dstillery

PEER-REVIEWED PUBLICATION

Privacy-Friendly ID-free Digital Ad Targeting Using URL Embeddings

IEEE BIG DATA 2020

dstillery

Table of Contents

Privacy-Friendly ID-Free Digital Ad Targeting using URL Embeddings

1st Amelia White
Dstillery
New York City, USA
awhite@dstillery.com

2nd Chris Jenness
Dstillery
New York City, USA
cjenness@dstillery.com

3rd Jason Kaufman
Dstillery
New York City, USA
jkaufman@dstillery.com

4th Roger Cost
Dstillery
New York City, USA
rcost@dstillery.com

5th Wickus Martin
Dstillery
New York City, USA
wmartin@dstillery.com

6th Justin Moynihan
Dstillery
New York City, USA
jmoynihan@dstillery.com

7th Patrick McCarthy
Dstillery
New York City, USA
pmccarthy@dstillery.com

8th Melinda Han Williams
Dstillery
New York City, USA
melinda@dstillery.com

2020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/BigData50022.2020.9378394

***Abstract*—Today, predictive digital ad targeting typically relies on detailed user profiles. As consumers deserve and expect more internet privacy, we aim to develop methods to effectively target advertising in a way that respects consumers' wishes. In particular, we have developed ID-Free targeting, a method to target advertising based on predicted user behavior, without reference to any profile, web browsing history, or any other identifier of the user being targeted. With the restriction that historical user behavior is not available, the only accessible view of the user is the information about the digital ad opportunity itself, such as the URL and the time of day. In this work, we focus on the problem of making brand-specific targeting decisions based on the URL where the ad will be shown, that is, predicting which ad inventories are most likely to lead to a conversion for a particular brand. In order to address this problem, we learn rich URL embedding features for ad inventory URLs. We share the design of our online learning algorithm, which is a practical implementation for training the URL embeddings with the large, dynamic "corpus" that is the internet. We show that this approach to feature representation gives clear performance improvements over a naive representation, and improvements are even more pronounced when brand-specific training data are scarce. The resulting model enables a fully ID-Free delivery of precise ad targeting to users with completely unknown identity and historical behavior.**

***Index Terms*—digital advertising, neural networks, embedding, privacy**

I. Introduction

As consumers increasingly demand a more private internet browsing experience, the digital advertising industry has been challenged with finding an approach to delivering relevant advertising to users who have no associated profile or identifiers. Most predictive ad targeting today relies on the use of third-party browser cookies, which allow targeting companies to associate the web browser with a persistent, cross-site browser ID. This anonymous ID can be used for both data collection and ad targeting. That is, a user's website visits across websites can be collected and stored in a profile associated with their browser ID, which can then be used to execute targeted ad delivery via real-time bidding (RTB) advertising. Predictive advertising built on detailed browsing history enables highly accurate ad targeting, but can also inadvertently produce a negative user experience. Users may feel they are being followed around the internet or can be targeted out of context, for example receiving ads for personal products while at work. As a result, there is a movement in the industry away from the support of third-party cookies. Over 40% of U.S. internet users today use browsers that do not have third-party cookies enabled [1]. In early 2020, Chrome announced its intention to phase out support for third-party cookies in two years [2], which would effectively end the use of this technology in the digital advertising ecosystem.

At Dstillery, we have developed an approach for privacy-friendly ID-Free targeting to effectively target digital advertising to users with no identifiers of any kind and no history or other stored information available [3]. We focus on a form of ad targeting called prospecting, which aims to identify and reach users likely to become new customers of a brand. A prospecting campaign aims to drive devices to take an action that a campaign is trying to drive; we call this event a conversion. A conversion action is typically a product purchase or visit to the brand's homepage. Our approach to privacy-friendly ID-Free targeting involves training prospecting models based solely on information seen in a bid request. In this work, we focus on the most predictive signal, the inventory URL. We learn rich embedding features for ad inventory URLs, allowing us to train models that perform even when training data are scarce. We learn URL embeddings from website visitation behavior rather than from the text or other website content. The resulting embeddings reflect the browsing intent of the user, as websites viewed with similar intent are similarly located in embedding space.

