SocialTransfer: Cross-Domain Transfer Learning from Social Streams for Media Applications*

Suman D. Roy †, Tao Mei ‡, Wenjun Zeng †, Shipeng Li ‡
† University of Missouri, Columbia, USA
‡ Microsoft Research Asia, Beijing, P. R. China
sdr5x8@mail.missouri.edu; {tmei, spli}@microsoft.com; zengw@missouri.edu

ABSTRACT

The usage and applications of social media have become pervasive. This has enabled an innovative paradigm to solve multimedia problems (e.g., recommendation and popularity prediction), which are otherwise hard to address purely by traditional approaches. In this paper, we investigate how to build a mutual connection among the disparate social media on the Internet, using which cross-domain media recommendation can be realized. We accomplish this goal through *SocialTransfer*—a novel cross-domain real-time transfer learning framework. While existing transfer learning methods do not address how to utilize the real time social streams, our proposed *SocialTransfer* is able to effectively learn from social streams to help multimedia applications, assuming an intermediate topic space can be built across domains. It is characterized by two key components: 1) a topic space learned in real time from social streams via Online Streaming Latent Dirichlet Allocation (OSLDA), and 2) a real-time cross-domain graph spectra analysis based transfer learning method that seamlessly incorporates learned topic models from social streams into the transfer learning framework. We present as use cases of *SocialTransfer* two video recommendation applications that otherwise can hardly be achieved by conventional media analysis techniques: 1) socialized query suggestion for video search, and 2) socialized video recommendation that features socially trending topical videos. We conduct experiments on a real-world large-scale dataset, including 10.2 million tweets and 5.7 million YouTube videos and show that *SocialTransfer* outperforms traditional learners significantly, and plays a natural and interoperable connection across video and social domains, leading to a wide variety of cross-domain applications.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Web-based services*

General Terms

Algorithms, Experimentation, Human Factors.

* This work was performed at Microsoft Research Asia.

MM'12, October 29–November 2, 2012, Nara, Japan.

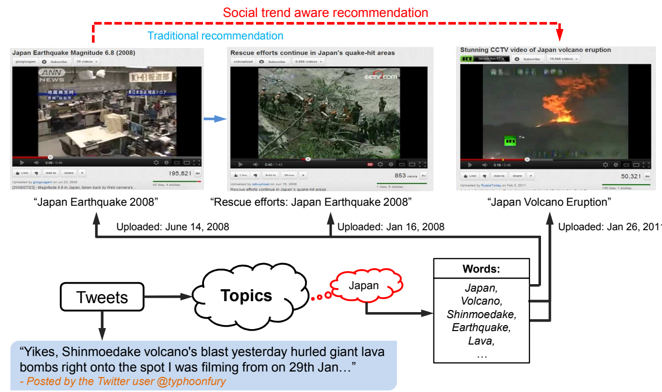


Figure 1: Example of using social topics in building social trend aware multimedia applications. In this example, we show that related video (i.e. video-video) recommendation can be enriched by using topics learned from the domain of social streams. This cross-domain transfer of knowledge is accomplished through a mutual topic space (e.g., the space includes the topics like "Japan" containing words like "volcano," "earthquake," and so on).

Keywords

Cross-domain media retrieval, recommendation, transfer learning, social media.

1. INTRODUCTION

Social media has become a disruptive platform for addressing many multimedia problems that could not be elegantly solved previously. For example, real-time social data is being utilized in semantic video indexing, image/video context annotation [17] and visualizing political activity and flu outbreaks [13]. Social streams like Twitter are a good indicator of crowd sourcing activity of a social community. The information in social streams is real time, thus it can be used to learn about real life events quickly. Major world events in recent times, such as the Egyptian Revolution, the London riots and the Japan Earthquake have been extensively captured using social streams such as Twitter and Facebook updates, as shown in Fig. 1.

One aspect of social micro-blogs like Twitter is its short text format, which is fast, real-time, and allows events to be instantly reported and broadcasted online [17]. Consider this in the light of a traditional media application like video recommendation, e.g., a user is watching an old video on "Egypt" (uploaded in 2008) on the eve of the "Egyptian revolutions" (Jan 29^{th} 2011). Also, consider a journalist who just uploaded a live video of the revolution in YouTube [10]. Most existing video recommendation systems have no way to *relate* these two videos, i.e., the seed video the user is watching and the related newly uploaded video [10, 26], in spite of their striking similarity in topics. It has been previously noted that use of social data could boost recommender systems [18]. Socialized recommendation using social streams has the potential to model factual world events in real time by topic extraction and subsequently perform interesting tasks such as video associations among the old and fresh videos belonging to similar topics. Learning fresh video associations is important to improve the performance of multimedia applications, such as video recommendation in terms of topical relevance and popularity. As shown in Fig. 1, fresh socially relevant videos (e.g., Japan's Shinmoedake Volcano Eruption in 2012) can be associated to older videos (e.g., Japan Earthquake in 2008) to facilitate social trend aware recommendations. Social trend aware recommendation includes videos belonging to related trending topics for recommendation.

However, social streams (like Twitter) and traditional media (like video publishing sites) exist across disparate domains on the Internet. Thus, the potential of these two resources is limited within the domain where it resides. For example, video recommendation [3, 16] and video popularity ranking [5] is often judged based only on view counts [10] or user activity (co-clicks) [26], but not on how socially trending the video topic is in real life. Thus, incorporating social knowledge into traditional media applications requires cross-domain information transfer, which contains the *wisdom of the crowds*. It is therefore important to develop a cross-domain knowledge transfer mechanism from the crowd-sourced social domain to traditional media (video) domain.

There are significant challenges in using social streams to perform cross-domain recommendations. The main concern is to transfer knowledge across domain and align features that are common to both domains (e.g., video tags and social stream topic words as shown in Fig. 1). Specifically, there are distinct challenges in cross domain socialized recommendations:

- A unified framework to combine the social and multimedia feature information which has different domain-specific properties.
- A transfer learning algorithm that can seamlessly propagate the knowledge (i.e., social topics) mined from the crowd-sourced social streams to the video domain.
- The scaling up and adaptation of the transfer learning algorithm to the ever bursty real-time nature of the social streams.
- Dealing with the noisy, incomplete, ambiguous, and short form nature of social stream data. Each tweet is limited to 140 characters and often improperly structured in grammar/syntax. Traditional language model (e.g., Bag-of-Words) would fail to scale up with such kind of data.

In this paper, we present *SocialTransfer*—a scalable technique for real-time transfer learning between the domains of social streams and traditional media (like video). *SocialTransfer* utilizes topics extracted from social streams to build an intermediate topic space in between the social and video domains. The topic space is an abstract space containing several clusters of words belonging to various topics that reflect world events in real time, including current and past trends. We use the Online Stream LDA model (OSLDA) to learn topics from social streams [1]. *SocialTransfer* uses a graph based framework to model the transfer learning problem (what feature information is transferable and how) between the social and the video domains. Spectral analysis of this graph fetches the eigenvectors, using which we can represent both the social and the video feature information as a combined feature representation [15]. Since the stream is temporal nature, *SocialTransfer* also allows progressively updating the topic space and seamlessly incorporating newer

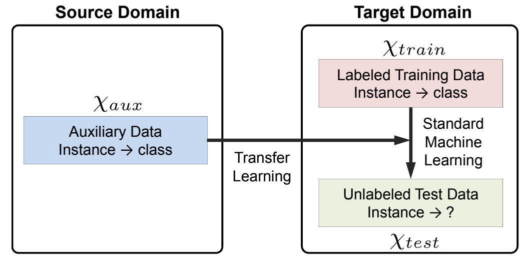


Figure 2: A basic framework for transfer learning. Cross-domain transfer learning assumes the class labels of instances of the source and target domain have the same overall category.

trends into the transfer learning framework for socially aware media recommendations. Fig. 1 shows an example of this kind. The framework we develop can be reused for several multimedia applications where social influence is capable of improving performance. Our results show that *SocialTransfer* considerably outperforms traditional learners without transfer learning.

