# Representing Documents Through Their Readers

Khalid El-Arini*
Facebook
kelarini@fb.com

Min Xu
Carnegie Mellon University
minx@cs.cmu.edu

Emily B. Fox
University of Washington
ebfox@uw.edu

Carlos Guestrin
University of Washington
guestrin@cs.washington.edu

## ABSTRACT

From Twitter to Facebook to Reddit, users have become accustomed to sharing the articles they read with friends or followers on their social networks. While previous work has modeled what these shared stories say about the user who shares them, the converse question remains unexplored: what can we learn about an article from the identities of its likely readers?

To address this question, we model the content of news articles and blog posts by attributes of the people who are likely to share them. For example, many Twitter users describe themselves in a short profile, labeling themselves with phrases such as "vegetarian" or "liberal." By assuming that a user's labels correspond to topics in the articles he shares, we can learn a labeled dictionary from a training corpus of articles shared on Twitter. Thereafter, we can code any new document as a sparse non-negative linear combination of user labels, where we encourage correlated labels to appear together in the output via a structured sparsity penalty.

Finally, we show that our approach yields a novel document representation that can be effectively used in many problem settings, from recommendation to modeling news dynamics. For example, while the top politics stories will change drastically from one month to the next, the "politics" label will still be there to describe them. We evaluate our model on millions of tweeted news articles and blog posts collected between September 2010 and September 2012, demonstrating that our approach is effective.

## Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

## Keywords

document modeling, Twitter, structured sparsity

*Work done while at Carnegie Mellon University.


*KDD '13,* August 11–14, 2013, Chicago, Illinois, USA.


## 1. INTRODUCTION

In today's world, it has become commonplace for readers to share news articles and blog posts with their friends and followers on social networking sites. Understanding that much of their future success depends on such traffic, news sites and blogs have made it easy for their readers to share articles they find interesting, from the ubiquitous "share" buttons alongside news content, bearing the logos of Facebook, Twitter and others, to so-called "social reader" apps built directly into Facebook. *The Guardian* newspaper, for example, recently announced that, for the first time, more visits to their site were coming through Facebook than through Google search.[1] This user behavior gives us an unprecedented chance to study the readers of news articles at a large scale by analyzing their public digital footprint.

In the past, work has been done that uses such data in the setting of personalized news, recommending articles to readers based on previous articles that they may have shared or liked [8, 11, 16]. However, in this paper, we seek to investigate a different question: rather than modeling a reader by the articles she shares, what can we instead learn *about an article* from the attributes of its readers? Specifically, can we build a valuable, general purpose document representation by representing new articles—never before seen or shared—with the predicted attributes of their likely readers?

To address this question, we utilize the microblogging site Twitter as a testbed, as it is widely used by readers as a public medium for disseminating articles and interesting links. In particular, Twitter users share articles by *tweeting* them. Moreover, many Twitter users also describe themselves in a short profile description, using words like "vegetarian" or "runner" (Figure 1). (Following the convention of previous work [10], we will refer to these user attribute labels as *badges*.) As most Twitter profiles are public, we can thus scan millions of tweets to learn the relationship between articles and the badges of users who share them.

To look at an article through the lens of its readers, one could directly analyze the profiles of all Twitter users who have shared the article. This approach, however, is impossible to extend to articles not shared extensively on Twitter. We thus take the more general approach of associating badges with the *content of the articles* rather than with the articles themselves. Specifically, we learn a sparse dictionary from a vast collection of tweeted news articles; each column in the dictionary—a weight vector over the vocabulary—

[1] `http://www.guardian.co.uk/gnm-press-office/changing-media-summit-tanya-cordrey`

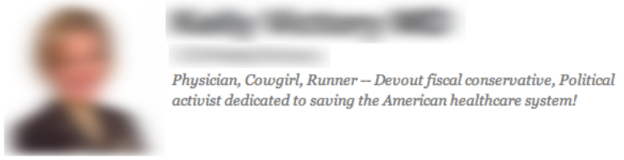



Figure 1: Example of a Twitter user profile. Labels such as "physician," "fiscal," or "healthcare" all describe the user's interests, and we refer to these as *badges*.

corresponds to a specific badge. For example, if users who have the word "vegetarian" in their Twitter profiles often share articles about health food, then we might learn that a "vegetarian" badge is associated with high weights on the words "tofu" and "kale." By learning such a labeled dictionary, we can use badges to represent new articles.

Modeling article content through user attributes, in contrast with user-oblivious approaches such as latent Dirichlet allocation (LDA) [4], offers a more interpretable representation of the articles for personalization and recommendation algorithms. For example, in content-based filtering, user preferences are commonly represented as weight vectors over a vocabulary of textual features, such as words or topics from a topic model [11, 24]. By using badges as these textual features, we expect to obtain a more natural representation of user preferences: vegetarian readers can be *directly* described by a "vegetarian" badge, rather than by potentially less focused topics from an arbitrary topic model.

Likewise, for personalization applications, a badge-based representation occurs at a more appropriate level of granularity than lower-level word-based representations, such as tf-idf. We see this in Figure 2, where we show both word-based and badge-based representations of an article from *The Guardian*.[2] This article is about a particular militant group operating on the Afghanistan-Pakistan border, and in Figure 2a, we see that the most important words in this document correspond to the name of this network: the *Haqqani group*. While informative, such a representation of the article is likely too specific; we expect a reader of this article to be broadly interested in Afghanistan and Pakistan, and not just singularly focused on the Haqqani group.

Another advantage of using badges to represent articles is that by associating the relatively stationary badges with the highly dynamic latent topics, one can naturally match the corresponding latent topics across different time periods. For instance, while what it means to be "liberal" changes from month to month, as expressed in what self-described liberals share on Twitter, the "liberal" badge is persistent, allowing us to produce a direct correspondence between "liberal" topics from different periods of time. In contrast, in a traditional topic modeling setting, we would be forced either to perform a heuristic bipartite matching on the topic-word distributions from the different time periods, to best match the unlabeled topics with each other, or to resort to a more complicated model that directly models the time stamp of each article, which can lead to inefficient inference [2].

