Information Retrieval Journal (2019) 22:447–475
https://doi.org/10.1007/s10791-019-09351-2

Optimizing the recency-relevance-diversity trade-offs in non-personalized news recommendations

Abhijnan Chakraborty[1,2] · **Saptarshi Ghosh**[2] · **Niloy Ganguly**[2] · **Krishna P. Gummadi**[1]

Received: 2 July 2018 / Accepted: 26 January 2019 / Published online: 27 February 2019

Abstract

Online news media sites are emerging as the primary source of news for a large number of users. Due to a large number of stories being published in these media sites, users usually rely on news recommendation systems to find important news. In this work, we focus on automatically recommending news stories to all users of such media websites, where the selection is not influenced by a particular user's news reading habit. When recommending news stories in such non-personalized manner, there are three basic metrics of interest—recency, importance (analogous to *relevance* in personalized recommendation) and diversity of the recommended news. Ideally, recommender systems should recommend the most important stories soon after they are published. However, the importance of a story only becomes evident as the story ages, thereby creating a tension between recency and importance. A systematic analysis of popular recommendation strategies in use today reveals that they lead to poor trade-offs between recency and importance in practice. So, in this paper, we propose a new recommendation strategy (called *Highest Future-Impact*) which attempts to optimize on both the axes. To implement our proposed strategy in practice, we propose two approaches to predict the future-impact of news stories, by using crowd-sourced popularity signals and by observing editorial selection in past news data. Finally, we propose approaches to inculcate diversity in recommended news which can maintain a balanced proportion of news from different news sections. Evaluations over real-world news datasets show that our implementations achieve good performance in recommending news stories.

Keywords News recommendation · Recency-relevance trade-off · Recommendation diversity · Coverage bias

This paper is an extended version of our earlier papers "Optimizing the Recency-Relevancy Trade-off in Online News Recommendations" (Chakraborty et al. 2017a) and "Editorial Algorithms: Optimizing Recency, Relevance and Diversity for Automated News Curation" (Chakraborty et al. 2018). The additional contribution of this paper is detailed in the Related Work section.

✉ Abhijnan Chakraborty
achakrab@mpi-sws.org

[1] Max Planck Institute for Software Systems, Saarbrücken, Germany

[2] Indian Institute of Technology Kharagpur, Kharagpur, India

1 Introduction

Recent years have witnessed a fundamental change in the news landscape. Today, *online news media sites*, be they mass media sites like The New York Times (`nytimes.com`, henceforth referred to as 'NYTimes'), or CNN (`us.cnn.com`), or social media sites like Facebook or Twitter, are emerging as the primary (and frequently *only*) sources of news for a large and rapidly growing fraction of people world-wide. On the other hand, the number of users receiving news via traditional offline methods, e.g., via print newspapers and weeklies, are in steep decline.[1] A recent survey by the Pew Research Center found that around 48% of American Internet users got political news on social media sites like Facebook, almost as many as those that got such news from local television channels (Mitchell et al. 2014).

Due to the round-the-clock (24/7) nature of online news and the need to keep their audience coming back to their sites,[2] online media sites publish news stories throughout the day. As a consequence, the number of news stories appearing in the media sites today are way more than what any user can possibly consume. For instance, more than 800 news stories on average appear on the NYTimes website every day. For social media sites like Facebook or Twitter, this number is many times higher. In the face of such information overload, the users of online media sites need to rely on news recommendations to find interesting stories and discover important events.

Such recommendations are primarily of two types: *personalized* and *non-personalized*. In personalized recommendations, news stories are recommended to individual users based on their interests as inferred from their past activities. Generally, as different users may have different interests, the recommended news can differ substantially from user to user. On the other hand, in non-personalized (or *global*) recommendations, the same content is broadcasted to all users (at least in a particular geographical area) of a media site. For instance, the same front-page stories are shown to all news readers visiting a media site, where *personalized factors* (e.g., interests of individual readers) are *not* used for selecting/recommending them. Personalized content recommendations have been studied extensively in the past literature (Agarwal et al. 2011; Li et al. 2011b, c; Liu et al. 2010), and many algorithms have been proposed in this regard. In this work, we focus on the lesser explored domain of non-personalized recommendations. As we show in the subsequent sections, recommending news is surprisingly tricky, even when they are *not* personalized.

While recommending news to the readers, there are three basic metrics of interest—recency, relevance and diversity of the selected stories. Recency captures a story's age, i.e., when the story is published. By relevance, we refer to the importance or the impact of a story, either judged from the editors' *notions of newsworthiness* (Shoemaker et al. 2009), or estimated through the audience-driven popularity measures, such as the number of people who read or liked the story. These two measures are somewhat complementary, sometimes audience preference for stories matches that of the editors; whereas, for some other stories, their preferences diverge. In this paper, we have taken into account both ways of estimating importance of different news stories. Finally, along with their newsworthiness,

[1] http://www.stateofthemedia.org/2013/newspapers-stabilizing-but-still-threatened/newspapers-by-the-numbers.

[2] Similar to most online websites, many online news media sites are also predominantly funded by their users watching advertisements on their sites.

the recommended news stories should avoid covering redundant topics, and instead have diverse topical coverage.

Ideally, recommender systems should select the most important stories while they are still recent, i.e., soon after they are published. However, estimating relevance for global recommendations poses a different set of challenges, compared to judging the relevance of personalized recommendations. While relevance of a news story for a particular user can be judged based on the past actions of the user, it is hard to assess the global impact of a story right after its publication (even for human editors, and particularly for automated recommendations). The impact of a story only becomes evident as the story ages, which creates a fundamental tension between selecting recent stories with uncertain importance or choosing very important stories that are not recently published. Our primary goal is to understand and optimize for this *recency-importance trade-off* when recommending news stories.

We begin by analyzing the recency-importance trade-offs offered by the recommendation strategies which are in use today. Specifically, we investigate

1. *Recent-Impact*-based recommendations (used in NYTimes recommendations), where stories which were most popular in the latest time interval are selected, and
2. *Rising-Impact*-based recommendations (used in Twitter trending topics), where stories that received the sharpest spike in popularity in the most recent time interval, compared to the previous time interval, are chosen.

These strategies are based on two key assumptions about popularity life-cycles of news stories (e.g., how the number of views a story receives evolves over time): (1) popularity life-cycles of all stories are somewhat similarly skewed (otherwise a low-impact story that receives all its views in a short time interval would be selected over a high-impact story that steadily accumulates views over a longer time interval), and (2) news stories achieve their peak recent-popularity or rising-popularity early in their life-cycles (allowing them to be chosen soon after their publications). Our analysis, using real-world news datasets, shows that these assumptions do not hold quite frequently, leading to poor trade-offs between recency and importance in practice.

To optimize recency-importance trade-off, we evaluate a simple, but previously unexplored, strategy called *Future-Impact*-based recommendations, where stories are selected based on how many views they are expected to receive in the future (and *not* in the past). Intuitively, future-impact of a story captures the extent to which the story is likely to be discussed in the future, and journalism studies have argued that it is a useful metric for selecting news stories in its own right (Novendstern 2011). Additionally, two properties of the future-impact metric help achieving better trade-offs between recency and importance: (1) a high-impact story has higher future-impact than a low-impact story, and (2) news stories have highest future-impact shortly after they are published.

We tackle the technical challenges related to the deployment of future-impact strategy. We utilize both crowd-driven popularity signals, where the idea is to *predict* the future-impact of a story at time *t* based on its popularity till time *t*, and by mimicking editorial judgements on past news data. Evaluation over real-world news datasets shows that our strategies achieve good performance trade-offs between recency and importance.

Next, we explore how recommending stories only based on future-impact may create topical coverage bias in the recommended stories. We show that focusing only on future-impact can result in considerable diurnal variation in coverage of news stories related to specific topics. For example, news stories related to Economy or Science can be recommended predominantly at certain times of the day and not at other times. We further show that such churn in recommended news stories can induce a significant bias in users' topical exposure, depending on her diurnal browsing behavior. To tackle this issue, we propose approaches to introduce diversity in recommended news, which can further maintain certain composition of stories from different news sections.

In summary, the paper makes the following four contributions: (1) we analyze the recency-importance trade-offs achieved by current news recommendation strategies and show them to be sub-optimal, (2) we propose a simple yet previously overlooked strategy that selects stories based on their future-impact, and show that it has the potential to achieve better recency-importance trade-offs than current strategies, (3) we propose a practical implementation of future-impact based recommendation strategy, by utilizing popularity signals as well as editorial judgements in predicting future-impact, and (4) we develop approaches to eliminate chances of having temporal coverage bias in the recommended stories. Evaluations over real-world datasets show that our implementation achieves good performance in recommending news stories.

2 Related work

As mentioned earlier, this paper is an extended version of our earlier papers: Chakraborty et al. (2017a), where we introduced the notion of recency-relevance trade-off in news recommendations that rely on popularity signals to select the news stories, and Chakraborty et al. (2018), where we developed an approach to mimic editorial selection done in media newsrooms. This paper brings together these two complementary ways to estimate relevance (i.e., importance) of stories in non-personalized news recommendations: (1) using crowd-driven popularity signals and (2) by learning from editorial judgments. Additionally, by gathering extensive longitudinal data from NYTimes, we highlight how the recommendations optimizing for only recency and importance can create temporal coverage bias (i.e., different users visiting a media site at different times may end up having highly different topical exposure). Finally, we propose approaches to counter this bias and maintain desired composition of stories from different news sections at different times. Next, we discuss other related research efforts.