The main contributions of this work are as follows:

- We propose the concept of bridging social media from disparate sources by building a common latent topic space, which represents one of the first attempts toward answering sociological problems using cross-domain social media.
- We propose *SocialTransfer*, a novel transfer learning framework based on efficient graph spectra analysis by seamlessly integrating the topic space learned from social stream in real time.
- We present several interesting media applications based on *SocialTransfer*, which could otherwise hardly be realized in conventional approaches, and evaluate through large-scale real-world social media data.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 introduces our proposed *SocialTransfer* and explains techniques employed to ensure scalable transfer learning from social streams. Section 4 describes two novel applications based on *SocialTransfer*. Section 5 describes the experimental data (videos and tweets). Section 6 presents experiments and evaluations, followed by conclusions in Section 7.

2. RELATED WORK

We discuss related work in the fields of transfer learning, mining social streams, and video recommendation.

Transfer Learning. Common machine learning techniques traditionally address isolated tasks. In contrast, transfer learning aims to transfer knowledge learned in one source domain and use it to improve learning in a related target domain. Fig. 2 shows the basic concept of transfer learning. The source domain data Z_{src} contains the auxiliary data, while target domain Z_{tar} contains the training and test data. A comprehensive survey of transfer learning techniques is provided in [21]. A unified framework for transfer learning in scenarios ranging from cross-domain, cross-category and self-taught learning is described in [8]. Transfer learning has been previously used in various cases including classification, image clustering, collaborative filtering, and sensor based location prediction [8, 21, 28].

Our *SocialTransfer* framework is inspired by the work in [8]. However, we distinguish ourselves from [8] in scaling transfer learning to specifically incorporate social stream data as source domain and show how topic learning can be smoothly combined with transfer learning in real-time. To the best of our knowledge, a framework that can handle social stream topics distinctively as source domain for cross-domain transfer learning has not been proposed before. This is challenging due to the unique characteristics of social stream data [17].

Domain-independent feature representation in transfer learning can also have significant effects on performance (e.g., to avoid negative transfer) [8]. Spectral techniques have been used to address the problem of combined feature representation [2]. However, such spectral techniques (e.g., eigenvector extraction [19]) should scale to dynamic social stream traffic, which is addressed in this paper. Although [24] attempts to use transfer learning for social recommendation, their model is not real time and limited to non-streaming data only. Instead, we show how to model transfer learning from streaming social data in real time, which is a significantly challenging problem not yet resolved.

Mining Social Streams. Social data from Twitter streams can be mined to build a relevant topic space using topic modeling [17, 22]. Such topic space can act as a bridge between the social and the traditional media domain, supporting multimedia applications like social video recommendation and social video popularity. Topic modeling aims to extract topics from large corpus of unlabeled documents by using generative models like Latent Dirichlet Allocation (LDA) [12]. There have been previous efforts to incorporate social data for recommendation [18, 24], but they do not use social streams specifically [22]. Social streams are more challenging to extract topics from; due to their dynamic, noisy, short and real-time nature [17]. Thus, large scale matrix decomposition is infeasible for social streams [18].

Previous research on mining social stream data assumes that the entire tweet stream is available to the algorithm at the beginning of the run. This assumption is only applicable in ideal case; it does not hold in real life situations. In our paper, we simulate the tweet stream in pseudo real-time, where the *SocialTransfer* algorithm has not seen the entire tweet stream in advance. Instead, the complete timeline is divided into time slots, and a certain number of tweets occupy each time slot as they are generated in real life, similar to the technique in [1]. Tweet chunks are fed to the *SocialTransfer* algorithm in time-sequential batches based on the time slots in which they are generated (pseudo real-time). We show later how *SocialTransfer* is a unique method to combine scalable social stream topic modeling and transfer learning; providing a natural interface for topic modeling to fit into the process of transfer learning and seamlessly integrate topic model and transfer learning.

Video Recommendation. Videos can be recommended based on several parameters, like multimodal fusion and relevance feedback [20, 25], user activity [10, 16], meta-data [6] and the social web [13]. In related video recommendation, the aim of the recommender is to recommend related videos with respect to a seed video that the user is currently watching. The list of related videos given a particular seed video can be considered to be a graph where the seed video is the root node and has an edge to every related video which are child nodes. Such a recommendation network of videos is called a Related Video Graph (RVG) [10], which is described in [26, 3] for the YouTube recommendation system. A detailed statistical analysis of the social network of YouTube videos is provided in [7]. Categorization of videos has also been a hot topic for research. There have been efforts to categorize videos based on related tags [23] and comments [11].

The fact that RVG-based recommendation has significant effect on video popularity (based on view counts etc.) is discussed in [5]. Efforts on video recommendation using certain traditional techniques without co-clicks fail to do any better. For example, recommendation using video modal features as described in [20, 25] might not be worthwhile since the calibration effort is too expensive [17]. However, recent research has shown that analyzing video meta-data such as tags, title, and comments can considerably help understand categorical association of videos [6, 11, 23].

Table 1: Notations of auxiliary, training, and test data for *SocialTransfer*

Dataset	Notation	Domain	Domain Type	Instances	Labels
Training	χ_{train}	Z_{tar}	Online video	$\{x_{tr}^{(m)}\}$	$\mathbb{C} = \{c^i\}$
Testing	χ_{test}	Z_{tar}	Online video	$\{x_{ts}^{(n)}\}$	N/A
Auxiliary	χ_{aux}	Z_{src}	Social stream	$\{x_{ax}^{(k)}\}$	$\mathbb{C} = \{c^i\}$

3. SOCIALTRANSFER

In this section, we will first present the problem statement and overview of *SocialTransfer*. Then, we will introduce online topic modeling, spectral graph learning, and algorithm of *SocialTransfer*.

3.1 Problem Statement

In *SocialTransfer*, we have two datasets in the target domain; the target training data $\chi_{train} = \{x_{tr}^m\}_{m=1}^M$ with labels and the target test data $\chi_{test} = \{x_{ts}^n\}_{n=1}^N$ without labels. The training data contains M instances whereas the test data contains N instances. Unlike traditional machine learning, we also have an auxiliary data set $\chi_{aux} = \{x_{ax}^k\}_{k=1}^D$. consisting of D tweets instances. We assume that the target data and the auxiliary data share the same categories (e.g., both a tweet and a video can be regarding music), but exist in different domains (e.g., tweet is social text-based micro-blogging while RVGs consist of videos).

Consider a set of B videos in the target domain. For a video v_i $(1 \leqslant i \leqslant B)$, we can represent the set of tags of v_i as $\{tags(v_i)\}$. Each tag in the set $\{tags(v_i)\}$ is a word, represented as $w_j^i, 1 \leqslant j \leqslant |tags(v_i)|$. Now consider a stream of D tweets picked from the source domain to be used for modeling the social topic space. For a tweet t_k $(1 \leqslant k \leqslant D)$, let $tpw(t_k)$ represent the topical words in the topic of t_k (we consider only the principal topic, i.e. topic for which the conditional probability of topic given tweet is maximum). Then each instance/label of the twitter stream data can be represented as $t_k \rightarrow tpw(t_k)$. These instances can be combined into the auxiliary data set $\chi_{aux} = \{x_{ax}^k\}_{k=1}^D$.

As shown in Table 1, all the instances $x \in \chi_{train} \cup \chi_{test} \cup \chi_{aux}$ are represented by the features in the feature space $\mathcal{F} = \{f_S^{(i)}\}_{i=1}^S$. Our goal is to learn an accurate classifier $f'(\cdot)$ from χ_{train} and χ_{aux} that can predict the testing data with minimum classification error. We call this classifier $f'(\chi_{test})$. Thus, the goal of transfer learning is to minimize the prediction error on χ_{test} by leveraging the auxiliary data from χ_{aux}.

