Research Article **Open Access**

Deep Learning Based Modeling in Computational Advertising: A Winning Formula

Mengmeng Chen* and Luis Rabelo

Department of Industrial Engineering and Management Systems, University of Central Florida, USA

Abstract

Targeting the correct customers is critical to computational advertising. Once you find your potential customer, you want to know who can offer the most suitable offer at the right time. There are other ways to visualize your perfect customer using deep learning, and we developed a method to target and attract the best online customers and get to know what they want before you start selling.

Keywords: Deep learning; Computational advertising; Intelligence

Introduction

Computational advertising is an emerging discipline that is driven by industry and gradually forms as a scientific system. Stanford University's description of it: "Computational Advertising is a comprehensive discipline involving large-scale search, text analysis, information retrieval, statistical modeling, machine learning, classification, optimization, and microeconomics. The core of computing advertising is to find the best match between the user and the ad in a given environment. The environment here can refer to the user entering a query in the search engine, or the user reading the web page, or the user watching the movie on a portable device. In traditional advertisements such as TV commercials, advertisers do not know who the audience is, what age group, and what they feel after watching the advertisement. Even the ratings of TVs cannot be accurately counted, let alone the effect of advertising. Computational advertising shows more intelligence: advertisers can understand where the target customers are, who sees the ads, the effects of the ads, etc., which can be roughly calculated. In general, computational advertising optimizes the entire advertising system, balancing advertisers, service providers (which can be search engines, or news portals) and the interests of users to maximize them. Therefore, for the effect of the advertisement, it needs to be calculated. For example, in the online advertising, the click rate, conversion rate, etc., advertisers can see the directly quantified advertising effect through calculation of advertising, and calculate the accurate ROI.

For users, most of them only want to see ads that they are interested in. Then the network service provider must first know who the user is, this through the user's usage behavior, log data mining, build user characteristics and interest models. Of course, this is often a very rough excavation, and the user's interest will also change dynamically. Therefore, sometimes the user's network usage behavior needs to be calculated in real time, and adjusted immediately, such as the user's search click behavior, can be used as a reference data. If the user does not click on the ad presented on the first page of the search, it means that the ad match may not be done well; then when the page is turned to the second page, the ad can be adjusted according to the previous behavior. Of course, this technology requires higher real-time calculations. Another thing is the cooperation between advertisers and service providers. There are related scientific issues such as game theory and auction mechanism. After all, from the perspective of the industrial ecosystem, each role can get enough benefits to make the entire industry chain prosper. If each role gets too many benefits and damages the partners, it will eventually affect the entire industry,

including itself. For example, the search engine's bidding ranking, the mechanism of bidding should be scientifically designed. According to the theory of game theory, the choice of no mechanism constraint is certainly not optimal. In the search auction ranking, assuming that there is no mechanism constraint, the second-ranked advertiser hopes to be slightly higher than the third-place bid, which can reduce the budget and will not change the ranking of the display. By analogy, the first place hopes to be slightly higher than the second place, and the third place hopes to be higher than the fourth place. The final result is that everyone's bid is close to zero, which actually destroys the business model of the bidding ranking. Inflicts damage on the entire industry. Therefore, the design of the bidding mechanism is crucial, and the interests of both parties should be fully balanced.

In addition, computational advertising also involves related disciplines such as natural language processing and machine learning. For example, using natural language processing can better identify the content of the web page and understand the context so that the "best match" advertisement can be displayed in combination with the user characteristics. Machine learning is more about helping to improve this match and constantly tuning the system.

Fundamentally, the main reason why computing advertising can arise is the Internet data capabilities of Internet companies. We define Computational advertising as following: Computational advertising is a programmatic based computational science, aiming at maximizing the comprehensive revenue of marketing, focusing on the relevance of user-advertising matching and the bidding model of advertising. Computational advertising involves the integration of natural language processing, data mining, revenue optimization, creative design and many other disciplines. For users, there are various forms such as business search advertisements, browsing page advertisements, and community crowd advertisements. The essence of computational advertising is to study how to use mathematical methods to solve various problems in activities.

