Leveraging Blogging Activity on Tumblr to Infer Demographics and Interests of Users for Advertising Purposes

Mihajlo Grbovic, Vladan Radosavljevic, Nemanja Djuric
Narayan Bhamidipati, Ananth Nagarajan
Yahoo Labs
{mihajlo, vladan, nemanja}@yahoo-inc.com
701 First Ave, Sunnyvale, CA, USA

ABSTRACT

As one of the leading platforms for creative content, Tumblr offers advertisers a unique way of creating brand identity. Advertisers can tell their story through images, animation, text, music, video and more, and they can promote that content by sponsoring it to appear as an advertisement in the streams of Tumblr users. In this paper, we present a framework that enabled one of the key targeted advertising components for Tumblr, specifically, gender and interest targeting. We describe the main challenges involved in the development of the framework, which include the creation of a ground truth for training gender prediction models, as well as mapping Tumblr content to an interest taxonomy. For purposes of inferring user interests, we propose a novel semi-supervised neural language model for categorization of Tumblr content (i.e., post tags and post keywords). The model was trained on a large-scale data set consisting of 6.8 billion user posts, with a very limited amount of categorized keywords, and was shown to have superior performance over the baseline models. We successfully deployed gender and interest targeting capability in Yahoo production systems, delivering inference for users that covers more than 90% of the daily activities on Tumblr. Online performance results indicate advantages of the proposed approach, where we observed a 20% increase in user engagement with sponsored posts in comparison to untargeted campaigns.

Categories and Subject Descriptors

H.2.8 [**Database applications**]: Data Mining

Keywords

data mining; computational advertising; audience modeling; algorithms

Published as part of the #Microposts2016 Workshop proceedings, available online as CEUR Vol-1691 (http://ceur-ws.org/Vol-1691)

#Microposts2016, Apr 11th, 2016, Montréal, Canada.

1. INTRODUCTION

In recent years, online social networks have evolved to become an important part of life for online users of all demographic and socio-economic backgrounds. They allow users to easily stay in touch with their friends and family, discuss everyday events, or share their interests with other users with the click of a button. Tumblr is one such social network, representing one of the most popular and fastest growing networks on the web. Hundreds of millions of people around the world come every month to Tumblr to find, follow, and share what they love. The Tumblr network is a gold mine of content, comprising of 200 million blogs on different topics such as travel, sports, and music, where 85 million user posts are published on a daily basis. This wealth of user-generated data opens a great opportunity for advertisers, allowing them to promote their products through high-quality targeting campaigns to both blog visitors and blog owners [16].

The standard, prevalent form of advertising on Tumblr is through *sponsored posts* that appear alongside regular posts in the user's *dashboard*, the central page for a Tumblr user, displaying the newest posts of followed blogs in the form of a stream. This form of advertising, in which advertisements resemble native content in the stream, is often referred to as *native advertising*. Native advertisements are usually aesthetically beautiful and highly engaging, which typically makes them more enjoyable than regular display ads [4]. Tumblr launched its native advertising product in May of 2012. Since then, the number of advertisers (or brands) on the platform has grown steadily and reached a milestone of 100 advertisers in April of 2013. Moreover, most of the biggest global brands have used Tumblr to advertise and sponsored posts have generated billions of paid ad impressions since the launch of the Tumblr advertising product[1]. In this paper, we further enhance ad targeting that Tumblr offers, allowing advertisers to specify new demographic or interest categories. This improved targeting provides advertisers with the control and flexibility to find the audience they most want to reach.

Building of interest targeting products on social and microblogging platforms is an important research topic, discussed previously by several researchers [11]. However, due to its distinct characteristics, Tumblr poses novel challenges,

[1] www.comscore.com, accessed June 2015

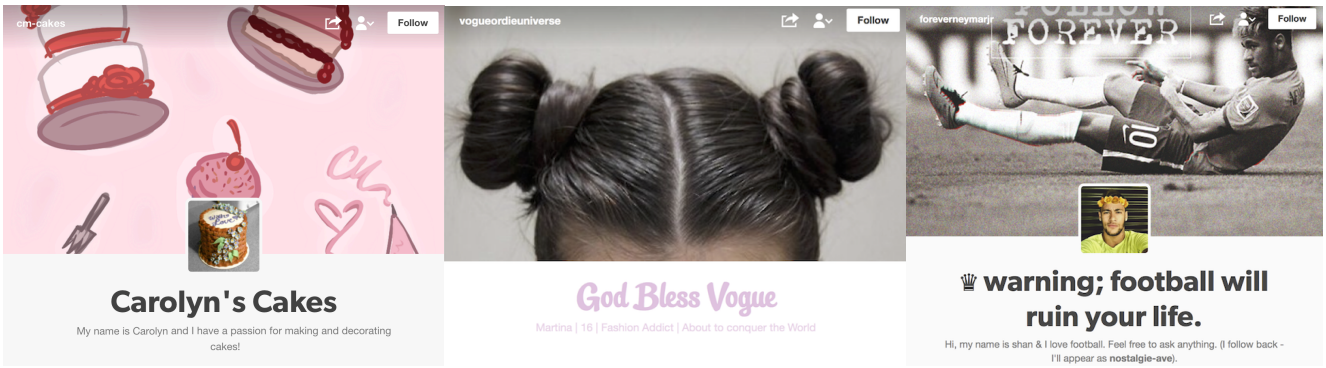

Figure 1: Examples of blog title (first line, bigger font) and blog description (bottom line, smaller font)

which we explain in detail in this paper. In particular, the content and language used on Tumblr have distinct characteristics that needed to be accounted for during the modeling. Users often use tags to summarize the text in their posts. However, the language styles used in the tags and post text are different (e.g., the tag "hp" and the word "hp" have different meanings when they appear in the posts, "Harry Potter" and "HP company", respectively). Moreover, unlike the popular social platform Facebook, which contains a large amount of social interactions but a limited amount of content, or the microblogging platform Twitter, which contains an intermediate amount of social interaction and content, Tumblr represents a unique combination of a rich and diverse content platform and a dynamic social network. To make use of this vast advertising potential, we propose to classify user-generated Tumblr content into a standard multi-level *general-interest taxonomy* [2] that advertisers commonly use for defining their targeting campaigns, opening doors to high-quality audience segmentation and modeling for purposes of ad targeting. However, inferring categories of user posts is a challenging task, given the huge quantities of unlabeled data being posted every day and the very limited amount of labeled data, typically obtained by human editorial efforts. To this end, we propose a novel semi-supervised neural language model, capable of jointly learning embeddings of post keywords, post tags and category representations in the same feature space. The neural model was trained on a large-scale data set comprising of 6.8 billion posts, with only a fraction of categorized content.

