Masthead Logo

University of Iowa
Iowa Research Online

Theses and Dissertations

Spring 2013

Knowledge transfer: what, how, and why

Si-Chi Chin
University of Iowa

Copyright 2013 Si-Chi Chin

This dissertation is available at Iowa Research Online: https://ir.uiowa.edu/etd/2456

Recommended Citation
Chin, Si-Chi. "Knowledge transfer: what, how, and why." PhD (Doctor of Philosophy) thesis, University of Iowa, 2013.
https://doi.org/10.17077/etd.9zdptf8l

Follow this and additional works at: https://ir.uiowa.edu/etd

Part of the Bioinformatics Commons

KNOWLEDGE TRANSFER: WHAT, HOW, AND WHY

by

Si-Chi Chin

An Abstract

Of a thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Informatics
in the Graduate College of
The University of Iowa

May 2013

Thesis Supervisors: Professor W. Nick Street
Associate Professor David Eichmann

ABSTRACT

People learn and induce from prior experiences. We first learn how to use a spoon and then know how to use forks of various sizes. We first learn how to sew and then learn how to embroider. Transferring knowledge from one situation to another related situation often increases the speed and quality of learning. This observation is relevant to human learning, as well as machine learning.

This thesis focuses on the problem of knowledge transfer in the context of machine learning and information science. The goal of knowledge transfer is to train a system to recognize and apply knowledge acquired from previous tasks to new tasks or new domains. An effective knowledge transfer system facilitates the learning processes for novel tasks, where little information is available. For example, the ability to transfer knowledge from a model that identifies writers born in the U.S. to identify writers born in Kiribati, a much lesser known country, would increase the speed of learning to identify writers born in Kiribati from scratch.

In this thesis, we investigate three dimensions of knowledge transfer: what, how, and why. We present and elaborate on these questions: What type of knowledge should we transfer? How should we transfer knowledge across entities? Why do we observe certain pattern of knowledge transfer? We first propose Segmented Transfer – a novel knowledge transfer model – to identify and learn from the most informative partitions from prior tasks. We apply the proposed model to the problem of Wikipedia vandalism detection and entity search and classification.

Based on the foundation of knowledge transfer and network theory, we propose Knowledge Transfer Network (KTN), a novel type of network describing transfer learning relationships among problems. This novel type of network provides insights on identifying ontological connections that were initially obscured. We analyze the correlation between node characteristics and network centrality metrics for a KTN. Our experiments on the problem of Wikipedia vandalism detection and entity search and classification show that the high task similarity does not always turn into high transferability. Task characteristics, such as the class balance of the task or diversity of predictive features, can outweigh task similarity in terms of task transferability.

Abstract Approved: ______________________________
Thesis Supervisor

____
Title and Department

____
Date

____
Thesis Supervisor

____
Title and Department

____
Date

KNOWLEDGE TRANSFER: WHAT, HOW, AND WHY

by

Si-Chi Chin

A thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Informatics
in the Graduate College of
The University of Iowa

May 2013

Thesis Supervisors: Professor W. Nick Street
Associate Professor David Eichmann

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

PH.D. THESIS

This is to certify that the Ph.D. thesis of

Si-Chi Chin

has been approved by the Examining Committee for the thesis requirement for the Doctor of Philosophy degree in Informatics at the May 2013 graduation.

Thesis Committee: ______________________________
W. Nick Street, Thesis Supervisor

____
David Eichmann, Thesis Supervisor

____
Padmini Srinivasan

____
Alberto Segre

____
Haowei Hsieh

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the invaluable guidance of my advisors and my committee members, unconditional love and support from my family and husband, and generous help from friends and colleagues. Thank you for being there for me in this long journey.

My utmost gratitude to Dr. Nick Street and Dr. David Eichmann, my academic advisors, whose continuing encouragement and support I will never forget. Dr. Street has been my inspiration, friend, guide, and philosopher as I hurdle all the obstacles in the completion of this research work. This thesis would have remained a dream had it not been for the support of my advisors.

I wish to thank my dissertation committee members Dr. Padmini Srinivasan, Dr. Alberto Segre, and Dr. Haowei Hsieh, who provided their expert advice for this research and served as mentors and role models.

I would like to thank my fellow doctoral students – those who have moved on, those in the quagmire, and those just beginning – for their support, feedback, and friendship.