In the remainder of this paper, we describe our approach for learning a badge-based representation of documents from the self-described attributes of their likely readers. We then perform an extensive evaluation of our approach, and show through both examples and quantitative experiments that

[2]http://www.guardian.co.uk/world/2012/sep/07/haqqani-network-blacklisted-terrorist-us


(a) word representation (b) badge representation

Figure 2: Here, we see the difference between the word representation and badge representation of the same article from *The Guardian*, "Haqqani network is considered most ruthless branch of Afghan insurgency" (September 7, 2012). In (a), the size of a word is proportional to its tf-idf weight, while in (b), the size of a badge is proportional to the weight it is assigned via the approach in Section 3.2. (Throughout this paper, we will display badges in blue and words in black.)

incorporating reader information into content analysis yields an article representation that is more interpretable for human understanding and more effective for personalization.

## 2. APPROACH SUMMARY

We give a succinct high-level summary of our model and algorithms in this section and provide full details in the following sections of the paper:

1. We collect a training data set of tweeted news articles from a specified time period. We represent the content of each training article as a bag-of-words vector, with more important words accruing larger weight.
2. We learn a labeled dictionary—whose columns correspond to badges and rows correspond to words in a vocabulary—by minimizing the (regularized) reconstruction error of training articles with respect to the badges of the users who shared them.
3. Given a new article from the same time period, we represent the article as the sparse linear combination of badges that most faithfully represents its content in terms of the labeled dictionary learned in the previous step. We incorporate the relationships between badges through a structured sparsity regularization.

If we have data from multiple time periods, we learn a separate dictionary per time period.

## 3. THE BADGE MODEL

The data we gather from Twitter is threefold: (1) we take each tweeted article, download its content, and represent it as a vector of words following the bag-of-words convention; (2) we associate each article with the users who have tweeted it; (3) we associate each user with a set of descriptive words from his or her profile, which we refer to as *badges*.

Given this data, we face two challenges: first, we must represent each badge as a weighted set of characteristic words, and, second, we seek to represent any article as a weighted set of badges whose characteristic words collectively best represent the content of the article.

We emphasize that any acceptable solution to these challenges must be *scalable*, while at the same time incentivizing *sparsity*. An approach that is not scalable could not handle the web-scale data sets we encounter in our setting; in a given month of tweets, we must learn thousands of badges from millions of news articles. Meanwhile, sparsity leads to an interpretable, parsimonious document representation, while simultaneously improving scalability. We emphasize

that we want sparsity in two parts of our model: each badge should have a small set of characteristic words, and each article should be described by a small set of badges.

As an obvious first step, we might consider the vast existing literature on probabilistic topic modeling [3]. A standard LDA-based topic model organizes a document collection into so-called topics, where each topic is a distribution over words in a vocabulary. Each document is then represented as a distribution over topics. Standard topic modeling approaches do not incentivize sparsity, leading to dense document and topic representations. More complex topic modeling-based approaches exist that incorporate sparsity (e.g., [23]), but they are not naturally scalable to web-scale data sets.

As such, we take an alternative approach to addressing these challenges that allows us to directly control the sparsity of the representation while maintaining scalability. Formally, we let $V$ denote the size of the vocabulary in our training data, $N$ the number of training documents, and $K$ the total number of badges. From a generative perspective, we think of the document $i$, represented as a $V$-dimensional vector over the words, $\mathbf{y}_i$, as formed by:

$$\mathbf{y}_i \approx \mathbf{B}\boldsymbol{\theta}_i.$$

$\mathbf{B}$ is a non-negative $V \times K$ matrix with a column for each badge, representing the weighted set of characteristic words for the badge. $\boldsymbol{\theta}_i$ is a $K$-dimensional vector that similarly represents the weighted set of characteristic badges associated with document $i$. We borrow a term from information theory and refer to $\mathbf{B}$ as our *badge dictionary*, where each column of $\mathbf{B}$ is an entry in the dictionary. Our sparsity assumptions translated to this setting mean that both $\mathbf{B}$ and $\boldsymbol{\theta}_i$ must have small numbers of non-zero entries. The training corpus of articles along with the user profile information provides us the $\mathbf{y}_i$'s and information about the $\boldsymbol{\theta}_i$'s for many documents with which we can learn the matrix $\mathbf{B}$; we refer to this phase as "learning the dictionary." We then can apply the dictionary $\mathbf{B}$ to analyze contents of new documents, estimating their $\boldsymbol{\theta}_i$ vectors corresponding to relevant badges; we refer to this phase as "coding the documents."

## 3.1 Learning the Dictionary

For each document $i$ in our training corpus, we observe the content vector $\mathbf{y}_i$, and the badges of the readers who shared document $i$ on Twitter. This set of badges—reported by the document's readers—does not give us direct access to $\boldsymbol{\theta}_i$, because it may contain irrelevant badges while potentially omitting important badges. However, this difficulty is not insurmountable; as we are only interested in a high-level association between badges and article content, drawn from a large collection of users and articles, it is reasonable and sufficient to assume that, on average, the documents shared by readers self-identified with a specific badge $k$ will be relevant to badge $k$, while documents shared by readers without badge $k$ will be irrelevant to badge $k$. Therefore, to learn the dictionary $\mathbf{B}$, we approximate $\boldsymbol{\theta}_i$ by taking each reader of document $i$, and assume uniform weights over the badges declared in his or her profile. We then aggregate over all of document $i$'s readers. More precisely, we assume $\theta_{ik} \propto \sum_{u \text{ tweeted } i} \delta_k^{(u)} / \sum_j \delta_j^{(u)}$, where $\delta_k^{(u)}$ is a 0/1 function indicating whether user $u$ identifies with badge $k$.