*Corresponding author: Mengmeng Chen, Department of Industrial Engineering and Management Systems University of Central Florida, USA, Tel: 4078232204; E-mail: CHENMM@Knights.ucf.edu

Citation: Chen M, Rabelo L (2018) Deep Learning Based Modeling in Computational Advertising: A Winning Formula. Ind Eng Manage 7: 266. doi:10.4172/2169-0316.1000266

Literature Review

Customers are the most important pillar of a company's success. Without customers, business has no prospects for survival. Old fashion marketing usually take survey or interview towards current and potential customers. However, most of your customers hate doing a survey and demographics info on household basis are extremely expensive to purchase. Facebook provides the very successful lookalike modeling service in their advertising product, help programmatic advertising buyers targeting potential audiences who is similar to existing customers, which is very helpful and easy to use, and produces billions revenue every year for Facebook. However, Facebook can be successful is because they have tons of real person data all over the world, they could easily target people by their like, habits, age, gender, education and all kind of personal information. For business buying advertising in other platforms a.k.a Google AdWords or Bing search, buyer has to create their own lookalike model and there's a deliberate conundrum about the accuracy. How to improve the audience targeting accuracy becomes a trending topic.

Audience targeting

For example, Heckman in 1979 mentioned sample selection bias might not be able to representative of a more general population. Kim and Rossi in 1994 indicated that more frequent and higher spending consumers are more price-sensitive and this can lead to inefficient prediction when apply models to unrepresented training sample. Similar examples can be found in other fields, including bioinformatics and brain computer interfacing, natural language processing, computer vision, robot control, software engineering.

Targeting models in machine learning are often trained based on aggregate data, while data aggregation topic is also very important in forecasting research. Many papers in this literature review contains different models trained at different levels of aggregation. Promotion sells with individual level data and found individual level models does not produce more accurate predictions. Eijte et al. [1] did the same prediction sale forecast as well. A comparison on different models established on disaggregates data and daily sponsored search data that includes ad position variation, and they found aggregate data on daily basis yields imbalanced estimation.

The accuracy of predictive models decays with lower quality predictors, but the rate of deterioration varies between methods. The authors found Lasso-driven methodologies could adjust the weak predictors by assigning zero weights to these subsets [2]. On the other hand, distance-driven methods such as Euclidean distance may be more sensitive as weak predictors would produce noise.

Machine learning methods such as logistic regression, decision trees, neural networks provides better performance in customer segmentation compare to basic recency, frequency and monetary analysis; and there exist variation between different data mining methods when predicting. Models using factor analysis in macroeconomic forecasting could reduce dimensionality and are more accurate than models which do not apply factor analysis, for example OLS regression and bagging. Bayesian networks and Naive Bayes have better performance compare to clustering while making sub-prime loan borrower approval decisions.

Finally, there also exist very rich literature in machine learning methods for predictions in other area, Decision tree (C5) outperformed logistic regression and neural network in predicting breast cancer survivability. Neural networks outperformed logistic regression nd tree-based classification algorithm (C5) for predicting death after injury. SVM performances better than discriminant analysis, fuzzy clustering, logistic regression, neural networks and random forests in predicting diabetes.

Click through rate prediction

A lot of different models was proposed to predict the click through rate in both academia and industry, which most of those are using machine learning. Basically those can be divided into 3 different types: linear regression, nonlinear and fusion models. Richardson et al. [3] developed a logistic regression model to predict the click through rate for searching ads includes numbers, positions and other features. A decision trees in analyzing the internal correlations among different features was dig out by Dembczynski et al. [4] applied and he found the nonlinear relationships between the target and other supporting factors. A Logistic Regression method predict the click through rate was developed by Chapelle [5] proposed for Yahoo. A Fusion model combined logistic regression and decision tree was developed by He et al. [6] for the advertisement system of Facebook. Menon et al. [7] applied a maximum likelihood algorithm to compute the probabilistic model of the click through rate. Because of the business needs most of existing methods of predicting click through rate do works well and provides somehow satisfied results based on massive data training.

Applied a rapid level by level training process in solving Deep Neural Network connection hardness. Based on this achievement we may train neural network in a much more faster way. In this paper, we will apply character level CNN architectures towards click through rate prediction helping filter out most valuable text ads using Tens or Flow.