2.1 Personalized versus non-personalized news recommendations

A lot of prior works have focused on developing *personalized* news recommendation systems, which recommend news stories tailored to individual users. For example, Liu et al. (2010) developed a Bayesian model to predict individual user's interests from her past activities, and the news trend reflected from the activities of a group of users, and then recommend stories according to the interests. Li et al. (2011b) designed a scalable personalized news recommender system by using a two-level representation, containing the topics relevant to user's preference at one level, and the news articles on these topics at the second level. Agarwal et al. (2011) proposed *click shaping* to jointly optimize the number of clicks and post-click downstream utilities for recommending news stories. Maksai et al. (2015) proposed metrics to evaluate the performance of such systems. However, except *Most Emailed, Most Viewed, Most Shared* stories (Chakraborty et al. 2016a) or *Trending Topics* (Twitter 2010; Chakraborty et al. 2017b) deployed in media sites today, there are not many research works to develop *non-personalized* news recommendation systems. In this paper, we attempt to fill this vacuum by presenting a systematic approach to recommend news stories in the non-personalized scenario.

2.2 Recency versus importance debate

Present approaches on designing content recommendation systems are putting increasing emphasis on the recency and realtimeness of content. Liang et al. (2012) proposed a time-aware content recommendation system, while Watanabe et al. (2011) proposed a framework to detect breaking news, and trending events from online social media in real-time. This focus on recency also leads to a growing concern over the long-term importance of the recommended contents, and many users view such content as potentially waste of time information (Clear 2015). Although this debate on recency versus relevancy is going on for some time, to our knowledge, in this paper, we are the first to propose a recommendation strategy which can simultaneously optimize for recency as well as the importance of the recommended news stories.

2.3 Predicting popularity of online contents

Prior works have attempted to predict the popularity of YouTube videos (Figueiredo et al. 2014), Flickr images (McParlane et al. 2014), or future citation count of research papers (Yan et al. 2011; Yu et al. 2012). Similarly, efforts have been made to understand dynamic popularity classes (Lehmann et al. 2012; Crane and Sornette 2008), and whether they lead to the emergence of self-fulfilling prophecies (Salganik and Watts 2008), or social influence biases (Muchnik et al. 2013). Complementary to the above works, in this paper, we develop tools to predict the user attention different stories are going to receive in future, and utilize them for recommending news stories.

2.4 Diurnal patterns in media browsing

Intuitively, different readers around the world (in different timezones) can be expected to access media websites at different times of the day. Golder et al. (2007) observed strong diurnal patterns in accessing messages and applications on Facebook. Similarly, Duarte et al. (2007) found hourly variations in the generation of blog posts, bookmarks, as well as answers in different Q & A websites. Benevenuto et al. (2009) analyzed browsing patterns of tens of thousands of users in social media sites, using click-stream data from a social network aggregator. They observed that most of the users accessed the sites only a few times and during certain periods of a day. Similarly, Yasseri et al. (2012) found circadian patterns in the activities of Wikipedia editors. In this work, we argue that such diurnal browsing patterns of news readers can introduce coverage bias in the news consumed by different readers if the recommendation puts too much emphasis on recency of news stories, exhibiting high churn in the recommended news.

2.5 Coverage differences in information retrieval systems

With the rapid adoption of information retrieval systems such as search or recommendation systems, there have been multiple attempts to examine the information coverage of such systems. For instance, Vaughan and Thelwall (2004) studied the geographical coverage in search engine results, i.e., whether webpages from certain countries are being included more in the search results, than those from other countries. Graham and Zook (2013) analyzed whether certain geographical locations have more annotations in online maps than other places. Kulshrestha et al. (2015) analyzed the topical coverage of recommendation systems deployed on Twitter, and found that the social recommendations add some topical diversity to the word-of-mouth consumption of many users.

Few prior works in this context have focused on the linguistic differences in the online content. For example, Hecht and Gergle (2010) analyzed the knowledge diversity across different Wikipedia language editions. Bao et al. (2012) proposed a system 'Omnipedia', which allows the users to access information from different language editions of Wikipedia simultaneously. Similarly, Hong et al. (2011) studied how features such as URLs, hashtags, mentions, replies, and retweets are adopted in different languages on Twitter. Complementary to the above works, we explore the diurnal pattern in the selection of news stories, and find that even stories from the *same media source*, and written in the *same language* can introduce significant differences in news coverage.

2.6 Multi-objective optimization

Even though there is a large body of work on multi-objective optimization (Deb 2014) and constrained optimizations (Bertsekas 2014), there has not been much attempt to utilize these approaches to recommend news stories. Agarwal et al. (2011) proposed 'click shaping' to jointly optimize for clicks and post-click downstream utilities for recommending news stories in the personalized setting. In this paper, we use constrained optimization framework to bound the uncertainty in prediction, whereas simultaneously maximizing the predicted future-impacts of the recommended news stories. Similarly, we also try to jointly optimize the newsworthiness inferred from editorial judgments and recency aspects in non-personalized news recommendations.

3 Understanding the recency-importance trade-off

In this section, we first introduce the datasets and the terminology used, then analyze the recency-importance trade-offs offered by the recommendation strategies which are in use today, and finally motivate the need for a new strategy. Note that in this section and next two sections, we treat popularity of a news story as its importance.

3.1 Datasets used

To explore the recency-importance tradeoff in recommendations, we used the following two datasets representing different interaction patterns between news stories and their readers. The first dataset contains the viewing patterns of different news stories, and the second dataset is regarding the news sharing patterns on Twitter. None of these datasets consider personalized factors for presenting the news stories.

3.1.1 Yahoo! News dataset

We used the user click log made publicly available by Yahoo!.[3] Specifically, the R6B dataset contains a fraction of the user click information for 652 news stories displayed on the 'Today Module' on Yahoo!'s front page during the consecutive 15-day period from October 2 to October 16, 2011. The dataset contains information regarding 28,041,015 user visits to the 'Today Module' during this period. During each visit, a story was chosen uniformly at random following the method developed in Li et al. (2011a), and shown to the user. The click log contains the information about whether a user clicked on the story or not. As the data contains the timestamps of the user visits, for each story, we extracted the sequence of clicks over time. However, one limitation of the dataset is that the stories are anonymized, and no additional information regarding the stories is provided. In absence of any information on the stories, we considered the timestamp of the first click to a story as its publish time.

3.1.2 NYTimes Tweets dataset

Apart from using the user click information, we also gathered how stories published by NYTimes are shared on Twitter. NYTimes maintains several Twitter accounts (e.g., *@nytimes, @nytpolitics, @nytopinion*), from which they regularly tweet the links to the stories published at `nytimes.com`. Using the Twitter streaming API,[4] we collected the tweets made by the NYTimes accounts, and all retweets of these tweets. We also gathered the replies posted by the Twitter users who follow any of these NYTimes accounts. In total, we collected 1,026,116 posts during March 1, 2016 to April 30, 2016, and extracted links to 11,629 unique NYTimes stories. From this data, we computed the sequence of tweets (and retweets) mentioning each news story during this 2 month period.

3.2 Lifecycle of a news story: terminology

Every news story in a media site goes through different phases in its **popularity lifecycle**, where the **popularity** of a story is usually based on some crowdsourced measure of readers' interest in that story. For instance, the popularity of a story at time t can be measured as the number of views (or likes or shares) the story gets in a unit time interval around t. Figure 1 shows the lifecycle of an example story s. s appears in the media site at time t_{birth}, then receives different amounts of popularity at different time instants. Finally, its lifecycle gets over at time t_{death}, after which it does not get any more views. Thus, the **lifetime** of a story is the interval between the time instant when the story first appeared in the website and the instant when its lifecycle is over. For example, lifetime of s is $|t_{death} - t_{birth}|$.

The **lifetime-impact** of a news story is measured by the number of views (or likes or shares) that the story gets during its entire lifetime. For example, the lifetime-impact of the story s is the *area under the popularity curve* (i.e., the total area of the regions A, B and C) in Fig. 1.

Now, assume that a set of news stories are to be recommended at a particular time instant t'. The candidate set of stories comprises of all the news stories published before

[3] https://webscope.sandbox.yahoo.com.

[4] https://dev.twitter.com/streaming/overview.

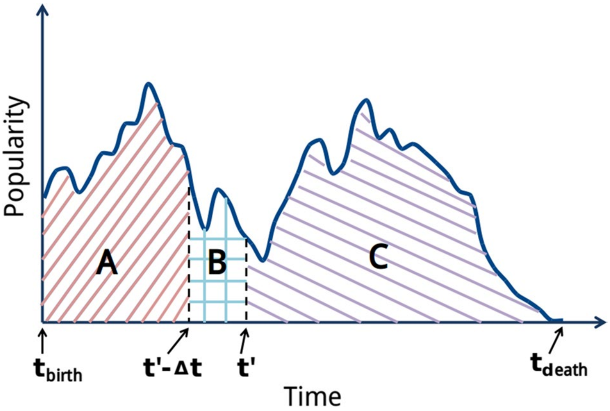


Fig. 1 Popularity lifecycle of a news story, where popularity can be measured as the number of views (or likes or shares) per unit time

time t', out of which different recommendation strategies would recommend different set of stories. The **age** of a recommended story is the difference between the time the story is published, and the time it is recommended. So, if the story s is recommended at time t', then the age of s at the time of this recommendation would be $|t' - t_{birth}|$.

