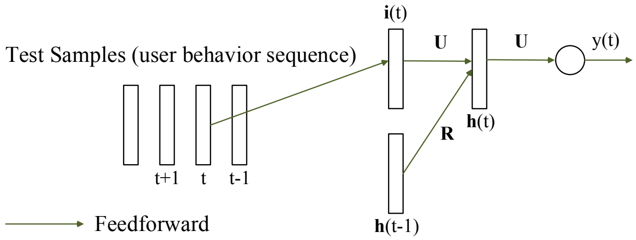


Figure 5: RNN testing process with sequential input samples. The hidden state of previous test sample will be used as input, together with the current sample features.

Table 1: Statistics of the dataset for training and testing click prediction models.

	Ad Impressions	Ads	Users
Training	3,740,980	1,363,687	235,215
Testing	3,741,500	1,379,581	235,215

Experiments

This section first describes the settings of our experiments, and then reports the experimental results.

Data Setting

To validate whether the RNN model we proposed can really help enhance the click prediction accuracy, we conduct a series of experiments based on the click-through logs of a commercial search engine. In particular, we collect half a month logs from November 9th to November 22nd in 2013 as our experimental dataset. And, we randomly sample a set of search engine users (fully anonymized) and collect their corresponding events from the original whole traffic. Finally, we collect over 7 million ad impressions in this period of time. After that, we use the first week's data to train click prediction models, and apply those models to the second week's data for testing. Detailed statistics of the dataset can be found in Table 1.

Evaluation Metrics

In our work, there are multiple models to be applied to predict the click probability for ad impressions in the testing dataset. We use recorded user actions, i.e., click or non-click, in logs as the true labels. To measure the overall performance of each model, we follow the common practice in previous click prediction research in sponsored search and employ Area Under ROC Curve (**AUC**) and Relative Information Gain (**RIG**) as the evaluation metrics (Graepel et al. 2010; Xiong et al. 2012; Wang et al. 2013).

Compared Methods

In order to investigate the model effectiveness, we compare the performance of our RNN model with other classical click prediction models, including Logistic Regression (LR) and Neural Networks (NN), with identical feature set as described aforementioned. We set LR and NN as baseline models due to the following reasons: 1) Quite a few previous studies (Richardson, Dominowska, and Ragno 2007; McMahan et al. 2013; Wang et al. 2013) have demonstrated that they are state-of-the-art models for click prediction in sponsored search. 2) LR and NN models ignore the sequential dependency among the data, while our RNN based framework is able to model such information. Through the comparison with them, we will see whether RNN can successfully leverage dependencies in the data sequence to help improve the accuracy of click prediction.

Table 2: Overall performance of different models in terms of AUC and RIG.

Model	AUC	RIG
LR	87.48%	22.30%
NN	88.51%	23.76%
RNN	**88.94%**	**26.16%**

Experimental Results

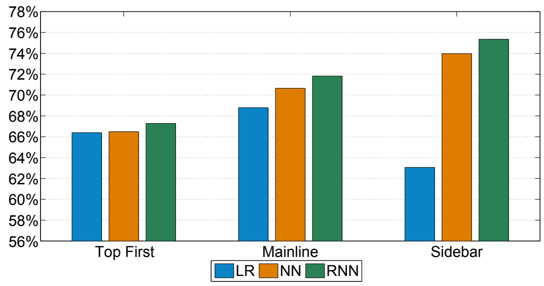
Overall Performance For fair model comparison, we carefully select the parameters of each model with cross validation, and ensure every model achieve its best performance respectively. To be more specific, parameters for grid search include: the coefficient of L2 penalty, the number of training epochs, the hidden layer size for RNN and NN models, and the number of unfolding steps for RNN. Finally, we get the best settings of parameters as follows: the coefficient of L2 penalty is $1e - 6$, the number of training epochs is 3, the hidden layer size is 13, and the number of unfolding steps should be 3 (more details will be provided later).

Table 2 reports the overall AUC and RIG of all three methods on test dataset. It demonstrates that our proposed RNN model can significantly improve the accuracy of click prediction, compared with baseline approaches. In particular, in terms of **RIG**, there is about **17.3%** relative improvement over LR, and about **10%** relative improvement over NN. As for the metric of **AUC**, we can find there is about **1.7%** relative gain over LR, and about **0.5%** relative gain over NN. In real sponsored search system, such improvement in click prediction accuracy will lead to a significant revenue increment.

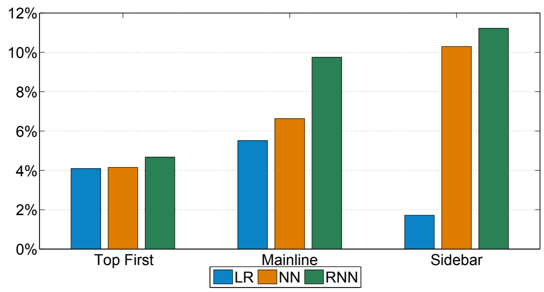
The overall performance above shows the effectiveness of our RNN model, which clearly transcends sequence independent models. Next, we will conduct detailed analysis on how the sequential information help to get more accurate click prediction.

