Manufacturing & Service Operations Management

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Optimization in Online Content Recommendation Services: Beyond Click-Through Rates

Omar Besbes, Yonatan Gur, Assaf Zeevi

M&SOM MANUFACTURING & SERVICE OPERATIONS MANAGEMENT
Vol. 18, No. 1, Winter 2016, pp. 15–33
ISSN 1523-4614 (print) | ISSN 1526-5498 (online)

informs
http://dx.doi.org/10.1287/msom.2015.0548

Optimization in Online Content Recommendation Services: Beyond Click-Through Rates

Omar Besbes
Columbia University, New York, New York 10027, ob2105@gsb.columbia.edu

Yonatan Gur
Stanford University, Stanford, California 94305, ygur@stanford.edu

Assaf Zeevi
Columbia University, New York, New York 10027, assaf@gsb.columbia.edu

A new class of online services allows Internet media sites to direct users from articles they are currently reading to other content they may be interested in. This process creates a "browsing path" along which there is potential for repeated interaction between the user and the provider, giving rise to a dynamic optimization problem. A key metric that often underlies this recommendation process is the click-through rate (CTR) of candidate articles. Whereas CTR is a measure of instantaneous click likelihood, we analyze the performance improvement that one may achieve by some lookahead that accounts for the potential future path of users. To that end, by using some data of user path history at major media sites, we introduce and derive a representation of content along two key dimensions: clickability, the likelihood to click to an article when it is recommended; and engageability, the likelihood to click from an article when it hosts a recommendation. We then propose a class of heuristics that leverage both clickability and engageability, and provide theoretical support for favoring such path-focused heuristics over myopic heuristics that focus only on clickability (no lookahead). We conduct a live pilot experiment that measures the performance of a practical proxy of our proposed class, when integrated into the operating system of a worldwide leading provider of content recommendations, allowing us to estimate the aggregate improvement in clicks per visit relative to the CTR-driven current practice. The documented improvement highlights the importance and the practicality of efficiently incorporating the future path of users in real time.

Keywords: online services; dynamic assortment selection; data-driven optimization; recommendation systems; content marketing; digital marketing; path data
History: Received: February 28, 2014; accepted: March 26, 2015. Published online in *Articles in Advance* September 23, 2015.

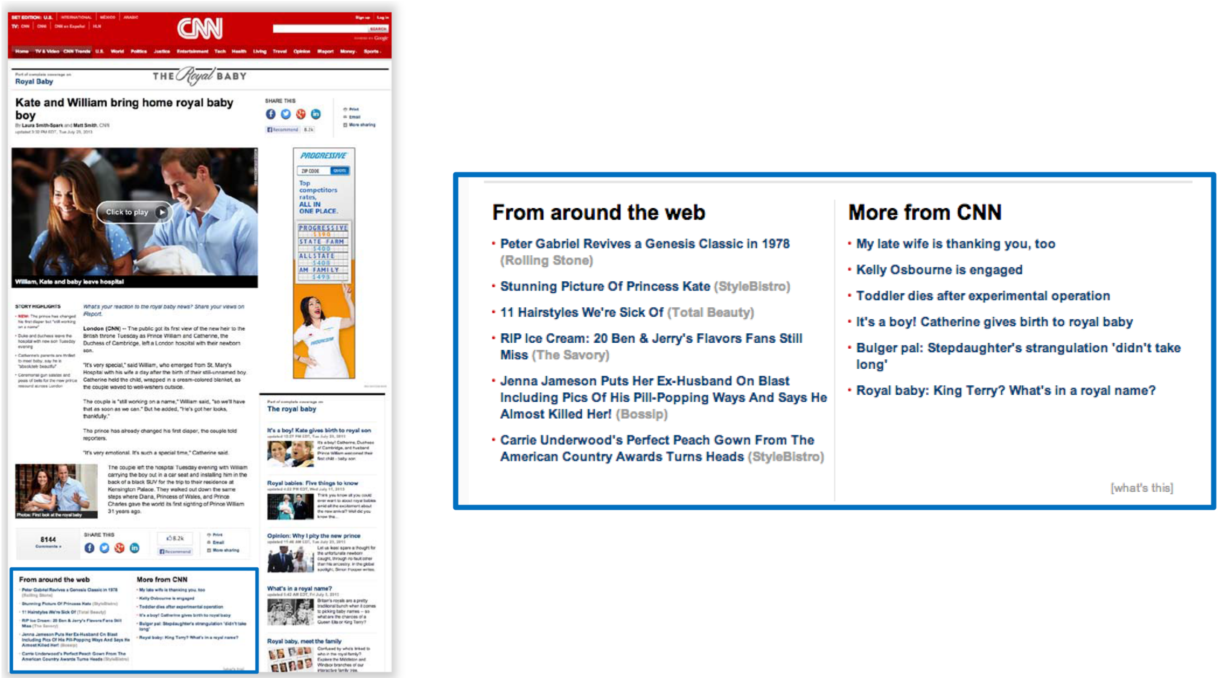
1. Introduction

1.1. Background and Motivation

Diversity and the sheer number of content sites on the World Wide Web has been increasing at an extraordinary rate. One of the great technological challenges, and a major achievement of search portals, is the ability to successfully navigate users through this complex forest of information to their desired content. However, search is just one route for users to seek content, and one that is mostly effective when users have a fairly good idea of what they are searching for. Recent years have witnessed the emergence of dynamically customized *content recommendations*, a new class of online services that complement search by allowing media sites to direct users from articles they are currently reading to other Web-based content they may be interested in consuming. This paper focuses on performance assessment and optimization for this new class of services.

1.1.1. Brief Overview of the Service. When a reader arrives to an online article (for example, by clicking a link placed at the publisher's front page), a customized recommendation is generated at the bottom of the article. (Figure 1 depicts such an example.) The recommendation typically contains 3 to 12 links that point readers to recommended articles, typically specifying the titles of these articles. By clicking on one of these links the reader is sent to the recommended article, at the bottom of which a new recommendation is often provided. In most cases these recommendations are generated by a service provider (not the media site). Recommended articles may be internal (organic), leading readers to other articles published in the host media site, or external (sponsored), in general leading readers to other publishers. Whereas internal recommendations are typically given as a service to the host publisher, external links are sponsored (either by the site on the receiving end of the recommendation, or by a third party that

Figure 1 (Color online) Online Content Recommendation



Notes. (Left) The position of the recommendation, at the bottom of a CNN article. (Right) The enlarged recommendation, containing links to recommended articles. The right side of this recommendation contains internal links to other articles on CNN's website, or CNN owned blogs. The left side of the recommendations contains external (sponsored) links to articles from other media sites.

promotes the content) based on a fee per click, which is split between the service provider and the publisher that hosts the recommendation. This simple revenue-sharing business model is predicated on the service's success in matching users with customized content that is relevant for them at the time of their visit. The problem of dynamically matching users and content lies at the heart of both the service provider's and online publishers' revenue maximization problems, and determines the value of the service to the publishers and their readers.

At a high level, the process of matching a reader with a bundle of recommended articles takes the following form. When a reader arrives to an article, a request for a recommendation is sent by the host publisher to the service provider. Such a request may include information on the host article as well as the reader. The service provider also holds a database of feasible articles, including information such as the topic classification, the publish date, or the click history. The available information is processed by several competing and complementary algorithms that analyze different aspects of it: the contextual connection between the host article and the candidates for recommendation, the reading behavior and patterns associated with articles and readers, and additional information such as general traffic trends in the content network. These inputs are combined to generate a customized content recommendation.

Various characteristics distinguish this process from more traditional product recommendation services (such as Amazon or Netflix). These features include the rate at which new "products" (articles) are added to the system (millions of articles added daily), and the typical short shelf life of many articles, which often lose relevancy in a matter of hours/days after publication. Moreover, the content recommendation service is mostly not subscription based, which limits the quality of the information that can be gathered on readers. Together, these introduce practical challenges that go beyond the traditional product recommendation problem (e.g., the need to base recommendations on dynamic and limited information on users' preferences and available products).

1.1.2. Main Questions. A key feature defining the content recommendation service is that it stimulates *ongoing user engagement* in each interaction. Whereas many online services are terminated after a single click, the content recommendation service is dynamic, as each successful recommendation leads to a new opportunity for interaction: following the first click, the user arrives to a new article, at the bottom of which a new recommendation is generated, and so on. Thus, content recommendations often serve as a navigation tool for readers, inducing a chain of articles. In such an environment, a central question is how to measure (and optimize) the performance of the recommendation service. This fundamental question is followed by practical ones: When constructing recommendations, what is the value of considering the potential future path of a reader as opposed to accounting only for her current position in the content network? How can one efficiently account for path factors in real time, while noting the computational and information availability limitations of the recommendation system?

1.2. Main Contributions

A key performance indicator that is widely used in practice to evaluate various articles as candidates for recommendation is the click-through rate (CTR): the number of times a link to an article was clicked, divided by the number of times that link was shown. The CTR metric is commonly adopted by online services that aim to generate a *single* click per user. In such a context, the focus is typically on integrating the click probability with the potential revenue generated by a click (for an overview, see Jansen and Mullen 2008, Feng et al. 2007). Following this common approach, content recommendation algorithms used in current practice are primarily designed to maximize the instantaneous CTR (or alternatively, the instantaneous revenue) of the generated recommendation. Although high CTR signals that the service is frequently being used, CTR also has an important limitation in the context of the dynamic service at hand: it measures the probability to click at the current step, but does not account for interactions that may come after the click, and in particular, *future* clicks along the potential visit path of the reader. The present paper aims to challenge this practice by demonstrating theoretically and practically (i) the value to be captured by accounting for the *future* path of users, and (ii) a relatively simple and practical approach for doing so. The study is based on a large data set of user path history at a few major media sites, as well as data from a live experiment. The present paper makes three main contributions: we model and demonstrate the value of a new dimension of content through predictive analytics, we develop heuristics to optimize dynamic recommendations based on this new dimension, and we validate the value of these heuristics theoretically, through simulation and also through a live experiment at a leading content recommendation provider. We next discuss each of these contributions in more detail.

1.2.1. From a Diagnostic to the Notion of Engageability. We first highlight the importance of accounting for the *future* path of users. Considering the path readers took through internal recommendations in our data set, we observe that between 15% and 30% of users' clicks on recommendations were generated *after* the first click, and that articles differ in how much they induce clicks when *hosting* recommendations. We then introduce and estimate a choice model, capturing key characteristics of the reader-recommendation interaction that impact click behavior. We calibrate this model based on a large data set, in a manner that accounts for the evolution of articles' relevancy over time. Based on our model, we develop a representation of content along two key dimensions: (1) *clickability*, the likelihood to *click to* an article when it is recommended; and (2) *engageability*, the likelihood to *click from* an article when it hosts a recommendation; the full meaning of this terminology will become apparent in what follows.