One challenge of learning URL embeddings for our use case is that we are working with a large, dynamic set of web browsing behavior. This means there are many parameters to learn, and in order to capture a new website the whole model must be retrained with a new dictionary, which is impractical given the size of the task. To overcome this problem, we use a modified feature hashing trick to minimize the effect of hash collisions while reducing the number of parameters in the model. This avoids the need for a dictionary and lets us continuously train the model with online learning, rather than run large batch trainings.

We developed this approach to ID-Free targeting both in response to the large number of internet users currently unaddressable by third-party cookies and anticipating the deprecation of third-party cookies in the future. The models are "ID-Free" in the sense that they are used to target ad delivery to users without any identifiers, in contrast to standard approaches to predictive ad targeting. To build the models, we require data from a sample of users with identifiers, and in this paper, model building is performed on data gathered using third-party cookies. We intend to explore and demonstrate these techniques using the data sources available today, and this methodology can be used currently to reach users without IDs. Later, in the event that third-party cookies are no longer available, these techniques will be applied to different data sources that do not rely on third-party cookies. Section III describes the data sources used in this work as well as the sources we anticipate using in a scenario where third-party cookies are no longer available.

The novel contributions of this work include:

- Dictionary-free URL embeddings that can be continuously updated using online learning.
- The use of URL embeddings as features for ad targeting models that score ad opportunities based on a single URL and perform across a wide range of dataset sizes.
- Empirical insight into how a meaningful metric space learned via URL embeddings facilitates learning by allowing models to generalize what is learned from websites in the training dataset to websites rarely or never seen in training.
- A method to deliver predictive targeted digital advertising to users with no ID that is effective even with small training datasets.

II. Related Work

On the problem of the targeting of advertising to users without identifiers or stored behavior, two main approaches are prominent: traditional contextual and inventory modeling.

Traditional contextual targeting is when advertising is targeted based on the text of the website itself. Typically, the advertiser chooses keywords associated with a campaign, and advertising is delivered to webpages containing those words [4]. This differs from our approach in that delivery is optimized towards human-selected keywords, rather than a campaign-specific outcome. Additionally, it is limited to ad inventory with rich text from which keywords can be extracted. However, we note that the information used in this approach (words on the website) is fully complementary to that used by ours (behavioral sequences of website visitation). As such, the keyword data extracted in the traditional contextual approach can be used as an enhancement to our URL embeddings approach. This is an area of ongoing research outside the scope of this work.

Inventory modeling targets ads based on the past performance of each inventory URL [5]. This approach is similar to our ID-Free approach, and in fact, the baseline approach used in section IV is equivalent to inventory modeling. We will show that a significant drawback of inventory modeling is that it requires a substantial amount of data on a given inventory (gathered at a cost) to make a good prediction on that inventory. Our method overcomes this drawback by using a richer feature representation that incorporates information from a separate training dataset.

Dimensionality reduction is a well studied field of machine learning, consisting of both linear [6], [7] and nonlinear [8] methods. Reducing the dimension of the feature space can improve predictive performance by lowering variance, especially when data are limited. More recently, the field of representation learning has developed [9], which aims to learn features that improve the performance of machine learning tasks. One very successful representation learning approach uses neural networks to learn numerical embeddings, significantly reducing the dimension of large sparse feature spaces. Examples of this work can be found in the field of Natural Language Processing (NLP) [10], [11], and these methods have been applied in industry in domains such as ecommerce, recommendations and retargeting [12]–[14]. Feature hashing has been used to train embeddings for word representation [15], though to our knowledge has not been used to train embeddings for individual websites using online learning.

Several variants of embedding browsing histories or URLs have been used to address problems adjacent to ours. Recurrent Neural Networks have been used to embed or model complete browsing histories [16], [17]. Our problem differs from this one in that for us, entire sequences of web browsing events are not available and one must make a targeting decision based solely on the information available in a single bid request. Work embedding single URLs includes [18], which uses the characters and structure of the URL string itself to embed URLs to facilitate phishing detection. In contrast, we seek to understand the intent of real users visiting URLs, with the goal of predicting interest in brands and products. As such, examination of the URL string itself is not applicable to our problem. The approach presented here uses browsing behavior to embed URLs resulting in embeddings representing the intent of the website user.