Targeting pipelines described in this paper are being used to show ads to millions of users daily, and have substantially improved Tumblr's business metrics following the launch. On our path to developing targeting capabilities for Tumblr, we first created user profiles, based on users' Tumblr activities that include publishing blog posts, following other blogs, liking posts, and other. Lastly, we aimed at building and delivering both demographic and interest predictive models based on the created profiles.

We note that the privacy of our users is of critical importance. Therefore, we were constrained in regards to what data we can use. Specifically, user profiles were created solely from data which users share publicly with others, including contents of blog posts, blog titles and descriptions, and follow, like and reblog actions. This data is publicly available through Tumblr Firehose data source[3]. Other user activities, such as user searches on Tumblr, which blogs they visited and where they clicked, are considered to be sensitive data and were not used in any way for the development of the ad targeting models.

2. RELATED WORK

Personalization is defined as "the ability to proactively tailor products and product purchasing experiences to tastes of individual consumers based upon their personal and preference information" [7], and it has become an important topic in recent years. Personalization of online content for individual users may lead to improved user experience and directly translate into financial gains for online businesses [14]. In addition, personalization fosters a stronger bond between customers and companies, and can help in increasing user loyalty and retention [2]. For these reasons it has been recognized as a strategic goal and is the focus of significant research efforts of major internet companies [8, 12].

We consider personalization through the domain of ad targeting [9], where the task is to find the best matching ads to be displayed for each individual user. This improves the user's online experience (as only relevant and interesting ads are shown) and can lead to increased revenue for the advertisers (as users are more likely to click on the ad and make a purchase). Due to its large impact and many open research questions, targeted advertising has garnered significant interest from the machine learning community, as witnessed by a large number of recent workshops[4]. and publications [5, 11].

One of the basic approaches in ad targeting is to target users with ads based on their demographics, such as age or gender. Historically, this approach has proven to work better than targeting random users. However, while for some products this type of targeting may be sufficient (e.g., women's makeup, women's clothing, man's razors, man's clothing), for others it is not effective enough and a more involved profiling of users is required. A popular method in today's ad targeting that addresses this issue is known as interest targeting, in which users are assigned interest categories, such as "sports" or "travel", based on their historical behavior [1]. Typically, a taxonomy is used to decide on the

[2] http://www.iab.net/QAGInitiative/overview/taxonomy

[3] gnip.com/sources/tumblr

[4] www.targetad-workshop.net

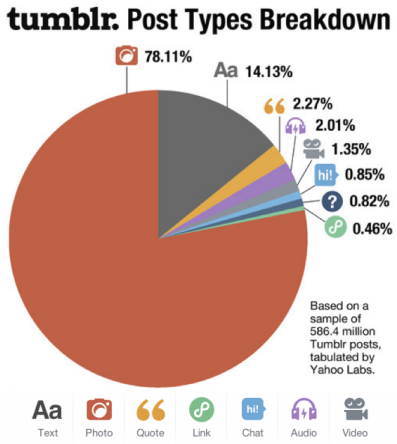

Figure 2: Distribution of Tumblr post types

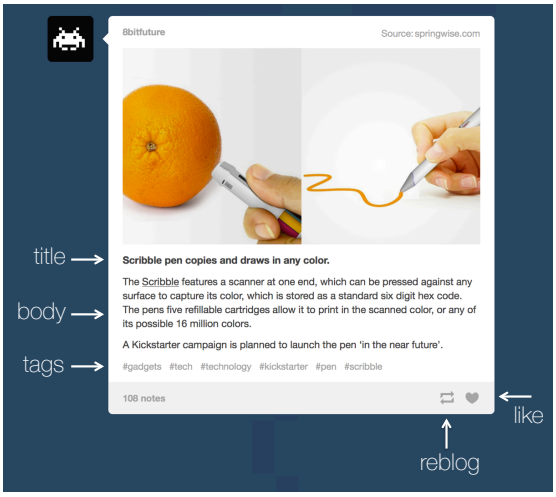

Figure 3: Example of Tumblr blog post

targeting categories, and a model is learned to categorize user activities and estimate their interest in each category. Interest targeting is known to build good brand awareness with relevant audience, which has already shown interest in the corresponding category. In this paper we follow this interest targeting approach. Alternatively, advertisers may be interested going a step forward and optimizing for intent, typically done by assigning categories to actual ads, and training a machine learning model to estimate the probability of an ad click in that category [17]. For each ad category a separate predictive model is trained, and evaluated on the entire user population, with N users with the highest score selected for ad exposure.

To the best of our knowledge, the Tumblr social network has been considered by only a few scientific studies. In [3, 15], the authors discuss the problem of blog recommendation, while in [6] they explore Tumblr social norms. However, our work is the first paper that addresses ad targeting on Tumblr.

3. WHAT IS TUMBLR?

Tumblr[5] is one of the most popular social blogging platforms, where users can create and share posts with the followers of their blogs. According to data from January 2015[6], there is a total of 221.6 million blogs on Tumblr, which jointly produced over 102.7 billion blog posts. With a large number of users signing up every day, Tumblr is currently the fastest growing social platform[7].

3.1 User activities on Tumblr

To register for a Tumblr account, a valid e-mail address is required, along with a primary username (which will become a part of the blog URL) and a confirmation of age. A Tumblr blog resembles a webpage, with a profile picture, blog title and blog description appearing at the top (see Figure 1), followed by a stream of blog posts bellow. The first blog created by a registered user is considered his or her primary blog. In addition, a very small portion of users maintain one or more secondary blogs. A Tumblr user is uniquely described by the blog ID of the primary blog, and throughout the paper we will use "blog" and "user" interchangeably.