Our suggested representation of the "space of articles" is compact, but captures a key new dimension (engageability) and therefore significantly enriches the one adopted by current practice (which, as we explain later, may be interpreted as focusing mainly on clickability). This new space quantifies both the likelihood to click on each article, and the likelihood to continue using the service in the next step, if this article is indeed clicked and becomes the host of a recommendation.

1.2.2. From Engageability to Path-Focused Optimization. We formulate the optimal content recommendation problem and show that it is NP-hard in general and practically intractable. We then formalize a myopic heuristic whose objective is to maximize the probability to click on the current recommendation, and we establish that the gap between the performance of optimal recommendations and that of the myopic heuristic may be arbitrarily large. In that sense, theoretically, myopic recommendations may perform poorly. To close this performance gap, and based on the aforementioned content representation, we propose a path-focused heuristic that considers both the clickability and the engageability of candidate articles.

1.2.3. Validating the Proposed Heuristic. We demonstrate through theoretical lower bounds and simulation results that this heuristic yields performance that is close to the one of the optimal recommendations (which are computationally intractable for large instances). Armed with this backing, we move to test the proposed heuristics in the system operated by Outbrain, our industry collaborator and a worldwide leading provider of content recommendations. The challenge in testing our ideas is twofold: (i) the information available to process in real time is limited, and (ii) one needs to isolate the impact of our proposed ideas and heuristics relative to the ones adopted by current practice.

To address the first challenge, we study an "adjusted-myopic" implementation of one-step look-ahead recommendation policies, using proxies that are observed in real time throughout the recommendation process, without increasing the computational complexity of existing practice. To address the second challenge, we ensure that we do not simply add another algorithm to the list of existing algorithms used by the operator, because in such a case it would be impossible to fully isolate the impact. Rather, from the existing algorithms, all of which aim at maximizing CTR, we select one class of algorithms that do so by estimating the CTR of candidate articles based on click observations. This class is considered successful, and is responsible for generating roughly one-third of the links that are recommended in each assortment.

We then design and implement a controlled experiment where we modify this class on a test group in a manner that isolates the effect of accounting for the future path of readers through an observable proxy of the engageability parameter. The experiment measures the impact of said change on the performance of that class, finding an aggregate increase of 9.86% in clicks per visit. The improvement was statistically significant at the aggregate level even though the improvement measured at a more disaggregate level is not always statistically significant, due to a small sample effect. The results of the experiment imply that there is value to be captured by efficiently accounting for the future path of users in real time. Moreover, capturing such value does not necessarily require the development of new methods or technologies; it can be done by adjusting existing policies that have been designed to maximize myopic objectives.

1.3. Relation to Literature

Aspects of our study relate to literature streams in operations, information systems, and marketing.

1.3.1. Assortment Planning. At a technical level, the service provider aims to dynamically select a list of recommended links from a broader set of available candidates. Thus, although our main focus is on practical aspects of this problem, our formulation has some similarities to the assortment planning problem studied in the operations management literature under various settings and demand models (for a comprehensive review, see Kök et al. 2009). Caro and Gallien (2007) have studied the trade-off between exploration and exploitation when assortment selection is dynamic and demand is unknown; see also Rusmevichientong et al. (2010), Alptekinoglu et al. (2012), and Saure and Zeevi (2013). A paper that studies dynamic assortment selection in an environment that is closer to the one of content recommendations is that of Caro et al. (2014). In their formulation the attractiveness of products decays over time once they are introduced in the assortment, and products can be introduced only once. Yet, our formulation is quite different from theirs; a key distinction is that in Caro et al. (2014) one needs to decide in advance the timing at which each product is introduced in the assortment, whereas in the current study decisions depend on the realized path of the reader. In addition, in our formulation relevant features of articles change independently of the decision sequence, and articles can be recommended many times.

1.3.2. Performance in Online Services. The current paper also relates to studies that focus on performance metrics and heuristics in online services; see, e.g., Kumar et al. (2006), Mehta et al. (2007), Ghose and Yang (2009), Araman and Fridgeirsdottir (2011), and Najafi-Asadolahi and Fridgeirsdottir (2014) for methodological as well as empirical approaches in the context of online advertising. The main distinction of the current study from this line of work lies in the dynamic nature that governs the service at hand, and thus, calls for performance metrics (and appropriate heuristics) that account for the future path of users.

1.3.3. Path Data. Our study also relates to papers that study operational challenges and benefits of using path data to model and analyze consumers' behavior in retail, e-commerce, and advertising; for an overview, see Hui et al. (2009). In the context of online marketing and retail, which is the closest to ours, various papers study the use of large data sets to better model and predict user decisions. Although using such data typically involves methodological and technical challenges (see, e.g., Bucklin et al. 2002), it may lead to concrete benefits. For example, Bucklin and Sismeiro (2003) use information on the browsing path of Internet users to calibrate a model of click behavior, and Montgomery et al. (2004) demonstrate that path data can lead to a more accurate prediction of purchase decisions in an online bookseller; for an overview, see a survey by Bucklin and Sismeiro (2009). Our study differs from that line of work in three main aspects. First, whereas these papers typically study the consideration of path data to maximize *instantaneous* objectives (such as the probability that a consumer will buy a suggested book), the current paper studies the use of data to maximize a *path* objective (the number of clicks along the visit of a reader). Second, contrasting with the focus of the above literature stream on the *past* path of users for optimization purposes, the present paper accounts for the potential *future* path of users. Finally, compared with the above line of work, the current paper also pursues performance optimization while considering the information stream and computational limitations of existing practice. This allows us to go beyond explanatory or predictive analysis and to validate the proposed approach through a live experiment that measures added value captured in practice.

1.3.4. Recommender Systems. An active stream of literature has been studying recommender systems, focusing on tactical aspects that concern modeling and maintaining connections between users and products, as well as implementing practical algorithms based on these considerations; see the book by Ricci et al. (2011) and the survey by Adomavicius and Tuzhilin (2005) for an overview. A typical perspective that is taken in this rich line of work is that of the *consumer*, focusing on an objective of maximizing the probability to click on a recommendation. Common approaches that are used for this purpose are nearest neighbor methods, relevance feedback methods, probabilistic (nonparametric or Bayesian) learning methods, and linear classification methods; for an overview of such so-called content-based methods, see Pazzani and Billsus (2007). Another common class of algorithms focuses on collaborative filtering; see the survey by Su and Khoshgoftaar (2009) and references therein, as well as the industry report by Linden et al. (2003) on Amazon's item-based collaborative filtering approach. The current paper does not focus on these tactical questions, but rather on the higher-level principles that guide the design of such algorithms when one aims to account for the future path of a user. By doing so, to the best of our knowledge, the current paper is the first to focus on the perspective of the *recommender system* (the service provider) in a context of a multistep service in which the system's objective is not necessarily aligned with that of the consumer.

1.3.5. User Engagement. The notion of engageability is concretely defined in this paper as an element of a choice model, capturing the impact an article may have on the likelihood of readers to click on recommendations that originate from it. It is worthwhile to identify early on the difference between engageability, as it is defined here, and *user engagement*, a common notion that has been (potentially ambiguously) mentioned in the media as well as in several academic studies. These studies define user engagement rather loosely and in various manners (on the Internet as well as in other contexts) and study the way it relates to several measurable factors. Among others, these factors include the format of the media (see, e.g., Webster and Ahuja 2006) and interactivity (see, e.g., Haywood and Cairns 2006); see also O'Brien and Toms (2008) and O'Reilly (2009) for an overview, as well as Lee et al. (2014) and references therein for more recent work in the context of social media. Based on such elements, there have been a few attempts to define and measure user engagement; see, e.g., O'Brien and Toms (2010). Nevertheless, we are not aware of any study directly connecting some form of user engagement to performance metrics that account for the future path of users. Establishing potential relations between aspects of user engagement and a performance-related quantity such as engageability (as it is defined in the current paper), is indeed a natural avenue for future research.

1.3.6. Structure of the Paper. In §2 we introduce the notion of engageability, validate it based on a rich data set, and construct a compact representation of articles. In §3 we formulate the content recommendation problem, highlight computational challenges, and suggest a path-focused heuristic based on the aforementioned representation. Section 4 suggests an efficient implementation of path-focused recommendations and studies its impact relative to current practice in a controlled experiment. Section 5 includes some concluding remarks. Proofs and other auxiliary material appear in the online companion (available as supplemental material at http://dx.doi.org/10.1287/msom.2015.0548).

2. Introducing and Validating the Notion of Engageability

2.1. Preliminaries

2.1.1. Available (and Unavailable) Data. Our database includes access to over five billion internal recommendations that were shown on various media sites (all generated by Outbrain), including some anonymous information about articles, readers, recommendations, and observed click/no-click feedback. Every article that was visited or recommended in the database has a unique ID. Using this ID, one may trace the publish date of the article and its main topic (which is classified into 84 topic categories; representative ones include "news: politics," "sports: tennis," and "entertainment: celebrities"). Each event in which a reader arrives to an article is documented by a unique recommendation ID, reader ID, and the ID of the article that hosts the recommendation (at its bottom). Using the recommendation ID, one may trace

- the ID list of articles that appear in the internal recommendation (ordered by position),
- the time at which the recommendation was created,
- the time that was spent by the reader on the host article before leaving it (for some media sites), and
- the ID of the recommendation on which the reader clicked to arrive to the current page (if the reader arrived by clicking on an internal recommendation).

Our data set does not include information about additional article features such as the number of paragraphs, appearance of figures/pictures, or links presented in the text. The data set neither includes the sponsored links that were shown nor clicks on them. In addition, we note that because of technical reasons (for example, this recommendation service is not subscription based), information on the preferences of readers is typically of limited detail and reliability and is not included in our data set.

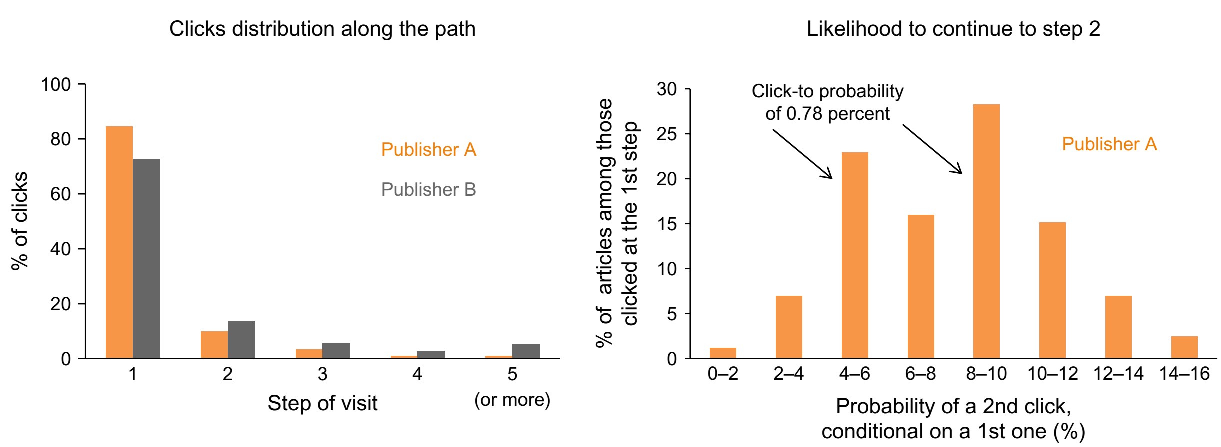
2.1.2. Preliminary Empirical Insights: Click Distribution and Host Effect. We construct the visit paths of readers from an arrival to some host article through a sequence of clicks (if such clicks take place) on internal links. The distribution of clicks along visit steps in two representative media sites is shown on the left part of Figure 2. We observe that a significant portion of the service is provided *along* the visit path: between 15% and 30% of clicks were generated *after* the first click; this range is representative of other publishers as well. The portion of clicks along the path is significant even though the future path is not taken into account by the first recommendation; one can only expect this portion of the business volume to grow if recommendations account for the future path of readers.