To measure how a particular recommendation strategy performs in terms of recency, we consider the average age of all stories recommended by this strategy. For measuring importance, we compute average lifetime-impact of all stories recommended by this strategy. Ideally, recommended stories should simultaneously have high recency and high importance. But, as we show in the rest of the section, it is very hard to jointly optimize for recency and importance, when selecting news stories. In practice, we observe that when existing recommendation strategies perform better in one aspect, they tend to perform poorly on the other—we refer to this observation as the ***recency-importance trade-off***.

3.3 Recency-importance trade-offs in existing recommendation strategies

We now describe some broad non-personalized recommendation strategies presently deployed in news media sites. While describing the strategies, for now, we assume the existence of an ‘*oracle*’, which knows the past as well as the future popularity of every news story published in the site. That is, at time t', the oracle knows exactly how many views (or likes or shares) a story has received till t', and how many it will receive after t', throughout its entire lifetime.

3.3.1 Optimizing for recency or importance

We start by describing some simple strategies that attempt to optimize for either recency or importance.

Latest stories In this strategy, the site simply recommends the most recent stories. All stories available at time t' are ranked based on $|t' - t_{birth}|$, and then the K latest stories are recommended.

Highest lifetime-impact stories Another strategy would be to recommend stories based on the lifetime-impact of the stories, i.e., based on the total number of views (or likes / shares) a story would receive during its entire lifetime (which we assume is known by the oracle). With respect to Fig. 1, this strategy would rank all stories based on the total area

Table 1 Comparing the performances of recommending latest and highest lifetime-impact stories

Recommendation strategy	Average age	Average lifetime-impact
10 latest stories	4.15 h	1684 views
10 highest lifetime-impact stories	4.09 days	5734 views

under their popularity curves during the interval $[t_{birth}, t_{death}]$ (i.e., the combined area of regions A, B, and C), and then recommend top K stories.

Clearly, the two strategies described above are the two extremes. The strategy of recommending latest stories does not take into account the lifetime-impact of the stories, and hence might end up recommending stories which never become much popular. Whereas, recommending the highest lifetime-impact stories does not consider the recency of the stories, resulting in often recommending older stories at the end of their lifecycles.

Table 1 compares between the top 10 news stories recommended by the two extreme strategies on the Yahoo! News dataset described in Sect. 3.1. While the 10 Latest Stories have small average age (only 4.15 h), their average lifetime-impact is also relatively low (1684 views). On the other hand, the 10 Highest Lifetime-Impact Stories have much higher lifetime-impact (5734 views) but are much older (4.09 days).

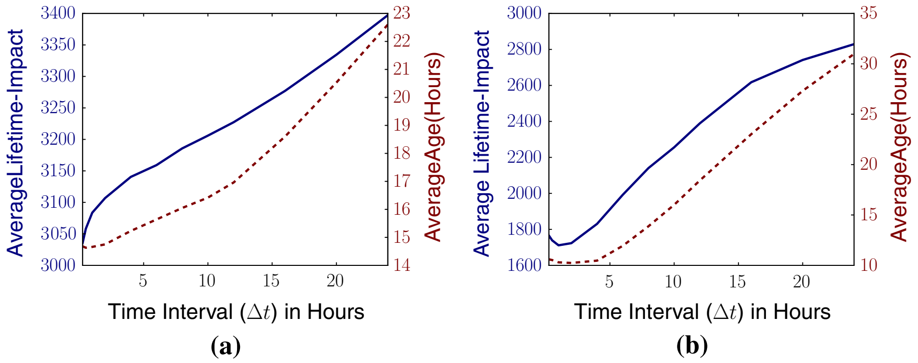
3.3.2 Trading between recency and importance

Between the above two extreme strategies, there are other strategies that attempt to balance both recency and lifetime-impact, by looking at the popularity of news stories around the time of recommendation t'. We describe two such strategies next.

Highest recent-impact stories This strategy attempts to identify the stories that have the highest popularity (e.g., most viewed, most liked, or most shared) over a certain duration of time Δt immediately before the recommendation instant t'. With respect to Fig. 1, the stories will be ranked based on the area under their popularity curves during the interval $[t' - \Delta t, t']$ (i.e., the region B), and the top K stories will be recommended. The choice of the interval Δt can vary widely, ranging from last few minutes to few hours, last 1 day, or even last 1 month. The choice of Δt can have large implications on the type of stories being recommended. If Δt is large, the recommended stories are older and the freshness of the stories are lost. Whereas, if Δt is too small, it is not clear whether popularity over Δt for a story is a good indicator of its lifetime-impact.

To bring out the implications of considering different values of Δt, Fig. 2a compares sets of top 10 stories with highest recent-impact, considering various values of Δt on the Yahoo! News data. As Δt is increased from 15 min to 24 h, the average age of the recommended stories increases, i.e., the stories gradually become less recent. But the average lifetime-impact of the stories increases, i.e., more important stories are recommended.

Trending, or highest rising-impact stories Yet another recommendation strategy is based on how the popularity is changing over a certain duration of time Δt immediately before the recommendation instant t'. This strategy is about picking the K stories having the highest derivative (over time) of the popularity, computed over the duration Δt. In other words, the stories with highest rise in popularity during the last Δt interval are recommended. Examples of this strategy include *Twitter Trending Topics* (Twitter 2010; Mathioudakis and Koudas 2010).



(a) **(b)**

Fig. 2 Comparing the performances of recommending 10 stories with **a** highest recent-impact, and **b** highest rising-impact over different time intervals Δt. The recent-impact and rising-impact of the stories are computed over 1 h (i.e., $\Delta t = 1$ h)

But even here, the choice of Δt is crucial in determining the type of stories being recommended. When Δt is large, this strategy is similar to recommending highest recent-impact stories. Whereas, if Δt is small, the recommendation strategy tends to pick *flash in the pan* stories which have high instantaneous peaks in popularity, and ignores stories gaining popularity more consistently. Figure 2b shows the average age and lifetime-impact for top 10 stories with highest rising-impact computed over different Δt on the Yahoo! News data. Similar to the case with recent-impact in Fig. 2a, even here we observe the trend of increase in the age as well as the lifetime-impact of recommended stories with increase in Δt.

Takeaway We see that different recommendation strategies attempt to balance between recency and importance in different ways. At a high level, the recency-importance trade-off always exists, and increasing one usually leads to a fall in the other. Most existing implementations of these recommendation strategies use somewhat arbitrary parameters like Δt, such as 15 min for Twitter trending topics, 1 day for NYTimes most viewed stories over the last day, and so on. But it is not clear which strategy yields the *'best'* result, as the objective of these recommendations are not explicitly stated. In this work, we argue that we can adopt a new recommendation strategy which would help to get better recency as well as higher importance. We present this strategy in the next section.

4 Highest future-impact recommendations: a new strategy

In this section, we propose a new recommendation strategy which will select stories based on their **future-impact**, i.e., *how much user attention (number of views or shares) each story is likely to receive in the future*. With respect to Fig. 1, this strategy will rank all the stories available at time t', based on the area under their popularity curves beyond time t' (i.e., the region C), and choose the top K stories to recommend. Note that, for now, we assume the presence of an *oracle* which has knowledge of the future. We will relax this assumption later, when we propose a practical implementation of the recommendation strategy in the next section.

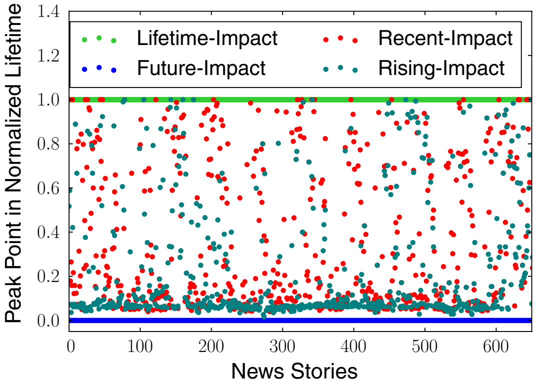


Fig. 3 The points in the normalized lifetimes where different news stories have highest values of lifetime-impact, future-impact, recent-impact and rising-impact. The recent-impact and rising-impact of the stories are computed over 1 h (Color figure online)

4.1 Why recommend stories based on highest future-impact?

We now describe two main motivations for recommending news stories having the highest future-impact.

1. *The normative argument* The effectiveness of recommending news stories based on their future-impact can be argued *normatively* using several communication theories, which consider the social aspects of reading news. Using the seminal work of Habermas et al. (1974) on the *'public sphere'*, Novendstern (2011) argued that news is a part of people's *'public discourse'*, using which they can participate in community discussions. Therefore, a reader should read those stories which will be largely discussed in future, rather than the stories which have already been discussed in the past.

On one hand, applying the *'knowledge gap hypothesis'* (Tichenor et al. 1970), we can argue that if some readers read interesting news stories ahead of their peers, such differences in knowledge acquisition help maintain the knowledge gap between different segments of the society. However, on the other hand, using the advance knowledge, such readers can play the roles of *opinion leaders* (Richins and Root-Shaffer 1988), and initiate discussions in their communities around the news stories. Such spreading of ideas from mass media to opinion leaders, and from them to the wider society, forms the basis of the *two-step flow of communication model* (Katz 1957). Therefore, a recommendation strategy *should* recommend to its readers the stories which would enable such information flow.