3.2 Approach Overview

Our goal is to combine the training, test and auxiliary data into a single transfer framework for prediction. There are two problems we particularly need to solve in this framework: (1) we must learn the interconnected pattern of shared features between the source and the target data, and (2) since the topics modeled from social stream (auxiliary data) changes with the real world trends, we need a transfer framework that can allow *progressive inclusion of topics in pseudo real-time*.

Let us focus on the first problem and understand how to learn the interconnected structure of shared features across the domains. The single transfer framework we use for this purpose is represented as a graph called the transfer graph G (see Fig. 5), which contains the videos, tweets, feature words and category information. To learn

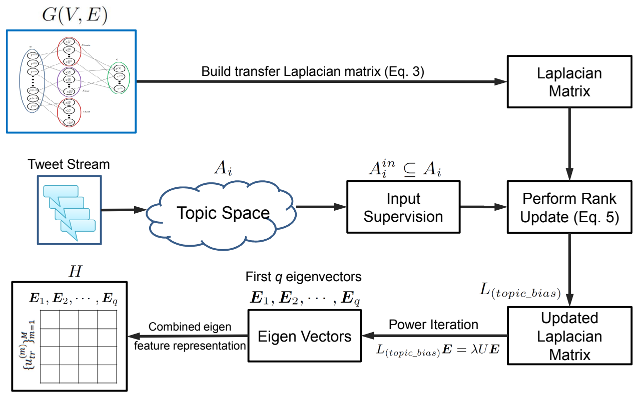


Figure 3: The flow diagram addresses the overall approach in solving the two key problems of *SocialTransfer*: (1) learning the shared feature representation across domains in terms of eigenvectors using Spectral Learning (Power Iteration), and (2) reflecting the progressive inclusion of topics by updating the transfer Laplacian matrix.

the interconnected pattern of shared features between the source and the target data, we perform spectral analysis [2] of the transfer graph. As shown in Fig. 3, spectral learning uses a technique called Power Iteration [14] to extract the eigenvectors from the Laplacian representation of the transfer graph. Spectral analysis of the transfer graph gives us the combined feature representation of the auxiliary and the training data using eigenvectors. This eigen feature representation reflects the intrinsic structure in terms of the principal components of the combined source and training data. Traditional learners, e.g., Support Vector Machines (SVM) [4], can then use the combined features for prediction rather than using only the training features.

Now, we focus on the second problem of how to progressively include social topics. Since the tweet stream is incrementally witnessed by the algorithm, the transfer graph needs to be updated in order to progressively include the twitter topics in pseudo-real time. Said alternately, in order to include topics as they are generated in real time, we must update the transfer graph and recalculate the graph spectra. This is achieved by treating the topics as input supervision before spectral learning (as shown in Fig. 3). In particular, to incorporate the new topical information to the existing transfer graph, we utilize selected topics from the topic space (described in Section 3.3) created from the tweets. We can treat the topical words of tweets and the corresponding topics as labeled instances, and then incorporate the new tweet information as a semi-supervised rank update on the existing Laplacian matrix (details in Section 3.5) as shown in the flow diagram Fig. 3 [1]. In other words, selected topics act as input supervision for the Laplacian matrix which allows for smooth incorporation of social topics into the transfer learning framework.

3.3 Learning Topics from Social Streams via OSLDA

We use the Online Streaming LDA (OSLDA) model for real-time topic learning from Twitter stream [1]. Each topic is comprised of a group of related words called *topical* words. Topic learning treats each tweet as a document and builds a generative model to connect the tweet to one or more topics. Thus, the topic of a tweet contains words (topical words) that are related to the tweet words but might not be explicitly present in the tweet itself. More precisely, the topic modeling generates two distributions, a tweet-topic distribution and a topic-word distribution.

As we have mentioned, extracting topics from social streams is non-trivial, due to the unique characteristics of social stream data [17]. Previous work has however shown that significantly popular topics (e.g. trending topics) can be extracted from social streams with reasonable accuracy [1]. Since every topical word in the topic space has an assigned topic label as shown in Table 2, the entire topic space can be treated as some sort of social bias for any semi-supervised learning task that requires social influence. Again, devising a natural way to incorporate this social bias into transfer learning is not trivial, which is one of the important issues addressed in this paper.

Note that each assigned topic consists of a cluster of topical words. Similarly, each topic can be considered a cluster in the topic space. We can limit ourselves to incorporating only selected topics from the topic space as input supervision (an additional set of labeled instances) for the transfer learning task. This choice will depend on factors such as whether we want to model only fresh (trending) topics or only video category specific topics. Thus for K topics in the global topic space, we can choose a particular set of topical words $A_i^{in} \subseteq A_i$, for $i = 1, 2, \ldots, K$ to act as the bias or input supervision to update the transfer graph before spectral learning. This sort of input topic supervision is fed into the transfer graph progressively, as is depicted in Fig. 4, where topics modeled in real-time from the social stream using OSLDA is used to update the transfer graph by means of a ranked update (Eq. 5) on the transfer Laplacian matrix representation of the transfer graph. This allows progressive and seamless inclusion of topics into the transfer graph as shown in Fig. 4, facilitating the social influence in transfer learning.

3.4 Transfer Graph

A general graph based framework for cross-domain transfer learning was proposed in [8], which includes the target and the auxiliary data with some common relations and attributes between them. We

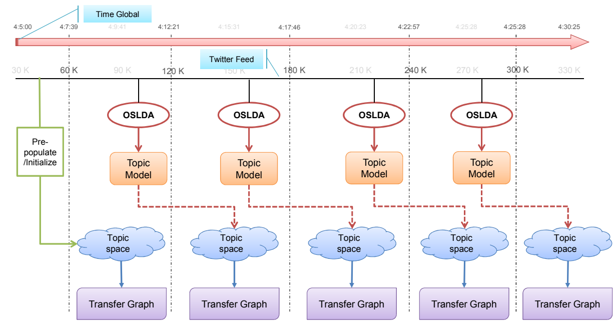


Figure 4: Updating transfer graph with topics modeled using OSLDA. The topic space is updated with new topics learned from previous tweet chunk, which is subsequently used to update the transfer graph.

Table 2: Example Topical Words and Related Topics

Topical Words	Assigned Topic	YouTube Category
dance, adventure, photography, visit	events	Travel & Events
anime, hero, online, celebrity, diva	films	Films & Animation
iphone, games, showcase	electronics	Sci. & Tech
war, economy, army, revolution, blog, egypt	politics	News & Politics
trailer, show, live, watch	entertainment	Entertainment
wow, rap, jam, gaga	music	Travel & Music

[1] A rank update refers to cases where a matrix is updated using outer product (as opposed to dot product).

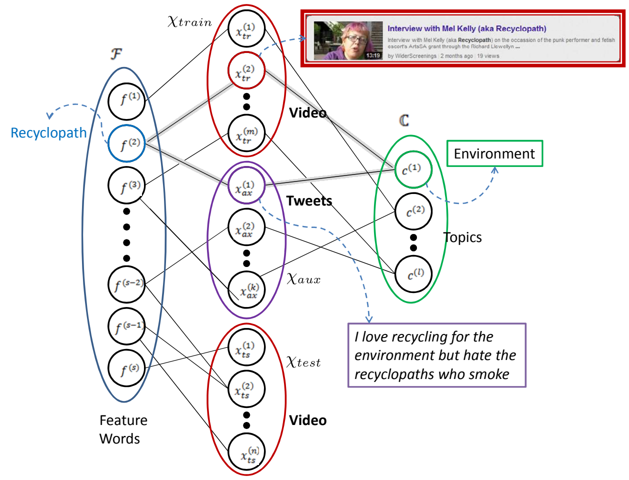


Figure 5: Transfer graph for *SocialTransfer* with connections among auxiliary and target data including features and class labels.

adapt that framework in our scenario. However, the graph in [8] cannot update itself to incorporate streaming tweets topic information in scalable fashion. Instead, the transfer graph in *SocialTransfer* is capable of updating itself with new tweets stream topics in real-time. The transfer graph's main purpose is to capture the cross-domain attributes of social streams and videos for using in the transfer learning task and model the relation between the auxiliary data from Twitter and the target video data. This 'transfer graph' (Fig. 5) contains the instances, features and class labels of the target data and the observed auxiliary data as vertices. The edges are set up based on the relations between the auxiliary and the target data nodes. The transfer graph presents a unified graph structure to represent the task of transfer learning from social domain to video domain.