With each $\mathbf{y}_i$ given and each $\boldsymbol{\theta}_i$ approximated, the badge dictionary $\mathbf{B}$ can be learned by choosing a loss function and minimizing the loss objective:

$$\min_{\mathbf{B}\geq 0} \sum_{i=1}^{N} l(\mathbf{y}_i, \mathbf{B}\boldsymbol{\theta}_i) + \lambda_B \sum_{j=1}^{V}\sum_{k=1}^{K} |\mathbf{B}_{jk}|.$$

We constrain all entries of $\mathbf{B}$ to be non-negative to make the results more interpretable, and use the well-studied $\ell_1$-regularization on the entries of $\mathbf{B}$ to encourage sparsity in the learned $\mathbf{B}$ matrix.

In our work, we let $\mathbf{y}_i$ be a *term frequency-inverse document frequency (tf-idf)* vector of the words in document $i$ (cf. [17]), normalized to have $\ell_2$-norm of 1. The $\boldsymbol{\theta}_i$ vector, as described before, gives a uniform weight to the set of all badges of the readers of document $i$, normalized also to have unit $\ell_2$-norm. We then minimize a square-error loss and choose the regularization parameter $\lambda_B$ that achieves a desired level of sparsity in the resulting $\mathbf{B}$ matrix:

$$\min_{\mathbf{B}\geq 0} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{B}\boldsymbol{\theta}_i\|_2^2 + \lambda_B \sum_{j=1}^{V}\sum_{k=1}^{K} |\mathbf{B}_{jk}|. \quad (1)$$

We optimize Eq. 1 using a simple projected stochastic gradient descent, described further in the supplemental material.[3] This approach to optimization allows us to operate on large, streaming, web-scale data sets. We further normalize each column of $\mathbf{B}$ to have unit $\ell_2$-norm, to prepare us for coding documents, as described in the next section.

Many techniques learn both $\mathbf{B}$ and $\boldsymbol{\theta}_i$ from the training corpus—non-negative matrix factorization and LDA, for example. However, joint estimation of both $\mathbf{B}$ and $\boldsymbol{\theta}_i$ is inherently more complex; with many more variables to learn, the estimation is slower and the solution quality is poorer. In contrast, our method uses the reader attribute information to guide the estimation of $\boldsymbol{\theta}$, thus drastically reducing the learning complexity.

## 3.2 Coding the Documents

A straightforward approach for representing a new document in terms of badges is to take the same loss-objective as in the dictionary learning phase, and optimize over $\boldsymbol{\theta}_i$ instead of $\mathbf{B}$. That is, given a new document $i$, we optimize:

$$\min_{\boldsymbol{\theta}_i\geq 0} l(\mathbf{y}_i, \mathbf{B}\boldsymbol{\theta}_i) + \lambda_\theta \|\boldsymbol{\theta}_i\|_1,$$

where we again encourage sparsity in the estimated $\boldsymbol{\theta}_i$ by the $\ell_1$-regularization.

With squared-error, our objective takes the form of the well-known non-negative lasso:

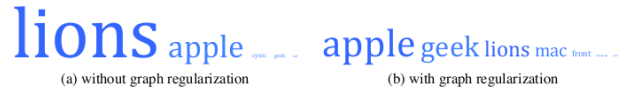
$$\min_{\boldsymbol{\theta}_i\geq 0} \|\mathbf{y}_i - \mathbf{B}\boldsymbol{\theta}_i\|_2^2 + \lambda_\theta \|\boldsymbol{\theta}_i\|_1. \quad (2)$$

We again borrow a term from information theory and refer to an optimization like in Eq. 2 as *coding the article* in terms of the badges. Eq. 2 can be solved efficiently through various algorithms, including coordinate descent and Shotgun [5].

## 3.3 Incorporating Relations among Badges

In practice, there is a subtle problem with the formulation in Eq. 2. Many badges tend to be highly related, such as "progressive" and "liberal," "school" and "student," and "vegan" and "vegetarian." These closely-related badges tend to model similar content and overlap in explanatory power. Thus, the estimated set of relevant badges—the non-zero

[3] `http://www.cs.cmu.edu/~kbe/badgepaper_supp.pdf`




(a) without graph regularization (b) with graph regularization

Figure 3: Badge representation of an article about Mac OS X Lion, with and without graph regularization. The size of a badge is proportional to its weight.

entries of the estimated $\boldsymbol{\theta}_i$ vector, encouraged to be as small as possible by the sparsity regularization—would arbitrarily include, e.g., either "progressive" or "liberal," but not both. The fact that these choices are arbitrary has undesirable consequences: for instance, given two very similar articles about the liberal political view on education, one may be represented by the badges "progressive" and "school" and the other by a completely disjoint set of badges, "liberal" and "student". Any learning algorithm that uses the selected badges as features would consequently be misled into treating the two articles as completely dissimilar.

Ameliorating this problem requires two steps: (1) we must detect similarity relations among the badges; and, (2) we must augment the article coding objective so that groups of closely-related badges—e.g., "progressive" and "liberal"—would be selected together in the article representation.

To determine whether two badges are related, we look at co-occurrence counts of the badges in the profiles of Twitter users. Closely related badges might either frequently co-occur in user profiles—in cases like "Obama" and "liberal"—or each frequently co-occur with some other common badge—in cases like "liberal" and "progressive," with the common badge perhaps being, e.g., "activist" or "blogger." To address both cases, we form a weighted undirected graph over the badges where each edge between two badges has a weight proportional to the frequency that these two badges co-occur in Twitter user profiles. More precisely, if $s$ and $t$ represent two distinct badges, we let the weight of the edge between $s$ and $t$ be $w_{st} \equiv \frac{\#s,\, t \text{ co-occur}}{(\#s \text{ occur})(\#t \text{ occur})}$. One can see then that highly related badges would either be neighbors in this graph or be connected by a very short path, where the weights of the edges on the path would be very high.