I am deeply indebted to my husband Aaron Hefel and my parents Chiu-Chau Lo and Yee-Huei Chin who dedicated their love and patience during the journey of doctoral studies. They have gien me the strength to finish the dissertation. This thesis is dedicated to them.

ABSTRACT

People learn and induce from prior experiences. We first learn how to use a spoon and then know how to use forks of various sizes. We first learn how to sew and then learn how to embroider. Transferring knowledge from one situation to another related situation often increases the speed and quality of learning. This observation is relevant to human learning, as well as machine learning.

This thesis focuses on the problem of knowledge transfer in the context of machine learning and information science. The goal of knowledge transfer is to train a system to recognize and apply knowledge acquired from previous tasks to new tasks or new domains. An effective knowledge transfer system facilitates the learning processes for novel tasks, where little information is available. For example, the ability to transfer knowledge from a model that identifies writers born in the U.S. to identify writers born in Kiribati, a much lesser known country, would increase the speed of learning to identify writers born in Kiribati from scratch.

In this thesis, we investigate three dimensions of knowledge transfer: what, how, and why. We present and elaborate on these questions: What type of knowledge should we transfer? How should we transfer knowledge across entities? Why do we observe certain pattern of knowledge transfer? We first propose Segmented Transfer – a novel knowledge transfer model – to identify and learn from the most informative partitions from prior tasks. We apply the proposed model to the problem of Wikipedia vandalism detection and entity search and classification.

Based on the foundation of knowledge transfer and network theory, we propose Knowledge Transfer Network (KTN), a novel type of network describing transfer learning relationships among problems. This novel type of network provides insights on identifying ontological connections that were initially obscured. We analyze the correlation between node characteristics and network centrality metrics for a KTN. Our experiments on the problem of Wikipedia vandalism detection and entity search and classification show that the high task similarity does not always turn into high transferability. Task characteristics, such as the class balance of the task or diversity of predictive features, can outweigh task similarity in terms of task transferability.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

CHAPTER 1
INTRODUCTION

Human beings learn from prior experiences, and so can automated systems. As humans, we first learn how to use a spoon and then know how to use forks of various sizes. We first learn how to sew and then learn how to embroider. We also find it easier to learn French after having learned Spanish. Transferring knowledge from one situation to another related situation often increases the speed and quality of learning. This observation is relevant to human learning, as well as machine learning.

This thesis focuses on the problem of knowledge transfer in the context of machine learning and information science. Knowledge transfer, which is more commonly known as transfer learning and domain adaptation, has received much attention in machine learning research and practice over the years [85, 23, 100, 74, 27, 77]. The lack of high-quality annotated examples creates a major challenge to train a learning model. Researchers in machine learning have found that transfer learning provides a solution to this problem. Transfer learning aims to train a system to recognize and apply knowledge acquired from previous tasks to new tasks or new domains. By reusing information from previously learned source task, transfer learning can reduce the cost of learning a model for a new target task.

Maximizing the utility of information provides opportunities to improve the process of knowledge discovery. In the field of machine learning and natural language processing (NLP), obtaining training labels is often expensive, while an enormous amount of unlabeled data are often available. Therefore, maximizing the utility of available label information would benefit the learning process. In light of this notion, this thesis studies transfer learning, emphasizing the reuse of previously acquired knowledge to other applicable tasks.

The discussion of how to efficiently utilize available information makes transfer learning valuable to information science studies. However, current research on transfer learning emphasizes the "outcome" of the transfer – learning better and faster – as opposed to analyzing the "reason" of the transfer – why we learn better and faster. For example, we may observe that it is faster to learn ballroom dancing after having learned figure skating. However, simply observing the fact is insufficient for understanding how and why the transfer of learning occurs. Building a better predictive model using transfer learning would be insufficient for understanding the happening of knowledge transfer. In this thesis, we aim to fill the gap by introducing interdisciplinary perspectives, crossing the domains of machine learning and information science.

In this thesis, we use "knowledge transfer" to refer to transfer learning from the machine learning community. Our definition of knowledge transfer concerns not only the improved outcome of transfer learning, but also the process of and reasons for effective transfer. We are interested in revealing the explicit knowledge that was transferred between the source and the target task. Traditionally, transfer learning assumes that transfer occurs among related tasks. However, the relatedness between tasks might be imperceptible from similarity measurements. For example, we may wonder whether the transfer of learning can be achieved between using a fork and using a pair of chopsticks. If the transfer is observed, we may want to know what contributes to the transfer of learning between the two task. Our approach of knowledge transfer goes beyond transfer learning and aims to explore and reveal new knowledge about the learning problem.