Common user activities of Tumblr users include the following: 1) creating a post on one's blog; 2) sharing a post created by another blog, called *reblogging* (a reblogged post will appear on the user's blog); 3) liking a post by another blog; and 4) following another blog. Similar to Twitter, the follow connections at Tumblr are uni-directional. However, unlike Twitter, users can create longer and richer content in the form of several post types, such as text, photo, quote, link, chat, audio and video. The most popular types of blog posts are photo posts and text posts, and, based on the analysis published in [18], together they cover more than 92% of all posts on Tumblr (see Figure 2). Any post type can be annotated with words starting with # that concisely describe the post and allow for easier browsing and searching (called *tags*). Additional metadata that describes a post includes photo captions in photo posts, post titles in text posts, and artists names in audio posts. An example post is shown in Figure 3. Tags, such as *#gadgets* or *#tech*, are displayed bellow the photo caption, while the buttons for reblog and like actions are located in the bottom right corner. Lastly, each user has a *dashboard* (i.e., a feed of blog posts published by followed users, which is ordered in time), with more recent posts appearing at the top.

3.2 Advertising at Tumblr

Advertising on Tumblr is implemented through the mechanism of sponsored (or promoted) posts shown in a user's dashboard. This is similar to how advertising works on Twitter and Facebook. A sponsored post can be a video, an image, or simply a textual post containing an advertising message. In Figure 4, we show an example of a sponsored post and how it appears on web and mobile dashboards. Similarly to organic (or non-promoted) posts, sponsored posts can propagate from user to user in the network by means of reblogs, and users can also "like" the promoted post. Both likes and reblogs can be seen as an explicit form of acceptance or endorsement of the advertising message. Moreover,

[5] www.tumblr.com
[6] www.tumblr.com/about
[7] http://t.co/3txHFRJreJ

Figure 4: Example of Tumblr sponsored post

just like any other posts, sponsored posts are supplemented with notes on who liked and reblogged the post.

Interestingly, while user-generated, organic posts are reblogged on average 14 times, sponsored posts are reblogged on average 10,000 times[8]. We have observed that 40% of engagements with sponsored posts are reblogs, likes, or follows. What is more, every fourth reblog of a sponsored post results in 6 downstream reblogs from followers, leading to content longevity, and one third of reblogs of sponsored posts are present for 30 days or more after the initial post.

4. TUMBLR DATA

In this section, we describe the data sources (user activities and post contents) utilized to create user profiles. In particular, user activities included actions such as posts, likes, follows and reblogs, while post contents included tags, the title and body for text posts, artist names from audio posts, as well as tags and captions for photo posts.

4.1 Data sources

Once signed in onto Tumblr, a user can follow other users' blogs. The follow action is one-directional as it does not require the follow back. For the purpose of this study, we collected a sub-graph which contained 96.9 million unique nodes (i.e., users), 5.1 billion edges (i.e., follows), out of which 36.4 million are bi-directional (18.2 million pairs of users that follow each other). The data set included more than 26.1 billion activities on Tumblr. As discussed earlier, an activity log is available through a data feed called Firehose.

To create user profiles for targeting, textual contents of all posts were collected, including photo captions, tags, titles and bodies. In addition, every time a user performs a post or reblog activity, Firehose lists the user's blog title and blog description, which we also used to represent a user. As we can see in Figure 1, a blog title and description often provide useful information with respect to targeting, such as the user's first name, age, and even declared interests (e.g., statements such as "fashion addict" or "I love football").

[8]http://yhoo.it/1vFfIAc

Table 1: User data extracted from Tumblr Firehose

Declared	Content	Actions
blog title	post tags	reblog
blog description	photo captions	like
	text post title	follow
	text post body	
	audio post artists	

4.2 Data Processing

In order to improve the representation of user profiles, we propose to extract keywords from available blog information. This requires a certain amount of data preparation and processing. Given the extracted blog data, including the title, content and tags, we first removed all the html tags, followed by the removal of stopwords and the formation of bigrams. It is common for certain words to appear together more often than some others (e.g., words "credit" and "card"), and we aim to capture those bigrams and use them in keyword-based user profiles. To detect bigrams, we use a procedure that counts the unigram and bigram appearances, and for each combination of words w_i and w_j it calculates the following score:

$$\text{score}(w_i, w_j) = \frac{\text{count}(w_i, w_j)}{\text{count}(w_i)\text{count}(w_j)}. \quad (4.1)$$

Finally, bigrams with a score above a certain threshold were extracted from the text in blog contents. Tags, on the other hand, naturally form n-grams, and we extracted them in their original form.

4.3 User profiles

Available data sources were used to create user profiles. In particular, we extracted three distinct groups of user-related data: 1) declared; 2) content of posts; and 3) actions. The specific components included in each of the data groups are listed in Table 1. From each group we extracted features to represent the users as described below.

Declared data consists of information which a user provided during sign-up, including keywords from the blog title and blog description, where keywords were extracted using the method in Section 4.2. To create user profiles we kept the most frequent keywords from blog titles and descriptions, after removing stopwords such as "a", "the", "where", "in". We counted the keyword frequency in a user's blog title and description, and stored the count along with the time stamp of the latest log-in as a part of the user profile.

Content features were formed from the textual contents of posts which a user either created or reblogged. The main content feature types included: 1) post tags; 2) keywords from the post title and body; 3) keywords from photo post captions; and 4) artist names from audio posts. Tags in posts were not tokenized, instead they were used in the form they appear, e.g., tag "food for a vegan" was one keyword. On the other hand, in order to extract keywords from text appearing in text post content, we again used the method from Section 4.2. We kept only the most frequently occurring keywords, excluding stopwords. In addition, we used the most popular artist names as keywords. In this way, we collected several millions of distinct keywords that were

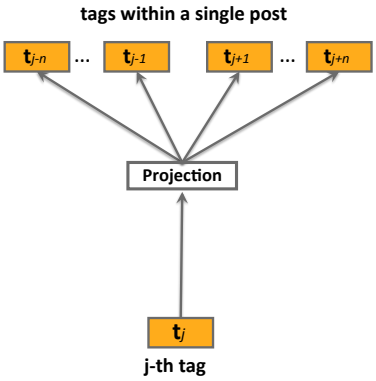

Figure 5: Unsupervised skip-gram model

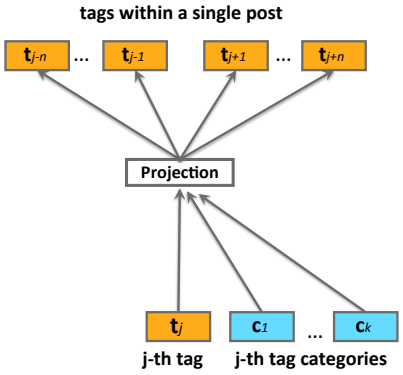

Figure 6: Semi-supervised skip-gram model

used to obtain rich representation of user profiles. To illustrate content keyword extraction from our dataset, consider that user u_i at time stamp t used tag *#hp* five times and tag *#nba* eight times, keyword *football* two times in post titles and posted an audio post with a song from artist *Shakira* ten times, then the resulting user profile would be: $u_i = \{tag : \{\#hp, t : 5, \#nba, t : 8\}, title : \{football, t : 2\}, artist : \{shakira, t : 10\}\}$.