Given such a graph, we augment our model with the *graph-guided fused lasso* regularization of Kim et al. [15]:

$$\min_{\boldsymbol{\theta}_i \geq 0} \|\mathbf{y}_i - \mathbf{B}\boldsymbol{\theta}_i\|_2^2 + \lambda_\theta \|\boldsymbol{\theta}_i\|_1 + \lambda_{\mathcal{G}} \sum_{(s,t)\in E(\mathcal{G})} w_{st}|\theta_{is} - \theta_{it}|, \quad (3)$$

where $w_{st}$ is the weight of the badge pair $(s,t)$ in the co-occurrence graph $\mathcal{G}$, as defined above. The graph fusion regularization encourages $\theta_{is}$ to be close to $\theta_{it}$ for all edges $(s,t)$ in the graph, where the strength of the regularization is proportional to the weight of the edge. In this way, highly related badges, closely connected in $\mathcal{G}$ by heavily weighted edges, are incentivized to be turned on or off simultaneously, since similar values of $\theta_{i\cdot}$ for such badges lowers the objective. The graph fusion regularization parameter, $\lambda_{\mathcal{G}}$, regulates how big a role the graph $\mathcal{G}$ should play in regularizing $\boldsymbol{\theta}$. We refer the readers to the recent work of Chen et al. [7] for a detailed discussion of the optimization algorithm for solving Eq. 3, which we use in our approach.

As an example of how this graph regularization addresses our problem, we can consider an article about Mac OS X Lion.[4] Coding this article with the vanilla lasso, without graph regularization, leads to a badge representation overwhelmed by the "lions" badge. This is problematic because, while the "lions" badge well explains the word "lion," which appears often in the article, the main usage of the "lions" badge occurs in the context of the Detroit Lions football team. As a result, the Mac OS X article could, with respect to the computed badge representation, be more similar to a football article than to a technology article. When using the graph-guided fused lasso however, we obtain a more balanced coding, with the badges "apple" and "geek" now being the most dominant, taking up nearly sixty percent of the squared two-norm of the badge vector (cf. Figure 3).

The reason for this improvement is evident when we consider the neighbors of "lions" and "apple" in our badge graph. The strongest links emanating from the "lions" badge are related to Michigan—e.g., "Detroit" and "mlive" (a Michigan news site)—or to animals—e.g., "jungle," "monkey" and "roar." These neighboring badges do not do a good job explaining the Mac OS X article, and so this forces "lions" to be downweighted. However, if we consider the strongest neighbors of "apple" in the badge graph, we see words such as "fanboy," "jailbreak" and "ipod," which are much more related to the content of the article.

[4] `http://www.macobserver.com/tmo/article/my_favorite_stealthy_lion_features/`

## 4. RELATIONSHIP TO PRIOR WORK

The idea of inferring information about documents from their readers is not new; there is a rich line of research on *collaborative filtering*, which classifies, filters, or recommends documents by detecting readership patterns which, in some sense, represent the collaborative effort of all the readers [21]. The most common approaches for collaborative filtering, such as matrix completion [6], leverage the intuition that similar readers tend to read similar documents, and thus recommend articles to users if they were read by users with similar past behavior. However, such approaches must overcome the *cold start problem*, where, for example, they are unable to infer much meaningful information about articles that do not have a large number of readers. In contrast, our approach avoids this problem altogether by associating user preferences with the *content* of the articles, and thus can be used to analyze articles which have never been read.

Popular methods for collaborative filtering often assume low-dimensional latent factors in readership patterns. Our approach also involves latent factors, but guides the latent variable discovery by associating each factor with a badge. Thus, our model can handle many latent factors without sacrificing much computational or statistical efficiency.

Because the latent factors in our model associate user preferences with topics in the document contents, our work draws upon the massive existing literature on *topic modeling* [3]. Of the countless varieties of topic models, the labeled LDA model [20] is particularly relevant, as it presents a method of associating each latent topic with an observed tag. Though it is reasonable to try to use labeled LDA to tie badges with topics, we prefer our dictionary learning algorithm, as it allows us to better promote sparsity and incorporate badge relations.

Likewise, while we can imagine an alternative *discriminative* formulation of our problem as a multi-label classifier (cf. [22]), we found that it was more natural to express the desired structured sparsity of the output in the form of a generative model.


(a) Olympics (#2) (b) soccer (#5) (c) Labour (#10) (d) views (#27)

Figure 4: **(a-c)** Word clouds representing three of the most frequently used badges in coding articles from *The Guardian* in our September 2012 test data set. The size of a word is proportional to its weight in the badge. **(d)** This word cloud represents "views," the 27th most heavily used badge when we code the September 2012 *Guardian* articles. This is the first badge in the ranking with an incoherent word representation, which is unsurprising, since a word like "views" does not naturally correspond to a specific user attribute, and is unlikely to be suitable as a badge (cf. Section 6).

Finally, our work is inspired by previous work that attempted to learn a latent badge representation of *individual users* based on their Twitter behavior [10]. Our work has a different goal, in that we seek to build a general-purpose document representation by learning associations between badges and *document content* from millions of users. Moreover, the prior work uses a pre-defined set of approximately 30 badges, while the badge dictionaries we learn using our methodology are comprised of thousands of badges.

# 5. EMPIRICAL ANALYSIS

We conduct an extensive empirical analysis of our badge-based document representation, focusing on the question we posed at the beginning of the paper. Specifically, we seek to show that by representing documents by attributes of their likely readers, we can create a document representation suitable for personalization.

We begin by describing the large data set we use for our evaluation, followed by both anecdotal descriptions and quantitative comparisons, showing that our badge-based document representation is useful and insightful.

## 5.1 Data Processing and Experimental Setup

To evaluate our method, we must obtain a training set of tweeted news articles. We achieve this with access to the Twitter Garden Hose stream, which is an approximately 10% random sample of all tweets. In our experiments, we consider three months-worth of tweets: September 2010, September 2011 and September 2012.[5] For each of these three months, we scan through every tweet in the Garden Hose and extract those that are: (1) a tweet of a link; and, (2) came from a user with a non-empty profile. This leaves us with over 120 million tweets across the three months.