In the next section we will introduce our re-search methodology, describe the data and model design of this experiments, and discuss the targeting variables and the output measure used to train and evaluate the models [8-15].
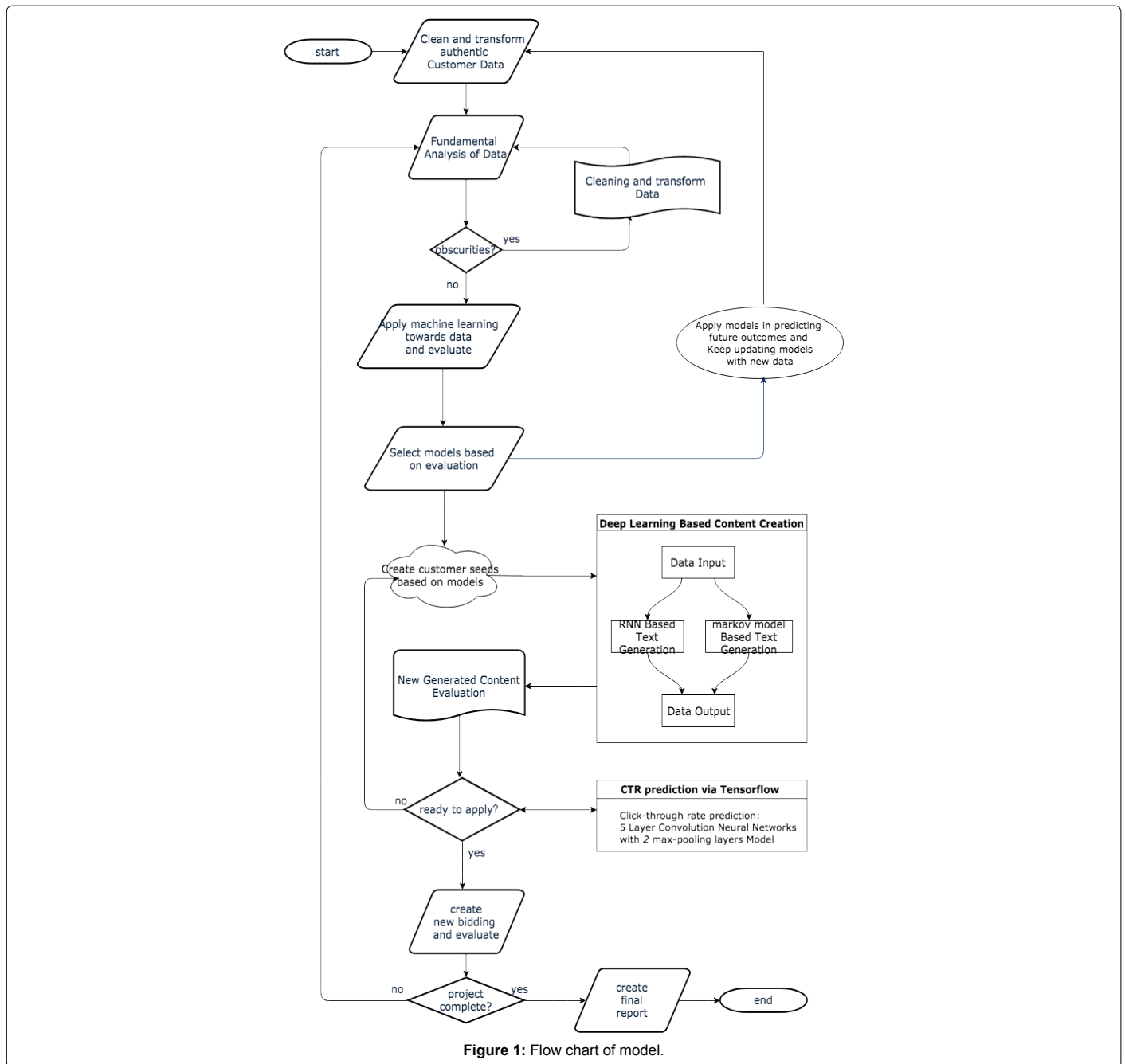
Methodology

Advertising to the most valuable audience group will benefit advertiser where here is the author, platform such as Google, Bing, Facebook as well as the customers. And this is the key problem in the field of online advertising. Inappropriate targeting will lead to the waste of marketing budget. As a result, we here build this deep learning based audience targeting model and test it with real business data.

Machine learning out the customer seeds

Our first step is applied different machine learning methodologies toward the sample data and predicting the outcome as accurate as possible. If data still obscurities after finishing fundamental process, we have to go back redo the cleaning and transformation process. If models passed evaluation we apply new incoming data to update the existing models.

The methodologies we applied here could be either supervised learning or unsupervised learning really depending on what kind of data we have: for supervised learning we will apply Decision Tree, Logistic Regression, Deepnets, Fusion, or Time series; for Un-supervised learning could be clustering or associations. This step will help us create customer seed groups based on existing data. For example after the process we may find from previous customer information, customers from specific age group, household income, hair color, reading preference or zip code are more willing to purchase, or even more, customers searched some topics are more likely to purchase. Those different attributions will bring us different level of targeting purpose and will make future marketing much more efficient [16-19] (Figure 1).