2. *Better recency-importance trade-off* The second motivation for recommending stories with highest future-impact comes from the perspective of recency-importance trade-off. Unlike the existing strategies which optimizes for one at the cost of the other, the Highest Future-Impact strategy can optimize for *both* recency and importance of the recommended news stories.

The future-impact of every story declines over time, from the maximum at its birth[5] to zero at its death. Hence, story selection based on its future-impact effectively captures the trade-off between its age (recency) and its lifetime-impact (importance). The strategy would like to pick stories with *high* lifetime-impact and that too *early* in their lifetimes, which is different from existing strategies of selecting stories based on their lifetime-impact, which

[5] At birth, future-impact of a story equals its lifetime-impact.

stays the same throughout a story's lifetime, or their recent-impact or rising-impact, that are *not* guaranteed to decrease over time.

To demonstrate this difference, we normalized the lifetime of every news story such that any time instant in its lifecycle would fall between 0 and 1. Then, we checked at what point in its lifecycle, the story has the highest value of recent-impact, rising-impact, and the future-impact. Figure 3 shows the highest points for different stories in the Yahoo! News dataset, where the y-axis shows the normalized lifetime of stories. We can see that the highest future-impacts for all stories are at time 0 (blue colored points at $y = 0$). Although the lifetime-impacts for all stories remain the same throughout the lifetime, it will be *fully known* only at time 1 (light green colored points at $y = 1$). Regarding recent-impact and rising-impact, different stories reach their highest values at different points of time during their life-cycle; often long after they are published, hence, the corresponding highest points are scattered throughout Fig. 3.

4.2 Comparing highest future-impact strategy with existing strategies

To compare different recommendation strategies mentioned earlier, we execute the strategies over the stories which first appeared during the initial 70% of our datasets (chronologically ordered), and received no views (or shares) during the last 10% of the data. Rest of the stories are not considered as the lifetimes of these stories may not be over; hence, it will not be possible to know the actual lifetime-impact and the future-impact values for them. We consider the lifetime of a story to be over when it does *not* receive any view (or share) during the rest of the datasets.

We execute different recommendations at every 15-min intervals over the time duration covered by the initial 70% of the datasets, and pick the top 10 stories as recommended by different strategies. We then compute the following performance metrics for the recommended stories, and the average value of these metrics are used for comparison:

1. Average age (which captures recency),
2. Average lifetime-impact (which captures importance), and
3. Average future-impact of the recommended stories.

Table 2 shows the average performance of recommending news stories according to different strategies over Yahoo! News and NYTimes Tweets datasets. Table 2 demonstrates the recency-importance trade-off. The strategy which achieves the maximum lifetime-impact (Highest Lifetime-Impact) suffers from high average age of the recommended stories, while the strategy which achieves lowest average age (Latest) has the lowest average lifetime-impact. Other strategies, like Highest Recent-Impact and Highest Rising-Impact, achieve some balance along these two metrics. However, the Highest Future-Impact strategy often achieves good performance with respect to both metrics. Additionally, the Highest Future-Impact strategy also gives stories which will get most attention in future.

5 Implementing highest future-impact recommendations

As stated earlier, we have assumed the existence of an omniscient oracle till now, which has the knowledge of the future-impact of every story. In this section, we focus on actually implementing the Highest Future-Impact recommendation strategy. To recommend the

Table 2 Comparing the performances of recommending 10 latest, highest lifetime-impact, highest rising-impact, highest recent-impact, and highest future-impact stories

Recommendation strategy	Yahoo! news			NYTimes Tweets		
	Avg age	Avg lifetime-impact	Avg future-impact	Avg age	Avg lifetime-impact	Avg future-impact
Latest	**4.72**	1729.51	1346.15	**2.28**	26.99	14.64
Highest lifetime-impact	71.79	**5211.95**	461.05	141.77	**269.72**	15.61
Highest rising-impact	10.96	1832.06	875.38	2.88	63.96	20.37
Highest recent-impact	22.93	3425.98	539.92	18.39	119.16	21.55
Highest future-impact	7.71	2841.06	**1835.76**	17.69	117.72	**30.91**

Age is measured in hours. Best values for each metric are highlighted in bold. The Δt duration for the highest rising-impact and highest recent-impact stories are taken to be 15 min and 24 h respectively stories having highest future-impact, in practice, we would need to *estimate* the future-impact of a story. Thus, we *predict the future-impact* of all stories at a particular time instant, and recommend the stories which have the highest predicted future-impact. Next, we present this strategy in detail.

5.1 Recommendations using future-impact predictions

When a news story is published at time t_{birth}, we only have the textual content and some meta-information of the story (e.g., its topical category, the event on which the story is reporting, the author of the story, and so on). As time progresses, we get the information on how the readers are interacting with the story. For example, we can divide the time starting from t_{birth} in different fixed t-sized time intervals (e.g., t can be 5, 15, 30 min or longer), and then compute its popularity (e.g., the number of views the story got) during these time intervals.

To predict the future-impact of a story, we first attempt to predict the lifetime-impact of the story. Then, the predicted future-impact can be computed as the predicted lifetime-impact, minus the number of views (or likes or shares) the story has received so far. Thus, for a given news story s, our task is to predict the lifetime-impact at time τ using the information available till time τ.

This prediction task falls under the broad class of estimating the amount of user attention for different online contents. There have been attempts to predict the user attention for Youtube videos (Figueiredo et al. 2014), Flickr images (McParlane et al. 2014) and so on. However, there is one distinction which makes the prediction for news stories different than other types of contents. The lifetimes of news stories are much smaller compared to the lifetimes of other types of contents, and due to this very nature of news, it is desirable to accurately predict the lifetime-impact as early as possible from the publish time t_{birth}, and with only a limited amount of data.

Due to this constraint, several past works on online news (Chakraborty et al. 2016b; Bandari et al. 2012; Reis et al. 2015) have attempted to predict user attention classes (e.g., whether a story is going to be viral or not) instead of predicting the exact amount of user attention. However, in our context, coarse grained user attention classes will be insufficient to give us an estimate of the lifetime-impact of stories.

Additionally, due to the limitation of the Yahoo! News data we are using in this work, we could not extract any content or meta-information for the news stories. Hence, for the sake of generality, regardless of the dataset, we attempt to predict the lifetime-impact of the news stories at time τ, only using the number of views (or shares in case of NYTimes data) that the stories received between their publish times and τ.

Specifically, for a news story s, we first compute the feature vector x_s of size m, where m is the number of 15 min intervals between t_{birth} and τ, and each feature in x_s is the number of views (or shares) s received during the corresponding interval. Then, we predict the lifetime-impact y_s using this feature vector x_s as input. We explore two methods to predict the lifetime-impact of news stories, as described next.

Method 1: ordinary least squares (OLS)

In the first method, we predict y_s assuming a linear model: $y_s = x_s^T \beta + \varepsilon_s$, where β is the vector of weights for different features including the intercept β_0, and ε_s is the random noise with zero mean and constant variance σ^2. β is then estimated by minimizing the *sum* *of squared errors* for a set of n stories (training data points), for which the lifetime-impact is known a priori (Faraway 2002). Specifically, the *least squares estimate* of β (denoted as $\hat{\beta}$) is measured as

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{2n} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 \tag{1}$$

where y_i and x_i are the lifetime-impact and the feature vector for story i respectively.

Once we get the estimated weight vector $\hat{\beta}$, then given the observed feature values x_s for story s, the predicted lifetime-impact $\hat{y}_s$ is computed as

$$\hat{y}_s = x_s^T \hat{\beta} \tag{2}$$

where $\hat{y}_s$ is the *conditional mean* $E(y_s|x_s)$.

Method 2: gradient tree boosting (GTB)

In the first method, we assume that y_s can be expressed as a linear combination of features in x_s. However, if this assumption is not valid in the real data, then the linear model will fail to capture the reality and as a result, OLS will have lower accuracy in predicting y_s. In method 2, we use non-parametric regression model *Decision Trees*, which does not assume anything about the nature of the underlying relationship between y_s and the features x_s (Breiman et al. 1984).

Although decision trees can work without having any underlying assumption of the data, Breiman (1996) showed that decision trees are *unstable* in the sense that small perturbations in the training set may result in large changes in the constructed predictor. To improve the accuracy, Breiman argued for using an ensemble of multiple decision trees (e.g., 10, 100, or 500 such trees) instead of using only one predictor, and then combining their individual predictions. Gradient Tree Boosting (GTB) (Friedman 2001) is one of the ways to do exactly that.

GTB starts with *short* decision trees (also called *weak learners*) to predict y_s, and gradually adds larger trees using a gradient descent like procedure. In each addition step, a tree is added to the model which minimizes a particular loss function computed over the training samples. The final predicted lifetime-impact $\hat{y}_s$ is computed as the weighted sum of the predictions from the sequence of trees being added. The benefit of GTB is that it can work with any differentiable loss function, e.g., least squares, least absolute deviation, etc. In this work, we particularly use least absolute deviation as the loss function for GTB.

Predicted future-impact Finally for both methods, after getting the predicted lifetime-impact of s, we compute the predicted future-impact $f_\tau(s)$ of s at time τ as

$$f_\tau(s) = \hat{y}_s - \sum_{t=t_{birth}}^{\tau} popularity_t(s) \tag{3}$$

where $popularity_t(s)$ is the number of views (or shares) obtained by s at time t.