Before diving into the details of the transfer graph, it is important we mention that the novelty of our approach lies in how we incorporate the learned social topics into this transfer graph. As described later in Section 3.5.1, we incorporate the learned topic model into the transfer graph by means of a ranked update on the Laplacian matrix representation of the transfer graph. SocialTransfer is a unique method to combine topic modeling and transfer learning; providing a natural interface for topic modeling to seamlessly fit into the process of transfer learning.

We can see the example in the transfer graph in Fig. 5. The feature word `'recyclopath'` occurs in the training video instance `'Interview with Mel Kelly' '(aka Recyclopath)'` shown in the top right. Since the video lacks any tags related to `'Environment'`, a traditional learner will find it difficult to extract the topic of this video to be related to `'Environment'`. However, the auxiliary data has a tweet instance belonging to the `'Environment'` topic having the word `'recyclopath'`. Thus, the transfer learner can label this video as 'Environment'-related and associate this video to another 'Environment'-related video. This is an example of discovery of video associations by understanding video topics with the help of social topics.

As shown in Fig. 5, the transfer graph $G(V, E)$ consists of vertices representing instances, features or class labels, and edges E denoting co-occurrences between end nodes in the target and the auxiliary data i.e.:

$$V = \chi_{train} \cup \chi_{test} \cup \chi_{aux} \cup \mathcal{F} \cup \mathbb{C} \tag{1}$$

The weight of each edge where one of the end nodes belongs to $\mathbb{C}$ indicates the number of such co-occurrences. Let $\omega_{(x,f)}$ represent the importance of the feature $f \in \mathcal{F}$ that appears in instance $x \in \chi_{train} \cup \chi_{test} \cup \chi_{aux}$. Then, the weight of an edge where one of the end nodes belongs to $\mathcal{F}$ is indicated by $\omega_{(x,f)}$. The importance of a feature word $\omega_{(x,f)}$ can be calculated using the topic-word probability distribution matrix obtained from OSLDA. The total number of features and class label nodes remain fixed in the transfer graph. Let $T(x)$ represent the true label of the instance. If e_{ij} denotes the weight of an edge between two nodes ϑ_i and ϑ_j in the transfer graph, then edge weights can be assigned as:

$$e_{ij} = \begin{cases} \omega_{\vartheta_i,\vartheta_j} & \vartheta_i \in \chi_{train} \cup \chi_{test} \cup \chi_{aux} \wedge \vartheta_j \in \mathcal{F} \\ \omega_{\vartheta_j,\vartheta_i} & \vartheta_i \in \mathcal{F} \wedge \vartheta_j \in \chi_{train} \cup \chi_{test} \cup \chi_{aux} \\ 1 & \vartheta_i \in \chi_{train} \wedge \vartheta_j \in \mathbb{C} \wedge T(\vartheta_i) = \vartheta_j \\ 1 & \vartheta_i \in \chi_{aux} \wedge \vartheta_j \in \mathbb{C} \wedge T(\vartheta_i) = \vartheta_j \\ 1 & \vartheta_i \in \mathbb{C} \wedge \vartheta_j \in \chi_{train} \wedge T(\vartheta_j) = \vartheta_i \\ 1 & \vartheta_i \in \mathbb{C} \wedge \vartheta_j \in \chi_{aux} \wedge T(\vartheta_j) = \vartheta_i \end{cases} \tag{2}$$

For all other cases except the ones mentioned in Eq. (2), we set $e_{ij} = 0$. The edge weights thus represent the occurrence/importance of a category or feature present in the auxiliary/target data, which will be eventually utilized as a distance metric during spectral clustering. Some nodes in the graph may be isolated with no edge connections. The matrix updating process (Section 3.5.1) adds new edges to the isolated nodes. The transfer graph G is usually sparse, symmetric, real and positive semi-definite, which allows the possibility of calculating its spectra efficiently [2]. The graph spectrum in terms of eigenvectors is the impression of the structure of relations among the source and target data. This structural relation between the cross domain data is the essence of transfer learning [8]. Thus, it is necessary to represent the source and target data as a transfer graph and then analyze their structural relation by learning the graph spectrum.

3.5 Learning Transfer Graph Spectra

The highlight of *SocialTransfer* is how it learns transfer graph spectra and incorporates new social topics into the transfer graph in real-time. This task is non-trivial, since if not properly done, it may incur substantial costs in terms of scalability (e.g., in eigen-feature extraction) and interoperability (in integration of topics) between topic modeling and transfer learning. In this section, we demonstrate how we achieve both these goals efficiently.

Once the transfer graph $G = (V, E)$ is built, we can use graph spectra analysis to form an eigen feature representation, which combines the principal component features from the training and the auxiliary data. In order to extract the top-q eigenvectors of the transfer graph $G = (V, E)$, we first need to convert the graph into a Laplacian matrix. Let $deg(\vartheta_i)$ denote the degree of the i-th vertex in G. Then the transfer graph Laplacian $L_{input} := (l'_{i,j})_{|V|\times|V|}$, can be obtained as:

$$l'_{i,j} := \begin{cases} deg(\vartheta_i) & if\ i = j \\ -1 & if\ i \neq j \wedge e_{ij} = 1 \\ 0 & otherwise \end{cases} \tag{3}$$

If the Laplacian eigen values are represented as $\lambda_0 = 1 \geqslant \lambda_1 \geqslant \cdots \geqslant \lambda_q$, then the eigen gap can be defined as: $eigengap = \frac{\lambda_q}{\lambda_{q-1}}$.

Since the Twitter stream is extremely dynamic, topics and trends change overtime. This requires a feature extraction scheme that can reflect and scale with the social stream. Previous approaches for spectral feature representation in transfer learning have suggested the use of the normalized cut (Ncut) technique for eigenvector extraction [8]. However, our experiments (Fig. 12 in Section 6.3) showed that the normalized cut technique is incapable of scaling with the twitter stream.

Therefore, we use a Power Iteration technique for computing the q largest eigenvectors of L_{input} [14]. The method begins with a random $|V| \times q$ eigenvector matrix and iteratively performs matrix multiplication and ortho-normalization until convergence [2]. The speed of convergence of this method depends on the eigen gap, i.e. the difference between successive eigen values. In fact, Bach *et.al.* mention that the number of steps required for the orthogonal convergence in the Power Iteration method is $O(\frac{1}{1-eigengap})$ [2].

Since topics are updated in the topic space with time, we need to devise a way to progressively incorporate these new topics into the transfer graph. These topics could be incorporated by picturing them to be a time-dependent labeled bias (like a semi-supervised bias) which is an additional set of labeled instances acting as input supervision. One option for incorporating the semi-supervised topic bias as input supervision into the Laplacian representation of the transfer graph (L_{input}) is by producing a ranked update on L_{input} (see Eq. 5). The update in effect recalculates the weights of edge/path between the features and the corresponding labels within the transfer graph, thus updating the characteristic of the Laplacian (Eq. 2 and 3). Essentially, the ranked update on the Laplacian using the topic bias adds positive weights between feature words that share the same topic and adds negative weights between feature words that belong to different topics. Thus, the target and the auxiliary data instances act as sort of virtual nodes enabling this re-weighing of the feature edges.