Next, we focus on the set of all the links that were *clicked on in the first step*, and more precisely, the articles these links lead to. In the right part of Figure 2, for all these articles, we construct a histogram based on the probability of a second click (conditional on the first click); the x axis has bins of conditional probabilities and the y axis represents the frequency (percentage of all the articles that were clicked on in the first step) of such occurrences. For example, roughly 28% of articles clicked on in the first step are characterized by a conditional click-from likelihood between 8% and 10%, and only 7% of such articles are characterized by a conditional click-from likelihood between 12% and 14%. We observe significant variation in this conditional probability in the space of articles. This points out that the link that is selected by a reader at the first step clearly affects clicks generated along the future path of the reader. Moreover, we note that the average CTR *to* articles along different bins may be similar; for example, the mean probability to click to articles at both the third and the fifth bins is ~0.78%.

As mentioned in §1, the key performance indicator that is commonly used in practice to evaluate each article as a candidate for recommendation is its estimated CTR, namely, the probability that a link that leads to it will be clicked, if recommended by the service provider. As the CTR does not account for *future* clicks along the potential visit path of the reader, the observations above imply that CTR-driven recommendations might be "leaving clicks on the table." Intuitively, while content recommendations aim to match readers with attractive links, the host article describes the environment in which the matching takes place, and therefore potentially affects the click behavior. These observations lead to the following questions. First, are readers indeed more likely to continue the service after visiting certain articles? Second, if so, to what extent can one identify such articles in real time and adjust the CTR indicator accordingly, in a manner that is compatible with the practical limitations of the service? Third, how much additional value can be captured in this manner?

In §2.2 we introduce the notion of engageability as an element of a choice model; engageability quantitatively captures the impact of the *host* article on the click behavior of readers. Our model leads to a compact representation of articles in a content space, which is introduced and discussed in §2.3. We validate that engageability is a significant click driver in §2.4, through out-of-sample testing.

Figure 2 **(Color online) Aggregate Analysis of Clicks Along the Visit Path**



Notes. (Left) The distribution of clicks along visit steps in two representative media sites (A and B). (Right) The distribution of the probability to click again among articles to which readers arrived after their first click (in media site A).

2.2. Choice Model

As the recommendation service aims to suggest attractive links, a key element on which recommendation algorithms focus is the ID of *recommended* articles. Other considerable elements include the position of links within recommended assortments, the topics of candidate articles, and the level of familiarity the reader has with the service. Our model accounts for these elements, as well as the ID of the article that *hosts* the recommendation, an element that has been overlooked so far.

As discussed in §2.1, our data set does not include access to some of the factors that may affect the likelihood to click on internal recommendations. Taking this limitation in mind, the purpose of our model will not be to entirely quantify the environment in which readers of media sites make clicking decisions, but rather to demonstrate that the host article significantly affects such decisions, while accounting for other reasonable click drivers that are considered in practice. In particular, the model is selected while keeping in mind important practical considerations. First, we aim at a parsimonious model to avoid overfitting. Second, as the relevancy of content changes over time, we will need to refresh estimates of key parameters. Finally, later on (in §4) we will aim at implementing practical algorithms that are based on this model (in particular, these will account for the host effect in addition to the link effect) via proxies that are observable in *real time*, to measure in a controlled experiment the value captured by accounting for the host effect (relative to current practice where it is not considered).

We consider a multinomial logit model to capture the impact of the following elements on clicking behavior. (These elements will be discussed after the model is set.) The dummy variable β_x captures the effect of a host article x. The dummy variable γ_y captures the effect of a recommended article y. The variable $\mu_{x,y}$ captures the interaction effect of (x, y). The variable θ_u captures the effect of user familiarity with the recommendation service. Given an article y that appears in a recommended assortment A, we denote the position of article y in the assortment by $p\colon (y, A) \to \{0, \dots, 5\}$, 0 being the highest position. Then, the variable $\lambda_{p(y,A)}$ captures the effect of the position of y in A.

Given a reader type $u \in \mathcal{U}$, a recommended assortment A, and a host article x, we denote the probability to click on a link to article y by $\mathbb{P}_{u,x,y}(A)$, and define

$$\mathbb{P}_{u,x,y}(A) = \begin{cases} \dfrac{\phi_{u,x,y}(A)}{1+\sum_{y'\in A}\phi_{u,x,y'}(A)} & \\ \quad \text{if } y \text{ appears in } A, & \\ 0 \quad \text{otherwise.} & \end{cases} \tag{1}$$

Whenever y appears in A, we define

$$\phi_{u,x,y}(A) = \exp\{\alpha + \beta_x + \gamma_y + \mu_{x,y} + \theta_u + \lambda_{p(y,A)}\}. \tag{2}$$

2.2.1. Engageability (Host Effect) and Clickability (Link Effect). The parameter β is associated with the likelihood to click from an article whenever it hosts a recommendation, and it is driven by the actual content of the article. We refer to β as the engageability of an article. Under our model, the engageability of an article may account for two potentially different effects. The first one is "homogeneous" with respect to all recommended links that may be placed at its bottom. The intuition underlying this parameter is that when an article is well written, interesting, relevant, and of reasonable length, the reader is more likely to read through it, arrive to its bottom, and click on a recommendation. On the other hand, when content is poor or irrelevant, a reader is more likely to terminate the service rather than scroll down through the article, and therefore she is less likely to see the recommendation and click on it. Engageability of an article may also capture in an aggregate fashion an effect that is "heterogeneous" with respect to different links.[1] The engageability of an article may change with time, along with its drivers, which may include the relevancy of its content.

The parameter γ is associated with the likelihood to *click to* an article whenever it is recommended, and is driven by the title of the article (which is typically the only information given on the link itself). We refer to γ as the clickability of an article. The clickability of articles may change with time as well.

2.2.2. Control Parameters. The parameter θ_u is a dummy variable that captures the effect of the user type, and in particular, of experienced users. We differentiate between two types of users: experienced users (that have clicked on an Outbrain recommendation before) and inexperienced users. Thus, we have $\theta_u \in \{\theta_{\text{exp}}, \theta_{\text{inexp}}\}$ for each user, where we normalize by setting $\theta_{\text{inexp}} = 0$ (treating unexperienced users as a benchmark) and estimate θ_{exp} from the data. Experienced readers were defined as ones that clicked on a recommendation during an initial period of 10 days. Then, during the 30 days over which the model was estimated, we update reader types from "inexperienced" to "experienced" once they click on an internal recommendation. The main motivation for distinguishing between experienced and inexperienced users stems from the aggregate data analysis

[1] Theoretically, such connections between articles may potentially be separated from the first, "homogeneous" engageability effect, by using a more complex description of contextual relation between articles/topics (an example of such a description is given in Appendix A.2 in the online companion). In this study, we do not aim to separate between homogeneous and heterogeneous effects, but rather focus on the value of accounting for the future path of users through the engageability parameter.

Table 1 Experienced vs. Inexperienced

User type	Population share (%)	Visits share (%)	Clicks per visit
Experienced	8.2	16.9	$2.3 \cdot \nu_{\text{inexp}}$
Inexperienced	91.8	83.1	ν_{inexp}

Notes. This table summarizes the difference between inexperienced users and experienced users, as was observed along the 30 days that followed the initial period; because of a nondisclosure agreement, the value of ν_{inexp} (clicks per visits for inexperienced readers) is not disclosed.

summarized in Table 1, indicating that although most of the users are inexperienced, on average, an experienced user visits the media site more than twice the times an inexperienced one visits it, and clicks more than twice the times (per visit) relative to an inexperienced one. Alternative methods of segmenting users are discussed in Appendix A.2 (see the online companion).

To formulate the contextual relation between the host article and a recommended one, we use $\mu_{x,y}$, a dummy variable that flags cases in which the recommended article relates contextually to the host article. We define articles that relate to each other as ones that share the same topic, using a classification to 84 topic categories. Thus, we have $\mu_{x,y} \in \{\mu_{\text{related}}, \mu_{\text{unrelated}}\}$ for each pair (x, y), where we normalize by letting $\mu_{\text{unrelated}} = 0$ (treating unrelated recommendations as a benchmark) and estimate μ_{related} from the data. Alternative formulations of the contextual relation are discussed in Appendix A.2 in the online companion.

The position effect is captured by the variables $\lambda_p \in \{\lambda_0, \ldots, \lambda_5\}$. We set $\lambda_0 = 0$ (treating the highest position as a benchmark), and we estimate the other five parameters to measure the effect of lower positions.

2.2.3. Estimation. We estimate the model using data collected by Outbrain in early 2013. The data set includes 40 consecutive days of internal recommendations presented on articles of one media site. We differentiated experienced readers from inexperienced ones using a training set that included the first 10 days, and we estimated the model using data from the remaining 30 days. Since clickability and engageability of articles may be time varying, the model was estimated independently over 360 batches of two hours. In each such batch, approximately 500 different articles hosted recommendations, and a similar number of articles were recommended (approximately 90% of host articles were also recommended in the same batch, and vice versa). Along each batch approximately 1,000 parameters where estimated (including the control parameters). Estimation in each batch was based on approximately 2M recommendations (of six links each) and 100,000 clicks. The model was estimated by applying a Newton step method to maximize the log likelihood of the model. The estimation

Table 2 Estimation of Control Parameters: Scaled Results

Effect	Parameter	Estimate	Standard error
Intercept	α	-1	$8.8 \cdot 10^{-7}$
User	θ_{exp}	0.254	$3.82 \cdot 10^{-4}$
Contextual relation	μ_{related}	$-2.25 \cdot 10^{-2}$	$4.49 \cdot 10^{-3}$
Position	λ_1	-0.247	$1.10 \cdot 10^{-4}$
	λ_2	-0.384	$3.15 \cdot 10^{-6}$
	λ_3	-0.456	$4.27 \cdot 10^{-6}$
	λ_4	-0.512	$4.72 \cdot 10^{-6}$
	λ_5	-0.515	$4.72 \cdot 10^{-6}$

Notes. The shifted control parameters' estimates and standard errors for one media site, in the first estimation batch; because of a nondisclosure agreement, all the values shown in the table were obtained by dividing original estimates and standard errors by the absolute value of α. All estimates are at significance level $p < 0.01$.