5.2 Comparing different methods for prediction

We now compare the performance of recommending stories using the predictions made by the above two methods. We first predict the future-impact of stories using both methods.

Then, all stories are ranked based on the predicted future-impact of the stories and top *K* stories are recommended.

As explained in Sect. 4.2, we only consider the stories appearing in the initial 70% of the datasets. Among them, we use the stories in first 40% of the datasets as training, and the stories appearing in the next 30% as the test data to compare the performances. We execute different recommendations at every 15 min intervals over the test data, and compute different performance metrics on the 10 stories recommended by different strategies.

We compare the performances along the three metrics introduced earlier—(1) average age, (2) average lifetime-impact, and (3) average future-impact of the recommended stories. Table 3 shows the average performance of recommending based on the future-impact values predicted by the two methods and all other strategies mentioned in the earlier section. We can see from Table 3 that the future-impact prediction using both methods work well, yielding results comparable to the highest future-impact stories. Between the two, prediction using OLS outperforms the prediction using GTB by achieving performances closer to the strategy of recommending highest future-impact stories, only except the recency of the recommended Yahoo! news stories.

6 Measuring future-impact by mimicking editorial judgement

In non-personalized recommendations, in absence of a measure of personalized *relevance* of a news story, the recommendation algorithm has to learn the importance of different news stories from how humans select them. Traditionally, this role has been played by the expert editors of different news media organizations. Lately, with the popularity of social media, users decide what stories to reach their peers. In this paper so far, we have relied only on this user-news engagements for predicting future-impact of news stories. Next, we plan to utilize the editorial judgments to measure the future-impact. Towards that, we consider the importance of a news story as judged from the editors' *notions of newsworthiness* (Shoemaker et al. 2009). Then, we try to predict the newsworthiness of a story from observing the editorial decisions on past news data (instead of relying on the story's popularity among the users).

6.1 Datasets gathered

To measure the future-impact of news stories by observing past editorial decisions, we undertook an extensive data collection drive covering a period of 1 year: July, 2015–June, 2016. During this period, we collected all news stories appearing on The Guardian and NYTimes, using their respective APIs.[6] We also collected the stories selected by the editors for the printed newspaper everyday by scraping the corresponding webpages[7] throughout this 1 year period. In total, we gathered 90,355 Guardian and 242,125 NYTimes stories; out of which, 13,580 Guardian stories and 40,419 NYTimes stories were part of their print editions (as shown in Table 4).

[6] Guadian API is available at https://open-platform.theguardian.com and NYTimes API can be found at https://developer.nytimes.com/article_search_v2.

[7] https://theguardian.com/theguardian and https://www.nytimes.com/section/todayspaper respectively.

Table 3 Comparing the performances of two future-impact prediction methods with other recommendation strategies

Recommendation strategy	Yahoo! News			NYTimes Tweets		
	Avg age	Avg lifetime-impact	Avg future-impact	Avg age	Avg lifetime-impact	Avg future-impact
Latest	4.89	1820.66	1375.5	1.47	24.2	11.48
Highest lifetime-impact	96.7	5583.02	456.69	168.43	221.4	13.29
Highest rising-impact	12.33	1949.11	926.67	3.22	76.02	14.27
Highest recent-impact	24.01	3528.75	576.74	19.61	132.64	17.87
Highest future-impact	8.05	2968.56	1912.01	16.74	151.59	27.6
Prediction using OLS	9.09	**2781.93**	**1621.8**	**8.88**	**145.29**	**22.62**
Prediction using GTB	**8.01**	2693.55	1619.37	15.37	143.16	19.91

Bold values present the best performance among different approaches

Table 4 No. of guardian and NYTimes stories published between 1st July, 2015 and 30th June 2016

Newspaper	All stories	Stories published in newspaper
The guardian	90,355	13,580
NYTimes	242,125	40,419

6.2 Estimating importance of news story

To calculate the importance of a story, we use the gathered datasets as training, and develop a supervised binary classifier (two classes denote whether a story is selected by the editor or not), and use the predicted selection probability as the importance score. To some extent, this score reveals the newsworthiness of the story. We use the following features for the classifier:

1. Abstract/summary of a story,
2. Name(s) of its author(s),
3. List of topics (or keywords) describing the story,
4. The section or category of the news (e.g., politics, sports), and
5. Number of stories on same topic(s) published in the last 7 days.

The classifier works in two stages. As the first four features listed above are textual features, we train four text classifiers for each of them in the first stage. Then, we use the output of these individual classifiers to train another classifier at the second stage. More specifically, we use the predicted probabilities for selected/not-selected classes as features for the second stage classifier. As the textual classifiers, we use Convolutional Neural Networks (CNN) based classifier for feature (1), and three Naive Bayes (NB) classifiers for features (2), (3) and (4).

Finally, a SVM classifier (with RBF kernel) is used at the second stage. Thus, the SVM classifier effectively uses nine numeric features—predicted probailities from the textual classifiers and the number of similar stories (after appropriate scaling). A story's importance score is then measured as the curation probability predicted by this SVM classifier (using the method proposed by Lin et al. (2007)).

The CNN architecture for the textual classifier over the abstract is similar to that used in Kim (2014), where every abstract is converted to a $m \times n$ matrix (m is the maximum abstract length, and $n = 50$ is the word vector dimension). A convolution operation is applied to every possible window of h words to produce a feature map. We then apply a max over time pooling operation over the feature map and take the maximum value as a feature. Multiple features are obtained by varying the value of h. These features form the penultimate layer and are passed to a fully connected softmax layer whose output gives the probability distribution over the selected/not-selected classes.

We experimented with different combinations for the classifier described above. For example, we tried applying Naive Bayes classifier instead of CNN for classifying the abstract, but we got worse results (prediction accuracy 68% for Naive Bayes compared to 72% for CNN). We could not replace Naive Bayes with CNN for other textual classifiers, because the amount of data to classify is very small (author names, keywords all are just a few words). Similarly, we tried different classification models (e.g., SVM, Random Forest, Decision Tree) as the second stage classifier, and found the proposed combination to work best.

6.3 Measuring future-impact of a story

While estimating importance using popularity signals, we could easily get the future-impact by subtracting the amount of views/shares a story has received so far. However, in the current context, where we estimate importance by mimicking past editorial decisions, we measure the future-impact of a story as a combination of its recency and importance.

Recency of a story i is measured as the difference between the recommendation time and the publish time of the story.

$$recency_i = \frac{1}{time\ since\ i\ is\ published} \tag{4}$$

where the time difference can be computed in seconds, minutes or hours depending on the particular recommendation. We then normalize $recency_i$ score of a story using the score of the most recent story published.

$$normalized\_recency_i = \frac{recency_i}{max(\{\forall i\ recency_i\})} \tag{5}$$

After computing the recency and importance scores for a story, we compute its future-impact (ϕ_i) as a combination of these scores:

$$\phi_i = importance_i \cdot normalized\_recency_i{}^{\lambda} \tag{6}$$

where λ is a hyper parameter, which controls the decay in future-impact with time. If $\lambda \geq 1$, the future-impact decreases rapidly as time progresses; whereas, $\lambda < 1$ represents much slower decay.

7 Temporal coverage bias in the recommended news

In the earlier sections, we proposed and implemented a new non-personalized news recommendation strategy of recommending stories based on their future-impact. Next, we try to explore its effect on the topical coverage of the recommended news. Because future-impact strategy tries to optimize both recency and importance, a new important story will replace an old story in the recommendation, creating *churn* in the recommended news. Specifically, if the recommended stories are changing throughout the day, and a reader is browsing the media site only a few times a day, she might get stories having topical coverage different from some other reader browsing the website at different times. We investigate such biases in this section.

7.1 Dataset gathered

To study the temporal coverage bias, we select a particular recommendation strategy deployed at the NYTimes website—'Top Stories', where the recommended stories are conceptually similar to having stories with high future-impact. To enable the readers to take a quick look at the most important news stories at a particular time, NYTimes editors

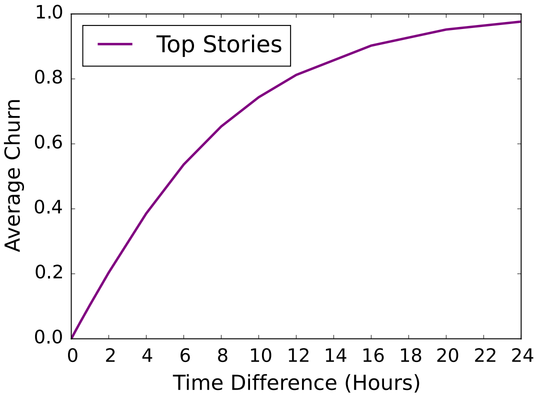


Fig. 4 Average churn in the NYTimes top stories

highlights around 20 stories in the top of the homepage. We denote these set of stories as 'Top Stories'. We undertook an extensive data collection drive to collect all news stories appearing on NYTimes Top Stories during a period of 8 months, July, 2015–February, 2016, by querying the Top Stories API[8] at every 5-min intervals throughout this 8 month period. Overall, we collected 10, 348 distinct news stories. Further, NYTimes Article Search API[9] provides detailed metadata regarding every news story (e.g, its author, headline, summary, section and the topics assigned to it by NYTimes). We gathered these metadata for all top stories, which we use extensively to analyze the coverage of news stories.