An additional reason for using the ranked update technique is that previous work has also rigorously demonstrated that when Laplacians such as L_{input} is positive semi-definite [19], a ranked update can improve eigenvector extraction speed by spreading the eigen gap. In the next, we will show the use of ranked updates to incorporate semi-supervised topic bias and update the transfer Laplacian.

3.5.1 *Incorporating Social Topics*

We know from topic modeling that the words in tweets can be clustered into topics. Let us consider there are K such topic clusters. The semi-supervised topic bias is implemented by assuming we know the correct topic labels for a subset of the feature words. Said in terms of the transfer graph, the supervised bias is an additional input of the correct cluster labels for a subset of the feature vertices. This input is learned by topic modeling using OSLDA, which was described in Section 3.3.

The semi-supervised bias consists of a set of topical words for each topic $A_i^{in} \subseteq A_i$ $(i = 1, 2, \ldots, K)$ that act as input supervision. Let us consider the simple case of two topic clusters A_1^{in} and A_2^{in}, such that $A^{in} = A_1^{in} \cup A_2^{in}$ denotes the set of labeled bias instances. Moveover, consider $d_i = \sum_j e_{ij}$ and $vol(A_k) = \sum_{i \in A_k} d_i$, then we can define a regularization vector δ_1 as:

$$\delta_1(i) = \begin{cases} \sqrt{\frac{d_i}{vol(A^{in})}} f(i), & i \in A^{in} \\ 0, & i \notin A^{in} \end{cases} \tag{4}$$

where $f(i) = \sqrt{\frac{vol(A_2^{in})}{vol(A_1^{in})}}$ if $i \in A_1^{in}$, and $f(i) = -\sqrt{\frac{vol(A_1^{in})}{vol(A_2^{in})}}$ if $i \in A_2^{in}$. The above Eq. 4 is to introduce a quadratic penalty if there is a violation in the topic bias label constraints. Said otherwise, this will cause vertices of features that belong to the same topic to cluster together while vertices of different topics will be assigned to separate clusters (due to the penalty). A rank-1 update on the original Laplacian can be made by

$$L_{topic\_bias} = L_{input} + \gamma \cdot \delta_1 \delta_1^T. \tag{5}$$

Algorithm 1 *SocialTransfer*: Transfer Learning from Social Stream

Input: A target classification task which includes the target training data χ_{train}, the source auxiliary data χ_{aux}, and the target test data χ_{test}.

Output: Classification result on χ_{test}

1: Construct the initial transfer graph $G(V, E)$ based on the social transfer clustering task (c.f. Section 3.2).
2: Calculate transfer Laplacian matrix: L_{input} from G by Eq. (3).
3: **for** each chunk of tweets entering the system **do**
4: Calculate the regularization vector δ_1 using the input supervision of social topics A^{in} by Eq. (4).
5: Perform semi-supervised topic bias update on transfer Laplacian: $L_{topic\_bias} = L_{input} + \gamma \cdot \delta_1 \delta_1^T$ by Eq. (5).
6: Use Power Iteration to calculate the first q eigenvectors of L_{topic_bias}: $\boldsymbol{E}_1, \boldsymbol{E}_2, \cdots, \boldsymbol{E}_q$ which satisfy the generalized eigenproblem: $L_{topic\_bias}\boldsymbol{E} = \lambda U \boldsymbol{E}$. The resulting eigenvectors will be used as initial eigenvectors for the next updated Laplacian matrix.
7: **end for**
8: Construct matrix H with $\boldsymbol{E}_1, \boldsymbol{E}_2, \cdots, \boldsymbol{E}_q$ as columns.
9: **for** each $\chi_{tr}^m \in \chi_{train}$ **do**
10: Let u_{tr}^m be the corresponding row in H w.r.t x_{tr}^m.
11: **end for**
12: Use a traditional classification algorithm (we use SVM) to train the classifier $f'(\chi_{test})$ based on $\mho_{tr} = \{u_{tr}^m\}_{m=1}^M$ instead of the original training set $\chi_{train} = \{x_{tr}^m\}_{m=1}^M$ and then classify $\chi_{test} = \{x_{ts}^n\}_{n=1}^N$ in the eigen feature space.

Similarly, if there are K topics, we can modify the original matrix L_{input} with a rank-*k* update [19] instead of a rank-1 update. This supervised ranked update firstly allows us to seamlessly incorporate streaming data progressively. Secondly, it aims at tuning certain algebraic properties of the input Laplacian matrix which are related to the convergence rate of the Power Iteration method, eventually speeding the eigen decomposition [19].

In summary, the input supervision using topics learned from the social stream allows us to implement rank-*k* updates on the transfer-Laplacian matrix as a similarity learning mechanism, where vertex similarities are adjusted on the basis of the topic bias. Note that the number of nodes in the graph is not changed during updating (dimension $|V|$ is fixed); instead the updates only introduce new edges or re-weights existing edges in the graph as it iteratively reuses the eigenvectors from previous update. Due to lack of space, we refrain from describing in detail how the rank-*k* update improves the speed of eigenvector extraction. Essentially, the ranked update increases the eigen gap, which accelerates the convergence of the Power Iteration method. For a detailed explanation of how a supervised bias using rank-*k* update accelerates the eigenvector extraction process, please refer to [19].

3.6 Algorithm for SocialTransfer

Once the first q eigenvectors $\boldsymbol{E}_1, \boldsymbol{E}_2, \ldots, \boldsymbol{E}_q$ have been found by iteratively using the Power Iteration method with the topic-based input supervision, we can form a combined feature representation that depends on both the training and the auxiliary data. Traditional learners like SVMs can use the combined features that include the transfer task to train a classifier $f'(\chi_{test})$. Algorithm 1 shows the proposed *SocialTransfer* for classification in the target domain based on auxiliary social streams.

4. SOCIAL APPLICATIONS

We present two applications which are difficult to realize purely by traditional multimedia techniques (i.e., without incorporating knowledge from social data). These applications, though seemingly unrelated, can both be boosted by learning from the social domain using the intermediate topic space. *SocialTransfer* can be potentially applied to a wide variety of new social applications by effectively transferring the knowledge from social to multimedia domain.

4.1 Socialized Query Suggestion for Video Search

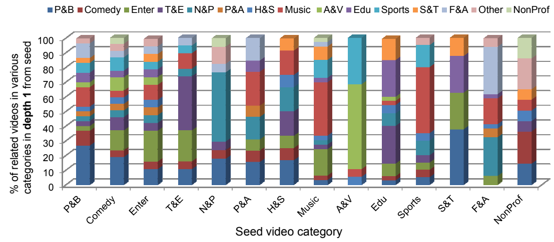
We first describe an application using the topic modeling via OSLDA within the *SocialTransfer* framework. Our intuition is that lack of a collaborative cross-domain recommendation environment compels users into unguided video search (pure querying rather than smart recommendation). One effect of such activity is that users will use the words of trending issues and topics (topical words) when performing video search queries on the Internet. Learning topical words in real time from social streams could be leveraged to suggest queries for video search. We believe this is an important application of real time topical analysis from social streams. In Section 6.1, we show experimental results which suggest: (1) user search queries in video search engines do contain words which we recovered as topical words from social streams using OSLDA. (2) There is a noticeable time lag between (a) OSLDA topic trend detection from social stream and (b) the increasing volume of search queries on that trend in an external (non-social stream) video search portal. This correspondence can be leveraged to augment user experience by socialized query suggestion for video search.

Socialized query suggestion for video search using the OSLDA model in *SocialTransfer* aims to recommend good query words in response to users query keywords. This will help searchers to better seek the more topic-relevant videos they are looking for, since the suggested topical words are connected to videos in the transfer graph. Said alternately, socialized query suggestion aims to localize the topic of the video the user is querying for by suggesting additional topical words. This is more effective in relevant video retrieval than just matching query keywords to video tags. Therefore, the prior knowledge of which query words the user will use for video search will not only enable the system to suggest better topical words for the user, but also improve the system capability in predicting which keywords the user will use for search and which videos the user will potentially watch in the future.