Next, as we are particularly interested in news articles, and not videos, photos, games and other such shared web pages, we filter the tweeted links to match one of 20,000 mainstream news sources, as defined by Google News. We then *download each news article* shared in this set of tweets that we believe to be written in the English language, resulting in a smaller, but extremely rich, data set of nearly 3 million tweeted news articles.

We use standard heuristics to extract the most meaningful unique words in these articles to create a vocabulary for each time period, as well as extract all badges that occur more frequently than a specified threshold. This leaves us with 4,460 unique badges in September 2010, 5,029 badges in September 2011, and 5,247 badges in September 2012, and vocabulary sizes of about 55,000 words.

Based on this training data, for each of the three months, we can compute the $\boldsymbol{\theta}$ and $\mathbf{y}$ vectors, as well as the undirected graph over badges with weights $w_{st}$, and commence with dictionary learning, as described in Section 3.1. We learn a separate badge dictionary for each of the three months; we expect many common badges (because, e.g., there are always "vegetarians"), but we expect the word representations of each badge to change over time. Moreover, it is important to note here the computational efficiency of our dictionary learning method as compared to training a standard topic model: on the largest of our data sets, our algorithm, running on a single core, finishes in 224 minutes, more than six times faster than a state-of-the-art distributed LDA implementation with the same number of topics (cf. Section 5.4).

For the quantitative comparisons, we require a test set of articles. While our training requires the analysis of tweets, any documents—including never-before-published ones—can be represented using our badge-based document representation. Thus, for our test set, we download eight entire sections from *The Guardian*, a leading British newspaper, over the three months considered in our training set, comprising nearly 14,000 articles. We represent each test article as a tf-idf vector over the time-specific vocabulary constructed during training. We then code each article by optimizing Eq. 3, using the dictionaries learned from the training data.

More details on the data processing pipeline and our optimization can be found in our supplemental material.

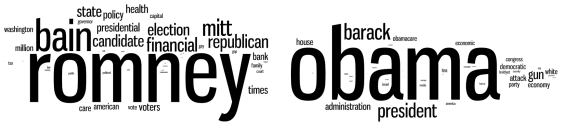
## 5.2 Examples

After learning badge dictionaries from the three training sets, we can ascertain how well the badge-labeled topics capture semantic themes in our data.

**Most Prevalent.** As a first example, we can examine the badges that we use most often (i.e., highest total weight) to code the *Guardian* articles from September 2012 in our test set. Three of these top badges are visualized in Figure 4: "Olympics" (ranked #2), "soccer" (ranked #5) and "Labour" (#10).[6] The characteristic words for these badges are precisely what we would expect; for example, the top words corresponding to the "Labour" badge are all related to British politics. We see such high quality associations between badges and their representative words throughout our dictionary. In fact, when ranking the badges by prevalence,

[5]Throughout the development of our approach and algorithms, we used a held-out validation set of tweets and tweeted articles, corresponding to July 2011 and July 2012.

[6]A full listing and visualization of the top ten badges can be found in the supplemental material.




(a) progressive > tcot (b) tcot > progressive

Figure 5: Here, we see the relationship between two related badges: "progressive" and "tcot" (Top Conservatives on Twitter). The word cloud on the left contains words that are more important for "progressive" than for "tcot," with the size of the word proportional to the difference in weights between the two dictionary elements. On the right, we see a word cloud containing the converse: words that are more important for "tcot" than for "progressive."

as above, we have to go down to the 27th position in the ranking before we find a badge with a poor representation: the "views" badge, which we visualize in Figure 4d.

**Dueling Badges.** An interesting exercise is to take a pair of antonymous badges, and see how their word representations compare. In Figure 5, we see a comparison of two popular badges related to American politics: "progressive" (a popular liberal badge) and "tcot" (**T**op **C**onservatives **o**n **T**witter). These dictionary elements were learned from the 2012 data, and thus come from the heat of the American Presidential race between Barack Obama and Mitt Romney. As this race was heavy on negative campaigning (cf., for example, [14]), it is not surprising to see that progressive supporters of Barack Obama were more likely than conservatives to share articles about Mitt Romney, and in particular, his controversial ties to Bain Capital, a financial firm he once headed. Likewise, conservatives are more likely than progressives to share articles about Barack Obama, presumably critical of him. We note that this analysis requires knowing how the users describe themselves, and is thus inaccessible to traditional topic models.

**Badges Over Time.** One motivation for using badges to represent documents is their persistence over time. For example, even if what it means to be liberal changes from year to year, the "liberal" badge is always there to represent liberal-leaning documents. Thus, it is instructive to consider examples of both static and dynamic badges.

In Figure 6, we find the "music" badge, which is one of the most static badges in our data set; its characteristic words barely change over the two year period from September 2010 to September 2012. Namely, the type of Twitter user who identifies herself with music in her profile is likely to share articles with the words "music," "band," "album" and "song."

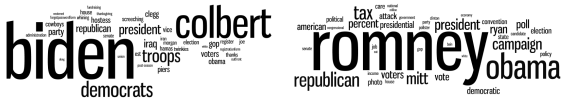
In contrast, Figure 7 shows one of the most dynamic badges in our data set: the one representing Vice President Joe Biden. The type of user who identifies himself with "Biden" shares rather different articles in 2010 and 2012. In September 2010, such a user focuses on the Vice President as well as comedian Stephen Colbert, who at the time was cohosting a political rally in Washington. However, by 2012, all signs of Joe Biden have diminished, and the primary focus of this badge is on the American Presidential race.

## 5.3 Case Study with Political Columnists

To demonstrate how our badge representation can provide insight on the makeup of a writer's likely readers, we use our model to analyze fourteen notable political columnists in the United States. These columnists each specialize in different topics, from economics to foreign policy, and are perceived to have different political leanings from very liberal to ultra-conservative. We show through various examples that, by understanding the writings of these political columnists through badges, we can characterize their target audiences in interesting ways. We emphasize that we only look at the *content* of the columnists' articles; only the badge dictionary is learned from documents shared on Twitter, and thus this analysis does not require that the columnists' articles appear on Twitter at all.