7.2 How quickly do top stories change?

First, we compute *the rate at which the list of top stories is changing at NYTimes*. If the list is fairly static, the churn-rate (i.e., the rate at which the recommended news stories is changing) will be very low. As a result, readers will receive similar information regardless of the time they are visiting NYTimes. However, if the churn-rate is high, then the readers browsing NYTimes at different times of the day will consume very different sets of stories.

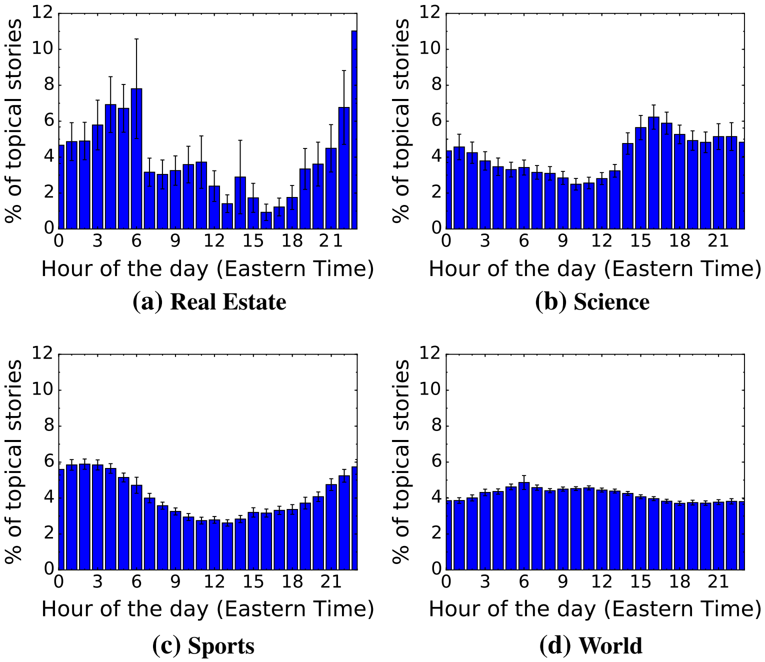
To compute the churn-rate in the top stories, we measure the fraction of *non-overlapping* stories between every pair of recommendation lists separated by time *t* in our dataset, where *t* varies from 15 min to 12 h. Figure 4 shows the average churn in top stories at NYTimes, and we can see that the churn is so high that two readers visiting NYTimes at 12 h time differences, would receive sets of stories that differ by almost 80%. In other words, these two readers, on average, would read 16 different stories, out of the 20 top stories recommended. Next, we look at how the churn in the top stories can affect the information consumption by the readers.

7.3 Diurnal pattern in the sectional coverage of top stories

To help the readers to easily navigate through the vast collection of news stories published, news organizations (e.g., NYTimes) assign a story to a particular *news section*, such as 'Business', 'Sports', or 'Arts'. To check whether there is indeed any *temporal bias* in the

[8] https://developer.nytimes.com/top_stories_v2.

[9] https://developer.nytimes.com/article_search_v2.



(a) Real Estate

(b) Science

(c) Sports

(d) World

Fig. 5 How top stories cover stories of different sections at different hours of the day. The error bars represent standard errors

sectional coverage of top stories, we consider all top stories from a particular section, and compute how these stories are recommended *at different hours of a day* during our measurement period. This distribution will be nearly uniform for a section if it gets uniformly covered throughout the day.

Figure 5 shows the hourly distribution of top stories on some sections. We see that while some topics are covered uniformly throughout the day (e.g., 'World' in Fig. 5d), there are several other topics which have huge diurnal variations (Fig. 5a–c). Especially for some niche topics like 'Real Estate' (Fig. 5a), there are several time periods in a day where there are very few articles on that topic among the top stories. Thus, if a reader is browsing the site at specific hours everyday, she might be missing the niche topics which do not get recommended at these hours.

7.4 Temporal bias in news coverage of readers

As different sections get non-uniform diurnal coverage in the top stories, the pertinent question to ask is *how readers' news consumption can get affected by these diurnal variations in coverage*. We use the notion of 'information diet' (Kulshrestha et al. 2015) to address this question. A reader's information diet is computed as the composition of all stories consumed by her.

To characterize the differences in information diets, we consider different readers who browse NYTimes regularly during different hours of a day. For simplicity, we assume that a reader browsing NYTimes at a particular hour will read all the content

Fig. 6 Overlap between the set of top stories appearing at different times of a day

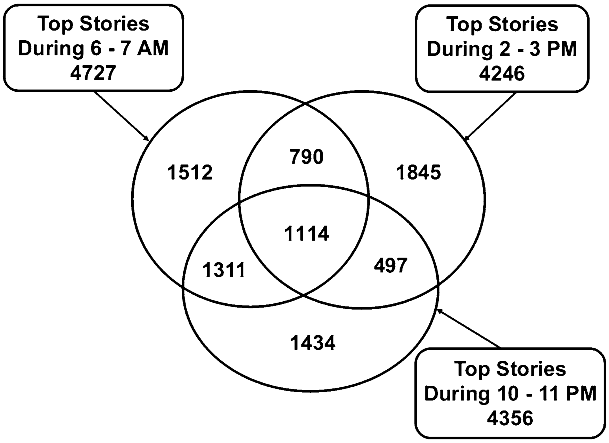

Top Stories During 6 - 7 AM 4727
Top Stories During 2 - 3 PM 4246
Top Stories During 10 - 11 PM 4356
1512
790
1845
1114
1311
497
1434

Table 5 Difference in information diet for readers who regularly browse top stories at different hours of the day

Topic	At 6–7 a.m.	Change during 2–3 p.m. (%)	Change during 10–11 p.m. (%)
Food	0.25	+ 240	+ 156
The upshot	1.1	+ 189.09	− 6.36
Real estate	0.42	− 71.43	− 33.33
Magazine	1.4	+ 47.86	+ 21.43
New York	13.86	− 37.16	− 15.66
Science	1.4	+ 36.43	+ 10.0
Health	1.31	+ 35.11	+ 31.3
Technology	1.52	+ 34.87	+ 32.89
Movies	1.23	+ 32.52	− 13.82
Travel	0.61	− 31.15	− 27.87
Sports	6.6	− 27.27	+ 3.64
Fashion	2.45	− 11.43	− 23.27
U.S.	14.62	− 9.3	+ 14.5
World	22.28	+ 9.74	− 7.45
Business	9.08	+ 8.7	+ 3.19
Politics	14.13	+ 4.53	+ 14.37

Changes in the sectional proportions are relative to their shares during 6–7 a.m.

recommended as top stories during that 1-h period. Lets suppose there are three news readers who habitually visit NYTimes either in the morning (6–7 a.m.), in the afternoon (2–3 p.m.), or in the night (10–11 p.m.). Figure 6 shows the overlap between the top stories consumed by these readers over our measurement period. We can see that even though all the readers are browsing the site during the same day, they would get only 25% stories in common, and the sets of stories they read would be very different from each other. For instance, over the 6 month period, the reader who habitually visits NYTimes during 6–7 a.m. would read around 60% different stories compared to the reader visiting at 2–3 p.m.

Table 6 Time-lag (hours) for news from different sections to become top stories (sorted on increasing order of median time)

Topic	Time to become top stories (h)		
	25th percentile	Median	75th percentile
New York	0.27	0.49	1.71
Politics	0.33	0.55	1.37
U.S.	0.31	0.55	1.59
World	0.33	0.64	2.5
Sports	0.38	1.1	3.72
Upshot	0.58	1.56	4.0
Arts	0.52	1.72	5.05
Movies	1.06	3.22	9.51
Fashion	1.13	7.48	21.82
Education	4.57	10.93	19.07
Magazine	4.47	11.58	29.52
Real estate	8.25	16.7	38.16

To understand whether these huge differences in the top stories consumed by different readers induce a bias in their information diets, we compute the topical composition of the news stories recommended at the hours of their visits. Table 5 shows how different readers visiting at different times will cover different topics in such largely different proportions, and hence have significantly different information diets. For example, as shown in Table 5, the reader browsing NYTimes during 6–7 a.m. every day will receive more 'New York' or 'Sports' related news, compared to the reader browsing during 2–3 p.m., who will read more stories from 'Food' or 'The Upshot' section.

We see that the variation in readers' diet, depending on their browsing time, is relatively less for topics of broad interest (e.g., 'Politics' or 'World') since these topics get recommended uniformly throughout the day. However, the variation is substantially higher for the niche topics like 'Science', 'Health' or 'Real Estate'. Therefore, for readers who don't have any specific topical interests, the temporal variation in the coverage of news stories on these niche topics can lead to significant imbalances in their information diets.

7.5 Potential sources of the bias

To understand the possible reasons for which the coverage of certain topics is different in different hours, we looked at the times of day stories on different topics get published at NYTimes. We observed that the stories on certain topics (e.g. 'World') get published throughout the day, whereas for other topics, news stories are published only during certain time-periods.

Further, depending on how the editors perceive the importance of *recency* in some type of stories, they might be recommended as top stories immediately after being published, or after some time-lag. Table 6 shows the 25th percentile, median and 75th percentile of the time-lag between the publish time of stories and the time at which the stories get recommended as top stories, for the various topics. We see that stories on topics like 'Politics', 'U.S.' or 'World' gets recommended within half-an-hour from being published; whereas, stories on 'Arts', 'Movies' or 'Fashion' take hours to get recommended. Thus, the bias in reader diets can be attributed to the complex correlation between the differences in publish times for different topics and the duration for which editors wait to pick them and recommend as top stories.