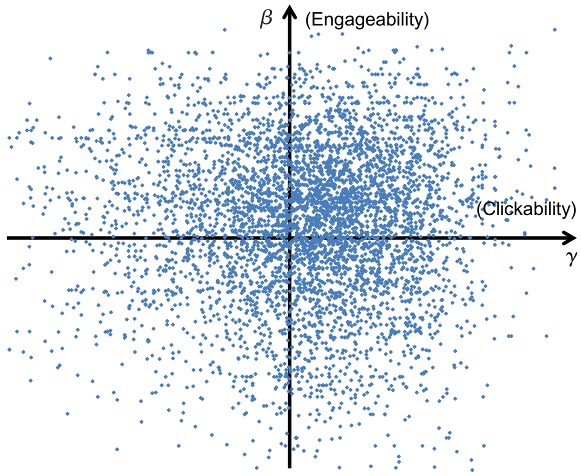
results in all 360 estimation batches were very similar in their statistical patterns. Scaled values of the control parameters' estimates from the first estimation batch are presented in Table 2.

The estimate of θ_{exp} quantifies the effect of user experience on the likelihood to click. It supports the statistics presented in Table 1: users that are familiar with the recommendation service tend to use it more often than inexperienced users. The estimate of μ_{related} quantifies the effect of the contextual relation between the host and recommended articles. Interestingly, it suggests that on average, users tend to click less when the recommended article directly relates to the article they just finished reading, relative to cases in which such direct relation does not exist. The estimates of $\lambda_1, \ldots, \lambda_5$ quantify the "cost of a lower position" in the recommendation box, relative to the highest position. As one would expect, the lower the position of the link is, the lower the likelihood of a reader to click on that link. Next, we discuss in more detail the estimates of the engageability (β) and clickability (γ) parameters.

2.3. Content Representation

To summarize the estimates of the clickability and engageability parameters, one may visualize the space of articles using a scatter plot of all articles (identified by their ID and the time window) along β and γ. The resulting plot is depicted in Figure 3. The plot leads to a representation of the articles available in a "content space" with two dimensions that have a "physical" meaning when examining articles as candidates for recommendation: it captures not only the likelihood to click on an article when it is recommended, but also the likelihood of readers to continue using the service if this article is indeed clicked, and thus hosts the recommendation in the following step. Figure 3 implies that engageability and clickability (and intuitively, the attractiveness of the title, and the actual content) are fundamentally different click drivers. In fact, the correlation between the two

Figure 3 (Color online) Articles in the Content Space

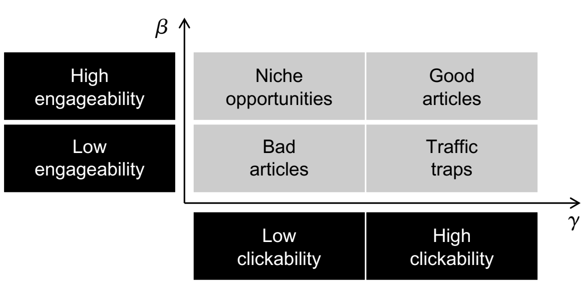


Notes. Every article is positioned in the content space according to its two estimates, β (engageability) and γ (clickability). The 5,012 articles that appear in the plot have at least 500 appearances as a host and as a recommended link during the estimation segment. The estimated values in the figure range from -3 to 3 along both axes.

is very low (0.03). Compared with current practice that focuses on clickability/CTR, this representation allows one to differentiate between articles that have similar clickability. We will demonstrate the value of the β-dimension in §§5 and 6.

2.3.1. The Content Matrix. The content space may be used as a visualization tool to depict and understand representative classes of articles (illustrated in Figure 4). In that spirit, one may develop a high level 2×2 view of content. We refer to articles with high clickability and high engageability as "good articles": readers are likely to click on them and are also likely to click from them and continue the service. On the other hand, the class of "bad articles" is characterized by low clickability and low engageability. We refer to articles with high clickability but low engageability as "traffic traps": these articles attract a lot of readers, but these readers tend to terminate the service upon arrival. Unlike bad articles, that are naturally screened out of the recommendation pool because of their low clickability, traffic traps represent a threat to a clickability-focused system: since they are being clicked on, they keep on being recommended despite potentially poor content that harms the service performance in the short term (along the path of readers) and in the long term (decreasing the value of the service for readers). We refer to articles with low clickability and high engageability as "niche opportunities": readers do not tend to click on these articles, but those who click on them tend to continue the service afterward. We finally note that the introduced content space may be useful for descriptive as well as prescriptive purposes; a few practical applications of this formulation are discussed in §5.

Figure 4 The Content Matrix



2.4. Validating the Notion of Engageability

2.4.1. In-Sample Testing. In each estimation batch, we perform a likelihood ratio test with the null hypothesis being that click observations follow a link-focused model. This model follows the structure in (1) with $\phi_{u,x,y}(A)$ defined by $\phi^{lf}_{u,x,y}(A) = \exp\{\alpha + \gamma_y + \mu_{x,y} + \theta_u + \lambda_{p(y,A)}\}$, where the control parameters $\mu_{x,y}$, θ_u, and $\lambda_{p(y,A)}$ are defined as in §2.2. In the link-focused model, engageability is always constrained to be zero. For each two-hour batch we measure

$$R = -2\ln\left[\frac{\text{likelihood for link-focused model}}{\text{likelihood for nominal model}}\right],$$

which is approximately distributed according to a chi-squared distribution with the number of degrees of freedom being the number of engageability parameters (which is the number of articles, roughly 500 in each batch). The obtained p-values of the various batches were all smaller than 0.05. We next turn to establish a *stronger* notion of validation through out-of-sample testing and predictive analysis.

2.4.2. Out-of-Sample Testing. We use each set of estimates generated in a batch to predict click/no-click outcomes in the following batch. We test the ability to predict a click on the whole recommendation (rather than on a specific link) in impressions where all the recommended articles were estimated in the previous batch. The procedure of testing the predictive power of the model is as follows.

Out-of-sample testing procedure. Input: $\delta \in [0, 1]$

1. For each two-hour data batch $1 \le j \le 359$:

(a) Estimate model parameters according to §2.2, using the data set of segment j.

(b) Test predictive power in the *following* two-hour batch: for any recommended assortment A in batch $j+1$, calculate the assortment click probability according to the estimates of batch j

$$\mathbb{P}_{u,x}(A) = \sum_{y \in A} \mathbb{P}_{u,x,y}(A),$$

where $\mathbb{P}_{u,x,y}(A)$ is defined according to (1) and $\phi_{u,x,y}(A)$ according to (2). Then, classify

$$C_\delta(A) = \begin{cases} 1 & \text{if } \mathbb{P}_{u,x}(A) \ge \delta, \\ 0 & \text{if } \mathbb{P}_{u,x}(A) < \delta. \end{cases}$$

2. Using the actual click/no-click reader's feedback, calculate throughout the entire data horizon:

(a) the false positive rate:

$$R_\delta^{fp} = \frac{\#\{A\colon \text{not clicked}, C_\delta(A)=1\}}{\#\{A\colon \text{not clicked}\}},$$

(b) the true positive rate:

$$R_\delta^{tp} = \frac{\#\{A\colon \text{clicked}, C_\delta(A)=1\}}{\#\{A\colon \text{clicked}\}}.$$

2.4.3. Benchmarks. We compare the predictive power of the model to one of the following benchmarks.

1. *Random click probabilities.* A random classifier in which $\mathbb{P}_{u,x}(A)$ is an independent uniform distribution over $[0,1]$.

2. *Link-focused model.* We estimated the model in (1) with $\phi_{u,x,y}(A)$ defined by

$$\phi_{u,x,y}^{lf}(A) = \exp\{\alpha + \gamma_y + \mu_{x,y} + \theta_u + \lambda_{p(y,A)}\}, \qquad (3)$$

where the control parameters $\mu_{x,y}$, θ_u, and $\lambda_{p(y,A)}$ are defined as in §2.2.

3. *Host-focused model.* We estimated the model in (1) with $\phi_{u,x,y}(A)$ defined by

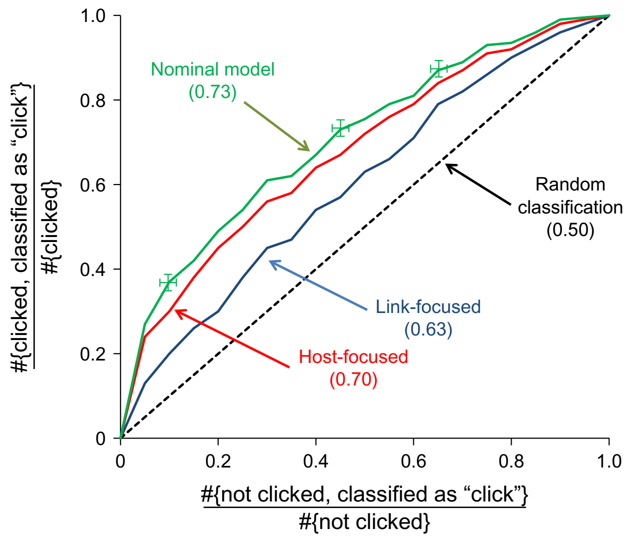
$$\phi_{u,x,y}^{hf}(A) = \phi_{u,x}^{hf}(A) = \exp\{\alpha + \beta_x + \theta_u\}, \qquad (4)$$

where θ_u is defined as in §2.2.

We repeat the above testing procedure for our model as well as for the three benchmarks for various values of $\delta \in [0,1]$. To put the predictive power of our model into perspective, we compare it with the three benchmarks in the receiver operating characteristic (ROC) space, in which the true positive rate is presented as a function of the false positive rate, for a spectrum of δ values in $[0,1]$.

2.4.4. Predictive Power. Figure 5 details the ROC curve of our nominal model, compared to that of the link-focused and the host-focused benchmarks, as well as the random classification diagonal. The large gap between the ROC curves of the nominal and the link-focused models suggests that indeed, host engageability has a key impact on clicking behavior of readers. Thus, developing heuristics and practical recommendation algorithms that leverage the engageability feature might lead to improved performance relative to myopic methods that ignore engageability. The importance of the host effect is also suggested by the relatively small gap between the ROC curves of the nominal and the host-focused models. Indeed, the predictive power of a model that accounts only for the host effect and the reader experience, significantly exceeded that of the *richer* link-focused model, that does not account for host engageability. We note that contrasting the predictive power for experienced versus inexperienced readers showed

Figure 5 (Color online) Quantifying Predictive Power in the ROC Space



Notes. The plot shows the ROC curve generated by our model, together with the link-focused model, the host-focused model, and the random classification diagonal. The area under each curve (AUC) appears in parentheses. All standard errors (with respect to both axes) are smaller than 0.02; three illustrative examples are depicted on the "nominal model" curve.

similar results; the two obtained ROC curves were statistically indistinguishable.