In order to situate knowledge transfer in a wider context, this thesis explores three dimensions of the area of study:

- *What* type of knowledge should we transfer?
- *How* should we transfer knowledge across tasks?
- *Why* do we observe a certain pattern of knowledge transfer?

The thesis is organized to address each of the three dimensions. Along the dimension of "what" and "how," we propose a novel method – segmented transfer – to learn from only the informative segments from the source tasks. Along the dimension of "why," we propose building a new type of network – a knowledge transfer network – to visualize the knowledge transfer relationship among tasks and to unveil the factors that contribute to the transferability of a source task. We test the proposed methods on two applications – Wikipedia Vandalism Detection and Entity Search and Classification.

The rest of the report is organized as follows. Chapter 2 summarizes the prior research of transfer learning, describing *what* to transfer and *how* to transfer, and surveys applications in the areas of information retrieval, data mining, and recommender systems. The end of the same chapter exemplifies the applications of network analysis to support the development of knowledge transfer networks. Chapter 3 describes the established work on how to select related source data to enhance learning performance in target data. Chapter 4 demonstrates the effort on constructing knowledge transfer networks and using the network analysis for the problem of Wikipedia Vandalism Detection and Entity Search and Retrieval. Chapter 5 concludes the thesis and indicates possible future research directions.

CHAPTER 2
BACKGROUND AND LITERATURE REVIEW

2.1 Introduction

This report discusses three dimensions of knowledge transfer: what, how, and why. In order to understand the *what* and *how* dimensions, this chapter defines transfer learning in Section 2.2 and surveys the state-of-the-art applications of transfer learning in Section 2.3. There exists opportunities to investigate a new type of transferable object (the *what*) and to enrich current transfer learning algorithms (the *how*). In support of the proposed knowledge transfer network to address the *why* dimension, Section 2.4 surveys the applications of network analysis, emphasizing on the social and information networks. The successful applications of network analysis indicate opportunities of using networks to understand the knowledge flow among various tasks, creating actionable knowledge in a given domain.

2.2 Transfer learning

Machine learning aims to discover interesting patterns from data, providing analytical models to explain and predict the data. Transfer learning is a research area in machine learning, emphasizing the reuse of previously acquired knowledge to another applicable task [74]. For example, one finds it easier to learn Spanish having learned French; or to perform ballroom dancing having already practiced figure skating. This area of research provides a promising solution to the issue of labeling costs. The method is particularly useful in the situations where labeled instances are absent or difficult to obtain. Transfer learning requires three components: the target task (e.g., the problem to be solved), the source task(s) (e.g., auxiliary data, previously studied), and criteria to select appropriate source tasks. Figure 2.1 illustrates the three primary steps involved in transfer learning:

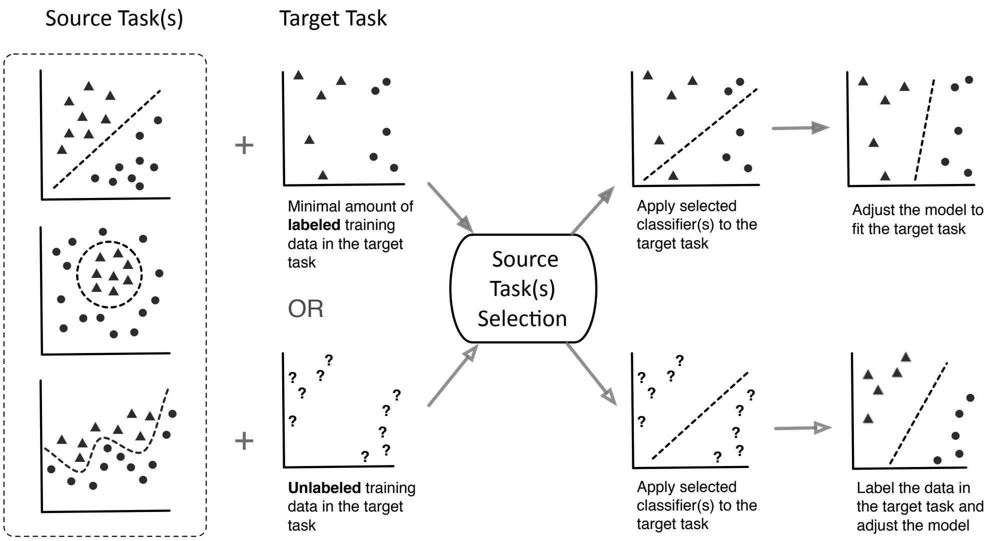

Source Task(s)
Target Task
+
Minimal amount of **labeled** training data in the target task
OR
+
Unlabeled training data in the target task
Source Task(s) Selection
Apply selected classifier(s) to the target task
Adjust the model to fit the target task
Apply selected classifier(s) to the target task
Label the data in the target task and adjust the model

Figure 2.1: Transfer learning

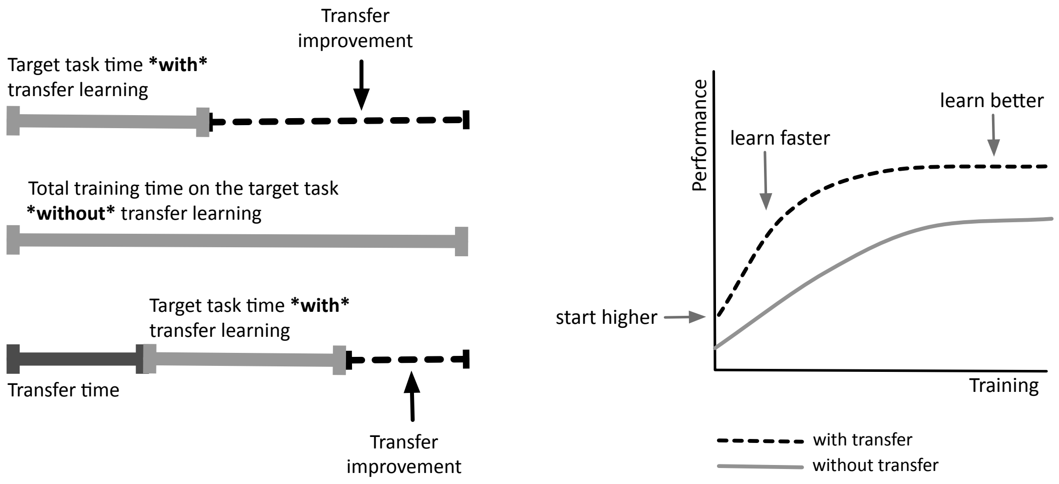
Note: Transfer learning reuses the previously acquired knowledge from source tasks and applies it on the target tasks. The first step of transfer learning is to select the most relevant source task(s) and then uses the source-task knowledge on the target task. A target task can be a partially labeled or unlabeled dataset. The transferred model is later adjusted based on available data from the target task.

- First, select one or more appropriate source tasks, given a target task.

- Second, transfer knowledge from the source task to the target task.

- Third, adapt the acquired knowledge to the target task.

Transfer learning leverages knowledge from the source task in the target task. It is useful when the data collection is expensive or impossible, when data is easily outdated, and when the test data are drawn from a different distribution or sample space of training data. The goal of transfer learning is to decrease the learning time of target task and to improve the generalization capacity of learned models (see Figure 2.2).


Transfer improvement
Target task time *with* transfer learning
Total training time on the target task *without* transfer learning
Target task time *with* transfer learning
Transfer time
Transfer improvement
Performance
learn better
learn faster
start higher
Training
with transfer
without transfer

(a) Training time scenario. Inspired by [100] (b) Training time scenario. Inspired by [101]
Figure 2.2: Transfer learning outcomes

Note: Transfer learning decreases the learning time of target task and improves the generalization learned models.

Research on transfer learning discusses how to use a prior knowledge learned from a source task to the target task and how to discover relevant prior knowledge to build a better classifier for the current task. Transfer learning believes that the generalization of a learned model may occur *across* tasks. In contrast, traditional machine learning limits the generalization to being *within* a task. Extensive research on transfer learning has continued to develop under many related names, e.g., inductive learning, multi-task learning, reinforcement learning, lifelong learning, knowledge transfer, domain transfer (or adaptation), knowledge reuse, information reuse, classifier reuse, and auxiliary classifier selection.