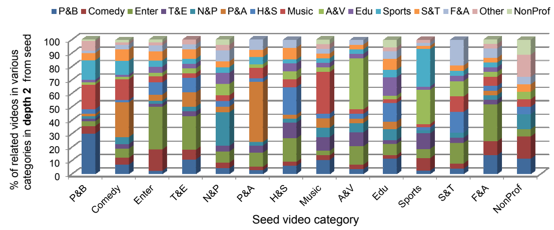
4.2 Socialized Video Recommendation

Modern video publishing sites like YouTube use related video recommendation techniques [7] based on RVGs to recommend a video to the user. The recommended video is related to the seed video which the user is currently watching by co-clicks or co-views, i.e. some signed-in user has clicked on both the seed and the related video [10] in sequence. In contrary, Socialized Video Recommendation recommends videos which bear similar topics to a seed video the user is watching. Thus, Socialized Video Recommendation is independent of the click through nature among videos, and has several advantages over traditional related video recommendation [7, 26], such as: (1) it considers video content/context (as in topics) while recommending videos, (2) its performance does not decrease as click data gets sparse, (3) it can recommend fresher videos that do not have significant user activity but are extremely relevant to the seed video and (4) it does not require signed-in user activity to learn and build RVGs.

For related video recommendation, the system must be able to predict which videos are 'related' to a seed video and are good candidates for recommendation. The first step in solving this is to detect the topic of the seed video. Thus, the job of the classifier is to classify the topic of a test seed video. Once we detect the topic



(a) depth = 1



(b) depth = 2

Figure 6: Distribution of video categories of recommended videos by RVG at the depth 1 and 2 from the seed video, showing the diverse nature of video recommendation in YouTube.

of the test seed video, we can assume all the videos belonging to that topic are candidates for related video recommendation of the seed. We then recommend only those videos from the candidate pool whose tags match the seed. A socialized video recommender can be developed by creating a learner that uses auxiliary tweet data by means of transfer learning. Given a set of RVG videos in the target domain, a traditional Non-Transfer learner like SVM [4] will aim to predict the related videos of a given seed video in the test data set by building a classifier only from the training data. Instead, *SocialTransfer* builds a classifier using both training video data and auxiliary tweet data.

5. DATA DESCRIPTION

Our study is based on a 5.7 million videos crawled from YouTube and 10.2 million tweets obtained from the NIST Twitter dataset [9]. The source domain is Twitter and the target domain is YouTube. The notations for data from each domain are included in Table 1.

5.1 YouTube Videos—Target Domain

We use a preliminary list of YouTube related video ids collected for experiments in [7]. Video meta-data includes values for entities such as video id, title, tags, view count, age (in days since uploading), category, related video ids (which comprises related videos at depth 1 of RVG) etc. As mentioned earlier, the videos related to a given seed video is captured using a directed graph, which is known as Related Video Graph (RVG) [10, 3]. Thus, if a video y is in the related video list of a seed video x, then there is a directed edge/path $x \rightarrow y$ in the RVG [10]. Moreover, for an edge $x \rightarrow y$ in the RVG, its tags can be represented as the instance: $\{tags(x)\} \rightarrow \{tags(y)\} - \{tags(x)\}$, where $\{tags(x)\}$ represents the set of tags of video x and '$-$' means set difference. All such instance/label combinations of the B videos have to be randomly divided into two sets for training and testing, called χ_{train} and χ_{test} respectively; where $|\chi_{train}| = M$ and $|\chi_{test}| = N$ represented as $\chi_{train} = \{x_{tr}^m\}_{m=1}^M$, $\chi_{test} = \{x_{ts}^n\}_{n=1}^N$ where $M + N = B$ and $\frac{M}{N} \approx 1.5$.

We have collected related video information up to five depths from an initial seed video ranging across the 14 main YouTube categories: Comedy, Entertainment ('Enter'), Education ('Edu'),

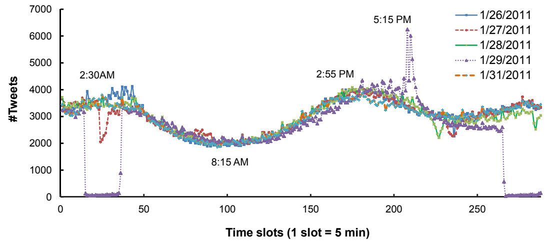


Figure 7: Daily tweet stream during 26^{th}—31^{st} Jan. 2011.

Music, Film & Animation('F&A'), Non-Profits & Activism ('Non-Prof'), Science & Technology ('S&T'), Travel & Events ('T&E'), Pets & Animals (P&A), HowTo & Style ('H&S'), Autos & Vehicles ('A&V'), News & Politics ('N&P'), Sports, and People & Blogs ('P&B'). Some videos are categorized unavailable, and in such cases we use the category of its parent video. Apart from these main categories, [11] has suggested around 75 sub-categories to the main YouTube categories. We include all of these as the pool of categories from which class labels can be drawn. Therefore, we tune the OSLDA to detect tweets where the tweet words fall into the tag space belonging to any of these category videos.

Since RVGs are essentially related recommendation networks, distribution of categories over videos changes as we move one depth to the next. This introduces some degree of intended diversity in the next video recommended [10], since it might be of a different category compared to the seed but somehow related. Although [10] mentions the existence of this diversity in YouTube recommendations, it does not provide quantitative results to capture the diversity. Instead, we have thoroughly analyzed the dataset and used a histogram to study the diversity issue in YouTube video recommendations. Fig. 6 shows the category distribution of related videos at depth 1 and 2 from the seed video. In other words, the X-axis represents the category of the seed video while the Y-axis represents the category distribution of the recommended video in succesive levels. On average, we found that the next recommended video has 25% chance of being in the same category as the seed video.

5.2 Twitter Streams—Source Domain

The Twitter dataset consists of 10.2 million tweets generated in the US and collected between Jan. 26^{th} 2011 and Feb. 11^{th} 2011. We simulate the twitter data as a stream, with each batch of tweets representing approximately five min. The resulting rate at which tweets streams over the last week of Jan, 2011 is shown in Fig. 7, where the 5 min batch time slots account for a total of 288 slots spanning 24 hours in the horizontal axis. We show the temporal stream volume (in No. of tweets generated) distribution only across seven days in order to avoid cluttering in Fig. 7.

From Fig. 7 we can conclude that under normal circumstances, the tweet rate distribution has a general pattern over 24 hours: there is a minima around 8:15 AM, followed by a gradual rise until 3 PM in the afternoon, where a local maxima is achieved. Interestingly, another spike is usually noticed in tweets around 2:30 AM in the morning. The drops to almost zero on Jan. 29^{th} can be accounted for by Twitter downtimes and the Blackberry outage in USA. The high spike around 5:15 PM on Jan. 29^{th} is caused due to a high volume of tweets during the onset of the Egyptian revolutions.

6. EXPERIMENTS

We focus our experiments for the first two applications mentioned in Section 4, namely socialized query suggestion for video search using OSLDA and socialized video recommendation.

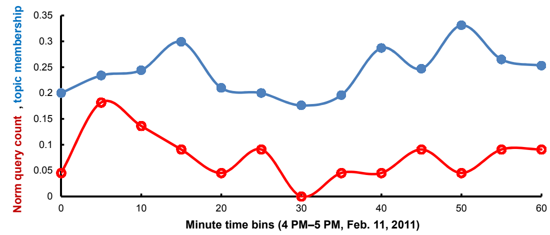


Figure 8: OSLDA trend detection on Twitter (top blue) vs. topical word search trend in a commercial video search engine (bottom brown).

6.1 Evaluations on Socialized Query Suggestion for Video Search

Experiments were conducted using video query logs from a commercial video search engine and 10.2 million tweets. The goal is to find a temporal pattern or common terms between tweet topic words and video search keywords from video logs. Fig. 8 shows the distribution of search queries with time in video query logs for the topic *'Egypt'* with real-time trend variation on Twitter as detected by OSLDA.