III. method

Our goal is to predict, using only data available in a bid request, whether an ad opportunity is likely to lead to a conversion for a brand's digital advertising campaign. The main feature of interest available in a bid request is the ad inventory URL, i.e. the website on which the ad will be displayed. This feature is predictive because it often signals intent. For example, a user on shoe websites may be shopping for shoes, and a user reading about sports may be interested in tickets or ways to live stream games. To use the inventory URL as a model feature, we first need to encode it numerically. A naive way to do this is to represent URLs as binary categorical variables using one-hot encoding. This results in an extremely large, sparse feature space with dimensionality equal to the number of unique URLs. Because this space lacks a distance metric to capture the relationship between websites, models trained with it have no ability to generalize what is learned on one website to other websites. Instead, we learn a lower-dimensional embedding space in which close distance corresponds to URLs with similar content. The space, therefore, inherits a meaningful metric and gives the URLs a rich numerical representation.

Building ID-Free models is then composed of two steps. First, we learn an embedding space, and next, we use it to train the models. The first step is performed in an unsupervised manner over a large dataset. We learn the embeddings online so that they evolve over time to reflect the changing behavior of people browsing the internet. This enables us to add new websites as soon as we see them without retraining the entire embedding space.

The second step, which is to train the ID-Free models using the embeddings as features, is then achievable with simple models on small datasets. Reusing the same embeddings for different ID-Free models allows for an efficient model building pipeline, a business requirement of ours given our need to build and update thousands of models daily.

To learn URL embeddings, we need data that capture sequences of website visits by individual users. In this work, these data have been gathered by observing website visits tied to browser IDs in RTB bid requests. In the future, this data can be gathered from a different, cookie-free source. For example, website visits could be collected from a sample of users who have opted in to participate in a panel study and are fully compensated for their data. It is worth noting that this data need not be tied to any personal identifiers and does not require the collection and storage of complete user browsing histories. At any one time we only need to save up to two website visits by any user, and in the current implementation this data ages out of our system within eight hours.

To train ID-Free models, we need a "conversion" dataset that consists of website visits, each labeled by whether or not the visit was followed by a conversion (for example, a purchase on a brand's website). In this work, we gather a sample of conversion data by observing website visits in real-time bidding (RTB) bid requests tied to third-party cookie browser IDs and recording whether or not each browser ID is subsequently linked to a product purchase (or other conversion event) within a specified time window afterward. Standards for measuring conversion without third-party cookies are still being developed, and this work assumes that some mechanism for this measurement will be available if third-party cookies are no longer in use. A possible mechanism to measure click through conversion has been proposed by Chrome [19]. In this proposal, one would be able to measure whether a user converts after viewing and clicking on an ad for a brand. Conversion data could then be gathered, for example, by showing ads at different websites and observing which ad placements result in conversions. Note that under this proposal, gathering conversion data would require the delivery of ads and therefore require ad spend, so this data comes at a cost.

A. *Step 1. Learning URL Embeddings*

We wish to embed URLs such that URLs viewed in browsing sessions of similar intent are close in embedding space. Data similar to ours are found in Natural Language Processing (NLP), where text data can be viewed as categorical variables representing millions of words and phrases. Much work has been done to embed words into a lower dimensional vector space where feature vectors can take on a continuous range of values and where semantically similar words are near one another. This work relies on the distributional hypothesis [20], which says that words that appear in the same context have similar semantic meanings. Similar to this hypothesis, we noticed that URLs with related content are often viewed directly before or after one another in time. This motivates adapting NLP methods to create embeddings for URLs.

We adapt the word2vec algorithm from NLP, [10], [11], which predicts a (target) word from its neighbors (context words) while learning an embedding vector for each word as parameters of a model. For any pair of websites visited by a user in sequence, we analogously define one website as the target URL and the other as the context URL. We then use a modified version of the neural network structure used in word2vec to predict a target URL from a context URL. As a result, URLs often visited close, in sequence, to a particular target URL will be close in embedding space.

To train the embeddings online, we cannot use a predefined dictionary of websites (a mapping from each website to a unique index), since it would change over time as new websites appear and disappear in our data stream. Although feature hashing can be used to remove the need for a predefined dictionary, using a hash function with a large enough range to avoid collisions would lead to a very large network.

