Outbrain Click Prediction
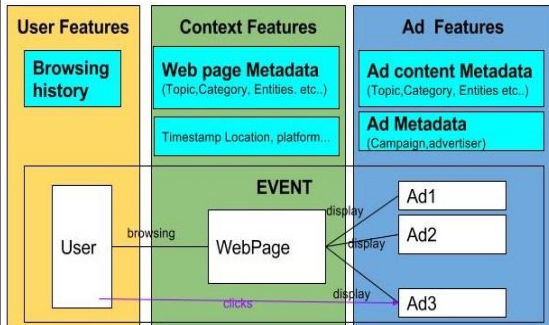
By Prashant Jaiswal(prjaiswa@stanford.edu), Vijayaraj Gopinath (vgopinat@stanford.edu) and Girish Limaye (glimaye@stanford.edu)

Problem : The key goal of the online display advertisers is to present users content they might be interested in while the they surf on their favourite sites. Our task is to predict click probability of Ads in a document by the users.

Key Data Considerations:
*We use data from Outbrain Kaggle contest
*Labelled data contains Ads, documents and click information. Documents have metadata like category, entity, publisher etc.
*Ads also have metadata like documents. User info like location, time is also available.
* We only use part of the massive data provided due to computational issues.

Method: Gradient boosted trees model is used after considering various models. We have sparse data & boosted trees model naturally handles missing data & categorical features very well. We fit a tree $\hat{f}^b$ in each iteration of the algorithm which is used to update the training algorithm.

$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x).$$

In subsequent iterations, we fit the residuals to get trees which are aggregated in the end.

$$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i). \qquad \hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x).$$

Intelligent Feature Conjugation:
We exploit the features that are shared across Users, Context and Ads and use them for pairwise interactions.
1. We use pairwise feature conjugation for category, topic and entity across the User, context and Ad using an IDF style feature.
2. We calculate features based on pairwise conjugation across features using Bayesian priors.
3. Organic & derived features sum to around 35. Memory & computational requirements for this dataset limit the number of features that can be used.

Results: Mean average precision MAP@12 is used to measure the performance of algorithm.

$$MAP@12 = \frac{1}{|U|} \sum_{u=1}^{|U|} \sum_{k=1}^{min(12,n)} P(k)$$

	Training	Testing
Observations	~4M	~1M
MAP@12	0.634	0.615

To put in context, highest score for $100K prize money in Kaggle is **0.695** so far.

Discussion:
*Looking for indirect relationships in the data certainly improved the prediction accuracy.
**Running the algorithm on cluster with lot of data would certainly improve MAP score.

Future:
*Explore the method's applicability beyond display ads in search related promoted content.
*Exploit ternary feature conjugation using a similar method.
*Compare the results to FieldAware Factorization machine that have seen success recently.

Lee et al. 2012; Yan et al. 2014; Chapelle et al. 2014

S. Rendle, "Factorization machines with libFM," ACM Transactions on Intelligent Systems and Technology (TIST), vol. 3, no. 3, p. 57, 2012.