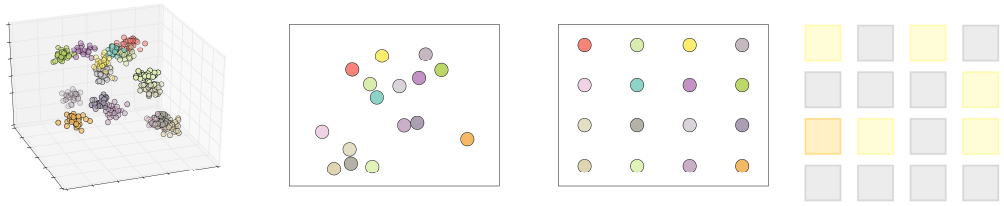
1 INTRODUCTION

Understanding and predicting the behavior of an entity over a large domain of different actions is a challenging problem. The problem is even more difficult when the behavioral data is massively collected with lots of noise. There are various studies in using behavioral data as a global indicator. For instance, large scale user activity data from Google is used to measure and track the user experience such as happiness and engagement (Rodden et al., 2010). The web behavioral data including searches and page views is used by Microsoft to decide the advertisement delivered to the user (Chandramouli et al., 2012). Similarly, Yahoo also conducts study on how education and other factors can affect the web browsing behavior, which can also be applied to improve advertisement targeting (Goel et al., 2012). The predictor to track stock index can be composed from the categorized moods based on the overall Twitter activities (Bollen & Mao, 2011). It is also possible to aim on lots of different business intelligence targets with the behavioral data at hand (Chen et al., 2012).

However, the aforementioned large scale behavioral analytics use cases have one aspect in common: they heavily simplified the response domain to have one or few learnable targets. In this work, we attempt to predict the response whose space is the same as the input space, using the historical behavioral data – not only from the target entity itself, but also from the peer entities. First, we organize the activities into topics. The topical activities on each topic is then quantified and measured for each entity. Over several periods of time, we observe the topical behavior over the same set of topics for all entities in the experiment. Several combinations of deep neural network are explored to predict topical behavior. Specifically, the long short-term memory units (LSTM) (Hochreiter & Schmidhuber, 1997) and other types of RNN (Funahashi & Nakamura, 1993) are employed to learn the temporal variation patterns of the topical behavior. The CNN and the locally-connected network (LCN) (LeCun et al., 1998) are used to learn the spatial composition of the topical behavior. The relationship between topics needs to be abstracted and evenly distributed like pixels for the CNN and the LCN to learn (Su, 2016). The experiment result is compared to the benchmark MLP result.