Figure 1: Flow chart of model.

Text generation using recurrent neural net works

After we successfully created valuable customer seeds group, we will be applying Recurrent Neural Networks or Markovify in generating new effective contents for audience targeting purpose. The reason is we want to target customer based on the keyword they searched on Google AdWords or Bing or Facebook. Recurrent Neural Networks (RNNs) are very effective models that have been widely used in many nature language processing tasks. Unlike Markovify, Recurrent neural networks do not use limited size of context, while information could cycle inside those networks until get enough outcome. We will explain RNN first then Markovify. For short term AdWords content/keywords generation: Assuming we have a network with an input layer x, context layer or hidden layer s and the output layer y. The input to the network

in time t is x(t), output is y(t), and s(t) is state of the hidden layer. We compute hidden and output layers as follows:

$$x(t)=w(t)+s(t-1)$$

$$s_j(t) = f\left(\sum_i x_i(t) u_{ji} \right)$$

$$y_k(t) = g\left(\sum_i s_i(t) v_{kj} \right)$$

Where $f(z)$ is sigmoid activation function:

$$f(z) = \frac{1}{1+e^{-z}}$$

And $g(z)$ is softmax function:

$$g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}}.$$

$s(0)$ can be set as very small variable such as 0.1, $s(t+1)$ is a copy of $s(t)$ in next time steps. Output layer $y(t)$ represents probability distribution of next word given previous word $w(t)$ and context $s(t-1)$. At each training step, error vector is computed and updated with the standard back propagation algorithm:

$$error(t) = desired(t) - y(t)$$

where desired is a vector using 1-of-N coding representing the word that should have been predicted in a particular context and $y(t)$ is the actual output from the network. And in order to performance improvement, we combine all text/words which less occurred in the training text into a special rare token. And chose Word probabilities computed as:

$$P(w_i(t+1) \mid w(t), s(t-1)) = \begin{cases} \frac{y_{rare}(t)}{C_{rare}} & if\ w_i(t+1)\ is\ rare, \\ y_i(t) & otherwise \end{cases}$$

Below is an example of Recurrent Neural Networks with 4-dimensional input layers and output layers, with one hidden layer of 3 neurons (Figure 2).

For the example we see that when the RNN found the "h" is assigned confidence of 1.0, 2.2 to e, -3.0 to l, and 4.1 to o. This chart shows when RNN fed the word:"HELL" as input, output layer would confidently assign for the ext word as "HELO" and we want the green numbers to be high and red numbers to be low. Since we can run the back propagation algorithm to figure out in what direction we should adjust every one of its weights to increase the scores of the correct targets (green bold numbers).

We applied textgenrnn model in this section which takes in an input of up to 40 characters, converts each character to a 100-D character embedding vector, and feeds those into a 128-cell long-short-term-memory (LSTM) recurrent layer. That output is mapped to probabilities for up to 394 different characters that they are the next character in the sequence, including uppercase characters, lowercase, punctuation, and emoji. (if training a new model on a new dataset, all of the numeric parameters above can be configured).

Markovify is another method to predict words. Mathematically speaking, the Markovify provides huge article data sets which full fill conditional probability distribution calculation of the next state words prediction which depends on the current word. Basically Markovify needs tons of previous sentence to train, and it will provides more specific reasonable generation. While RNN doesn't rely on the dataset but less readable. When to use which tool to generate content really depends on your aim [20,21].

Character-level convolutional networks

Once we finished with content generation and those words are good to be applied towards bidding, we have to estimate their click through rate in order to choose specific bidding price for each group. And we will apply the Character level CNN in click-through rate prediction.