We notice that there is few min's time lag between a trend topic appearing on Twitter, and the same topical words being searched on the commercial video search engine. This means as trends rise and fall in Twitter, the volume of queries on the same topic rises and falls for video search. To further support our claim that people search for Twitter trends outside Twitter, Fig. 9 shows the query keywords used in a commercial video search engine on Feb 11, 2011. If we eliminate daily searches such as 'cats,' 'movies,' and 'funny commercials' which are common (green dotted circles), then it is hard to miss that topical words (red solid circles) take up a significant portion of the remaining video search keywords. In the video search engine logs and for all queries on Feb 11^{th} that are not daily search terms (like 'cats'), 63% of query words were detected by OSLDA.

In fact, this technique of socialized query suggestion can be extended beyond video search. We used Google Insights to understand search patterns for web and image search on Feb 11, 2011. It was not surprising that *'Egypt'* was the hottest search topic that day. In fact, Google Web Insights provided us with the top 10 web search keywords related to *'Egypt'*; 7 of which had already been detected by OSLDA earlier. For Google Image search, 6 of the top 10 search keywords were detected by OSLDA. This is convincing evidence that the OSLDA detects relevant socially active topics within the *SocialTransfer* framework.

6.2 Evaluations of Socialized Video Recommendation

For socialized video recommendation, we test *SocialTransfer* against a traditional learner like SVM [4], where *SocialTransfer* uses auxiliary social data in combination with training data, whereas a traditional learner uses only the training data for prediction (called Non-Transfer) and serves as our benchmark. Here, the classification task is simple: given a seed test video, classify whether another video is a related video of the seed or not. This in aggregation is same as the case: given a seed test video, predict the list of related videos for the seed test video.

For the experiments, we set $\gamma = 1.25$, limit the power method to extracting top-25 eigenvectors and include 60% of the topic space for input supervision. The reasoning of these choices is explained over the following sections. We have three datasets for transfer learning—the target training data, the target test data and the source

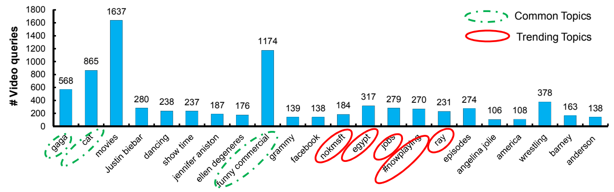


Figure 9: A significant majority of video search keywords come from trending topical words (red circles).

Table 3: Number of video instances in popular categories.

Category	#Training video	#Testing video
Comedy	254,254	169,502
Film & Animation	249,697	166,464
Entertainment	740,225	493,484
Sports	221,411	147,607
People & Blogs	252,617	168,412
Music	720,554	480,362

auxiliary data. The target dataset consists of 5.7 million videos in total along with their RVGs (contains a list of related videos for each seed video). However, some videos do not have categories or are removed from YouTube, and therefore we experiment on a reduced set of 4.8 million videos. Our training data consists of 60% videos randomly picked from the 4.8 million YouTube videos. The rest 40% videos ($\sim$ 2 million) are used for testing. As auxiliary data, we use the 10.2 million tweets from the Twitter stream. We ensure to extract topics from tweets based on approximately 90 categories (16 main + 75 other) so that the source and target domains share same categories. Additionally, we also evaluate category-specific predictions based on six popular categories (Comedy, Film & Entertainment, Sports, People & Blogs, Music). Table 3 shows the number of video instances used for evaluation in some of the popular categories.

In Table 4, we report the average error in prediction for the Non-Transfer cases (SVM on training only) vs. *SocialTransfer*. Non-Transfer refers to application of the traditional SVM learner to the original target dataset with no social influence (only training features are used); *SocialTransfer* means to apply SVM on the combined feature representation learned using transfer learning from social data (training + auxiliary).

The performance in Table 4 is measured in error rate by averaging 10 random repeats on each dataset by the two evaluation methods. For each repeat, we randomly select $5,000$ instances per category as target training data. We report the prediction error rate in each of the main categories, along with the overall error for the entire data set. We also report the standard deviation of the repeats in Table 4. The two methods are well-tuned using 10-fold cross validation. The overall gain using *SocialTransfer* is around 35.1% compared to non-transfer settings. Please note that the overall error rate is averaged over all the main categories and not just the six categories shown in Table 4. Performance improvement using transfer learning is most in category 'People & Blogs.' In all the major categories, *SocialTransfer* performs better than a traditional non-transfer learner. We further test the prediction error in the *SocialTransfer* framework based on several factors including scalability and learning capability per day of stream inflow.

Accuracy variation with tweet stream inflow. We test the rate at which the prediction error decreases with incoming stream of tweets (Fig. 10) every day across 12 days of the social data (Jan 26^{th}—Feb 7^{th}). It is clear from Fig. 10 that the influence of Twitter is significant (rapid decrease in error in first three days), and the chatter over a week consists of substantial number of words in

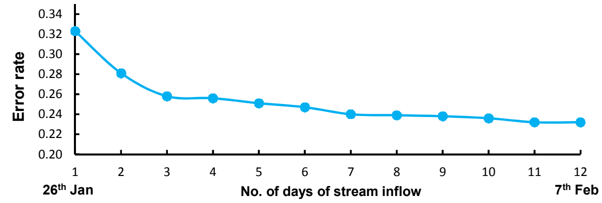


Figure 10: Drop in prediction error rate with daily stream inflow from Twitter.

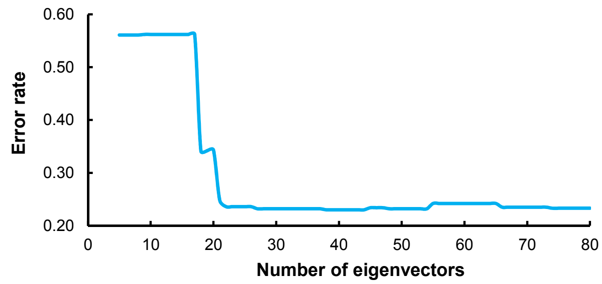


Figure 11: The influence of the number of eigenvectors extracted on the error rate.

every video category. However, some topical words co-occur in different topics, for example the word 'live' might refer to music or entertainment. On course of the 12 days, we see a 41.4% net reduction of error. Interestingly, within the first three days (26^{th}—29^{th} Jan.), the error is reduced by almost 20%, while the rest of the nine days involve an almost equal amount of error reduction.

Eigenvectors. Previously we mentioned that for the experiments, we fix the number of eigenvectors to be extracted from the transfer Laplacian to 25. The reason for this choice is due to results of Fig. 11, which shows the variation of the error rate with the number of eigenvectors extracted. We see that when the number of eigenvectors extracted is greater than 23, the error rate is almost constant. However, there is a trade-off between the time required for extraction vs. error rate of prediction for a certain number of eigenvectors to be extracted. Thus, since the variation of reduction in error rate is not significant beyond $22-25$ eigenvectors, we can safely assume that the extraction of more than 25 eigenvectors is not necessary.

Scalability. The speed at which the incoming stream of tweets is explored for topics by OSLDA together with the time required for eigen feature extraction from the transfer graph using spectral learning is important for maintaining scalability with the real-time social stream. In our system, the topic modeling is done in parallel with the eigenvector extraction to save time. Thus, our main aim should be to limit the time required to complete either of these tasks within the incoming tweet flow time.

Fig. 12 shows the comparison of runtimes for various settings of OSLDA, eigenvector extraction using power iteration (PI) and eigenvector extraction using Normalized cut (Ncut) with the time taken on average for an incoming chunk of tweets to stream in. For OSLDA, '$20k$' (in legend) refers to 20 topics extracted and '$50i$' refers to 50 iterations of the generative process. Experiments were run on a IBM server with 2.67 GHz processor and 8 GB RAM.