Action features include follows, likes, and reblogs. If user u_i follows user u_j at time stamp t, we create an indicator feature $follows : \{j, t : 1\}$ and add it to the u_i's user profile. Similarly, if user u_i likes or reblogs user u_j's post, we create a feature that keeps record of the number of likes m and reblogs n, as $likes : \{j, t : m\}$, $reblogs : \{j, t : n\}$, respectively, and update the user profile accordingly.

The described approach resulted in user profiles for a total of 81.8 million users. The total number of unique features was 1.4 million, and an average user had 379.9 non-zero features. Most of the features described above are represented as either binary indicators or counts of occurrences.

5. INTEREST PREDICTION

The goal of our work is to infer user demographics (described in the following section) and identify user groups with interests in certain topics, such as music, travel, cooking or books, in order to allow advertisers to target segmented Tumblr audiences. As the topics may be defined at various levels of granularity, to avoid sparsity problems while still providing useful and actionable interest categories, user interests are often classified into a pre-determined hierarchical interest taxonomy that the advertisers commonly use. However, to be able to create effective user interest classifiers, one requires a sufficient amount of labeled data. Yet, for the problem of the scale of Tumblr interest prediction, this can be a daunting task for human editors. For that reason we propose a novel semi-supervised classification approach, based on the recently proposed word2vec model [13], which efficiently and seamlessly makes use of large amounts of unlabeled and a limited amount of editorially labeled data for learning effective content classifiers.

5.1 User interest taxonomy

We decided to classify keywords into the General Interest Taxonomy (GIT), used by the Yahoo Gemini advertising platform for native advertising[9]. The GIT is carefully derived based on Interactive Advertising Bureau (IAB) taxonomy recommendations, in order to meet advertiser needs and protect Yahoo's interests. The GIT has a two-level hierarchical structure, such that advertisers can adjust the audience reach by utilizing broader or narrower interest categories. The top level of the taxonomy contains 23 nodes (e.g., "Automotive", "Business", "Pets", "Travel"), while the second level contains 130 nodes which represent more precise interests (e.g., "Automotive/SUV", "Automotive/Luxury", "Pets/Dogs").

5.2 Proposed semi-supervised classification

In this section, we present a novel classification approach based on the recently proposed skip-gram model [13], which is used to categorize keywords into the GIT taxonomy. For conciseness, we describe the proposed model on the assumption that it is applied to tag categorization. However, we used the same methodology for categorization of keywords originating from blog titles, descriptions and text, audio and image posts.

We consider the task of tag classification, where the goal is to classify tags into a pre-defined taxonomy of interest categories. In order to address this problem, we propose to learn tag representation in a low-dimensional space using neural language models that are applied to historical Tumblr posts. Let us assume that we are given N posts. In the post logs found in Firehose, every post p_i is recorded along with the tags t_{ij}, $j = 1...M$. We collected data in the form $p_i = \{t_{ij}, j = 1...M_i\}$, where M_i represents the number of tags in the i-th post. Given the data set $\mathcal{D} = \cup_{i=1}^{N} p_i$, the objective is to find a representation of tags in which semantically similar tags are nearby in the representation space. For this purpose, we extend ideas originating from recently proposed language models, as described in the remainder of this section.

The skip-gram (SG) model involves learning representations of tags in a low-dimensional space from post logs in an unsupervised fashion, by using the notion of a blog post as a "sentence" and the tags within the post as "words", borrowing the terminology from the Natural Language Processing (NLP) domain (see Figure 5). Tag representations using the skip-gram model [13] are learned by maximizing the objective function over the entire $\mathcal{D}$ set of blog posts,

[9] gemini.yahoo.com

Table 2: Examples of categorized tags

Tag	Category
music	Arts and Entertainment/Music
fashion	Style and Fashion
song	Arts and Entertainment/Music
art	Arts and Entertainment
disney	Arts and Entertainment/Movies
style	Style and Fashion
photography	Hobbies/Photography
teen wolf	Arts and Entertainment/TV
food	Food and Drink

defined as

$$\mathcal{L} = \sum_{p \in \mathcal{D}} \sum_{t_j \in p} \sum_{-n \leq m \leq n, m \neq 0} \log \mathbb{P}(t_{j+m}|t_j). \quad (5.1)$$

The probability $\mathbb{P}(t_{j+m}|t_j)$ of observing a neighboring tag t_{j+m} given the current tag t_j is defined using the soft-max,

$$\mathbb{P}(t_{j+m}|t_j) = \frac{\exp(\mathbf{v}_{t_j}^\top \mathbf{v}'_{t_{j+m}})}{\sum_{t=1}^{T} \exp(\mathbf{v}_{t_j}^\top \mathbf{v}'_t)}, \quad (5.2)$$

where $\mathbf{v}_t$ and $\mathbf{v}'_t$ are the input and output vector representations of tag t of user-specified dimensionality d, n defines the length of the context for tag sequences, and T is the number of unique tags in the vocabulary. From equation (5.2) we see that tags that co-occur often and tags with similar contexts (i.e., with similar neighboring tags) will have similar vector representations as learned by the word2vec model.

The semi-supervised skip-gram (SS-SG) model assumes that some tags are labeled with categories from the GIT taxonomy. Next, we introduce a dummy category vector for each node of the taxonomy, and leverage the tag contexts in blog posts to jointly learn tag vectors and category vectors in the same feature space. Given such a setup, after learning the representations, every tag from the vocabulary can be categorized by simply looking up the closest category vector in the joint embedding space.