Performance on Specific Ad Positions It is well-known that the click-through rate on different ad positions varies a lot, which is often referred to as the position bias. To further check the performance of models within specific positions, in our experiments, we separately analyze the performance of RNN model and two baseline algorithms on different ad positions: top first, mainline and sidebar.

Figure 6 shows the evaluation results on different positions. In Figure 6(a), RNN outperforms NN and LR measured by AUC on all positions. In Figure 6(b), in terms of



(a) AUC on specific ad positions



(b) RIG on specific ad positions

Figure 6: Performance on specific ad positions.

RIG, RNN achieves impressive relative gain over NN and LR, especially on mainline positions, where RNN beats NN by **3.12%**. According to our statistics on daily traffic data, most of revenue comes from the mainline ad clicks, where RNN can achieve significantly better performance. While for sidebar positions, the ads shown there are easily to be ignored by users, so that the clicks or positive instances are very rare. This may drastically hurt the performance of LR model. Nevertheless, the RNN model still performs the best, which indicates that even in rare cases, sequential information can still help.

Effect of Recurrent State for Inference We have shown the inference method of RNN in Figure 5. To further verify the importance of utilizing historical sequences in the inference phase, we remove the recurrent part of the RNN model after training, and feedforward the testing samples as a normal NN, which means that we just ignore the sequential dependencies in the testing phase. Finally, the AUC is 88.25% and RIG is 18.95%. Compared to Table 2, we can observe a severe drop of the performance of RNN model, which shows that the sequential information is indeed embedded in the recurrent structure and significantly contribute to the prediction accuracy.

Performance with Long vs. Short History In this part, we conduct experiments to check the model performance with different length of history. We first collect all available user sequences whose length is larger than a threshold T. In these sequences, the first T samples in each sequences are fed into model to serve as the "accumulation period". Then, we continue feeding and testing samples for the rest part of each sequence, and calculate the AUC and RIG on

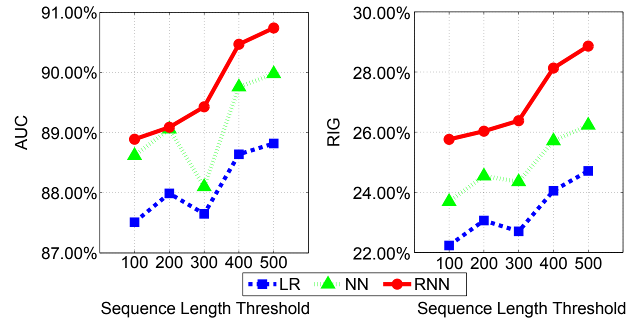


Figure 7: Performance with different history length T.

all those rest parts. In such setting, the user sequences which are selected with a larger threshold T have longer history to feed as "accumulation period". By doing so, we aim to verify whether our RNN model can maintain more robust sequential information in longer sequences.

Figure 7 shows the results. Just as expected, it turns out that our RNN model performs the best in all settings. Moreover, when the "accumulation period" gets longer, our RNN model tends to achieve even more relative gain compared to baseline models. This result validates the capability of RNN to capture and accumulate sequential information, especially for long sequences, and help further improve the accuracy of click prediction.

Effect of RNN Unfolding Step As described in the framework section, the unfolding structure plays an import role in RNN training process with BPTT algorithm. Since unfolding can directly incorporate several previous samples, and the depth of such explicit sequence modeling is determined by the steps of unfolding, it is necessary to delve into the effect of various RNN unfolding steps. This further analysis can help us better understand the property of the RNN model.

According to our experimental results, the prediction accuracy surges along with the increasing unfolding steps at the beginning. The best AUC and RIG can be achieved when unfolding 3 steps, after which the performance drops. By checking the error terms during the process of BPTT, we discover that the backpropagated error vanishes after 3 steps of unfolding, which explains why larger unfolding step is detrimental.

With these observations, intuitively, our RNN based framework can model sequential dependency in two ways: short-span dependency by explicitly learning from current input and several of its leading inputs through unfolding; long-span dependency by implicitly learning from all previous input, accumulated or embedded in the weights of the recurrent part. Meanwhile, in sponsored search, user's behavior is also affected by both the very recent events as explicit factor and the long-run history as implicit (background) factor. This reveals the intrinsic reason why RNN works so well for the sequential click prediction.