To reduce the number of parameters in our network while also taking advantage of feature hashing, we use a method proposed in [15] to modify the word2vec network structure. In addition to an initial input hash function with a range of size N, we use k intermediate hash functions, each with a smaller range of size M, that map each URL to k intermediate embedding vectors. We then use k trainable parameters to select, for each input hash value, the best linear combination of these k vectors to produce the final embedding vector. This creates a trainable mechanism to reduce the effect of collisions.

The network structure is shown in Figure 1, and consists of the following trainable variables:

Compression subnetwork

- An embedding matrix B of size M x d, where d is the size of the embedding vector (128) and M is the size of the intermediate hash range (250 thousand).
- A trainable matrix P of importance parameters of size N x k, where k is the number of intermediate hash functions (2) and N is the size of the input hash range (2.5 million).

Decompression layer

- A weight matrix of size N x d.

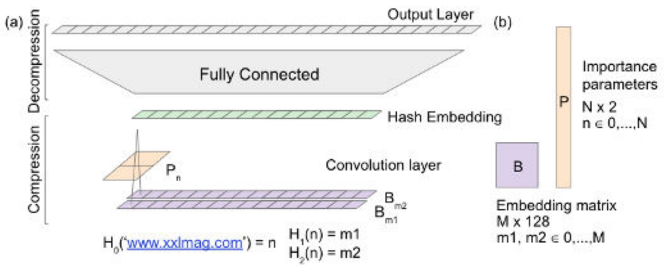

Fig. 1. Hash embedding network architecture (with k=2, d=128). a) Intermediate hashes (m1 and m2) of the input hash index (n) select embedding vectors from the embedding matrix B, which are then linearly combined in the convolution layer to produce the fully compressed hash embedding vector. b) Trainable parameters on the compression side of the network consist of the embedding matrix B and the importance parameters P.

In comparison the number of trainable variables in the standard word2vec algorithm is:

Compression subnetwork

- An embedding matrix of size N x d.

Decompression layer

- A weight matrix of size N x d.

To avoid collisions we choose a large input hash range N. We tune the size of the intermediate hash range, M, to find the best trade-off between embedding quality and space-efficiency. This structure therefore allows us to embed large vocabularies with many fewer parameters in the compression subnetwork.

We feed pairs of sequential URL visits by each user that occur within a short time limit into a training batch. While in this work we chose to record website visits by the domain rather than the full URL, it is also possible and procedurally equivalent to embed URLs at deeper levels, as long as there is adequate visitation volume to the chosen URL and level. The network is trained on 600 million visit pairs per day. The result is a numerical embedding of approximately 250K domains, reducing the dimensionality of our feature space from 250K to 128.

B. Step 2: Training ID-Free Models

We train ID-Free models to predict which ad opportunities are most likely to be followed by a conversion, that is, the action a campaign is trying to drive. The conversion action is typically a product purchase or visit to the brand's homepage. To train ID-Free models, we can use any information found in the bid request, which in addition to inventory URL includes features such as the time of day and the device location. This paper demonstrates our method using models built solely using ad inventory URL features and leaves the inclusion of additional features and subsequent feature selection as future work.

IV. Experiments and Results

First, we compare ID-Free models built using URL embeddings as features with models built using a baseline approach in which URLs are one-hot encoded. In the baseline approach, we refer to URL features as sparse categorical since URLs are represented in a high-dimensional, sparse space. For all experiments we trained logistic regression models, using cross-validation to choose the best regularization parameters. We chose to train logistic regression models as these models are simple and perform well. By significantly lowering the dimension of our feature space, we enable the use of more complex model types that would be impractical with a higher dimensional feature space; our results, therefore, give a lower bound on performance increase achievable with embedding features.

A. ID-Free Model Performance

To measure ID-Free model performance, we built models for 40 advertising campaigns; each model is a classifier that ranks bid requests by the probability that they lead to a conversion. As training data, we use inventory URLs from a week of bid requests, labeling bid requests with conversions from each advertising campaign. The data for each campaign were progressively downsampled to create six different sized datasets, with 488K, 244K, 97.6K, 48.8K, 24.4K, and 9.76K examples. Each dataset contains the same ratio of negative to positive examples (60:1). We sampled a test set from a future day of bid requests with respect to the training set, labeling a bid request as positive if the device seen on the bid request was seen taking the conversion action in the following seven day time period. The performance of the models on the test set is shown in Figure 2. Here we use two metrics to evaluate model performance: the area under the ROC curve (AUC), and the lift at 1% (Lf1), that is, the ratio of the precision of the top 1% of predictions to the precision of a random classifier.