(a) music (2010) (b) music (2012)

Figure 6: The "music" badge is one of the most static badges in our data set; its characteristic words barely change over the two year period from September 2010 to September 2012, as can be seen in this pair of word clouds.




(a) Biden (2010) (b) Biden (2012)

Figure 7: The "Biden" badge is a dynamic one. In 2010, readers with the badge share articles about Joe Biden and Stephen Colbert, while in 2012, the focus turns to Barack Obama and Mitt Romney, due to the Presidential campaign.

As a first analysis, we take each article written by each of the fourteen columnists in July 2012, and code the article text in terms of badges, using our methodology. For each columnist, we then average the badge representations of the columnist's articles, resulting in an aggregate badge representation for each columnist. Examples can be found in Figure 8. We find that the badge representation, in almost all cases, accurately reflects the topics of expertise of the columnists; for instance, the words "aid" and "Africa" appear prominently in the badge representation for Nicholas Kristof, which makes sense because a reader who is self-described to be interested in "aid" or "Africa" would be quite likely to read Kristof's analyses of the various humanitarian crises in third world countries. Likewise, the badge representation for Maureen Dowd accurately shows that her likely readers are "progressive." It is critical to point out that Dowd does not in fact use the word "progressive" in any of her columns throughout this time period; rather, this coding corresponds to the attributes of her likely readers. Additionally, the badges "Irish" and "Ireland" appear prominently because Maureen Dowd was on assignment in Ireland in July 2012, writing prolifically about the country.

As a second analysis, we compare the political leanings of the likely readers of the fourteen columnists, by coding the columnists' articles in terms of *only* the "progressive" and "tcot" badges. In Figure 9, we place the columnists on a spectrum, where the location of each columnist is based on the relative weight of the "tcot" badge to the "progressive" badge in his or her average badge representation. Thus, columnists appearing on the left side of the spectrum are


(a) Nicholas Kristof (b) Maureen Dowd

Figure 8: Word clouds depicting the prominent badges that best represent the writings of two *New York Times* columnists from July 2012. The badge representations of the writers match well with the subject matter of their columns.

more likely to appeal to readers self-identified as "progressive" than to readers self-identified as "tcot." For example, the location of ultra-conservative writer Ann Coulter on the far right of the spectrum indicates that, based on her writings in July 2012, her likely readership during that month is almost exclusively conservative.

Overall, our ranking of the political columnists roughly lines up with the public perception of the columnists' political alignments, with outspoken conservative voices like Coulter and Charles Krauthammer placed on the far right, and well known liberal voices like Maureen Dowd on the (not as extreme) left. Likewise, Thomas Friedman, Fareed Zakaria and David Ignatius are clustered together, as they all write primarily about foreign policy. It is important to emphasize, however, that our approach does not attempt to directly classify the political alignment of the columnists. Rather, we instead try to identify what kind of readers would be interested in reading each of the columnist's editorials. For instance, if we take the example of Kathleen Parker, although she is a conservative columnist with the *Washington Post*, she often levels strong criticism against the Republican Party, and even supported Barack Obama for the 2008 presidential elections. It is thus sensible that she is placed more to the left in our chart despite being conservative, because politically liberal readers often enjoy reading her columns.

However, our ranking is not perfect. If we consider the location of William Kristol, a neo-conservative icon, we find that he is incorrectly placed on the left side of this spectrum. We posit that this behavior arises from the phenomenon visualized in Figure 5, whereby progressives are more likely to write about Mitt Romney than conservatives. While an unabashed conservative, in the month of July 2012, Kristol writes about Romney nearly twice as much as he writes about Obama, which may explain the discrepancy.

Finally, it is interesting to note the relationship between our case study and the well-studied *ideal point model* from quantitative political science, which assumes legislators and bills lie in a low-dimensional Euclidean space indicating political positions, and the affinity of a legislator for a bill is a function of the distance between their two locations [19]. Early work learned such ideal points from roll call votes, whereas more recent work in machine learning has combined roll call data with text analysis [12]. While perhaps visually similar, Figure 9 is computed based on a completely different signal than traditional ideal point models.

## 5.4 Quantitative Comparisons

In this section, we evaluate the utility of our methodology as a document representation for personalization. We compare our badge-based approach to two commonly used representations: tf-idf and an LDA-based representation.

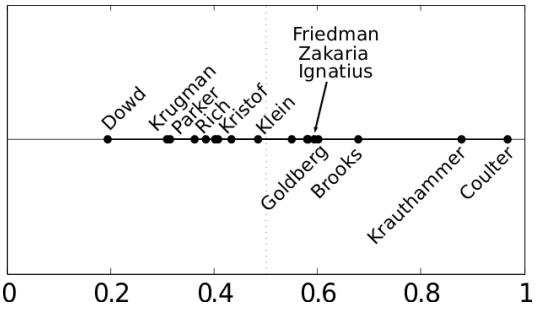



Figure 9: Predicted political alignments of the likely readers of fourteen political columnists based on results from representing the columnists' writing with only two badges: "progressive" and "tcot." The columnists are ranked from left to right based on how well the content of their writing is captured by the left-wing "progressive" badge as opposed to the right-wing "tcot" badge.

**Coherence.** As a first comparison, we test our hypothesis that the topics we learn with our badge-based representation are more semantically coherent than topics learned through a topic model, such as LDA. Our belief stems from the intuition that there is no explicit incentive for a topic model to produce coherent topics corresponding to human interests, while each of our dictionary elements corresponds to a badge directly used by a Twitter user to describe himself.