From Fig. 12, we can safely conclude that the model scales to incoming bursts of tweets, since the matrix decomposition with Power Iteration and the topic modeling using OSLDA require less time than the speed of incoming tweets. Note that Ncut does not scale as it takes longer time to extract eigenvectors than the speed of the incoming burst of tweets. Moreover, for more than $40,000$ tweets, Ncut causes our system to run out of memory.

Table 4: Experimental results in error rate for predicting relevant videos for Socialized Video Recommendation. The results are the averages of 10 random repeats along with their standard deviations. Both methods are tuned with 10-fold cross validation.

Approach	Overall	Comedy	Film & Animation	Entertainment	Sports	People & Blogs	Music
Non-Transfer	0.357 ± 0.049	0.429 ± 0.059	0.334 ± 0.063	0.386 ± 0.023	0.394 ± 0.072	0.247 ± 0.066	0.356 ± 0.036
SocialTransfer	0.232 ± 0.043	0.397 ± 0.065	0.242 ± 0.051	0.219 ± 0.015	0.282 ± 0.082	0.112 ± 0.032	0.230 ± 0.029

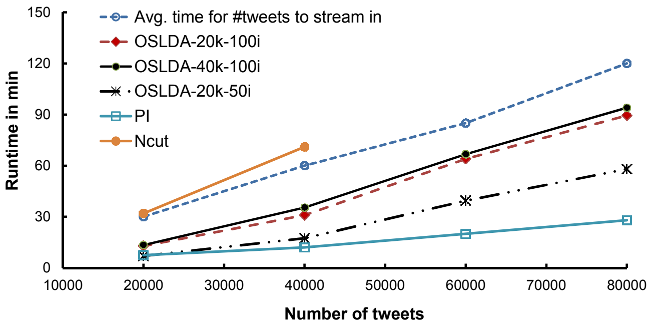


Figure 12: Runtime comparison for topic modeling and eigen decomposition with incoming tweet stream in *SocialTransfer*.

7. CONCLUSION

This paper presents *SocialTransfer*, a novel cross-domain real time transfer learning approach based on social streams. *SocialTransfer* can be applied to various multimedia applications which are boosted by the knowledge acquired from cross-domain data. We demonstrate the use of the *SocialTransfer* in realizing two unique applications, namely socialized video recommendation and socialized query suggestion for video search. Evaluation of video popularity prediction using *SocialTransfer* is left for future work. To aid the transfer process across domains in scaling with dynamic social streams, we propose a tweet topic based semi-supervised bias/ input supervision on the transfer graph that can help seamlessly incorporate new trend topics and accelerate the extraction of combined social and video features using eigen decomposition. To the best of our knowledge, such a framework that unites transfer learning and topic modeling in a scalable fashion for social stream data has not been proposed before. Experimental results show that *SocialTransfer* can outperform traditional learners by almost 35.1% reduction of error rate in predicting related videos during recommendation. The main contribution of this work is the scalable model for cross-domain real time transfer learning from social streams and using social topics in building novel multimedia applications that can hardly be realized by traditional multimedia techniques alone.

8. REFERENCES

[1] S. D. Roy, T. Mei, W. Zeng and S. Li. Empowering Cross-Domain Internet Media With Real-Time Topic Learning From Social Streams. In *Proc. of ICME*, 2012.

[2] F. R. Bach and M. I. Jordan. Learning spectral clustering, with application to speech separation. *Journal of Machine Learning Research*, 7:1963–2001, 2006.

[3] S. Baluja, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, and M. Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In *Proc. of WWW*, 2008.

[4] B. E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. of Workshop on Computational Learning Theory*, 1992.

[5] G. Chatzopoulou, C. Sheng, and M. Faloutsos. A first step towards understanding popularity in youtube. In *Proc. of INFOCOM Workshop*, 2010.

[6] Z. Chen, J. Cao, Y. Song, J. Guo, Y. Zhang, and J. Li. Context-oriented web video tag recommendation. In *Wide Web Conference*, pages 1079–1080, 2010.

[7] X. Cheng, C. Dale, and J. Liu. Statistics and social network of youtube videos. In *Proc. of Inter. Workshop on Quality of Service*, pages 229–238, 2008.

[8] W. Dai, O. Jin, G.-R. Xue, Q. Yang, and Y. Yu. Eigentransfer: a unified framework for transfer learning. In *Proc. of ICML*, page 25, 2009.

[9] http://trec.nist.gov/data/tweets/.

[10] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. V. Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston, and D. Sampath. The youtube video recommendation system. In *Proc. of International Conference on Recommender Systems*, pages 293–296, 2010.

[11] K. Filippova and K. B. Hall. Improved video categorization from text metadata and user comments. In *Proc. of SIGIR*, 2011.

[12] M. D. Hoffman, D. M. Blei, and F. R. Bach. Online learning for latent dirichlet allocation. In *Proc. of Neural Information Processing Systems*, pages 856–864, 2010.

[13] X. Jin, A. C. Gallagher, L. Cao, J. Luo, and J. Han. The wisdom of social multimedia: using flickr for prediction and forecast. In *Proc. of ACM Multimedia*, pages 1235–1244, 2010.

[14] F. Lin and W. W. Cohen. Power iteration clustering. In *Proc. of ICML*, pages 655–662, 2010.

[15] X. Ling, W. Dai, G.-R. Xue, Q. Yang, and Y. Yu. Spectral domain-transfer learning. In *ACM Conference on Knowledge Discovery and Data Mining*, pages 488–496, 2008.

[16] H. Luo, J. Fan, D. A. Keim, and S. Satoh. Personalized news video recommendation. In *Proc. of MMM*, pages 459–471, 2009.

[17] M. Namaan. Social multimedia: highlighting opportunities for search and mining of multimedia data in social media applications. *Multimedia Tools and Applications*, 56(1):9, 2012.

[18] H. Ma, T. C. Zhou, M. R. Lyu, and I. King. Improving recommender systems by incorporating social contextual information. *ACM Transactions on Information Systems*, 29(2):9, 2011.

[19] D. Mavroeidis. Mind the eigen-gap, or how to accelerate semi-supervised spectral learning algorithms. In *Proc. of IJCAI*, pages 2692–2697, 2011.

[20] T. Mei, B. Yang, X.-S. Hua, and S. Li. Contextual Video Recommendation by Multimodal Relevance and User Feedback. *ACM Trans. on Information Systems*, 29(2):10, 2011.

[21] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Trans. on Knowledge and Data Engineering*, 22:1345–1359, 2010.

[22] D. Ramage, S. T. Dumais, and D. J. Liebling. Characterizing microblogs with topic models. In *Proc. of AAAI Conf. on Weblogs and Social Media*, 2010.

[23] G. Toderici, H. Aradhye, M. Pasca, L. Sbaiz, and J. Yagnik. Finding meaning on youtube: Tag recommendation and category discovery. In *Proc. of CVPR*, pages 3447–3454, 2010.

[24] Q. Xu, E. Xiang, and Q. Yang. Social-behavior transfer learning for recommendation systems. *Proc. of Workshop on Social Web Mining*.

[25] B. Yang, T. Mei, X.-S. Hua, L. Yang, S.-Q. Yang, and M. Li. Online video recommendation based on multimodal fusion and relevance feedback. In *Proc. of CIVR*, pages 73–80, 2007.

[26] R. Zhou, S. Khemmarat, and L. Gao. The impact of youtube recommendation system on video views. In *Proc. of International Conference on Internet Measurement*, pages 404–410, 2010.

[27] L. Wu, L. Yang, N. Yu, and X.-S. Hua. Learning to tag. In *Proc. of WWW*, pages 361–370, 2009.

[28] P. Zhao, S. C. H. Hoi. OTL: A Framework of Online Transfer Learning. In *Proc. of ICML*, pages 1231–1238, 2010.