The plots of both absolute and relative model performance show that URL embedding features outperform sparse categorical features for all dataset sizes, and the magnitude of the performance increase grows as the dataset size decreases. For smaller datasets the performance of models built with categorical features deteriorates until the models are no longer viable. Embedding features allow us to build viable models even on smaller datasets, which will be crucial in a scenario where third-party cookies are no longer available and conversion data become harder to obtain.

To further interpret these results, we focus on models trained with the largest datasets (488K) and examine model performance on different parts of each test set. Figure 3 shows performance grouped by the number of times each URL occurred in the training set. This breakdown gives insight into why models with embedding features outperform those with categorical features. Models built with categorical features are unable to generalize; AUC is 50 for websites never seen in training and increases only as the number of times the website was seen in training increases. In contrast, AUC for models built with embedding features is roughly the same regardless of how many times the website was seen in training, showing that these models can generalize what is learned from one website to other websites. This leads to a large difference in model performance for websites rarely seen and a decreasing difference in performance as the number of times the website

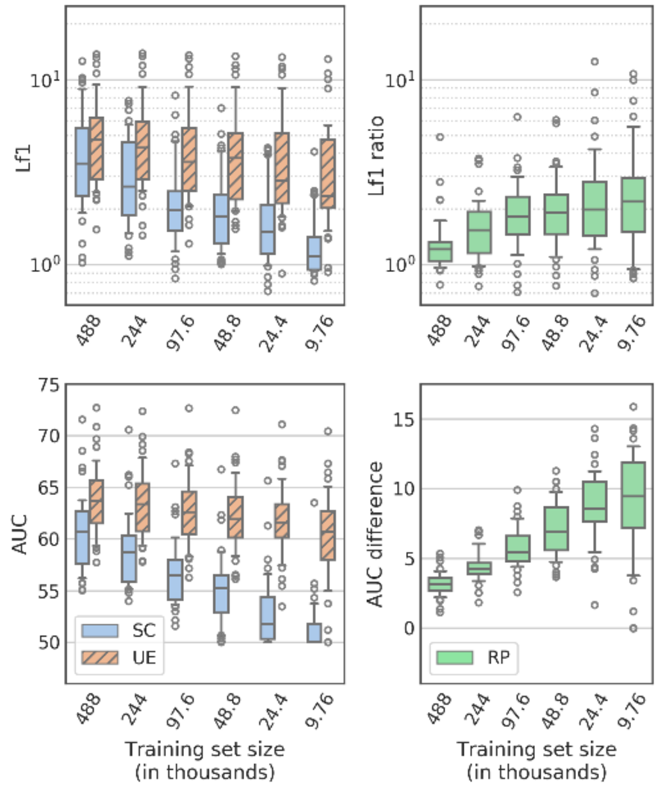

Fig. 2. Dependence of model performance on training set size for models with sparse categorical features (SC, blue) and URL embedding features (UE, orange). The relative performance is also shown (RP, green). The ratio of negatives to positives is 60. The whiskers indicate the 10th and 90th percentiles.

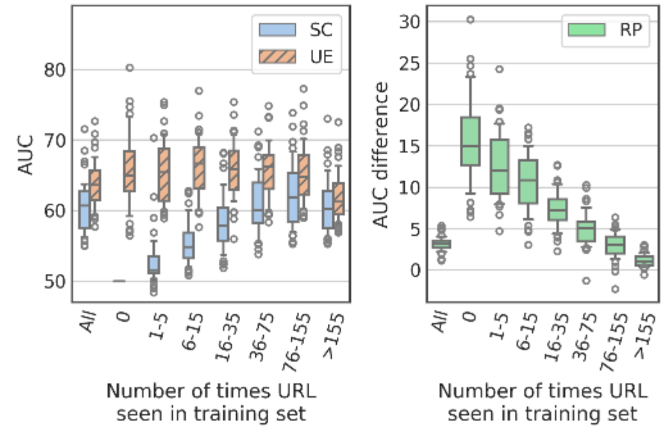

Fig. 3. Model performance on subsets of the test set for models with sparse categorical features (SC, blue) and URL embedding features (UE, orange). The relative performance is also shown (RP, green). The size of each training set is 488K and the ratio of negatives to positives is 60. The whiskers indicate the 10th and 90th percentiles. Performance is shown both for all examples in the test set (All) and bucketed by the number of times the URL occurred in the training set.