To quantitatively measure such a notion of coherence, we use the methodology of Mimno et al. [18], and compute a statistic based on how frequently the top words in each topic co-occur in a reference corpus. Mimno and colleagues show that this statistic strongly correlates with human notions of coherence, as validated by a user study. Figure 10a shows that, using Mimno's metric, our learned badge dictionary produces more semantically coherent concepts than LDA, when trained on the full September 2012 training data set.

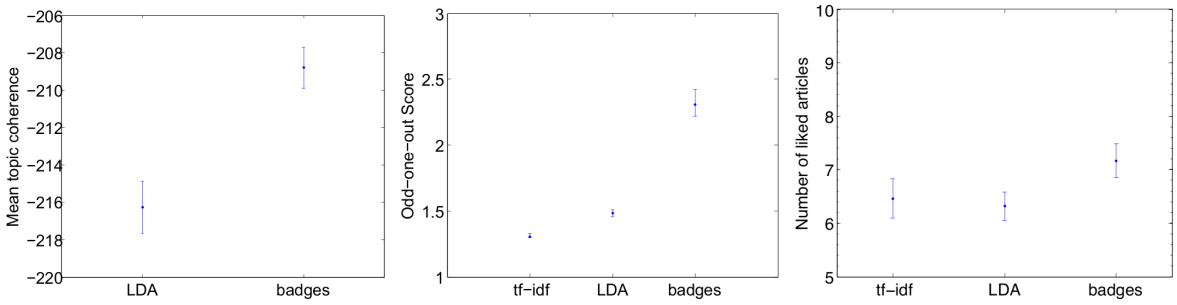
For a fair comparison, we run LDA with the same number of topics as we have badges for this time period (i.e., 5,247).[7] Moreover, we compute Mimno's coherence statistic using the top 15 words in each topic, following the convention used in their paper. Due to the scale of this problem (over 1.5 million documents, 5,000 topics, 55,000 word vocabulary and 375 million total tokens), we run a distributed implementation of collapsed Gibbs sampling for LDA, over 114 cores, provided as part of GraphLab [13].

**Odd-one-out.** Our second quantitative comparison examines our hypothesis that the badge-based document representation is better suited than competing techniques to represent coherent semantic concepts *over time*. This belief is due to the persistent nature of badges, which allows us to straightforwardly bind together concepts across different time periods. For example, the "football" badge in September 2010 can be directly matched to the "football" badge in September 2012, whereas topics from an unsupervised topic model, separately trained over disjoint data sets, must be matched in less straightforward ways.

To evaluate our hypothesis, we conduct the following intrusion detection experiment on our *Guardian* test data:

1. Pick two newspaper sections at random. Call the first one the *home* section and the second the *intruder*. For example, we might pick "world" and "sport" as the home and intruder sections, respectively.

[7]Similar results were obtained for 100-topic LDA.




(a) Semantic coherence (b) Odd-one-out (c) User study

Figure 10: **(a)** Dictionary elements from our learned badge dictionary are more semantically coherent than topics from an LDA topic model. Both models use the same number of topics/badges (5,247), and are trained on the September 2012 training data set. The coherence numbers reported are computed based on the methodology of Mimno et al. [18]. **(b)** Odd-one-out metric showing that our badge-based document representation does a better job at preserving the semantic similarity of articles within the same newspaper section over time than the competing representations. Results reported are the median of 56,000 independently drawn triplets of articles, and 95% confidence intervals are computed using the normal approximation to the binomial (cf. [1]). **(c)** Results from the news recommendation user study, showing that our badge-based document representation leads to better article recommendations than competing document representations.

2. Pick an article $h_1$, uniformly at random, from the home section of the September 2010 *Guardian* data.
3. Pick an article $h_2$, uniformly at random, from the home section of the September 2012 *Guardian* data.
4. Pick an article $i_2$, uniformly at random, from the intruder section of the September 2012 *Guardian* data.
5. We compute the ($\ell_2$-normalized) document representations for each of these three articles.
6. For a given representation (e.g., for LDA), we compute the following cosine similarity ratio: $(\mathbf{h}_1^\mathsf{T}\mathbf{h}_2)/(\mathbf{h}_1^\mathsf{T}\mathbf{i}_2)$, where, e.g., $\mathbf{h}_1$ is the vector representation of $h_1$. We call this the "odd-one-out" score for this triplet of articles and this document representation, as it tells us how much more similar the two documents from the same section are to each other, versus the two documents from different sections.

A document representation with a high "odd-one-out" score indicates that the semantic similarity between articles from the same section is preserved across time. A lower "odd-one-out" score indicates that a representation can more easily conflate the content of different news sections, leading to thematic incoherence over time.

We compute this score for our badge-based representation, as well as a 100-topic LDA topic representation and a tf-idf word representation. Specifically, for each pair of home and intruder sections, we draw 1,000 random article triplets, and compute the median odd-one-out score for each method. Figure 10b shows that, overall, aggregating over all pairs of sections, the badge representation significantly outperforms the two competing techniques on this metric. Moreover, in the supplemental material, we show that this significant advantage holds true not just at an aggregate level, but in about 80% of the individual section pairings.

**User Study.** Our final quantitative evaluation addresses the fundamental question: can we develop a document representation that works well for personalization?

To answer this question, we conduct a news recommendation user study on Amazon Mechanical Turk, comparing our badge-based document representation to tf-idf and LDA. We use each of the three as concept representations in the Interactive Concept Coverage framework of El-Arini [9], allowing us to recommend a diverse set of related articles based on user feedback. Our study is in two phases: first, a user provides feedback on a random set of articles that allows us to quickly estimate his interests, and then we recommend articles to the user and measure how many of them he likes.