Specifically, given the labeled tags, we extend $\mathcal{D}$ to obtain data set $\mathcal{D}_{ss}$ where categories are imputed into post "sentences" p_i, where available. In particular, labeled tags are accompanied by assigned categories, and every time a vector of a labeled central tag t_j is updated to predict the surrounding tags, the vectors of categories assigned to t_j are updated as well. More formally, assuming the central tag t_j is labeled with C_j of C categories in total, $\zeta_j = \{c_1, \ldots, c_{C_j}\}$, the semi-supervised skip-gram learns tag and category representations by maximizing the following objective function,

$$\sum_{p \in \mathcal{D}_{ss}} \sum_{t_j \in p} \sum_{-n \leq m \leq n, m \neq 0} \Big(\log \mathbb{P}(t_{j+m}|t_j) + \sum_{c \in \zeta_j} \log \mathbb{P}(t_{j+m}|c) \Big). \quad (5.3)$$

The probability $\mathbb{P}(t_{j+m}|c)$ of observing tag t_{j+m}, given label c of the current tag t_j, is defined using the soft-max,

$$\mathbb{P}(t_{j+m}|c) = \frac{\exp(\mathbf{v}_c^\top \mathbf{v}'_{t_{j+m}})}{\sum_{t=1}^{T} \exp(\mathbf{v}_c^\top \mathbf{v}'_t)}. \quad (5.4)$$

This procedure allows us to seamlessly incorporate labeled and unlabeled data, and learn tag and category vectors in the joint embedding space. Then, the classification of tags amounts to a simple nearest-neighbor search among the category vectors. In Figure 6 we show the graphical representation of the semi-supervised skip-gram model.

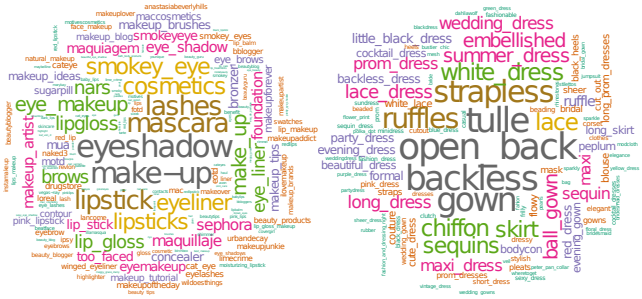

Figure 7: Nearest neighbors of tags: a) *#makeup*; b) *#dress*

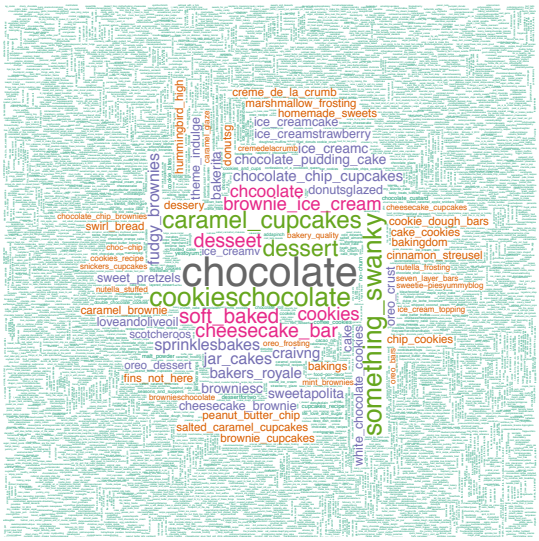

Figure 8: Nearest tags to "Food and Drink/Desserts"

5.2.1 *Training*

The models are optimized using stochastic gradient ascent, suitable for large-scale problems. However, computation of gradients $\nabla\mathcal{L}$ in (5.1) and (5.3) is proportional to the vocabulary size T, which may be computationally expensive in practical tasks seeing as how T could easily reach several million tags. As an alternative, we used a negative sampling approach [13], which significantly reduces the computational complexity.

The data set used during the model training comprised 6.8 billion posts that contained tags. To collect category labels for some of the tags, we sorted the tags in a decreasing order of popularity and the editors labeled the top ones with one or more categories. Following this process, we obtained 8,400 categorized tags. We show several examples in Table 2.

The models were trained using a machine with 96GB of RAM memory and 24 cores. The dimensionality of the embedding space was set to $d = 300$, and the context neighborhood size was set to $n = 5$. Finally, we used 10 negative samples in each vector update for negative sampling. Similarly to the approach in [13], most frequent tags were sub-sampled during the training.

5.2.2 *Inference*

When the vector representations of all tags are learned, we can find similar tags for a given tag by straightforward

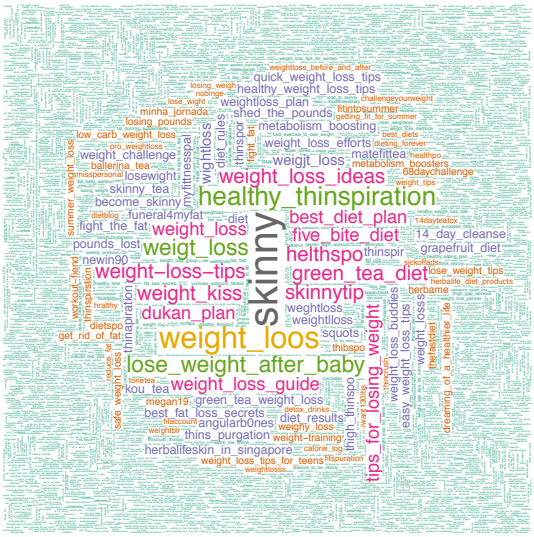

Figure 9: Nearest tags to "Health and Fitness/Weight Loss"

Table 3: Precision and recall of categorization methods

Method	Precision	Recall
LR-SG	0.71	0.65
k-NN-SG	0.82	0.62
SS-SG	**0.85**	**0.63**

k-nearest neighbor (k-NN) searches in the low-dimensional representation space. We use cosine distance [13] as a measure of similarity. To illustrate the usefulness of our approach, examples of similar tags to the tags *#makeup* and *#dress* are shown in Figure 7, where we can see that semantically related tags are grouped in the same part of the embedding space.

Similarly, we can find the most likely category for any tag by finding the nearest category in the vector space. To produce a confident set of categorized tags, we retrieved only tags with a cosine distance of 0.7 or higher to the corresponding category vectors. This threshold was obtained through editorial evaluation of the results. In total, more than 380,000 tags were confidently categorized into one or more categories. We show examples of categorized tags for the categories "Food and Drink/Desserts and Baking" and "Health and Fitness/Weight Loss" in Figures 8 and 9, respectively, with tag sizes in descending order of relevance.