The reason we use this CNN based prediction no other logistical regression or maximum likelihood is because our content are words. Another method to predict CTR based on keyword level is using Latent Dirichlet allocation model. Our Character level networks process text as a succession of characters, need zero of any knowledge of words, semantic or syntactic structures of any language. Our model is a 5-layers Convolution Neural Networks with: 2 max-pooling layers of kernel size 5 and 3 respectively, 2 dropout layers of probability 0.4, and ends with a sigmoid. It was then trained to minimize the squared error loss between ground truth and predicted CTRs. We can thus process batches of different dimensions between min size and max size (here set to 96 and 156 respectively) characters by padding each title with $(0,0,...,0)$ vectors to reach the maximum title length within the batch.
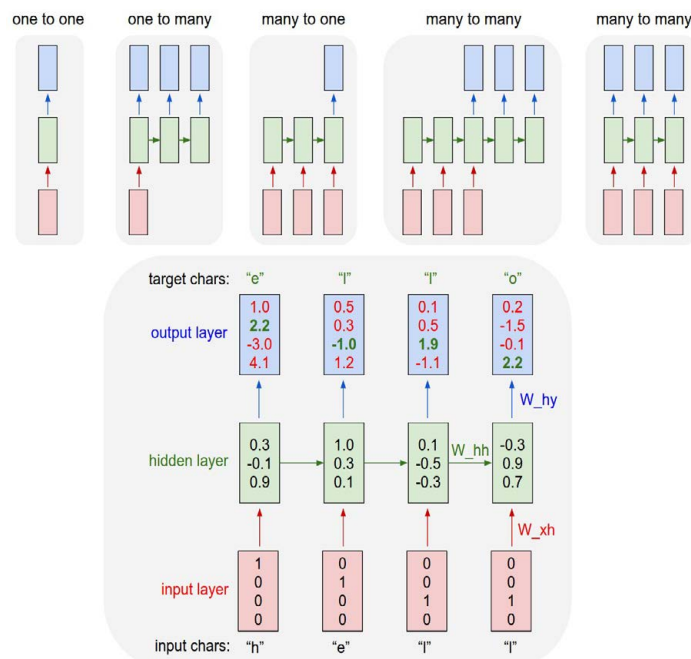

Figure 2: Text generation in recurrent neural networks.

When the title length was greater than max size, we randomly choose a subset of it (Figure 3).

The Experimental Dataset and the Evaluation Criteria

The data set we applied in this model is from a Private university in US. We collected enrollment of university between Jan 2017-June 2018, data includes all enrolled students' info, Google AdWords marketing data (leads, searched keywords, cost, etc.) during the same time. We first applied machine learning methodologies toward the enrollment data set and trying to train customer seeds group. The objective field will be student enrollment status and we got pretty decent model results after evaluations, with high AUC, High accuracy, no matter from logistic regression, decision tree or deep nets (Table 1).

We select just one evaluation as example showing as follow. The next step will be evaluating the importance of different subjects. We may weight the importance of subjects and can create customer seeds from different weight based on different groups (Table 2).

Based on the importance analysis we found green group plays as most important factor in predicting enrollment status, yellow group plays as medium important role and the red group may not effective the decision (Table 3).

We then applied Recurrent Neural Networks generate text content for searching purpose (Figure 4).