This section adopts the notations and the formalized definition described in Pan and Yang [74]. There is a feature space $\mathcal{X}$ where $X = \{x_1, ..., x_n\} \in \mathcal{X}$. Taking document classification as an example, x_i is the term vector representation of the ith document. X is a set of n documents in a feature space $\mathcal{X}$ that contains all possible term vectors. Another example is Wikipedia vandalism detection, in which, $\mathcal{X}$ is the set of features (e.g., perplexity, entropy, out-of-vocabulary frequency etc.) generated by statistical language models. The notation x_i is vector of statistical feature values for the ith revision and X is the complete revision history of a given article. $\mathcal{X}_s$ denotes the feature space of the source task and $\mathcal{X}_t$ denotes the feature space of the target task. If $\mathcal{X}_s = \mathcal{X}_t$, the source and target task have the same feature definitions.

There also exists a label space $\mathcal{Y}$, denoting the set of all class labels. Each data point is a pair of $\{x_i, y_i\}$ where $y_i \in \mathcal{Y}$. In a binary classification task, y_i takes only two values such as "Relevant/Non-relevant," "Positive/Negative," or "True/False." In a multi-class classification task, y_i is the set of class labels. $\mathcal{Y}_s$ denotes the label space of the source task and $\mathcal{Y}_t$ denotes the feature space of the target task. If $\mathcal{Y}_s = \mathcal{Y}_t$, both the source and target task use the same class labels, for example, both tasks are binary class.

The restricted definition of transfer learning assumes $\mathcal{X}_s = \mathcal{X}_t$ and $\mathcal{Y}_s = \mathcal{Y}_t$. Transfer learning assumes that $P(X_s) \neq P(X_t)$ and that $P(Y_s|X_s) \neq P(Y_t|X_t)$, while traditional learning assumes $P(X_s) = P(X_t)$ and that $P(Y_s|X_s) = P(Y_t|X_t)$. On the other hand, domain adaptation focuses on problems where $P(X_s) \neq P(X_t)$ but $P(Y_s|X_s) = P(Y_t|X_t)$.

Based on the specificity of transferred knowledge, transfer learning can be loosely divided into low-level knowledge transfer and high-level knowledge transfer [100]. Low-level knowledge acquired from the source task includes the experience instances, prior distributions, functions, or classifiers, for improving the starting point for the learning in the target task. High-level knowledge provides guidance during the learning in the target task. Silver [93] considered two types of knowledge transfer for the neural network learner: representational and functional. Pan and Yang [74] describe four types of transfer learning: instance-transfer, the feature-representation-transfer, the parameter-transfer, and the relational-knowledge-transfer. This section adds *model-transfer* as a new category, emphasizing the transfer of learning models, e.g., the reuse of classifiers, from the source task to the target task. Table 2.1 summarizes the five transfer learning categories, including the definitions from [74] to complete the section.

Several approaches are available to reuse previously learned classifiers. Predictions from classifiers trained on one or more source tasks can be used as additional features for the target task (feature enrichment) [30], or the classifiers can be selectively employed directly on the target task.

Table 2.1: Five categories of transfer learning.

Category	**Description**
Model	Generate appropriate models (e.g., classifiers) from source tasks (or part of them) that can be directly applied to the target task.
Instance	Train on re-weighed instances from the source tasks to use on the target task.
Feature	Discover feature representations to bridge the gap between the source and the target task.
Parameter	Identify parameters or priors shared by the source and the target task.
Relational	Map the relational knowledge between the source and the target task.

Other available approaches choose among candidate solutions from multiple available source tasks. For example, Zhang et al. [116] constructed an ensemble of decision trees trained from related tasks to improve prediction on the problem with limited labeled data. Yang et al. [110] described three methods to select auxiliary classifiers from an existing set. The first method is to use the Expectation Maximization (EM) algorithm to estimate the distribution respective to each class and then select the classifier that can best separate the between-class score distribution. The second method is to identify the "average" of multiple classifiers, assuming the average is better than any individual one. The method aggregated the predictions from multiple classifiers to create pseudolabels to evaluate each classifier. The pseudolabels formed a posterior distribution of the output prediction and can be used to compute average precision for each classifiers. The best classifiers are selected based on the average precision results. The third method is to build a regression model to predict a classifier's average precision score on the target task (the problem of interest) and then select the classifier with best performance.