A demonstration video of our tag categorization tool is available online at `http://youtu.be/ygn5oUBydfM`.

5.2.3 Evaluation

In order to quantify the benefits of our approach, we evaluated the method by excluding a random 1,000 tags from the editorially labeled set, and training the model using the remaining 7,400 labeled tags. We compared the SS-SG classification to the state-of-the-art logistic regression (LR) and k-NN, trained on the vectors learned by the original SG model. For LR classification (we refer to the method as LR-SG), we trained one classifier per interest category, while in k-NN (we refer to the method as k-NN-SG), for each test tag we found $K = 50$ nearest categorized neighbors and predicted the category by a majority vote. We report the results following a 5-fold cross-validation in Table 3. The results indicate that a classification based on our approach achieves higher precision than the competing methods, while at the same time maintaining the competitive recall measure.

Table 4: Language differences in post tags and post text

tag "hp" neighbors	word "hp" neighbors
harry potter	hewlett packard
hp movies	hp.com
hp books	hp computers
hp book quotes	hp company
harry potter facts	dell computers
hogwarts	hp printers

5.2.4 Model extensions

To be able to map more of Tumblr content to the GIT taxonomy, we trained two more semi-supervised skip-gram models, for: 1) keywords from post titles and bodies and 2) keywords from blog titles and descriptions. To train the models, we followed a similar procedure as before. Editors provided 4,700 categorized keywords, which were used to form training data sets for SS-SG model learning. Post keyword vectors were trained using a data set comprising $N = 6.8$ billion posts, while blog title and description keyword vectors were trained using $N = 37.1$ million blogs. These models were trained separately because of the language differences in these three domains, between the language used in post tags, post text, and blog title and description text. To justify our claim, in Table 4 we show the nearest neighbors of the tag "hp" from the tag SS-SG model and the word "hp" from the post text SS-SG model. As we can see in the table, "hp" has two different meanings in post tags and post text domains, referring to "Harry Potter" and "Hewlett-Packard", respectively.

To find the most confident keywords for each category, we calculated the cosine distances between keyword vectors and category vectors from the vocabulary. We retrieved 184,000 text blog keywords with a cosine distance of 0.7 or higher. We repeated the same procedure for keywords from blog titles and descriptions, resulting in 173,000 categorized keywords.

5.3 Forming interest segments

The goal of the task of interest prediction is to identify groups of users with an interest in certain topics, such as music, travel, cooking, or books, in order to allow advertisers to target the Tumblr audience according to their interests. Below we describe the method for predicting user interests which was used in this study.

In particular, after we obtained categorized tags and keywords, the interest score for user u_i in the k-th category at time t was calculated as

$$u^t_{i,cat=k} = \sum_{act \in \mathcal{A}_i} \sum_{kw \in act} \alpha^{(t-t_{act})}\, w_{kw}\; I(kw \text{ is of class } k), \tag{5.5}$$

where $\mathcal{A}_i$ is the set of all activities of user u_i, w_{kw} is the value of the keyword feature (e.g., if the post contains two mentions of the keyword "shakira" in the same time stamp, then the value is equal to 2, as explained in Section 4.2), while the indicator function $I(\cdot)$ returns 1 if the keyword

Table 5: Examples of interest inference based on categorized user features

User	Inferred interest	User profile
user 1	Arts and Entertainment/Movies	tag:{spoilers:30, shrek:18, hercules:12, cinderella:3, hobbit:123, hulk:21, pokemon:7, thor:58, ... disney:500, tarzan:8, marvel:385, wolverine:21, twilight:2, pixar:87, godzilla:1, x-men:53, ... pocahontas:4, avengers:134} txt:{aladdin:28, batman:10, bambi:12, movies:100} desc:{oscar:1, animation:12, comedy:1, movie:1, dvd:1}
user 2	Style and Fashion	tag:{ womensfashion:110, curls:6, fashiondiaries:133, redhair:2, menswear:125, chanel:4 ... springfashion:50, style:132, streetstyle:132, hairstylist:134, dapper:3, mensfashion:124} txt:{fashion:108}
user 3	Food and Drink	tag:{food:11, dessert:4 , soup:1, brunch:1, fruit:2, chicken:3, smoothie:1, cake:2, breakfast:2, ... ginger:2, salad:5, avocado:1} txt:{food:16, meals:6} follows:{user4542:1, user84852:1, user9332:1, user4524424:1 }
user 4	Home and Garden	tag:{daisies:2, kitchen:20, chair:3, art:81, outdoor:20, chandelier:12, lamp:8, window:2, bath:1 ... floral:17, home:3, wildflowers:1, flowers:102, interior:201, tree:1, flower:49, table:1, stairs:2, ... bedroom:56, wood:2, bathroom:26} txt:{garden:32, interior:17, home:41}
user 5	Automotive/Motorcycles	tag:{cars:24, ride:9, vehicle:22, riding:8, road:18} txt:{bike:8, motorcycle:10, riding:5, ride:9, road:10, vehicle:18, bikes:6, bicycle:2, scooter:1}

extracted from an activity is of class k, and 0 otherwise. In addition, we used the time stamp t_{act}, representing the day in which the activity happened, to exponentially decay less recent activities to account for passing interest (we used $\alpha = 0.99$ in our experiments). Note that the set $\mathcal{A}_i$, in addition to the user's original content, also included posts reblogged by user u_i.

The value of $u^t_{i,cat=k}$ represents an exponentially time-decayed count of all the activities in the k-th category. To effectively store the user profile for interest targeting, instead of storing all possible activities and their time stamps, we maintain a decayed sum of the activities and update the $u^t_{i,cat=k}$ daily.

Using this approach, we are able to qualify top K users in each category by sorting the interest score $u^t_{i,cat=k}$. Depending on the advertiser's goals and the category, the choice of K varies from campaign to campaign. We note that a single user can be qualified into one or more interest categories (e.g., a user can be categorized in "Sport", "Sport/Basketball", and "Health and Nutrition/Vitamins" at the same time), and, when the system was deployed, each user was assigned to 13 categories on average. An example of user profiles qualified into certain categories is given in Table 5.