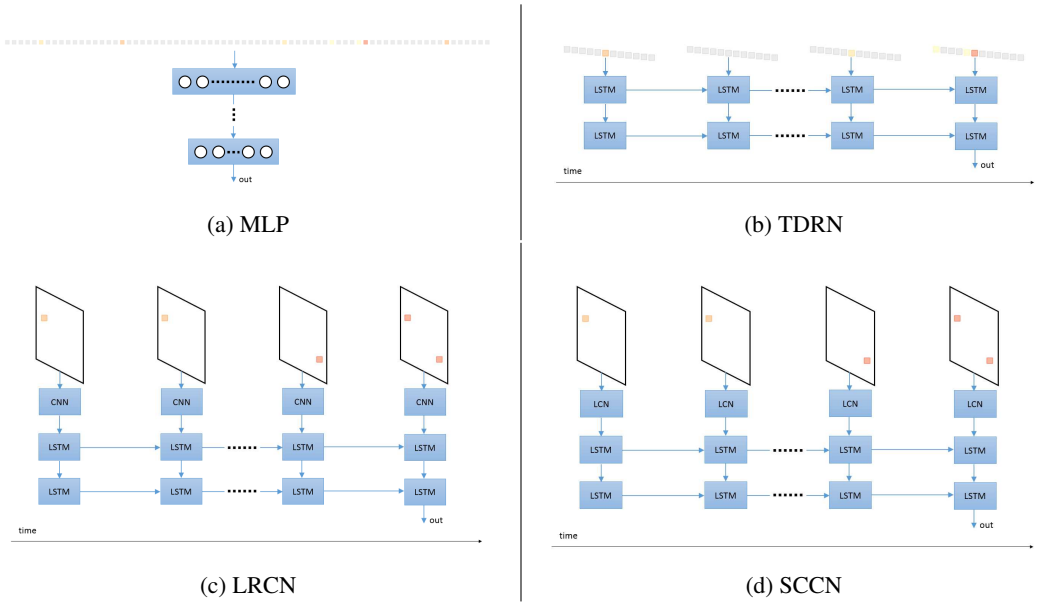
2 METHOD

To keep the behavior prediction within a trackable scope, we summarize the input behavioral data into topics. Starting from the activity log of all entities in the system, the clustering algorithm such as the latent latent Dirichlet allocation (Blei et al., 2003) is applied to find the topics in a high dimension word vector space. For each entity, the vectorized log entries are summarized on these



(a) (b) (c) (d)

Figure 1: Topical behavior. (a) data points in high dimensional space; (b) cluster centers (topics) after dimension reduction; (c) topics after homogeneous mapping; (d) topical metrics for an entity



(a) MLP (b) TDRN

(c) LRCN (d) SCCN

Figure 2: Different learning architecture for topical behavior prediction.

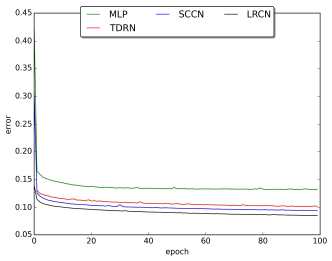
topics to form quantitative metrics. For example, the topical volume over topic t of entity e can be measured as

$$V_t^{(B_{e,T})} = \log(\sum_{a \in B_{e,T}} r_a + 1)), \tag{1}$$

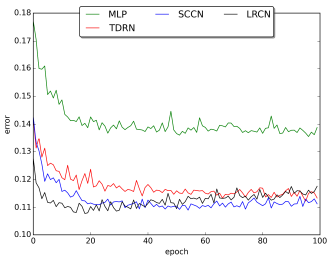
where r_a is the relevancy for activity a to topic t, and B is the set of activities defined by the unique content documents of all activities logged within the time period T.

To better explored the intra-topic relationship in the behavioral data, we want to capture the co-occurrence detail between any pair of topics. Furthermore, we want to learn the detail in the order of the distance between the topic pair - the co-occurrence means more when the two topics are closer to each other. CNN has shown great success in classifying images (Krizhevsky et al., 2012), text (Kalchbrenner et al., 2014), videos (Karpathy et al., 2014). In order to arrange the topical metrics to be similar to the pixels in an image into the CNN, the topical metrics need to go through the following two steps, as illustrated in Figure 1.

1. Dimension reduction: this step maps the topical metrics into a 2D or 3D space, while maintaining the spatial relationship on topics before the mapping. Some popular methods include principal component analysis and t-SNE (Van der Maaten & Hinton, 2008).
2. Homogeneous mapping: on the visualization space that the CNN can digest, the topical metrics also need to be placed evenly. The spatial relationship among the topics also needs to be maintained with best efforts. One way to achieve this goal is the split-diffuse (SD) algorithm (Su, 2016).



(a) RLE on training



(b) RLE on testing

	ΔRLE	ΔRLE_5
MLP	–	–
TDRN	16.51%	16.73%
LRCN	19.92%	20.53%
SCCN	19.85%	19.63%

Figure 3: Performance on various learning architectures

3 Temporal and Spatial Learning

We adopt various architectures to study how the temporal and the spatial information help the topical learning, as in Figure 2. In the MLP, the topical metrics over different time periods are cascaded into one single 1D vector for each sample. The number of neurons reduces over layers, with the output layer being the number of topics. Separately, we use one layer of LSTMs to track the topical metrics for each time period, and then another layer of LSTMs to track the output states of the LSTMs from the previous layer, forming the time distributed recurrent network (TDRN) in Figure 2(b).