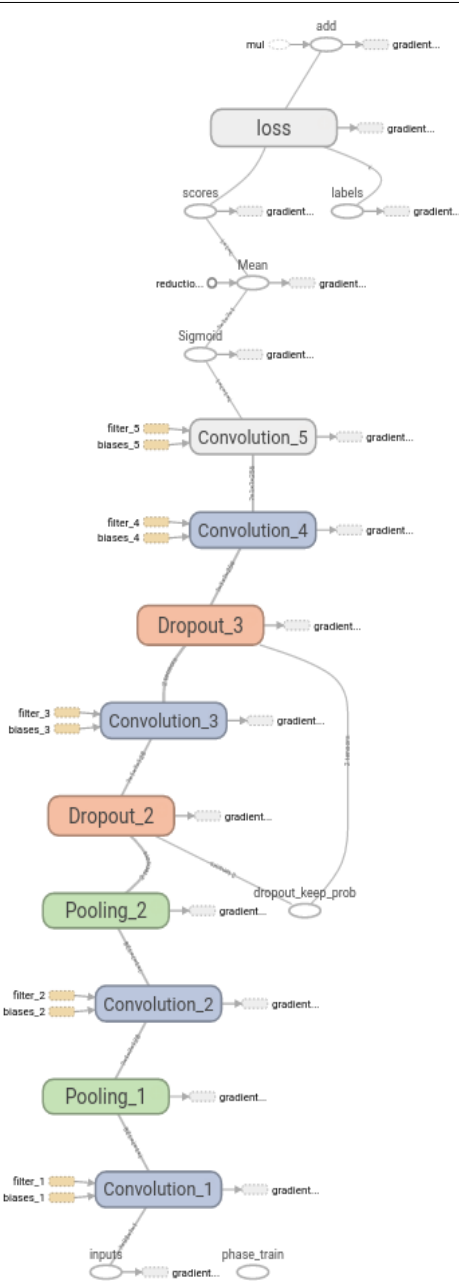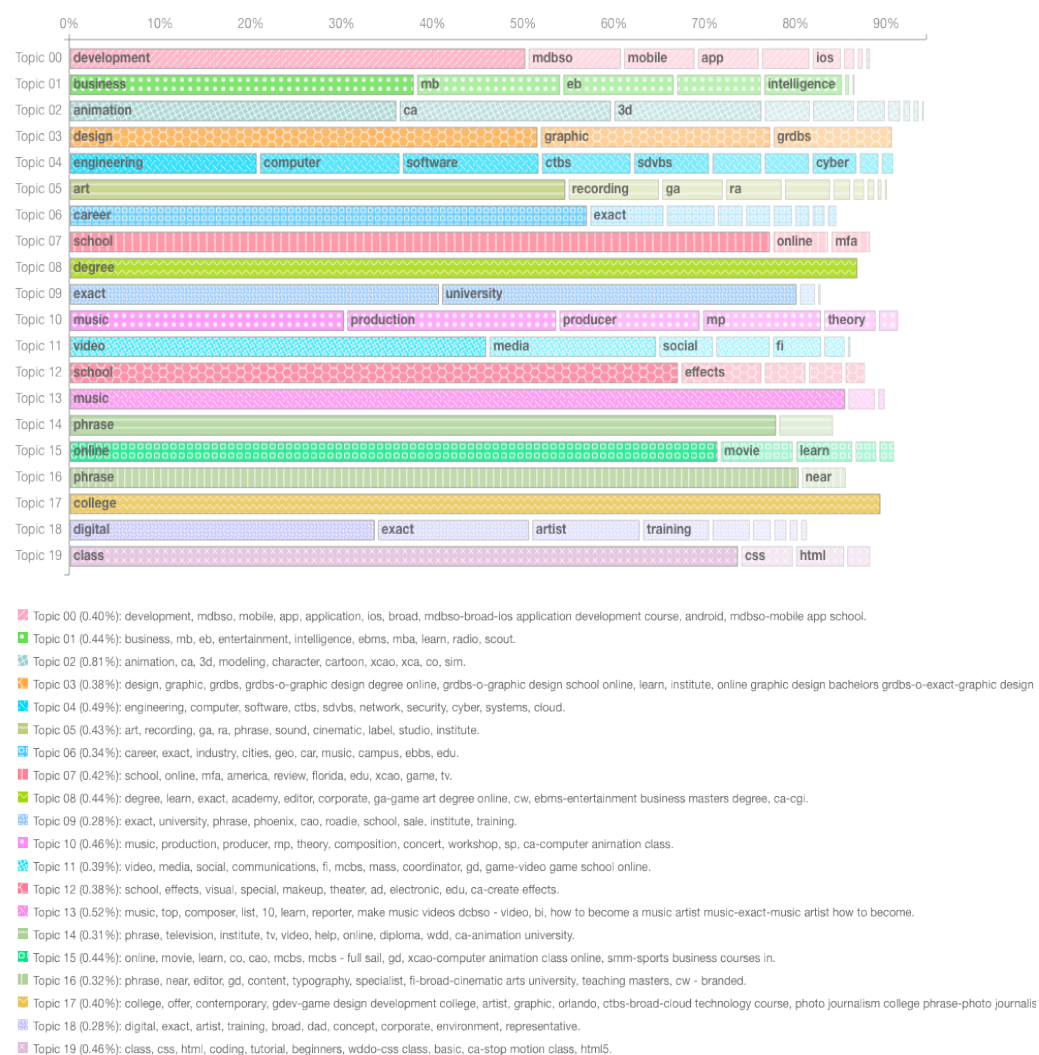The last step will be using character-level CNN predicting Click-

Figure 3: Character-level CNN for CTR prediction.