To reduce this bias, in the next section, we propose mechanisms to inculcate diversity in recommended news such that the recommendation can cover different topics and also have definite compositions of news from different sections, regardless of the time of recommendation.

8 Inculcating diversity in recommended news

Apart from the temporal bias identified in the last section, recommending stories based only on their future-impact values can overwhelm the recommendation stream with similar news. This is especially true for the *fast news days* when some important events occur. For example, in a day with a major political development or a natural disaster, almost all highest future-impact stories will cover the same event, because they will be both important and most recently published. Thus, the recommendation may potentially include only those stories, and nothing else.

To avoid such a situation, a recommendation system should also introduce topical diversity in the recommended news. A news story potentially covers a number of different entities (e.g, persons, locations, business organizations, etc.). For a set of stories, its diversity captures how many unique topics are covered by the set. More formally, we measure the diversity of a recommendation by the following function *f*(*S*) over the set of recommended stories *S*.

$$f(S) = \sum_{i \in S} \left(\phi_i \cdot \sum_{t \in \tau_i} \frac{1}{freq_t} \right) \tag{7}$$

where τ_i is the list of topics covered by *i*, and $freq_t$ is the number of stories in *S* which cover topic *t*.

To ensure topical diversity, a recommendation system should try to maximize *f*(*S*) while selecting the set *S* of news stories for recommendation:

$$maximize\, f(S) \tag{8}$$

subject to

$$|S| \le K \tag{9}$$

where *K* is the number of stories to be recommended. Before solving Eq. (8), we observe some properties of *f*(*S*).

Theorem 1 *A set function F is* **monotone and submodular** *iff for all articles a and sets* $A \subseteq B$:

1. $F(A \cup \{a\}) \ge F(A)$
2. $F(A \cup \{a\}) - F(A) \ge F(B \cup \{a\}) - F(B)$

The first condition is the condition for **monotone**, which denotes that if we add a new item to a set, the utility of the set either increases or remains the same, but doesn't decrease. The second condition is the condition for **submodularity**, which says that the benefit of adding an item to a smaller set is larger than the benefit of adding the item to a larger set. This property is also known as *diminishing returns*.

It can be easily seen that *f*(*S*) is *both monotone and submodular*, since the gain of adding a second article covering similar topics is smaller than the gain of adding the first, and *f*(*S*) doesn't decrease with the addition of a new article. Thus, solving Eq. (8) maps to maximizing submodular functions w.r.t *cardniality constraints*, and such maximization has been proved to be NP-Hard (Feige et al. 2011). We implement the $\frac{1}{3}$-*approximation* algorithm proposed in Feige et al. (2011) to solve Eq. (8). Intuitively, we first build *S* by taking *K* stories with highest ϕ_i scores. Then, we update *S* if removing a story from *S* and adding another story from outside *S* improves the overall diversity score. This process is repeated until no further change in *S* is possible.

8.1 Maintaining sectional composition

The above formulation does not take into account any constraint on the distribution of different sections in the selected news stories (similarly any distribution of hard vs soft news). If there are *J* sections, and typically a media site produces b_j fraction of stories for section *j*, then a recommendation designer may like to select the stories for recommendation such that for each section *j*, $b_j \cdot K$ stories are recommended.

We are going to express these constraints using **Matroids**, a combinatorial structure that generalizes the notion of linear independence in matrices (Schrijver 2002). A matroid can be defined as follows:

Theorem 2 *A matroid is a pair*, M = (Z, I), *defined over a finite set (the ground set) Z and a family of sets (the independent sets) I, that satisfies the following three axioms*:

1. **Non-emptiness** *The empty set* $\emptyset \in I$.
2. **Heredity** *If* $Y \in I$ *and* $X \subset Y$, *then* $X \in I$.
3. **Exchange** *If* $X \in I; Y \in I$ *and* $|Y| > |X|$, *then there exists* $z \in Y \setminus X$ *such that* $X \cup \{z\} \in I$.

There is a particular type of matroids, known as *Partition Matroid*, which is of interest to us in the current scenario. In partition matroid, the ground set Z is partitioned into disjoint subsets $Z_1, Z_2, ..., Z_Q$ for some *Q*, and

$$I = \{S \mid S \subseteq Z \textit{ and } |S \cap Z_q| \leq u_q,\ \forall q = 1, 2, ..., Q\} \tag{10}$$

for some values of $u_q;\forall q$.

In our context, Z is the set of all stories worthy of selection. $Q = J$, i.e., there are *J* sections and each of the stories belong to only one of these sections, which in effect partitions the set of news stories Z into J disjoint sets. We can then define a partition matroid $M' = (Z, I')$ according to the desired sectional composition, such that

$$I' = \{S \mid S \subset Z \textit{ and } |S \cap Z_j| \leq (b_j \cdot K),\ \forall j = 1, 2, ..., J\} \tag{11}$$

Then the problem of selection of news stories can be formulated as

$$maximize_{S \subseteq Z}\, f(S) \tag{12}$$

subject to

$$S \in I' \tag{13}$$

Table 7 Accuracy (Acc), Precision (P) and Recall (R) in predicting the editorial decision of selecting stories for next day's newspaper

Dataset	The guardian			NYTimes		
Approach	Acc	P	R	Acc	P	R
Most recent	0.747	0.180	0.180	0.639	0.066	0.086
Most diverse	0.688	0.083	0.106	0.648	0.086	0.013
Most important	0.737	0.415	0.651	0.823	0.605	0.614
Highest future-impact	0.815	0.528	0.652	0.866	0.776	0.787
Future-impact + diversity	0.823	0.609	0.723	0.917	0.827	0.798
Future-impact + diversity + sectional composition	**0.841**	**0.627**	**0.742**	**0.923**	**0.847**	**0.806**

Bold values present the best performance among different approaches

We have already shown that $f(S)$ is submodular. Hence, solving Eq. (12) now translates into maximizing a submodular function with matroid constraints, which is known to be NP-Hard (Du et al. 2013). A prior work by Du et al. (2013) has proposed an approximate solution with provable guarantees (see Du et al. 2013 for details). In this work, we utilize the method proposed in Du et al. (2013) to solve Eq. (12), where first the stories are sorted based on their future-impact values (ϕ_i) and then we pick K stories according to the sorted order to form S which simultaneously maintain the matroid property. We update S if removing a story from S and adding another story from outside S improves the overall diversity score but doesn't violate the partition matroid property. This process is repeated until no further change in S is possible.

8.2 Experimental evaluation

We compare our proposed diversity inclusion methods with several baselines:

1. Most recent stories,
2. Most important stories,
3. Most diverse stories (proposed in Abbassi et al. 2013), and
4. Stories with highest future-impact values.

To compare the performance of different methods, we consider the selection of stories for the daily newspaper of The Guardian and NYTimes from 1st January, 2016 to 30th June, 2016. For each day, we consider all stories published in last 3 days as the candidate set, and different methods would predict which stories made it to the print edition. Training data is selected on a sliding basis, i.e., to make a prediction for the newspaper on day m, we consider last six month's data upto day $m-3$ as training. We also estimate the sectional distribution of news stories in the print edition by taking the average fraction of them in this six month training data.

Table 7 shows the results of each of these approaches. We notice that only considering most recent, most important or most diverse articles result in poor precision and recall. Considering future-impact achieves considerable performance gains. However, as we can observe in Table 7, our proposed approaches perform better for both datasets by capturing all three salient aspects of editorial curation-recency, importance and diversity of the stories, with maintaining the sectional composition in the recommended stories providing best performance.

9 Conclusion

Online news media sites are increasingly deploying automated recommender systems to constantly update their audience with important and breaking news stories. In this paper, we focused on the fundamental tension faced by any recommendation strategy between choosing the *most recent* stories versus the *most important or impactful* stories. In personalized recommendation, the importance or relevance is measured based on how a story can appeal to individual user's interests. However, in non-personalized recommendation, due to the absence of any personalized interests, the system needs to recommend stories interesting to a broader group of users. One option is to learn from the editorial judgments (i.e., the way editors have selected stories in print newspapers for years). The other option is to observe how popular different stories are among the audience, and utilize this popularity signal for estimating importance. We considered both these options in this paper.

We conducted a systematic analysis of the recency-importance trade-offs achieved by the currently deployed recommendation strategies. After inferring the reasons for their poor performance, we proposed a simple yet previously overlooked strategy of recommending stories based on their *future-impact*. We developed practical implementation of the future-impact based recommendation strategy, using both editorial judgment and audience-driven popularity in predicting the future-impact of stories.

While evaluations suggested that these approaches can optimize for both recency and importance, focusing only on these two factors has undesirable consequences. We observed that during *fast news days*, stories on same topic can overwhelm the recommendation streams since they will be both recent and important. Moreover, due to the complex correlation between when the news stories on a particular section are generated, and when the users access the news website, different users visiting media sites at different times of a day will end up having different sectional exposure. To counter these biases, we further developed approaches to incorporate both topical and sectional diversity in the recommended news.

Although, we have tried to utilize data from a variety of mainstream media outlets such as Yahoo! News, NYTimes or The Guardian, there can be niche media websites with different user engagement patterns or editorial selection strategies. While our proposed method will work regardless of the type of media site, the analysis results may not be exactly the same. Similarly, there are other forms of user-news engagements (such as commenting on the stories), which we have not explored in this work. Our future work lies in investigating whether considering other features leads to increase in the recency of the recommended news stories. Finally, in this paper, we did not make any explicit attempt to consider the quality of the recommended stories. That would be another dimension for future works.