5.3.1 Leveraging the follower graph

To be able to target Tumblr users who do not create much content, but who actively follow and engage with other blogs, we leverage the follower graph to create additional categorized features. Using equation 5.5 we can identify frequent bloggers in certain interest categories by focusing on a small percentage of users with a maximum $u_{i,cat=k}$. Following and liking posts created by social influencers in the k-th category serves as additional evidence of one's interest in that category.

In each interest category, we label the 5% of users with the highest number of activities in that category as frequent bloggers. Next, we update the interest score of all users u_i in the k-th category, in the following manner

$$u^t_{i,cat=k} += \sum_{b \in \mathcal{F}_i} \sum_{eng \in b} \alpha^{(t-t_{eng})}\ w_{eng}\ I(b \text{ is of class } k), \tag{5.6}$$

where $\mathcal{F}_i$ is a set of all frequent blogs followed by user u_i, eng are all engagements with the b-th blog, i.e. likes or follow actions, along with their weights w_{eng} (e.g., if the posts created by the b-th blog were liked ten times, then the value was set to 10; if a user followed the b-th blog, then the value was set to 1), while the indicator function $I(\cdot)$ returns 1 if the blog is of class k, and 0 otherwise. Similarly to other activities, we applied the exponential decay to the sum, based on the time stamps of follow and like actions.

We have observed that additional signals, in the form of follow and like engagement with frequent bloggers, increase our segment sizes, making it possible to efficiently target a greater number of users.

Table 6: A/B test results on 10% of user population

Campaign	Control	Targeted
Home and Garden	–	+9.71%
Style & Fashion	–	+42.53%
Sports/Outdoor Sports	–	+19.86%
Arts & Enter./Television	–	+24.37%
Arts & Enter./Video Games	–	+19.02%
Pets/Dogs	–	+27.21%
Arts & Entertainment 1	–	+9.08%
Arts & Entertainment 2	–	+6.54%

5.4 Results

In order to evaluate the generated user interest segments, we performed online A/B testing and worked with several advertisers who ran concurrent interest-targeted and untargeted campaigns. We tracked user engagement with their ads in terms of sponsored post likes, reblogs and follows, and show the results for 8 targeting campaigns in Table 6. We observed an average increase of 20% in user engagement (aggregate of 3 metrics) with sponsored posts in comparison to untargeted campaigns. This performance result represents a significant improvement over the baseline approach.

6. GENDER PREDICTION

In this section, we explain the details of our gender prediction model, based on the user profiles described in the previous sections. We first describe the generation process of a golden set of labeled users, which is used to train a predictive model that generalizes well on the remaining unlabeled users. This is followed by the model's description and a discussion of the results.

Table 7: Matching names in blog description (* represents matched name; 564K female and 395K male users found)

regex	count
my name is *	783,564
my name's *	291,811
me llamo *	47,663
the name's *	38,065
mi nombre es *	9,751
mi chiamo *	9,181
mein name ist *	1,025
meu nome e *	512
mon nom est *	215
mio nome e *	185

6.1 Collecting ground-truth labels

In order to train the machine learning method for gender prediction, in addition to user profiles, we also require labels that present the ground truth (i.e., "male" or "female"). However, Tumblr does not collect gender information when users sign-up, leaving open the question of how to obtain such data.

To address this problem, we proposed to leverage highly informative blog description data in order to infer user gender information. In particular, very often users declare their name in the blog description, as illustrated in Figure 1. To extract the users' declared names, we used several regular expression rules that we found to result in very high precision. The obtained results from a large set of name-matching regular expressions were editorially tested for quality. It was found that regular expressions reported in Table 7 yielded the most reliable extracted names (valid names were extracted in more than 95% of the cases). Next, in order to generate the gender ground truth, we used the US census data of popular baby names[10] from year 1880 to 2013 to create a "name → gender" mapping. Seeing as how certain names are given to both males and females, we used the empirical counts of babies with certain names from census data to generate the labels. More specifically, we used male/female empirical ratios as soft labels, with 1 indicating 100% confidence in a male and 0 indicating 100% confidence in a female name.

6.2 Proposed approach

Let $\mathcal{D}_g = \{(\mathbf{x}_i, y_i), i = 1, ..., N\}$ denote our gender data set, where N is the total number of labeled users, $\mathbf{x}_i$ is the K-dimensional user feature vector generated from the user profiles, and $y_i \in [0, 1]$ is the user label (real-valued number, ranging from 0 to 1). The feature vectors were generated from the user profiles described in Section 4.2, by disregarding time stamps (due to the fact that, unlike the users' interests, their gender does not fluctuate), and directly using the feature counts as values. To handle large counts, we normalized the counts by applying log transformation: assuming that the count is x, we replace the count by the value $\log(1 + x)$.

Our goal is to learn a gender-predictive model, $f : \mathbf{x} \rightarrow y$. As a classification model, we used logistic regression, parameterized by weight vector $\mathbf{w}$. We assume that the posterior gender probabilities can be estimated as a linear function of input $\mathbf{x}$, passed through a sigmoidal function,

$$\mathbb{P}(y = 1|\mathbf{x}) = f(\mathbf{x}, \mathbf{w}) = \frac{1}{1 + \exp(-\mathbf{x}^{\mathrm{T}}\mathbf{w})}, \tag{6.1}$$

and $\mathbb{P}(y = 0|\mathbf{x}) = 1 - \mathbb{P}(y = 1|\mathbf{x})$. To estimate the parameters $\mathbf{w}$, we minimize the following loss function,

$$\min_{\mathbf{w}\in\mathbb{R}^K} \frac{1}{N} \sum_{i=1}^{N} \big(y_i - f(\mathbf{x}_i, \mathbf{w})\big)^2 + \lambda \|\mathbf{w}\|_1, \tag{6.2}$$

where hyper-parameter λ controls the ℓ_1-regularization, introduced to induce sparsity in the parameter vector and reduce the feature space to a subset of features that are the most predictive. For data sets with a large number of features, as we have in our use case, it is common that many features are not useful for producing the desired learning result. For this reason, the ℓ_1-regularization was a critical part of our training procedure. In addition, we experimentally observed that the model generalizes better when we trained an initial model with ℓ_1-regularization to find which features have non-zero weight, and then do another round of training without ℓ_1-regularization, by only using features with non-zero weights from the first round to learn a better classifier.