Resource ID	Metric_variance	Average_kendalls_tau_b	Average_balanced_accuracy	Average_spearmans_rho	Average_area_under_roc_curve	Average_ks_statistic	Accuracy
Ensemble/5	0.00122	0.17129	0.7318	0.18677	0.91815	0.84981	0.88792
Ensemble/5	0.00159	0.16086	0.67952	0.17599	0.97794	0.94663	0.90312
Deepnet/5bl	0.00003	0.15766	0.755	0.17904	0.97687	0.93639	0.8017
logisticregre	0.00026	0.18099	0.70921	0.19784	0.9663	0.91214	0.8903
logisticregre	0.00016	0.14427	0.70831	0.17087	0.96131	0.89943	0.88928
Deepnet/5bl	0.00002	0.15709	0.67883	0.17222	0.97886	0.94669	0.89756
logisticregre	0.00011	0.14348	0.70631	0.17026	0.96116	0.90112	0.88894
Model/5bb0	0	0.18524	0.70949	0.19974	0.94996	0.86561	0.89041
Ensemble/5b	0.00315	0.0959	0.69917	0.11742	0.71394	0.63117	0.85876

Table 1: Evaluation Matrix 1.

Actual vs. Predicted	Active	Negative Classes	Actual	Recall
Active	767	11	778	98.59%
Negative classes	117	7,920	8,037	98.54%
Predicted	884	7,931	8,815	98.57% Avg. Recall
Precision	86.76%	99.86%	93.31% Avg. Precision	98.55% Accuracy

Table 2: Evaluation sample.

Figure 4: Recurrent neural networks generate text content for searching purpose.

Field	Importance	Field	Importance	Field	Importance
MNC Keyword	0.31116	Highest level of education obtained	0.02178	Person Account Birthdate.day-of-month	0.00884
Status Category	0.26666	Person Account: Birthdate.year	0.02121	Person Account: Birthdate.month	0.00778
Attendance Status	0.06467	Grad Year Formula	0.02023	1stMilCertDate.day-of-month	0.00585
Lead Age Date	0.05807	Application Date.month	0.01178	Application Date.day-of-week	0.00504
CIP Code	0.04037	Application Date.day-of-month	0.01026	Person Account: Birthdate.day-of-week	0.00498
SAP Status	0.03281	Limbo Dates	0.00976	Affiliate Code	0.00459
Application Date. Year	0.03168	1st MilCertDate.month	0.00938	Gender	0.00447
Lead Group	0.0311	Lead Category	0.00907	Actual Start Date.month	0.00434

Table 3: Importance Ranking.

Week	Conv	Cost/Conv	Cost	Clicks	Impr	Avg. CPC	Conv. Rate
31 Aug	486	$178.06	$86,537.36	7,995	152,095	$10.82	6.08%
3 Sep	1,598	$175.04	$279,712.78	21,981	415,223	$12.73	7.27%
10 Sep	1,572	$173.15	$272,185.11	21,165	410,890	$12.86	7.43%
17 Sep	1,197	$158.99	$190,308.48	15,428	293,898	$12.34	7.76%

Table 4: CNN predicting Click-Through Rate on topic purpose.
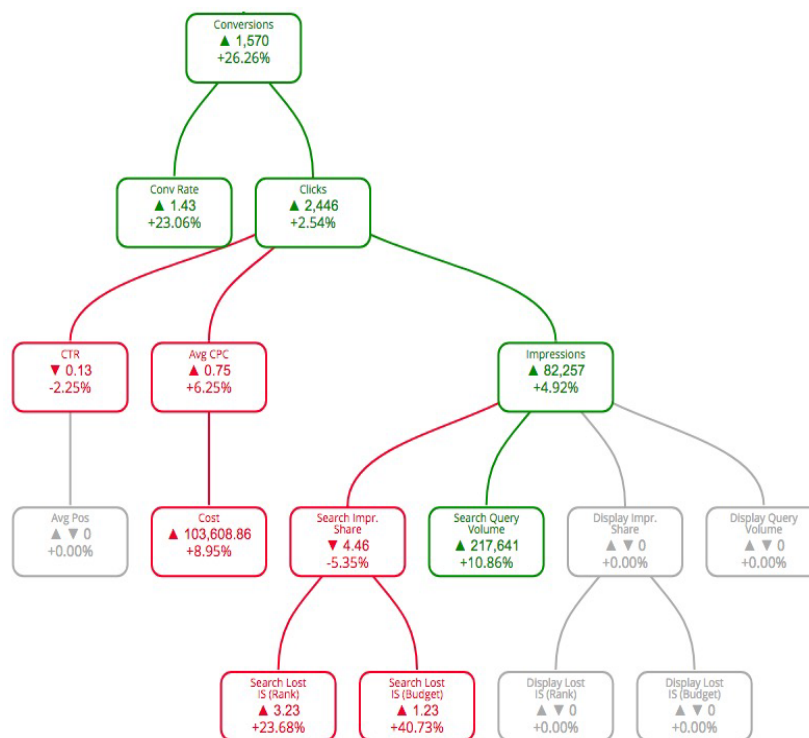

Figure 5: Results shows very significant improvement.