was seen increases. Figure 3 also shows a slight decrease in performance, for both model types, for the group of websites seen the greatest number of times. This could be due to very common websites being less predictive; a website such as yahoo.com, for example, with a large number of visitors, is unlikely to be an informative feature for a distinct behavioral trait.

B. URL Embedding Interpretability

We evaluate the URL embeddings themselves qualitatively by checking that their relative positions in the embedding space are interpretable. Table I shows, for example, the closest websites in embedding space to four distinct travel websites. This demonstrates that the embedding space is even able to differentiate between travel subtopics such as cruises, adventure travel, travel deals and wedding destinations.

TABLE I
EMBEDDING SPACE METRIC

Website	**Closest websites in embedding space**
princess.com	directlinecruises.com cruiseweb.com royalcaribean.com ncl.com cruiseadvice.org
boardingfare.com	theflightdeal.com travelsort.com thepointsguy.com frugaltravelguy.com travelskills.org
destinationweddings.com	couples.com hardrockhotelpuntacana.com iberostar.com barcelo.com riu.org
fodors.com	escapehere.com frommers.com budgettravel.com grayline.com independenttraveler.org

V. CONCLUSION

We have described a method for ID-Free targeting of digital advertising that can effectively target users with no identifiers or user history of any kind. The use of URL embeddings to encode the URL data is central to our approach, providing a robust performance improvement compared to a naive feature representation. This performance improvement is increasingly dramatic for smaller training datasets, and in the extreme case URL embeddings enable the training of useful models with datasets too small to produce a viable model using the naive encoding. We showed that when predicting whether an ad inventory URL will lead to a conversion, ID-Free models using URL embeddings make predictions of similar quality regardless of how many times the URL was seen in training, including for URLs never seen in the training dataset. This suggests that for training datasets of all sizes, the URL embeddings approach is so effective because it allows the models to generalize by applying what was learned on one URL to improve the prediction on another URL.

The process of training embeddings is both time and dataintensive, and in order to capture recent trends and new websites, the URL embeddings must be trained on recent data. We shared our approach to building an efficient ID-Free model building pipeline using a neural network architecture that enables online learning of URL embeddings without a predefined dictionary. With continuous availability of up-to-date embeddings, training the ID-Free models themselves is quick, allowing us to train many of these models per day easily.

The URL embeddings approach to encoding URL data is rooted in learned user behavior; the embeddings capture the patterns underlying common sequences of user web browsing behavior. As with other applications of the word2vec approach, it is impressive how well behavioral patterns alone capture the meaning of each website, creating a rich map of the relationships between websites. (In fact, at Dstillery we maintain a version of the embeddings reduced to two dimensions using t-SNE that we internally refer to as the "Map of the Internet" and which has many uses [21].) The behavioral-only nature of this feature representation also presents a promising opportunity for future work because it means the representation could potentially be improved by incorporating other, entirely complementary sources of information. In particular, the behavioral approach completely disregards the text on the page, a likely source of additional information for pages with rich textual content.

The method for ID-Free targeting presented in this work can be used to reach the many internet users without third-party cookies enabled today. It is currently in use at Dstillery, using the cookie-based data sources described here. Additionally, this approach will be widely applicable in a future scenario where third-party cookies are entirely unavailable and all internet users carry no identifiers when visiting websites without first-party logins. We have described how both steps of this approach can be adapted in a straightforward way to use data from cookie-free sources. In this scenario, acquisition of labeled training data will likely come at a cost per example, so any practical implementation of a predictive targeting solution will require an approach that learns more from less data. Our URL embeddings approach accomplishes this by supplementing the information in the training dataset with an information-rich feature representation; the model borrows information from the URL embeddings. Without third-party cookies, we anticipate a substantial proportion of all digital advertising impressions will be delivered without any user identifiers. This approach to ID-Free targeting, which is practical to implement and enables precise predictive targeting, provides a way for advertisers to continue to deliver performance advertising campaigns at scale, regardless of the availability of third-party cookies.