Table 2.2: Transfer Learning Application Category

Task type	**Object type**	**Example Tasks**
Classification	Instance	• Activity recognition [86, 46] • Cross language web page categorization [105] • Text categorization [112]
	Feature	• Sentiment classification [73, 36, 39] • Opinion mining [6] • Text categorization [22] • Disease reporting events detection [95] • Sequence labeling (e.g., POS tagging) [24]
	Parameter	• Information retrieval system [109]
	Model	• Activity recognition [113] • Social tag personalized recommendation [48] • Wikipedia vandalism detection † (Chapter 3)
Clustering	Feature	• Image clustering [111]
Collaborative Filtering	Feature	• Predict user rating [75, 56, 57, 102]
	Model	• Link prediction [7]

Note: Transfer learning application category. The table categorizes recent research on the applications of transfer learning by the task types (e.g., classification or clustering) and the transferred object type. The table also identifies a few example tasks for each category. † indicates the proposed applications in this report.

2.3 Applications of transfer learning

This section reviews selective research works since 2009 on the application of transfer learning, as a comprehensive study on transfer learning applications before 2009 can be found in [74]. Table 2.2 categorizes the applications of transfer learning. The example tasks share a common characteristic – the labeled data are available and often abundant in one area (e.g., movie rating) but is either unavailable or hard to obtain in another area (e.g., book rating). Among the five transfer learning types described in the previous section (i.e., instance, feature, parameter, relation, and model), instance-transfer and feature-representation-transfer are the two most common methods for the transfer learning applications. The following sections each concentrate on the area of information retrieval, collaborative filtering, and other representative data mining tasks.

2.3.1 Information Retrieval

Information retrieval is the area of study to trace and recover documents that satisfy an information need from large text collections. An information retrieval task can be considered as a binary classification problem that determines the relevance of a document to a query. Therefore, each query can be viewed as an individual classification task. A transfer learning approach to information retrieval problems explores methods to leverage knowledge acquired from the previously known queries to new queries. Yan and Zhang [109] incorporated task-level features into a probabilistic transfer learning model to enhance information retrieval performances. The authors extracted task-level features from properties of user queries (e.g., number of named entities referred in queries) and user profiles (e.g., the age of users). Their proposed model used hidden source variables sampled from a multinomial distribution to identify the related task clusters. The parameters previously learned from source queries are then transferred to the new hierarchical Bayesian model for the target query.

Applicable tasks for transfer learning in the area of Information Retrieval include sentiment classification (or opinion mining) and text categorization. This section describes the two tasks in more details.

2.3.1.1 Sentiment classification and opinion mining

This subsection discusses the application of transfer learning to sentiment classification and opinion mining. This area of study aims to identify subjective information in the text, determining if an expression is positive or negative (the polarity of text). The increasing amount of online reviews provides ample opportunities to the study of sentiment classification and opinion mining. However, review data often comes from a large variety of sources, from different products reviews to political opinions. Therefore subjective expressions often vary across domains, exhibiting different distribution of term features. For example, while the word “hilarious” is an informative indicator in movie reviews, it is irrelevant to nutrition supplements. Given the amount and the variety of available reviews, obtaining labeled training data are expensive. The challenge provides opportunities for the application of domain adaptation as well as transfer learning.

Numerous recent studies have demonstrated the advantages of applying domain adaptation methods in sentiment classification [58, 104, 59, 73, 36, 39]. A widely studied approach is to discover a latent feature representations to bridge the gap between source and target task. The re-engineered features can be used to train new classifiers or incorporate into a variation of matrix factorization framework. For example, Pan et al. [73] used a spectral clustering algorithm to match domain-independent and domain-specific term features. The authors constructed features in a common latent space to map the source and target domain and to train a linear classifier. Gao and Li [36] identified the common topic space between the source and target domain using cross-domain indexing. The authors built pivot features to bridge the source and target domain. Glorot et al. [39] used Deep Learning system to perform an unsupervised feature extraction. The proposed method aims to learn features that help “disentangle the underlying factors of variation” and thus help identify concepts and characteristics shared by product reviews across domains (the invariant properties). Distinct from most related studies in the area, Calais Guera et al. [6] adopted a social network approach to mine sentiments and opinions. The authors leveraged information from social media (e.g., Twitter) to construct endorsement network (Opinion Agreement Graph) and propagated bias information from persons to terms.