Given a trained logistic regression model, the posterior class probabilities are estimated as $f(\mathbf{x}_i, \mathbf{w}) \in [0, 1]$. Then, the classification predictions are made by thresholding, as $\hat{y}_i = \mathrm{sign}(f(\mathbf{x}_i, \mathbf{w}) - \theta)$, where threshold θ is set between 0 and 1 to ensure the desired precision and recall according to specific advertisers requirements.

Table 8: Accuracy of gender model on hold-out set

Gender	Precision	Recall
female	0.806	0.838
male	0.794	0.689

Table 9: Editorial evaluation of random user predictions

Class Prediction	Correct	Wrong	Not sure
female	429	4	298
male	144	5	127

6.3 Results

To evaluate the accuracy of our gender prediction framework, we trained a logistic regression model on 70% of the golden set and tested on the remaining 30%. We used the Vowpal Wabbit [10] implementation on Hadoop to train the model. To illustrate the performance of our gender classifier, the performance results in terms of precision and recall measures are presented in Table 8. The threshold value θ was set to a value which ensured precision of 0.8.

In addition to evaluation on the hold-out set, we also editorially evaluated gender predictions on the unlabeled data set of user profiles. We randomly picked 1,007 gender predictions from the population of 64.1 million users and asked editors to visit their profiles and verify their gender. They were instructed to mark our predictions as "correct", "incorrect", or "not sure". The "not sure" grade is to be used when the visual inspection of a profile is inconclusive, as we found was often the case. The editorial judgment came back with 573 "correct" (429 females and 144 males), 9 "incorrect", and

[10] www.ssa.gov/oact/babynames/limits.html

425 "not sure" grades (see Table 9). The fact that there are so many "not sure" grades indicates that in many cases it is hard to infer the gender even after manual efforts, further indicating the benefits of the proposed approach and its superior performance in comparison to humans. Finally, we retrained the model with 100% of the golden set and deployed it in Yahoo production systems.

A demonstration video of the most predictive tags in each gender group is available online at `https://www.youtube.com/watch?v=jXGJ0TpOlhg`.

7. DEPLOYED SYSTEM

Due to the rapid growth of Tumblr and the large number of activities generated by the existing users, we implemented daily scoring of users in Yahoo production servers. We store the activities, i.e. raw counts as well as decayed counts, in Hive tables[11] for efficient retrieval. The decayed counts used in interest prediction are updated on a daily basis by multiplying the old feature values by the decay factor α and adding new activities. In order to infer the gender of new users we implemented daily scoring by leveraging MapReduce on Hadoop[12]. Both interest and gender models are retrained on a regular basis.

After thorough editorial evaluation of the inferred gender and interest targeting, both targeting frameworks were enabled through Gemini self-serve tool[13]. Advertisers can choose to use gender and/or interest targeting with custom segment sizes, allowing for effective targeting campaigns.

8. CONCLUSIONS

We have presented the steps in the development of a large-scale Tumblr gender and interest targeting framework, where we used historical Tumblr activities to create rich user profiles. We described the methodology, including a novel semi-supervised neural language model, as well as the high-level implementation details behind the deployed system. Currently, our gender and interest predictions cover users that generate more than 90% of Tumblr's daily activities, and are heavily leveraged by advertisers. In our ongoing work, we are concentrating on creating custom keyword-targeted advertising segments, specifically tailored for a particular advertiser, which include addressing of the problems of keyword discovery and expansion.

9. REFERENCES

[1] A. Ahmed, Y. Low, M. Aly, V. Josifovski, and A. J. Smola. Scalable distributed inference of dynamic user interests for behavioral targeting. In *KDD*, pages 114–122, 2011.

[2] J. Alba, J. Lynch, B. Weitz, C. Janiszewski, R. Lutz, A. Sawyer, and S. Wood. Interactive home shopping: consumer, retailer, and manufacturer incentives to participate in electronic marketplaces. *The Journal of Marketing*, pages 38–53, 1997.

[3] N. Barbieri, F. Bonchi, and G. Manco. Who to follow and why: link prediction with explanations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1266–1275. ACM, 2014.

[4] F. Bonchi, R. Perego, F. Silvestri, H. Vahabi, and R. Venturini. Efficient query recommendations in the long tail via center-piece subgraphs. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 345–354, New York, NY, USA, 2012. ACM.

[5] A. Z. Broder. Computational advertising and recommender systems. In *Proceedings of the ACM conference on Recommender systems*, pages 1–2. ACM, 2008.

[6] Y. Chang, L. Tang, Y. Inagaki, and Y. Liu. What is tumblr: A statistical overview and comparison. *ACM SIGKDD Explorations Newsletter*, 16(1):21–29, 2014.

[7] R. K. Chellappa and R. G. Sin. Personalization versus privacy: An empirical examination of the online consumer's dilemma. *Information Technology and Management*, 6(2-3):181–202, 2005.

[8] A. S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: Scalable online collaborative filtering. In *WWW*, pages 271–280. ACM, 2007.

[9] D. Essex. Matchmaker, matchmaker. *Communications of the ACM*, 52(5):16–17, 2009.

[10] J. Langford, L. Li, and T. Zhang. Sparse online learning via truncated gradient. *The Journal of Machine Learning Research*, 10:777–801, 2009.

[11] A. Majumder and N. Shrivastava. Know your personalization: Learning topic level personalization in online services. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 873–884, 2013.

[12] U. Manber, A. Patel, and J. Robison. Experience with personalization on Yahoo! *Communications of the ACM*, 43(8):35, 2000.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013.

[14] D. Riecken. Personalized views of personalization. *Communications of the ACM*, 43(8):27–28, 2000.

[15] D. Shin, S. Cetintas, and K.-C. Lee. Recommending tumblr blogs to follow with inductive matrix completion. In *RecSys 14 Poster Proceedings*, 2014.

[16] T. Singh, L. Veron-Jackson, and J. Cullinane. Blogging: A new play in your marketing game plan. *Business Horizons*, 51(4):281–292, 2008.

[17] S. K. Tyler, S. Pandey, E. Gabrilovich, and V. Josifovski. Retrieval models for audience selection in display advertising. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 593–598. ACM, 2011.

[18] C. Yi, T. Lei, I. Yoshiyuki, and L. Yan. What is tumblr: A statistical overview and comparison. arXiv:1403.5206v2, 2014.

[11] https://hive.apache.org
[12] https://hadoop.apache.org
[13] https://gemini.yahoo.com