References

[1] (2020) Browser market share United States of America. [Online]. Available: https://gs.statcounter.com/browser-market-share/all/united-states-of-america

[2] (2020, Jan.) Building a more private web: A path towards making third party cookies obsolete. [Online]. Available: https://blog.chromium.org/2020/01/building-more-private-web-path-towards.html

[3] M. H. Williams, R. Daniel-Weiner, A. G. White, and C. Reisz, "U.S. patent application publication No. 2020/0104738." September 2018.

[4] (2020) Glossary of terminology. [Online]. Available: https://www.iab.com/insights/glossary-of-terminology

[5] C. Perlich, B. Dalessandro, R. Hook, O. Stitelman, T. Raeder, and F. Provost, "Bid optimizing and inventory scoring in targeted online advertising," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY: ACM Press, August 12-16, 2012 2012, pp. 804–812.

[6] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, p. 417–441, 1933.

[7] W. Torgerson, "Multidimensional scaling I: Theory and method," *Psychometrika*, vol. 17, p. 401–419, 1952.

[8] J. A. L. Verleysen, *Nonlinear Dimensionality Reduction*. New York, NY, USA: Springer, 2007.

[9] Y. Bengio, A. C. Courville, and P. Vincent, "Unsupervised feature learning and deep learning: A review and new perspectives," *CoRR*, vol. abs/1206.5538, 2012. [Online]. Available: http://arxiv.org/abs/1206.5538

[10] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[11] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Neural and Information Processing System (NIPS)*, 2013. [Online]. Available: https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

[12] M. Grbovic, V. Radosavljevic, N. Djuric, N. Bhamidipati, J. Savla, V. Bhagwan, and D. Sharp, "E-commerce in your inbox: Product recommendations at scale," *CoRR*, vol. abs/1606.07154, 2016. [Online]. Available: http://arxiv.org/abs/1606.07154

[13] F. Vasile, E. Smirnova, and A. Conneau, "Meta-Prod2Vec - Product embeddings using side-information for recommendation," *CoRR*, vol. abs/1607.07326, 2016. [Online]. Available: http://arxiv.org/abs/1607.07326

[14] O. Barkan and N. Koenigstein, "Item2Vec: Neural item embedding for collaborative filtering," *CoRR*, vol. abs/1603.04259, 2016. [Online]. Available: http://arxiv.org/abs/1603.04259

[15] D. Svenstrup, J. M. Hansen, and O. Winther, "Hash embeddings for efficient word representations," *CoRR*, vol. abs/1709.03933, 2017. [Online]. Available: http://arxiv.org/abs/1709.03933

[16] P. Jenkins, "Clickgraph: Web page embedding using clickstream data for multitask learning," in *Companion Proceedings of the 2019 World Wide Web Conference (WWW '19 Companion)*. New York, NY: ACM Press, May 13 - 17, 2019 2019, pp. 37–41. [Online]. Available: https://doi.org/10.1145/3308560.3314198

[17] D. Gligorijevic, J. Gligorijevic, and A. Flores, "Time-aware prospective modeling of users for online display advertising," in *AdKDD 2019 workshop at the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[18] H. Yuan, Z. Yang, X. Chen, Y. Li, and W. Liu, "URL2Vec: URL modeling with character embeddings for fast and accurate phishing website detection," in *2018 IEEE Intl Conf on Parallel Distributed Processing with Applications, Ubiquitous Computing Communications, Big Data Cloud Computing, Social Computing Networking, Sustainable Computing Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, 2018, pp. 265–272.

[19] (2020) Conversion measurement API. [Online]. Available: https://github.com/WICG/conversion-measurement-api

[20] J. Firth, *A Synopsis of Linguistic Theory 1930-1955*, reprinted in F.R. Palmer ed. London: Longman: Oxford: Philological Society, 1968, in Studies in Linguistic Analysis, pp. 1-32.

[21] M. H. Williams. (2019) Artificial intelligence on human behavior. [Online]. Available: https://www.youtube.com/watch?v=LMIBM0KebGE