Through Rate on topic purpose. We combined predicted click through rate and new generated content, setting the new bidding price on Google AdWords and test for 30 days (Table 4).

The results show very significant improvement as above (Figure 5).

Conclusion

We have presented a novel mechanism for computational advertising as the buyers side. It is based on machine learning, deep neural network which is known to improve the training of computational advertising systems, also we applied this deep learning based model toward real practice and good results are presented. This system model is proven to be useful in sponsored search marketing platform and theoretically could be applied toward other area such as social marketing, affiliate marketing, etc. Further study could focus on different content generation other than just text.

References

1. Eijte WF, Leeflang PSH, Wittink DR (1994) A comparison and an exploration of the forecasting accuracy of a loglinear model at different levels of aggregation. Int J Forecast 10: 245-261.

2. Jerome F, Hastie T, Tibshirani R (2009) The elements of statistical learning. Springer series in statistics.

3. Richardson M, Dominowska E, Ragno R (2007) Predicting clicks: Estimating the click-through rate for new ads. In: Proceedings of the 16th International World Wide Web Conference, WWW '07, pp: 521-530.

4. Dembczynski K, Kotlowski W, Weiss D (2008) Predicting ads click-through rate with decision rules. In: Proceedings of the Workshop on Targeting and Ranking in Online Advertising.

5. Chapelle O (2014) Modeling delayed feedback in display advertising. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'14, USA, pp: 1097-1105.

6. He X, Pan J, Jin O, Xu T, Liu B, et al. (2014) Practical lessons from predicting clicks on ads at Facebook. Proceedings of the 8th International Workshop on Data Mining for Online Advertising, ADKDD 2014 - Held in Conjunction with SIGKDD 2014.

7. Menon AK, Chitrapura K, Garg S, Agarwal D, Kota N (2011) Response prediction using collaborative filtering with hierarchies and side-information. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD'11, pp: 141-149.

8. Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. Neural Computation 18: 1527-1554.

9. Max Woolf : https://minimaxir.com/

10. Zhang X, Zhao J, LeCun Y (2015) Character-level Convolutional Networks for Text Classification. Cornell University Library.

11. Conneau A, Schwenk H, Le Cun Y (2016) Very Deep Convolutional Networks for Natural Language Processing. Cornell University Library

12. Karsten MB, Gretton A, Rasch MJ, Kriegel HP, Schlkopf B, et al. (2006) Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics 14: e49-e57.

13. Tatjana E, Lang B (2006) Efficient optimization of support vector machine learning parameters for unbalanced datasets. Journal of Computational and Applied Mathematics 196: 425-436.

14. Erdem T, Keane MP (1996) Decision-making under uncertainty: Capturing dynamic brand choice processes in turbulent consumer goods markets, Marketing Science 15: 1-20.

15. Benot F, Verleysen M (2014) Classification in the presence of label noise: A survey. IEEE Trans Neural Netw Learn Syst 25: 845-869.

16. Joo G, liobait I, Bifet A, Pechenizkiy M, Bouchachia A (2014) A survey on concept drift adaptation. ACM Computing Surveys (CSUR) 46: 44.

17. Arthur G, Borgwardt KM, Rasch MJ, Schlkopf B, Smola A (2012) A kernel two-sample test. J Mach Learn Res 13: 723-773.

18. Won J, Lowe L, Rabelo L, Lee G, Kwon O (2017) A Methodology on Guiding Effectiveness- Focused Training of the Weapon Operator Using Big Data and VC Simulations. SAE Int J Aerosp SAE International Journal of Aerospace 10: 57-67.

19. Mykola P, Bakker J, Liobait I, Ivan-nikov A, Krkkinen T (2010) Online mass flow prediction in CFB boilers with explicit detection of sudden concept drift. SIGKDD Explor 2: 109-116.

20. Keathley-Herring H (2017) An Approach to Quantify the Factors That Affect Performance Measurement System Implementation. EMJ-Eng Manag J 29: 63-73.

21. Rahal AD, Rabelo LC (2006) Assessment framework for the evaluation and prioritization of university inventions for licensing and commercialization. EMJ-Eng Manag J 18: 28-36.