Acknowledgements Open access funding provided by Max Planck Society. This research was supported in part by a European Research Council (ERC) Advanced Grant for the project "Foundations for Fair Social Computing", funded under the European Union's Horizon 2020 Framework Programme (Grant Agreement No. 789373). A. Chakraborty was a recipient of Google India PhD Fellowship and Prime Minister's Fellowship Scheme for Doctoral Research, a public-private partnership between Science and Engineering Research Board (SERB), Department of Science and Technology, Government of India and Confederation of Indian Industry (CII).

References

Abbassi, Z., Mirrokni, V. S., & Thakur, M. (2013). Diversity maximization under matroid constraints. In *ACM KDD*.

Agarwal, D., Chen, B. C., Elango, P., & Wang, X. (2011). Click shaping to optimize multiple objectives. In *ACM KDD*.

Bandari, R., Asur, S., & Huberman, B. A. (2012). The pulse of news in social media: Forecasting popularity. In *AAAI ICWSM*.

Bao, P., Hecht, B., Carton, S., Quaderi, M., Horn, M., & Gergle, D. (2012). Omnipedia: Bridging the Wikipedia language gap. In *ACM SIGCHI*.

Benevenuto, F., Rodrigues, T., Cha, M., & Almeida, V. (2009). Characterizing user behavior in online social networks. In *ACM IMC*.

Bertsekas, D. P. (2014). *Constrained optimization and Lagrange multiplier methods*. Cambridge: Academic Press.

Breiman, L. (1996). Bias, variance, and arcing classifiers. Statistics.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. Boca Raton: CRC Press.

Chakraborty, A., Ghosh, S., Ganguly, N., & Gummadi, K. P. (2016a). Dissemination biases of social media channels: On the topical coverage of socially shared news. In *AAAI ICWSM*.

Chakraborty, A., Paranjape, B., Kakarla, S., & Ganguly, N. (2016b). Stop clickbait: Detecting and preventing clickbaits in online news media. In *ACM/IEEE ASONAM*.

Chakraborty, A., Ghosh, S., Ganguly, N., & Gummadi, K. P. (2017a). Optimizing the recency-relevancy trade-off in online news recommendations. In *Proceedings of WWW*.

Chakraborty, A., Messias, J., Benevenuto, F., Ghosh, S., Ganguly, N., & Gummadi, K. P. (2017b). Who makes trends? understanding demographic biases in crowdsourced recommendations. In *AAAI ICWSM*.

Chakraborty, A., Luqman, M., Satapathy, S., & Ganguly, N. (2018). Editorial algorithms: Optimizing recency, relevance and diversity for automated news curation. In *Companion Proceedings of WWW*.

Clear, J. (2015). Stop overdosing on celebrity gossip, the news, and low quality information. https://jamesclear.com/brain-food. Accessed Dec 2018.

Crane, R., & Sornette, D. (2008). Robust dynamic classes revealed by measuring the response function of a social system. *PNAS*, *105*, 15649–15653.

Deb, K. (2014). Multi-objective optimization. In *Search methodologies*. Springer.

Du, N., Liang, Y., Balcan, M., & Song, L. (2013). Continuous-time influence maximization for multiple items. CoRR. arXiv:abs/1312.2164.

Duarte, F., Mattos, B., Bestavros, A., Almeida, V., & Almeida, J. (2007). Traffic characteristics and communication patterns in blogosphere. In *ICWSM*.

Faraway, J. (2002). Practical regression and anova using r.

Feige, U., Mirrokni, V. S., & Vondrak, J. (2011). Maximizing non-monotone submodular functions. *SIAM Journal on Computing*, *40*(4), 1133–1153.

Figueiredo, F., Almeida, J. M., Gonçalves, M. A., & Benevenuto, F. (2014). On the dynamics of social media popularity: A YouTube case study. *ACM Transactions on Internet Technology*, *14*, 24.

Friedman, J. (2001). Greedy function approximation: A gradient boosting machine. *Annals of statistics*, *29*, 1189–1232.

Golder, S., Wilkinson, D., & Huberman, B. (2007). Rhythms of social interaction: Messaging within a massive online network. In *ICCT*.

Graham, M., & Zook, M. (2013). Augmented realities and uneven geographies: Exploring the geolinguistic contours of the web. *Environment and Planning A*, *45*(1), 77–99.

Habermas, J., Lennox, S., & Lennox, F. (1974). The public sphere: An encyclopedia article. *New German Critique*, *3*, 49–55.

Hecht, B., & Gergle, D. (2010). The tower of babel meets web 2.0: User-generated content and its applications in a multilingual context. In *ACM SIGCHI*.

Hong, L., Convertino, G., & Chi, E. H. (2011). Language matters in twitter: A large scale study. In *ICWSM*.

Katz, E. (1957). The two-step flow of communication: An up-to-date report on an hypothesis. *Public opinion quarterly*, *21*(1), 61–78.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Empirical methods in natural language processing (EMNLP)*.

Kulshrestha, J., Zafar, M. B., Noboa, L. E., Gummadi, K. P., & Ghosh, S. (2015). Characterizing information diets of social media users. In *ICWSM*.

Lehmann, J., Gonçalves, B., Ramasco, J. J., & Cattuto, C. (2012). Dynamical classes of collective attention in twitter. In *WWW*.

Li, L., Chu, W., Langford, J., & Wang, X. (2011a). Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *ACM WSDM*.

Li, L., Wang, D., Li, T., Knox, D., & Padmanabhan, B. (2011b). Scene: A scalable two-stage personalized news recommendation system. In *ACM SIGIR*.

Li, L., Wang, D. D., Zhu, S. Z., & Li, T. (2011c). Personalized news recommendation: A review and an experimental investigation. *Journal of Computer Science and Technology*, *26*(5), 754.

Liang, H., Xu, Y., Tjondronegoro, D., & Christen, P. (2012). Time-aware topic recommendation based on micro-blogs. In *ACM CIKM*.

Lin, H. T., Lin, C. J., & Weng, R. C. (2007). A note on platt's probabilistic outputs for support vector machines. *Machine Learning*, *68*(3), 267–276.

Liu, J., Dolan, P., & Pedersen, E. R. (2010). Personalized news recommendation based on click behavior. In *ACM IUI*.

Maksai, A., Garcin, F., & Faltings, B. (2015). Predicting online performance of news recommender systems through richer evaluation metrics. In *RecSys*.

Mathioudakis, M., & Koudas, N. (2010). Twittermonitor: Trend detection over the twitter stream. In *ACM SIGMOD*.

McParlane, P. J., Moshfeghi, Y., & Jose, J. M. (2014). Nobody comes here anymore, it's too crowded; predicting image popularity on flickr. In *ACM ICMR*.

Mitchell, A., Gottfried, J., Kiley, J., & Matsa, K. E. (2014). Social media, political news and ideology. https://pewrsr.ch/1tJAhMi. Accessed Dec 2018.

Muchnik, L., Aral, S., & Taylor, S. J. (2013). Social influence bias: A randomized experiment. *Science*, *341*, 647–651.

Novendstern, M. (2011). Why do we read the news? Harvard Political Review. http://harvardpolitics.com/online/hprgument-blog/why-bother-to-read-the-news/. Accessed Dec 2018.

Reis, J., Benevenuto, F., Vaz de Melo, P., Prates, R., Kwak, H., & An, J. (2015). Breaking the news: First impressions matter on online news. In *ICWSM*.

Richins, M. L., & Root-Shaffer, T. (1988). The role of evolvement and opinion leadership in consumer word-of-mouth: An implicit model made explicit. *North American Advances in Consumer Research*, *15*, 32–36.

Salganik, M. J., & Watts, D. J. (2008). Leading the herd astray: An experimental study of self-fulfilling prophecies in an artificial cultural market. *Social Psychology Quarterly*, *71*, 338–355.

Schrijver, A. (2002). *Combinatorial optimization: Polyhedra and efficiency* (Vol. 24). Berlin: Springer.

Shoemaker, P. J., Vos, T. P., & Reese, S. D. (2009). *Journalists as gatekeepers. The handbook of journalism studies* (Vol. 73).

Tichenor, P. J., Donohue, G. A., & Olien, C. N. (1970). Mass media flow and differential growth in knowledge. *Public Opinion Quarterly*, *34*(2), 159–170.

Twitter. (2010). To trend or not to trend. https://blog.twitter.com/2010/to-trend-or-not-to-trend. Accessed Dec 2018.

Vaughan, L., & Thelwall, M. (2004). Search engine coverage bias: Evidence and possible causes. *Information Processing and Management*, *40*(4), 693–707.

Watanabe, K., Ochi, M., Okabe, M., & Onai, R. (2011). Jasmine: A real-time local-event detection system based on geolocation information propagated to microblogs. In *ACM CIKM*.

Yan, R., Tang, J., Liu, X., Shan, D., & Li, X. (2011). Citation count prediction: Learning to estimate future citations for literature. In *CIKM*.

Yasseri, T., Sumi, R., & Kertész, J. (2012). Circadian patterns of wikipedia editorial activity: A demographic analysis. *PLoS ONE*, *7*(1), e30091.

Yu, X., Gu, Q., Zhou, M., & Han, J. (2012). Citation prediction in heterogeneous bibliographic networks. In *SIAM ICDM*.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.