2.4.5. Robustness of the Model. The model that was estimated in §2.2 considers basic inputs in our data, in a manner that allows one to demonstrate the predictive power that is captured when accounting for the engageability of the host article. In Appendix A.2 in the online companion we consider two variants of this model that use additional information. Repeating the predictive analysis that was performed here, we demonstrate that enriching our model by considering more information does not necessarily increase the established predictive power. This might be due to overfitting the data because of the larger number of parameters.

3. The Content Recommendation Problem

In §2 we modeled the structure of click probabilities based on some characteristics of the user, the host article, and the recommended set of links. In this section we formulate the provider's content recommendation problem (CRP) of planning a dynamic schedule of recommendations to maximize the value generated by clicks along the *entire visit* of a reader. Note that the CRP needs to be solved each time a reader arrives to some initial (landing) article. The reader can terminate the service at any stage, by leaving the current article she reads through any mean other than clicking on a content recommendation (e.g., closing the window, clicking on a link that is not a content recommendation, or typing a new URL). The CRP is formulated in §3.1 and analyzed in §3.2 considering a general click probability structure; the specific model studied in the previous section will be leveraged in §3.3 in the context of the CRP.

3.1. General Formulation of the CRP

The CRP is formalized as follows. Let $1, \ldots, T$ be the horizon of the CRP throughout a visit of a single reader. We denote by l the number of links that are introduced in each recommendation. We denote by x_{t-1} the article that hosts the recommendation at epoch t (for example, x_0 denotes the article that hosts the recommendation at $t=1$; the article from which the reader starts her journey). We denote by $\mathscr{X}_t$ the set of articles that are available to be recommended at epoch t. The initial set of available articles is $\mathscr{X}_0$, and we assume this set is updated at each epoch by $\mathscr{X}_t = \mathscr{X}_{t-1}\backslash\{x_{t-1}\}$ (for example, at $t=1$ all the articles that are initially available can be recommended, except for x_0, that hosts the first recommendation). We assume $T \leq |\mathscr{X}_0| - l$, meaning that there are always enough available articles.

We denote by $\mathscr{U}$ the set of reader (user) types. We denote by u_0 the initial reader type. This type may account for various reader's characteristics (e.g., geographical location), as well as her reading and clicking history. We assume the reader type to be updated at each epoch according to $u_t = f_t(u_{t-1}, x_{t-1})$. This update may account for articles being read, as well as for epoch-dependent effects such as fatigue (for example, u_1, the type at $t=1$, may account for the initial type u_0, the initial article x_0, and the fact that the reader sees the recommendation after having read one article). Although we do not specify here a concrete structure for $f_t(\cdot, \cdot)$, we note that a special case of this update rule was used in §2.2, where $u_0 \in \{u_{\text{exp}}, u_{\text{inexp}}\}$ was set based on a training set; whenever $u_0 = u_{\text{exp}}$ one had $u_t = u_0$ for all t, and whenever $u_0 = u_{\text{inexp}}$ one had $u_1 = u_0$ and $u_t = u_{\text{exp}}$ for all $t \geq 2$.

A recommendation assortment is an ordered list of l links to articles that are available for recommendation. We denote by $\mathscr{A}^l(\mathscr{X}_t)$ the set of all possible assortments at epoch t (all ordered lists of items from $\mathscr{X}_t$). At each epoch $t = 1, \ldots, T$ the recommendation provider selects a recommendation assortment $A_t \in \mathscr{A}^l(\mathscr{X}_t)$ to present the reader with. For a given user type u, a host article x and a recommendation assortment A, we denote by $\mathbb{P}_{u,x,y}(A)$ the click probability to any article $y \in A$.[2] Finally, we denote by $w(x)$ the value (for the service provider) generated when a reader clicks on a link to article x. We assume that the value of clicking on each article is known; it may represent actual revenue (in sponsored links) or tangible value (in organic links that drive publishers to partner with the provider).

The structure described above assumes Markovian dynamics, which are considered in the following. Given an initial reader type u_0, an initial set of articles $\mathscr{X}_0$, and a host article x_0, the CRP of maximizing the value generated by clicks throughout the visit is defined by the following Bellman equations:

$$V_t^*(u_t, \mathscr{X}_t, x_{t-1}) = \max_{A \in \mathscr{A}^l(\mathscr{X}_t)} \left\{ \sum_{x_t \in A} \mathbb{P}_{u_t, x_{t-1}, x_t}(A) \cdot \left(w(x_t) + V_{t+1}^*(u_{t+1}, \mathscr{X}_{t+1}, x_t) \right) \right\}, \quad (5)$$

for $t = 1, \ldots, T$, where $V_{T+1}^*(u_{T+1}, \mathscr{X}_{T+1}, x_T) = 0$ for all u_{T+1}, $\mathscr{X}_{T+1}$, and x_T.

3.2. Complexity and Myopic Policy Performance

Since the CRP accounts for the future path of the reader, the computational complexity that is associated with finding its optimal solution increases rapidly when the set of available articles gets large.

Proposition 1 (Hardness of CRP). *The content recommendation problem defined by* (5) *is NP-hard.*

In the proof of the proposition, we establish that the Hamiltonian path problem, a known NP-hard problem (see Gary and Johnson 1979; see also Uehara and Uno 2005 and Karger et al. 1997 for various relaxations and approximation approaches to this problem), can be reduced to a special case of the CRP, and therefore, even when click probabilities from hosting articles to recommended articles are known for each arriving reader, the CRP is NP-hard. Given the large number of available articles and the high volume of reader arrivals, Proposition 1 implies that it is impractical for the service provider to look for an optimal solution for the CRP for each arriving reader. This motivates the introduction of customized recommendation algorithms that, although lacking performance guarantees for arbitrary problem instances, perform well empirically given the special structure of the problem at hand.

3.2.1. The Myopic Heuristic. One class of such algorithms is CTR driven, with the objective of recommending at each epoch t (until the reader terminates the service) an assortment of links that maximizes the instantaneous performance, without accounting for the future path of the reader. We refer

[2] With some abuse of notation we sometimes denote assortments as sets of links and note that the probability to click on a link that belongs to an assortment depends on all the links in the assortment as well as on the way they are ordered. Therefore, $y \in A$ and $y \in A'$ do *not* imply $\mathbb{P}_{u,x,y}(A) = \mathbb{P}_{u,x,y}(A')$. Similarly, A and A' containing the same articles does *not* imply $\sum_{y \in A} \mathbb{P}_{u,x,y}(A) = \sum_{y \in A'} \mathbb{P}_{u,x,y}(A')$, as articles may be ordered differently in each assortment.

to this approach as the *myopic* content recommendation problem (MCRP), and formally define it by

$$V_t^m(u_t, \mathscr{X}_t, x_{t-1}) = \sum_{x_t \in A_t^m} \mathbb{P}_{u_t, x_{t-1}, x_t}(A_t^m)(w(x_t) + V_t^m(u_{t+1}, \mathscr{X}_{t+1}, x_t)); \quad t = 1, \ldots, T, \tag{6}$$

where

$$A_t^m \in \arg\max_{A \in \mathscr{A}^l(\mathscr{X}_t)} \left\{ \sum_{x_t \in A} \mathbb{P}_{u_t, x_{t-1}, x_t}(A) w(x_t) \right\}; \quad t = 1, \ldots, T,$$

and where $V_{T+1}^m(u_{T+1}, \mathscr{X}_{T+1}, x_T) = 0$ for all u_{T+1}, $\mathscr{X}_{T+1}$, and x_T. The MCRP can be solved at each epoch separately, based on the current host article, reader type, and set of available articles.

3.2.2. The Suboptimality of Myopic Recommendations. Although recommending articles myopically is a practical approach, simple problem instances reveal that myopic recommendations may generate poor performance compared to the optimal schedule of recommendations. In one such instance that is depicted in Figure 6, myopic recommendations generate only two thirds of the expected clicks generated by optimal recommendations.

Figure 6 depicts an instance of significant gap between the performance of myopic recommendations and that of optimal recommendations. In fact, this performance gap may be arbitrarily large in general. To demonstrate that, we consider a special case of the problem where $l = 1$ (single link recommendations), and the user type u is fixed over time. Then, we can denote the probability to click from article x to a recommended article y simply by $\mathbb{P}_{u,x}(y)$. Let $\mathscr{G}_{T+1}$ denote the class of all networks induced by sets $\mathscr{X}_0$ of $T+1$ articles and sets of transition probabilities $\mathscr{P}_0^u = \{\mathbb{P}_{u,x}(y) \mid x \in \mathscr{X}_0, y \in \mathscr{X}_0\}$. The following result shows that myopic recommendations may yield arbitrarily poor performance compared to optimal recommendations when the size of the network and the problem horizon grow.

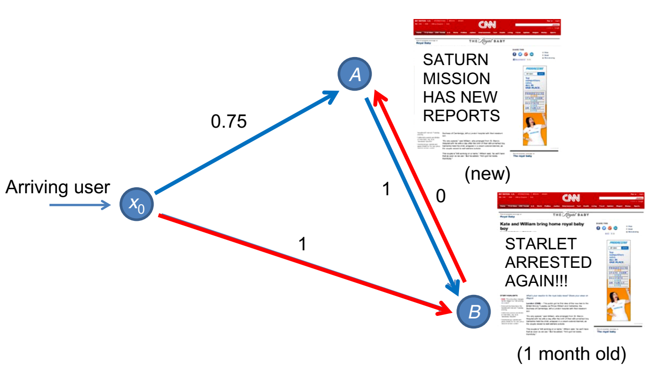
Proposition 2 (Performance Gap for MCRP). *Let $\mathscr{X}_0$ be an initial set of articles, let $\mathscr{X}_1$ be a set of articles available for recommendation from an initial article $x_0 \in \mathscr{X}_0$, and let u be a user type, as defined in §3.1. Let $\mathscr{P}_0^u$ be the set of transition probabilities induced by $\mathscr{X}_0$ and u, as defined in §3.2. Then,*

$$\inf_{\substack{(\mathscr{X}_0, \mathscr{P}_0^u) \in \mathscr{G}_{T+1}, \\ x_0 \in \mathscr{X}_0, u \in \mathscr{U}}} \left\{ \frac{V_1^m(u, \mathscr{X}_1, x_0)}{V_1^*(u, \mathscr{X}_1, x_0)} \right\} \to 0, \quad \textit{as } T \to \infty.$$

We note that the type u is fully specified by the set $\mathscr{P}_0^u$; with some abuse of notation, we omit the explicit dependence of the value function on the transition probabilities. The proof of the proposition is constructive; we explicitly exhibit a set of available articles with appropriate transition probabilities such that when the number of articles grows large the performance of the myopic heuristic, defined by (6), becomes arbitrarily poor compared to the one of optimal recommendations, defined by (5).