Conclusion and Future Work

In this paper, we propose a novel framework for click prediction based on Recurrent Neural Networks. Different from traditional click prediction models, our method leverages the temporal dependency in user's behavior sequence through the recurrent structure. A series of experiments show that our method outperforms state-of-the-art click prediction models in various settings. In this future, we will continue this direction in several aspects: 1) The sequence is currently built on user level. We will study different kinds of sequence building methods, e.g. by (user, ad) pair, (user, query) pair, advertiser, or even merge all users on the level of whole system. 2) We are going to deduce the meaning of dependency learnt by RNN via deep understanding of RNN structure. This may help up better utilize the property of the recurrent part. 3) Recently, some research work (Hermans and Schrauwen 2013) has been done on Deep Recurrent Neural Networks (DRNN) and shows good results. We plan to study whether "deep" structure can also help in click prediction, together with the "recurrent" structure.

References

Auli, M.; Galley, M.; Quirk, C.; and Zweig, G. 2013. Joint language and translation modeling with recurrent neural networks. EMNLP.

Bengio, Y.; Schwenk, H.; Senécal, J.-S.; Morin, F.; and Gauvain, J.-L. 2006. Neural probabilistic language models. In *Innovations in Machine Learning*. Springer. 137–186.

Box, G. E.; Jenkins, G. M.; and Reinsel, G. C. 2013. *Time series analysis: forecasting and control*. Wiley. com.

Fain, D. C., and Pedersen, J. O. 2006. Sponsored search: A brief history. *Bulletin of the American Society for Information Science and Technology* 32(2):12–13.

Graepel, T.; Candela, J. Q.; Borchert, T.; and Herbrich, R. 2010. Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 13–20.

Graves, A.; Liwicki, M.; Fernández, S.; Bertolami, R.; Bunke, H.; and Schmidhuber, J. 2009. A novel connectionist system for unconstrained handwriting recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31(5):855–868.

Hermans, M., and Schrauwen, B. 2013. Training and analysing deep recurrent neural networks. In *NIPS*, 190–198.

Hillard, D.; Manavoglu, E.; Raghavan, H.; Leggetter, C.; Cantú-Paz, E.; and Iyer, R. 2011. The sum of its parts: reducing sparsity in click estimation with query segments. *Information Retrieval* 14(3):315–336.

Jansen, B. J., and Mullen, T. 2008. Sponsored search: An overview of the concept, history, and technology. *International Journal of Electronic Business* 6(2):114–131.

Kirchgassner, G.; Wolters, J.; and Hassler, U. 2012. *Introduction to modern time series analysis*. Springer.

Kombrink, S.; Mikolov, T.; Karafiát, M.; and Burget, L. 2011. Recurrent neural network based language modeling in meeting recognition. In *INTERSPEECH*, 2877–2880.

McMahan, H. B.; Holt, G.; Sculley, D.; Young, M.; Ebner, D.; Grady, J.; Nie, L.; Phillips, T.; Davydov, E.; Golovin, D.; Chikkerur, S.; Liu, D.; Wattenberg, M.; Hrafnkelsson, A. M.; Boulos, T.; and Kubica, J. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*.

Mikolov, T.; Karafiát, M.; Burget, L.; Cernockỳ, J.; and Khudanpur, S. 2010. Recurrent neural network based language model. In *INTERSPEECH*, 1045–1048.

Mikolov, T.; Kombrink, S.; Burget, L.; Cernocky, J.; and Khudanpur, S. 2011a. Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 5528–5531. IEEE.

Mikolov, T.; Kombrink, S.; Deoras, A.; Burget, L.; and Černockỳ, J. 2011b. Rnnlm-recurrent neural network language modeling toolkit. In *Proc. IEEE workshop on Automatic Speech Recognition and Understanding*, 16.

Mikolov, T. 2012. *Statistical Language Models Based on Neural Networks*. Ph.D. Dissertation, Ph. D. thesis, Brno University of Technology.

Radlinski, F.; Broder, A.; Ciccolo, P.; Gabrilovich, E.; Josifovski, V.; and Riedel, L. 2008. Optimizing relevance and revenue in ad search: a query substitution approach. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 403–410. ACM.

Richardson, M.; Dominowska, E.; and Ragno, R. 2007. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*, 521–530. ACM.

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 2002. Learning representations by back-propagating errors. *Cognitive modeling* 1:213.

Shaparenko, B.; Çetin, Ö.; and Iyer, R. 2009. Data-driven text features for sponsored search click prediction. In *Proceedings of the Third International Workshop on Data Mining and Audience Intelligence for Advertising*, 46–54. ACM.

Wang, T.; Bian, J.; Liu, S.; Zhang, Y.; and Liu, T.-Y. 2013. Exploring consumer psychology for click prediction in sponsored search. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

Xiong, C.; Wang, T.; Ding, W.; Shen, Y.; and Liu, T.-Y. 2012. Relational click prediction for sponsored search. In *Proceedings of the fifth ACM international conference on Web search and data mining*, 493–502. ACM.

Xu, W.; Manavoglu, E.; and Cantu-Paz, E. 2010. Temporal click model for sponsored search. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, 106–113. ACM.