In Figure 2(c), the long-term recurrent convolutional networks (LRCN) (Donahue et al., 2015) combines the convolutional layers with the temporal recursion, to exploit both the temporal and the spatial relationship among the topics. In the proposed spatially connected convolutional networks (SCCN) in Figure 2(d), the convolutional units in LRCN are replaced by the LCNs. The LCNs do not share the trained weights between different position. Instead, the same set of weights is applied to the same position of different samples. The regulation is more effective on the locally customized patch dictionaries in the LCN, compared to that on a global dictionary in the CNN.

In predicting the trending or risky topic, the cost of missed future trend is higher than the cost of false positives. One of the possible loss metrics, the risk loss error (RLE), is defined as

$$RLE = \frac{1}{|\mathcal{V}|} \sum_{\forall v \in \mathcal{V}} v(\hat{v} - v)^2, \tag{2}$$

The data set comes from more than 150 million activity entries of $98,881$ network entities, which generate around 9.5 million topical trails. These entities are split into $69,407$ training samples, $7,712$ validation samples, and $21,762$ testing samples. The predicted target values in the testing samples are from the time periods that are later then all the time periods in the training samples. The learning architectures in our implementation are built with Keras (Chollet, 2015) with Theano (Theano Development Team, 2016) backend. Both dropout (Hinton et al., 2012) and L_2 regulation (Ng, 2004) are applied to all the architectures to keep the learned models generalized.

4 Result

Figure 3 shows the experimental result on loss metrics RLE. Experiments were also conducted on other metrics (the results being omitted). The epochs when the validation data has the best five RLEs are chosen to form the RLE_5 by averaging the corresponding RLEs from the testing data. With only the temporal relationship explored by the TDRN, the prediction gain against the MLP ranging from 11.32% to 16.73% depending on the loss metric and the evaluation scenario. The LRCN explores both temporal relationship and spatial relationship over topics. The additional spatial information among topics tracked by CNN further improve the prediction gain from 13.73% to 20.53%. Replacing the CNN spatial tracking with the LCNs, the SCCN provides a comparable 14.20% to 19.85% prediction gain to the LRCN. It is faster to train and to make prediction. Meanwhile, it is better regulated, making it more suitable for larger scale behavioral learning.

REFERENCES

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

Johan Bollen and Huina Mao. Twitter mood as a stock market predictor. *Computer*, 44(10):0091–94, 2011.

Badrish Chandramouli, Jonathan Goldstein, and Songyun Duan. Temporal analytics on big data for web advertising. In *2012 IEEE 28th international conference on data engineering*, pp. 90–101. IEEE, 2012.

Hsinchun Chen, Roger HL Chiang, and Veda C Storey. Business intelligence and analytics: From big data to big impact. *MIS quarterly*, 36(4):1165–1188, 2012.

François Chollet. Keras. `https://github.com/fchollet/keras`, 2015.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2625–2634, 2015.

Ken-ichi Funahashi and Yuichi Nakamura. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural networks*, 6(6):801–806, 1993.

Sharad Goel, Jake M Hofman, and M Irmak Sirer. Who does what on the web: A large-scale study of browsing behavior. In *ICWSM*, 2012.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732, 2014.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Andrew Y Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 78. ACM, 2004.

Kerry Rodden, Hilary Hutchinson, and Xin Fu. Measuring the user experience on a large scale: user-centered metrics for web applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2395–2398. ACM, 2010.

S. Su. Interacting with massive behavioral data. In *KDD 2016 Workshop on Interactive Data Exploration and Analytics*, pp. 127–129, 2016.

Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016. URL `http://arxiv.org/abs/1605.02688`.

L Van der Maaten and G. E. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.