To summarize, we have established that although finding an optimal recommendation schedule may not be practical, theoretically, algorithms that follow a myopic heuristic may significantly underperform relative to optimal recommendations. These results add context to the empirical insights discussed in §2.1, because these insights suggest that a gap between the performance of myopic recommendations and that of the optimal recommendations may appear not only in theory, but in real content networks as well.

Figure 6 (Color online) Suboptimality of Myopic Recommendations



Notes. A content recommendation instance, with $l = 1$, $T = 2$, $\mathscr{X}_0 = \{x_0, A, B\}$, and uniform click values. x_0 is the initial host. The click probabilities illustrate a scenario where article B has attractive title but irrelevant content that drives users to terminate the service. Myopic schedule first recommends B and then A, generating a single click. An optimal schedule first recommends A and then B, generating $0.75 + 0.75 \times 1 = 1.5$ expected clicks.

3.3. Leveraging Engageability: Path-Focused Approach

Having established in §2.4 the importance of engageability in predicting click behavior, we next turn to study the value one may capture by accounting for engageability for the purpose of optimizing recommendations. Adopting the model studied in §2, we suggest a heuristic that accounts for one step forward in a reader's path when creating each recommendation, and we analyze its performance.

3.3.1. One-Step Look-Ahead Heuristic. We suggest recommending articles with the objective of solving the *one-step look-ahead* recommendation problem, defined by the following set of equations:

$$V_t^{\text{one}}(u_t, \mathscr{X}_t, x_{t-1}) = \sum_{x_t \in A_t^{\text{one}}} \mathbb{P}_{u_t, x_{t-1}, x_t}(A_t^{\text{one}}) \big(w(x_t) + V_{t+1}^{\text{one}}(u_{t+1}, \mathscr{X}_{t+1}, x_t) \big), \tag{7}$$

for $t=1,\ldots,T-1$, where

$$A_t^{\text{one}} \in \underset{A\in\mathcal{A}^l(\mathcal{X}_t)}{\arg\max}\left\{\sum_{x_t\in A}\mathbb{P}_{u_t,x_{t-1},x_t}(A)\cdot\left(w(x_t)+\max_{A'\in\mathcal{A}^l(\mathcal{X}_{t+1})}\left\{\sum_{x_{t+1}\in A'}\mathbb{P}_{u_{t+1},x_t,x_{t+1}}(A')w(x_{t+1})\right\}\right)\right\},$$

for $t=1,\ldots,T-1$, where $V_T^{\text{one}}(u_T,\mathcal{X}_T,x_{T-1})=V_T^m(u_T,\mathcal{X}_T,x_{T-1})$ for all u_T, $\mathcal{X}_T$, and x_{T-1}, that is, in the last time slot one-step look-ahead recommendations are simply myopic. We first provide some backing for these heuristics, and in §4 we document a controlled experiment that demonstrates the value captured in practice by implementing such heuristics.

3.3.2. Simulation. We conducted a simulation based on our model estimates, to evaluate the performance gap (measured in clicks per visit) between optimal and myopic recommendations, and the portion of this gap that may be captured by one-step look-ahead recommendations. The setup and the results of the simulation are detailed in Appendix A.1 (see the online companion). Our results show that although optimal recommendations that account for the entire future path of readers may generate an increase of approximately 50% in clicks per visit relative to myopic recommendations, a major part (between 70% and 90%) of this performance gap may be captured by one-step look-ahead recommendations. In addition, the impact of one-step look-ahead recommendations is similar for experienced and inexperienced readers.

3.3.3. Theoretical Near Optimality. We show that under mild structural assumptions, the performance gap between the one-step look-ahead policy and that of the optimal recommendation policy is suitably small. For simplicity we consider a special case of the CRP in which the user type does not update ($u_t = u_0 = u$ for all $t=1,\ldots,T$), every recommendation consists of a single link, $w(x)=1$ for any available article x, and that the set of available articles $\mathcal{X}$ to be continuous and convex (and therefore it is not updated throughout the problem horizon). Specifically, we assume the set $\mathcal{X}$ is defined by

$$\mathcal{X}=\{(\gamma,\beta)\colon -1\le\gamma\le 1, -1\le\beta\le 1, \beta\le 2-\varepsilon-\gamma\}, \quad (8)$$

for some $\varepsilon\in[0,1]$. The set $\mathcal{X}$ is depicted in Figure B-4 in the online companion, which appears in the online companion. Intuitively, ε represents the trade-off between clickability and engageability in the efficient frontier set of available articles; when ε is small, this trade-off is mild in the sense that there are "many" articles with high clickability and high engageability.

Proposition 3 (Near Optimality of One-Step Look-Ahead Recommendations). *Let $\mathcal{X}$ be the set of available articles defined in (8), and assume that $\mathbb{P}_{u,x}(y)\le\bar{p}$ for all $x\in\mathcal{X}$, $y\in\mathcal{X}$, and $u\in\mathcal{U}$. Then, for any $u\in\mathcal{U}$ and $x_0\in\mathcal{X}$,*

$$\frac{V_1^{\text{one}}(u,\mathcal{X},x_0)}{V_1^*(u,\mathcal{X},x_0)}\ge e^{-2\varepsilon}\left(\frac{1+e^{-2\varepsilon}\bar{p}}{1+\bar{p}}\right)^{T-1}.$$

Recall that T is the horizon of the content recommendation problem, $\bar{p}$ is an upper bound on the transition probabilities, and ε introduces a trade-off between engageability and clickability over the efficient frontier of available articles, as is captured by the definition of $\mathcal{X}$ in (8). Thus, proposition 3 implies that the performance of the one-step look-ahead policy approaches that of the optimal recommendation policy when the (optimal) click probabilities are small, and when the efficient frontier set of available articles satisfy a mild trade-off between engageability and clickability.

4. Pilot Study: A Controlled Field Experiment

The analysis presented in §3.3 implies that there might be significant value in accounting for a single future step in the potential path of readers. To test this, together with Outbrain, we designed a live experiment that compares the performance of recommendations that account for a single future step with the performance of those that myopically aim at maximizing CTR. To take into account the information and complexity limitations of the operating recommendation system, we consider a practical and simple class of one-step look-ahead recommendation policies that suitably adjusts the approach described in §3.3.

4.1. Methodology

4.1.1. Process Overview. Whenever a reader arrives to an article, an assortment of recommended links is produced, through a process that involves various classes of recommendation algorithms. Outbrain uses several different classes of algorithms to make organic recommendations and mixes the recommendations of its different algorithms in the final assortment of recommended articles. At a high level, these algorithms operate as index policies: first, they assign grades to candidate articles; once grades are assigned, a typically randomized selection process takes place, where articles with higher grades are more likely to be recommended. Different algorithms may vary one from the other by their ranking system as well as by their randomized selection process. For example, whereas a certain ranking system may use estimates of CTR based on past clicks, other ranking systems may use characteristics of articles (such as their main topic), the extent of relation to the article the reader is currently positioned at, as well as documented performance of similar articles. To maintain variability in the produced assortment, a supervising mechanism typically ensures that each recommendation contains links selected by different classes of algorithms; yet, more successful classes have, on average, more "representatives" in recommended assortments.

To test the impact of accounting for engageability of candidate articles throughout the recommendation process, we will focus on a certain class of algorithms. As we later explain in more detail, in a test group we will adjust the ranking system of this class to account for engageability, and will measure the impact of the performance of that class relative to a control group where the class will maintain its original ranking system. The class we will adjust is designed to maximize the CTR of recommended articles; moreover, it ranks articles by estimating their CTR directly from click observations. It is considered a very successful class, and therefore typically generates a relatively large portion (roughly one third) of the organic links that are recommended in each assortment. It is worthwhile to mention that all the rest of the algorithms (that will not be adjusted) are also designed and tuned to maximize the CTR of recommended articles, but do so following different methods and using other inputs such as the ones discussed above.

4.1.2. Myopic Approach: CTR-Based Class. At a high level, the class we adjust operates as follows.

CTR-based procedure $\mathcal{P}$. Input: a set $\mathcal{X}$ of available articles.

1. For each article $x \in \mathcal{X}$ calculate CTR(x) along a window of recent observations.
2. For each $x \in \mathcal{X}$ assign a weight $q(x) = \psi[\mathrm{CTR}(x)]$, where $\psi\colon \mathbb{R} \to \mathbb{R}$ is some strictly increasing mapping.
3. For each $x \in \mathcal{X}$ assign a probability $p(x) = q(x)/\sum_{x' \in \mathcal{X}} q(x')$.
4. Draw an article to recommend according to the distribution $\{p(x)\}_{x \in \mathcal{X}}$.

The mapping ψ (together with the normalization that follows it) transforms an ordered list of observed CTRs into a distribution from which articles are sampled. The essential characteristic of ψ is in being strictly increasing: an article with higher observed CTR has higher probability of being recommended.[3] We further note that the set $\mathcal{X}$ considers some system constraints (for example, the article that currently hosts the recommendation cannot be recommended).

[3] Because of a nondisclosure agreement the structure of ψ is not explicitly disclosed here.

4.1.3. Adjusted-Myopic Proxy for the One-Step Look-Ahead Heuristic. Finding a solution for the one-step look-ahead problem involves worst-case computational complexity of order $|\mathcal{X}|^2$, compared with order $|\mathcal{X}|$ that is required to find the best myopic recommendation. Since the set of available articles is typically very large, a first step toward implementation was to find a proxy for the one-step look-ahead policy that requires computational complexity of order $|\mathcal{X}|$, and that follows a procedure similar to the policy currently in place. Moreover, since the clickability and the engageability of articles (the sequences of γ and β estimates) are obtained by an off-line estimation and currently are not available online, we use proxies that are collected and measured in an online fashion throughout the recommendation process. An intuitive proxy for probability to *click to* an article is the CTR of the article, defined by

$$\mathrm{CTR}(x) = \frac{\#\{\text{clicks to } x\}}{\#\{\text{times } x \text{ is recommended}\}},$$

for each article x. The CTR of each article is calculated over some time window along which offerings and click observations are documented. The correlation between the values of $\mathbb{P}_{u_t, x_{t-1}, x_t}(A)$, when constructed by our estimators (considering the recommended article (x_t), the host article (x_{t-1}), the reader type (u_t), and the whole assortment that was offered) and the values of CTR(x_t) (of the recommended article, x_t) that were calculated based on observations from the same estimation batch is 0.29. Similarly, a potential proxy for probability to click from an article is the exit-CTR of an article, defined by

$$\text{exit-CTR}(x) = \frac{\begin{array}{c}\#\{\text{events of at least one additional}\\ \text{page-view after reading } x\}\end{array}}{\#\{\text{times } x \text{ was viewed}\}}$$

This exit-CTR measure accounts not only for clicks on organic links, but also for other events, such as an additional article that was read at the same publisher shortly after visiting article x (for example, after a short visit in the front page of the media site). We found the correlation between the values of $\max_{A' \in \mathcal{A}^l(\mathcal{X}_{t+1})}\{\sum_{x_{t+1} \in A'} \mathbb{P}_{u_{t+1}, x_t, x_{t+1}}(A')\}$, when constructed by our estimators (considering the recommended article (x_t), the host article (x_{t-1}), the reader type (u_t), the whole assortment that was offered, as well as the set of articles that were available for recommendation at the following step), and the values of exit-CTR(x_t) (of the host article, x_t) that were calculated based on observations documented in the same estimation batch to be 0.25.