Specifically, our study involves the following:

1. Pick at random two time periods from the set: { September 2010, September 2011, September 2012 }. Assign one time period to the first phase of the study, and the other time period to the second phase.
2. From the first time period, draw 20 news articles, uniformly at random, from our *Guardian* data set.
3. Present these 20 news articles, one at a time, to the user, asking him to mark each article as interesting or not. This is the first phase of the study.
4. Draw a random document representation from the set: { tf-idf, LDA, badges }. Based on the representation we select, compute the average vector of the articles marked as interesting in the first phase. For example, if the user has indicated interest in just two articles, one on Manchester United and another on the London Olympics, and tf-idf was selected, we would average together the tf-idf vectors of the two articles, leading to high weights on words like "London," "football," "Olympics," "Manchester," etc.
5. Align the average document representation computed in the previous step to match with the corresponding representation in the second year of the study. With LDA, this involves the Hungarian algorithm for bipartite matching over the topic-word distributions, while for tf-idf and badges, it involves simply matching the words or labels from one time period to the other.
6. Use the transformed average document vector, indicating the user's interests from phase one of the study, as *concept weights* for Interactive Concept Coverage. Specifically, this entails using these weights to describe the relative importance of concepts (i.e., words, topics or badges) in a probabilistic max-cover setting, resulting in a diverse set of ten articles from the second time period relevant to the user's interests.[8]
7. Show the recommended articles to the user, one at a time, obtaining feedback on which ones are interesting.

[8] cf. Chapter 3 of El-Arini's thesis for more details [9].

Figure 10c shows that our badge-based representation significantly outperforms both tf-idf and LDA on this fundamental news recommendation task. On average, users find the articles we recommend to them to be more interesting than the articles recommended via the competing document representations. This is what we expected, and backs our hypothesis that the badge-based representation is a preferable document representation for personalization tasks—particularly ones that cut across periods of time. While tf-idf is excellent at detecting article similarity within a time period, it is worse at detecting similar articles from two completely different periods of time. Meanwhile, LDA is at the mercy of a successful bipartite matching. The badge-based representation can overcome both challenges, leading to improved performance. (More details on the study can be found in the supplemental material.)

# 6. DISCUSSION

In this work, we proposed a new document representation based on associating articles with attributes of their likely readers. Our approach of learning a labeled dictionary from a large-scale Twitter data set, which we then use to code new articles via a structured sparsity optimization, led to a document representation that was both human interpretable and useful for personalization. Experimentally, we demonstrated that our methodology leads to thematically coherent topics that are more consistent over time than popular alternative approaches, leading to better performance on a live personalization task. Moreover, our representation allows us to provide interesting insights about writers and the state of political discourse, confirming some widely held beliefs.

However, some challenges remain:

- Not every word that a user writes in his or her Twitter profile should be considered worthy of being a badge (cf. Figure 4d). Deeper linguistic analysis of user profiles will be necessary to identify words or phrases that are most suited to representing user attributes.
- The simple badges we gathered from Twitter users work well for news recommendation, but how do we transfer this success to other domains?

Despite these challenges, we believe that representing documents by their readers is an important, novel contribution.

# 7. ACKNOWLEDGMENTS

We are grateful to Brendan O'Connor and Noah Smith for providing us with access to the Twitter Garden Hose. This work was partially supported by ONR grant PECASE N000141010672 and NSF grant NETS SCAN CNS0721591.

# 8. REFERENCES

[1] M. Bland. *An Introduction to Medical Statistics*. Oxford Medical Publications, 3rd edition, 2000.

[2] D. M. Blei and J. Lafferty. Dynamic topic models. In *ICML*, 2006.

[3] D. M. Blei and J. Lafferty. Topic models. In A. Srivastava and M. Sahami, editors, *Text Mining: Theory and Applications*. Taylor and Francis, 2009.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[5] J. K. Bradley, A. Kyrola, D. Bickson, and C. Guestrin. Parallel coordinate descent for L1-regularized loss minimization. In *ICML*, 2011.

[6] E. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Tran. on Information Theory*, 56(5):2053–2080, 2010.

[7] X. Chen, Q. Lin, S. Kim, J. Carbonell, and E. P. Xing. Smoothing proximal gradient method for general structure sparse regression. *Annals of Applied Statistics*, 6(2):719–752, 2012.

[8] G. De Francisci Morales, A. Gionis, and C. Lucchese. From chatter to headlines: Harnessing the real-time web for personalized news recommendation. In *WSDM*, 2012.

[9] K. El-Arini. *Beyond Keyword Search: Representations and Models for Personalization*. PhD thesis, Carnegie Mellon University, 2013.

[10] K. El-Arini, U. Paquet, R. Herbrich, J. V. Gael, and B. A. y Arcas. Transparent user models for personalization. In *KDD*, 2012.

[11] K. El-Arini, G. Veda, D. Shahaf, and C. Guestrin. Turning down the noise in the blogosphere. In *KDD*, 2009.

[12] S. Gerrish and D. M. Blei. Predicting legislative roll calls from text. In *ICML*, 2011.

[13] J. Gonzalez, Y. Low, H. Gu, D. Bickson, and C. Guestrin. PowerGraph: Distributed graph-parallel computation on natural graphs. In *OSDI*, 2012.

[14] N. Greenstein. Negative ads: A shift in tone for the 2012 campaign. *TIME Magazine*, July 17, 2012.

[15] S. Kim, K.-A. Sohn, and E. P. Xing. A multivariate regression approach to association analysis of a quantitative trait network. In *ISMB*, 2009.

[16] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *WWW*, 2010.

[17] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[18] D. Mimno, H. Wallach, E. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. In *EMNLP*, 2011.

[19] K. T. Poole and H. Rosenthal. A spatial model for legislative roll call analysis. *American Journal of Political Science*, 29(2):357–384, May 1985.

[20] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *EMNLP*, 2009.

[21] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *WWW*, 2004.

[22] G. Tsoumakas, I. Katakis, and I. Vlahavas. Mining multi-label data. In *Data Mining and Knowledge Discovery Handbook*, pages 667–685, 2010.

[23] S. Williamson, C. Wang, K. A. Heller, and D. M. Blei. The IBP compound Dirichlet process and its application to focused topic modeling. In *ICML*, 2010.

[24] Y. Yue and C. Guestrin. Linear submodular bandits and their application to diversified retrieval. In *NIPS*, 2011.