Based on these findings, and assuming a uniform article value $w(\cdot) = 1$, we suggest the following *adjusted-myopic* recommendation policy that recommends the l articles with the highest index value:

$$\text{Index}(y) = \text{CTR}(y)[1 + \text{exit-CTR}(y)].$$

Recalling the one-step look-ahead heuristic in (7), the adjusted myopic policy uses observable proxies of the heuristic's elements to recommend articles based on a proxy of their one-step look-ahead value. This policy accounts for the potential future path of the reader upfront, without increasing the computational complexity of index policies that are currently used by the system.

4.1.4. Accounting for the Future Path of Readers.

As an alternative to the procedure $\mathcal{P}$ we suggest a class of recommendation policies that account for the engageability of candidate articles.

A simple look-ahead procedure $\tilde{\mathcal{P}}$. Input: a set $\mathcal{X}$ of available articles.

1. For each article $x \in \mathcal{X}$ calculate $\text{CTR}(x)$ and $\text{exit-CTR}(x)$ along a window of recent observations.
2. For each $x \in \mathcal{X}$ assign a weight $\tilde{q}(x) = \psi[\text{CTR}(x) \cdot (1 + \text{exit-CTR}(x))]$, where $\psi[\cdot]$ is the same mapping as in the procedure $\mathcal{P}$.
3. For each $x \in \mathcal{X}$ assign a probability $\tilde{p}(x) = \tilde{q}(x)/\sum_{x' \in \mathcal{X}} \tilde{q}(x')$.
4. Draw an article to recommend according to the distribution $\{\tilde{p}(x)\}_{x \in \mathcal{X}}$.

We note that since the same mapping ψ is used both in $\mathcal{P}$ and in $\tilde{\mathcal{P}}$, the only difference between these recommendation procedures is in the argument of ψ: in $\mathcal{P}$ it is the CTR of candidate articles; in $\tilde{\mathcal{P}}$ this argument is adjusted by the engageability of candidate articles, through the exit-CTR proxy.

To motivate the implementation of the adjusted-myopic procedure, and to evaluate the potential improvement one could hope for when applying it (relative to the CTR-based procedure), we repeated the simulation mentioned in §3.3 (and detailed in Appendix A.1 in the online companion) to measure the performance of recommendations that were selected to maximizes the adjusted-myopic objective in every step, observing that the adjusted-myopic policy achieved roughly 12% improvement compared with the myopic policy. Alternatively, roughly 25% of the improvement achieved by the one-step look-ahead policy over the myopic policy was captured by the adjusted-myopic one. The performance gap between the one-step look-ahead policy and the adjusted-myopic one can be explained by the imperfect correlation between the elements of the one-step look-ahead objective and those of the adjusted-myopic objective.

4.2. Experiment Setup

Each Outbrain reader has a unique user ID (a number that is uniquely matched with an Internet Protocol address) that typically does not change with time and is assumed to be independent of the clicking behavior of the reader. In the experiment, each reader was assigned either to a test group or to a control group based on this unique ID, in an arbitrary, predetermined manner that was designed to create a control group that is roughly four times larger than the test group. Each reader was assigned to the same group (test or control) in all the reader's arrivals throughout the entire time over which the experiment took place.

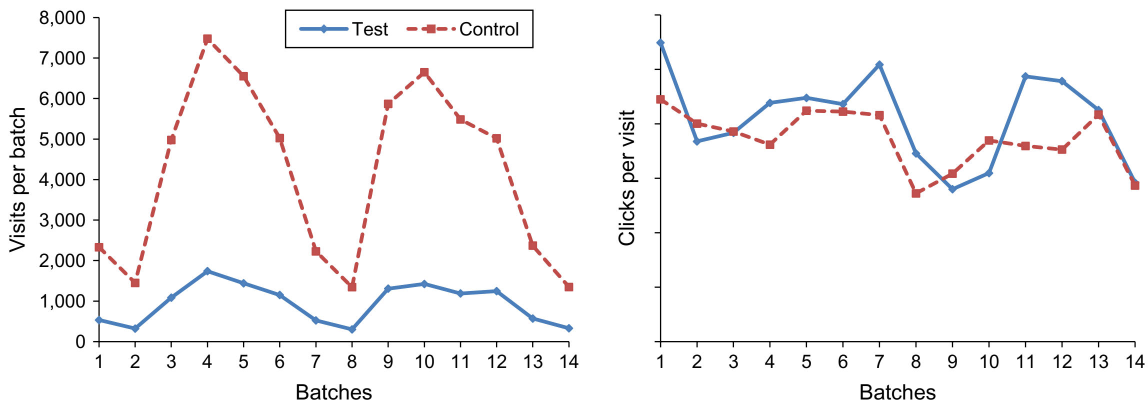
Whenever a reader arrived to an article, roughly one-third of the shown organic links were generated by the class described above. When the reader belonged to the control group, these links were generated by the procedure $\mathcal{P}$, that is, considering the CTR of candidate articles. When the reader belonged to the test group, these links were generated by $\tilde{\mathcal{P}}$, that is, considering both the CTR and the exit-CTR of candidate articles. The rest of the organic links were generated by other CTR-maximizing algorithms, whose process was not affected by the group each user belonged to.[4] The exact number of links that were generated in each impression by the algorithm class described above was determined independently of the group that the user belonged to. The group to which a reader belonged did not impact the sponsored links that were offered, or any other characteristics of the Web page.

We emphasize that the classes $\mathcal{P}$ and $\tilde{\mathcal{P}}$ are identical except for the objective their produced indexed lists maximize: CTR (in $\mathcal{P}$) versus a combination of CTR and exit-CTR (in $\tilde{\mathcal{P}}$). Therefore, comparing the performance of the class $\tilde{\mathcal{P}}$ with the one of the class $\mathcal{P}$, and having the rest of the algorithms not affected by the group that each user belonged to, the experiment isolates the impact of accounting for the future path of users from the potential effect of the algorithmic method itself.

To limit the scope of the experiment, it was designed to focus on *active* readers that have just clicked on an organic recommended link (active readers represent a special subset of experienced readers). A reader "entered" the experiment after the first click, and we do not differentiate with respect to the algorithm that generated the first clicked link. From that point, we tracked the path of the reader throughout organic recommendations generated by the described algorithm class, and compared the performance of that class of algorithms in the test group relative to the control

[4] Because of a nondisclosure agreement with Outbrain, we do not further disclose these algorithms here.

Figure 7 (Color online) Clicks per Visit, Test vs. Control



Notes. Number of visits recorded along each four-hour batch (left) and the average number of clicks per visit observed in each batch (right) in the test group and in the control group. Because of a nondisclosure agreement with Outbrain, the clicks per visit units have been removed from the plot.

group. In both test and control groups CTR and exit-CTR values were updated every three hours, based on observations documented in the previous three hours. The experiment took place during December 2013, over 56 consecutive hours beginning at midnight between a Monday and a Tuesday, in a single media site.

4.2.1. Performance Indicators. We follow the number of consecutive clicks made by each active reader on links that were generated by the algorithm class on which we focus. When the reader clicks on a sponsored link or an organic link that was not generated by that class, or when the reader terminates the session in any way without clicking on a recommended link, the path of the reader ends. We partition the experiment period into 14 batches of four hours each. Along each batch we calculate, in both groups, the average clicks per active reader's visit (not counting the first click after which the reader "entered" the experiment). We denote by $\nu_{\text{control}}(t)$ the average clicks per visit in the control group along batch t, and by $\nu_{\text{test}}(t)$ the average clicks per visit in the test group along batch t. We further denote by $r(t)$ the relative difference in performance in the test group relative to the control group, in batch t:

$$r(t) = 100 \cdot \frac{\nu_{\text{test}}(t) - \nu_{\text{control}}(t)}{\nu_{\text{control}}(t)}.$$

4.3. Results

4.3.1. Batch-by-Batch Analysis. Throughout the experiment, 58,116 visits of "active" readers were recorded in the control group and 13,155 visits were recorded in the test group. The experiment documents a 9.86% improvement in clicks per visit in the test group compared with the control group. The volume of visits and the documented clicks per visit values in the two groups along different batches are depicted in Figure 7. The relative differences in clicks per visit appear in Table 3.

Since the experiment took place over a relatively short time period and in a media site with relatively low volume of readers, some of the differences in the performance are not statistically significant. Nevertheless, in most of the batches there was an improvement in the test group relative to the control group; in three batches (4, 11, and 12) this performance improvement is statistically significant.

4.3.2. Aggregate Analysis. To estimate the aggregate impact of the treatment, we estimate a linear regression over the 58,116 documented visits. We denote by ν_i the number of clicks along visit i. We denote by TEST_i a dummy variable indicating whether visit i was in the test group (equals one if it was in the test group, zero otherwise); the associated coefficient g_{test} captures the impact of the treatment on the performance relative to the control group. For $t \in \{2, \ldots, 14\}$ we denote by $\text{BATCH}_{i,t}$ a dummy variable indicating whether visit i occurred in batch t;

Table 3 Relative Improvement, Test Compared to Control

Batch	Visits (control)	Visits (test)	$r(t)$ (%)
1	2,329	532	23.4
2	1,449	321	−8.2
3	4,977	1,085	−0.6
4	7,487	1,740	21.1*
5	6,551	1,439	5.5
6	5,024	1,151	3.3
7	2,227	523	22.3
8	1,345	301	27.0
9	5,868	1,308	−9.3
10	6,649	1,422	−16.2
11	5,484	1,189	35.8**
12	5,018	1,246	35.6**
13	2,370	569	2.0
14	1,347	329	1.8

*At confidence level $p < 0.1$; **at confidence level $p < 0.05$.

Table 4 **Linear Regression: Scaled Results**

Effect	Parameter	Estimate	Standard error
Intercept	α	1*	0.066
Group	g_{test}	0.079*	0.035
Batch	b_2	−0.153	0.105
	b_3	−0.170*	0.078
	b_4	−0.192*	0.074
	b_5	−0.078	0.078
	b_6	−0.087	0.078
	b_7	−0.109	0.096
	b_8	−0.389*	0.109
	b_9	−0.349*	0.079
	b_{10}	−0.227*	0.079
	b_{11}	−0.179*	0.079
	b_{12}	−0.188*	0.079
	b_{13}	−0.100	0.092
	b_{14}	−0.384*	0.109

Note. Because of a nondisclosure agreement, values in the table were obtained by dividing original estimates and standard errors by the absolute value of the intercept estimate.

*At confidence level $p < 0.05$.

the coefficients $b_2, \ldots, b_{14}$ capture the various batches effects relative to the first batch. We denote by ϵ_i the error term associated with visit i. We estimate the following linear model:

$$\nu_i = \alpha + g_{\text{test}} \cdot \text{TEST}_i + \sum_{t=2}^{14} b_t \cdot \text{BATCH}_{i,\,t} + \epsilon_i.$$

Scaled results of the regression are detailed in Table 4 (divided by the absolute value of the intercept estimate). The estimate of the treatment coefficient g_{test} is positive at confidence level $p < 0.05$, indicating that adjusting the CTR objective by the engageability proxy improved the performance of the algorithm class in a statistically significant manner.

4.3.3. Discussion. We note that the improvements above are witnessed even though the adjusted myopic policy could potentially be fine-tuned to enhance performance; for example, one could only suspect that the mapping ψ could be tuned to optimize the adjusted-myopic objective. Moreover, we note that the exit-CTR proxy accounts not only for clicks on Outbrain's links but also for other events of future page views. Indeed, other proxies that have higher correlation with elements of the one-step look-ahead heuristic may yield better performance. One example is the following exit-CTR measure, which accounts only for clicks on the recommendation that is hosted by the article:

$$\text{exit-CTR}'(x) = \frac{\#\{\text{clicks on recommendation from } x\}}{\#\{\text{times } x \text{ hosts a recommendation along } \tau\}}.$$

We found the correlation between the values of $\max_{A' \in \mathcal{A}^l(\mathcal{X}_{t+1})} \{\sum_{x_{t+1} \in A'} \mathbb{P}_{u_{t+1},\, x_t,\, x_{t+1}}(A')\}$ and the values of the above exit-CTR(x_t) values calculated based on observations documented in the same estimation batch to be 0.36 (higher than the one of proxy that was used in the pilot). The latter suggests that the performance improvement that was documented in the pilot may be further increased by using alternative and more suitable engageability proxies.

5. Concluding Remarks

This study introduces the emerging practice of dynamic content recommendation services and suggests a practical approach to improve the performance of such services by accounting for the potential future path of users. Our proposed path-focused approach was validated by theoretical bounds, simulation, and predictive analysis. Furthermore, we tested the approach in a live pilot experiment to compare its performance to myopic CTR-maximizing recommendations that are widely used in current practice. The pilot experiment documented a statistically significant aggregate improvement of 9.86% in clicks-per-visit when recommendations accounted also for the future path of readers through a proxy of the engageability parameters. These results indicate that accounting for the future path of users in real time may significantly improve performance. Moreover, such an improvement does not necessarily require the development of new recommendation technologies; it may be obtained by adjusting existing algorithms that have been fine-tuned to maximize a myopic objective such as CTR. Indeed, the key trade-off between instantaneous payoff and potential future payoff may exist in many dynamic/sequential services (a related example being YouTube's video recommendations, but other online services such as Amazon's recommendations may exhibit similar trade-offs). Thus, although our analysis quantifies this trade-off in the context of a dynamic service that recommends articles, the insights demonstrated by the introduced content space and the practical approach of approximating a one-step look-ahead heuristic via observable proxies may have broader implications.

To visualize the aforementioned trade-off between instantaneous and future payoffs in the context of articles, we developed a representation of articles in a compact *content space* along two dimensions: clickability and engageability. Various practical applications arise from this new representation. For example, our framework allows one to track the dynamics of clickability and engageability, which may be referred to as *the aging process* of articles. Since most of the articles lose relevancy rapidly from the time they are published, tracking this process is a practical challenge that is crucial for the provider's ability to screen out articles that became irrelevant. Tracking the varying coefficients shows that most of the articles exhibit a decreasing trend in both dimensions from the time they are published until they are not recommended anymore (typically because of their declining clickability). However, some articles exhibit a decrease in engageability while maintaining high clickability, becoming potential "traffic traps." In addition, topics may be visualized and better understood through the lens of the content space dimensions. In Appendix A.3 (see the online companion) we suggest a representation of topics in the content space through confidence sets and demonstrate this for two representative topics.

Various avenues for future research also emerge. In particular, a broader question is how to adequately measure performance/quality of a service. Although engageability might be one important measure, there are other potential metrics to consider. For example, in the present case, we also measured the average time that was spent by readers on articles (time spent is an independent and common characteristic of online services; see, e.g., Shahabi et al. 1997, Gündüz and Özsu 2003). Although both engageability and time spent potentially capture different aspects of quality, an observed positive correlation (0.28) seems to indicate a relation between the two. Developing a framework for measuring the quality of a service is indeed an ambitious avenue of future research.

Supplemental Material

Supplemental material to this paper is available at http://dx.doi.org/10.1287/msom.2015.0548.

Acknowledgments

The authors are grateful to Alan Scheller-Wolf, an associate editor, and three reviewers for their feedback that helped to improve the paper. The authors deeply thank their industrial partner, Outbrain Inc., with special thanks to Itai Hochman and Rani Nelken, who participated in this study from its inception. This work was supported by the National Science Foundation [Grant 0964170], the United States–Israel Binational Science Foundation [Grant 2010466], the Eugene M. Lang support fund, and the W. E. Deming Center for quality, productivity, and competitiveness.

References

Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engrg., IEEE Trans.* 17:734–749.

Alptekinoglu A, Honhon D, Ulu C (2012) Learning consumer tastes through dynamic assortments. *Oper. Res.* 60:833–849.

Araman V, Fridgeirsdottir K (2011) A uniform allocation mechanism and cost-per-impression pricing for online advertising. Working paper, American University of Beirut, Beirut, Lebanon.

Bucklin RE, Sismeiro C (2003) A model of Web site browsing behavior estimated on clickstream data. *J. Marketing Res.* 40: 249–267.

Bucklin RE, Sismeiro C (2009) Click here for Internet insight: Advances in clickstream data analysis in marketing. *J. Interactive Marketing* 23:35–48.

Bucklin RE, Lattin JM, Ansari A, Gupta S, Bell D, Coupey E, Little JDC, Mela C, Montgomery A, Steckel J (2002) Choice and the Internet: From clickstream to research stream. *Marketing Lett.* 13:245–258.

Caro F, Gallien J (2007) Dynamic assortment with demand learning for seasonal consumer goods. *Management Sci.* 53:276–292.

Caro F, Martinez-de-Albeniz V, Rusmevichientong P (2014) The assortment packing problem: Multiperiod assortment planning for short-lived products. *Management Sci.* 60:2701–2721.

Feng J, Bhargava HK, Pennock DM (2007) Implementing sponsored search in Web search engines: Computational evaluation of alternative mechanisms. *INFORMS J. Comput.* 19:137–148.

Gary MR, Johnson DS (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W. H. Freeman and Company, New York).

Ghose A, Yang S (2009) An empirical analysis of search engine advertising: Sponsored search in electronic markets. *Management Sci.* 55:1605–1622.

Gündüz S, Özsu MT (2003) A Web page prediction model based on click-stream tree representation of user behavior. *Proc. Ninth ACM SIGKDD Internat. Conf. Knowledge Discovery and Data Mining* (ACM, New York), 535–540.

Haywood N, Cairns P (2006) Engagement with an interactive museum exhibit. McEwan T, Gulliksen J, Benyon D, eds. *People and Computers XIX—The Bigger Picture* (Springer, London), 113–129.

Hui S, Fader P, Bradlow E (2009) Path data in marketing: An integrative framework and prospectus for model building. *Marketing Sci.* 28:320–335.

Jansen BJ, Mullen T (2008) Sponsored search: An overview of the concept, history, and technology. *Internat. J. Electronic Bus.* 6:114–131.

Karger D, Motwani R, Ramkumar GDS (1997) On approximating the longest path in a graph. *Algorithmica* 18:82–98.

Kök GA, Fisher ML, Vaidyanathan R (2009) Assortment planning: Review of literature and industry practice. *Retail Supply Chain Management* (Springer, New York).

Kumar S, Jacob VS, Sriskandarajah C (2006) Scheduling advertisements on a Web page to maximize revenue. *Eur. J. Oper. Res.* 173:1067–1089.

Lee D, Hosanagar K, Nair H (2014) The effect of social media marketing content on consumer engagement: Evidence from Facebook. Working paper, Tepper School of Business, Carnegie Mellon University, Pittsburgh.

Linden G, Smith B, York J (2003) Amazon.com recommendations: Item-to-item collaborative filtering. *Internet Comput., IEEE* 7:76–80.

Mehta A, Saberi A, Vazirani U, Vazirani V (2007) Adwords and generalized online matching. *J. ACM* 54, Article 22.

Montgomery AL, Li S, Srinivasan K, Liechty JC (2004) Modeling online browsing and path analysis using clickstream data. *Marketing Sci.* 23:579–595.

Najafi-Asadolahi S, Fridgeirsdottir K (2014) Cost-per-click pricing for display advertising. *Manufacturing Service Oper. Management* 16:482–497.

O'Brien HL, Toms EG (2008) What is user engagement? A conceptual framework for defining user engagement with technology. *J. Amer. Soc. Inform. Sci. Tech.* 59:938–955.

O'Brien HL, Toms EG (2010) The development and evaluation of a survey to measure user engagement. *J. Amer. Soc. Inform. Sci. Tech.* 61:50–69.

O'Reilly T (2009) *What Is Web* 2.0 (O'Reilly Media, Sebastopol, CA).

Pazzani MJ, Billsus D (2007) Content-based recommendation systems. Brusilovsky P, Kobsa A, Nejdl W, eds. *The Adaptive Web* (Springer-Verlag, Berlin), 325–341.

Ricci F, Rokach L, Shapira B (2011) Introduction to recommender systems handbook, Ricci F, Rokach L, Shapira B, Kantor PB, eds. *Recommender Systems Handbook* (Springer, New York), 1–35.

Rusmevichientong P, Shen ZM, Shmoys DB (2010) Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Oper. Res.* 58:1666–1680.

Saure D, Zeevi A (2013) Optimal dynamic assortment planning with demand learning. *Manufacturing Service Oper. Management* 15:387–404.

Shahabi C, Zarkesh AM, Adibi J, Shah V (1997) Knowledge discovery from users Web-page navigation. *Proc. IEEE Seventh Internat. Workshop Res. Issues in Data Engrg.* (IEEE Computer Society Press, Los Alamitos, CA), 20–29.

Su X, Khoshgoftaar TM (2009) A survey of collaborative filtering techniques. *Adv. Artificial Intelligence* 2009, Article 4.

Uehara R, Uno Y (2005) Efficient algorithms for the longest path problem. *Algorithms and Computation* (Springer, Berlin), 871–883.

Webster J, Ahuja JS (2006) Enhancing the design of Web navigation systems: The influence of user disorientation on engagement and performance. *MIS Quart.* 30:661–678.