2.3.1.2 Text categorization

Text categorization is an area of study concerned with assigning one or more predefined categories to documents. The research challenges of this area come from the highly unbalanced number of training examples across a large number of document categories. The challenge brings opportunities for transfer learning, domain adaptation, and multi-task learning. Previously studied methods for text classification can be categorized into feature-representation adaptation and instance-weight adaptation [73]. The first approach explores methods to reuse features from the source domain. The application of the approach is similar to its application in sentiment analysis research. For example, Dai et al. [22] map both the features and category labels (if available) from the source and target task to a common eigenvector representation.

The authors applied spectral graph theory on the constructed features to assign categories to documents. Stewart et al. [95] transferred tokens and linguistic structures from the source task to detect disease reporting events. The authors filtered feature space using learned tokens from the source task. They then classified instances in the target task using the structure-based features learning kernel function (i.e., SVM classifier). The second approach explores methods to reweigh instances from the source domain to use in the target domain. For example, Yang et al. [112] leveraged the labeled examples from auxiliary data and the correlation between the target and the auxiliary data to accomplish knowledge transfer. The authors used a generalized maximum entropy model and the estimated expectation of feature functions to transfer labels from auxiliary data to target data.

2.3.2 Collaborative Filtering

Collaborative filtering is one of the most commonly used methods for recommender systems. The method, modeling the "taste" of people, makes recommendations based on similar behavior patterns. The two major challenges for collaborative filtering method are the data sparsity and the "cold-start" situation [44]. The problem of sparsity comes from the limited number of users' ratings; the problem of "cold-start" occurs on the new items with only a few available ratings. Both constraints limit the available techniques of collaborative filtering, such as k-NN search, probabilistic modeling, or matrix factorization.

Several studies have attempted to tackle the problems using transfer learning

[48, 56, 57, 7, 76, 75]. Kamishima et al. [48] studied personalized recommendation for social tags. The authors developed TrBagg (Transfer Bagging) to transfer tags from non-target users. The proposed method samples the merged set of source and target data to train numerous weak classifiers that were then filtered based on either the full or a part of the target data. The final predictions were made by aggregating the results from the filtered set of classifiers by majority voting. Li et al. [56, 57] addressed the issue of data sparsity by mapping the cluster-level rating patterns to bridge the auxiliary (source) and target data. The authors used the framework to transfer movie ratings to book ratings. Cao et al. [7] used non-linear matrix factorization to predict potential links between users and items. The authors introduced a link function leveraging the task similarity learned from kernel method. Pan et al. [76] proposed a two-sided transfer learning method (Coordinate System Transfer) to transfer both user and item knowledge from an auxiliary domain. The authors used sparse matrix tri-factorization to discover the transferred knowledge (i.e., the principle coordinates) and then used a regularization technique to adapt the proposed coordinate systems. The same authors later generalized the framework to incorporate heterogeneous user feedback [75], predicting the missing scale rating from auxiliary like/dislike information. Vasuki et al. [102] used friendship networks for affiliation recommendation task. The authors leveraged user-side information from a combined graph of users and communities, using graph proximity and latent factor modeling to transfer knowledge to predict affiliation links.

2.3.3 Other data mining tasks

Recent research of transfer learning has investigated other data mining tasks such as activity recognition and image clustering. Activity recognition aims to recognize or infer individual's activities from sensor data. However, training a recognition model requires considerable human effort to annotate the sensor data and presents a major challenge to the area of study. To address the problem, research [86, 46] has suggested automatic approaches to reweigh labeled examples from the source task and transfer the labeled knowledge to new domain. Rashidi and Cook [86] proposed a semi-EM framework to estimate the mapping probability from each source activity to the target activities. The authors assigned labels to the target activities based on the learned probability mapping matrices. Hu et al. [46] leveraged Web pages associated with each activity to implement knowledge transfer. The authors extract web content from a search engine (e.g., Google) and computed a tf-idf feature vector for each activity. The similarity between activities is computed based on the tf-idf vectors and is used as the confidence to construct pseudo training data. The proposed method trained a weighted SVM on the pseudo training data to perform multi-class classification.

The area of image clustering has also captured the attention of transfer learning research. Image clustering aims to group related images so that the cluster can provide a summary for a set of images. However, the distribution of the labeled data are highly unbalanced among heterogeneous feature spaces. The situation provides opportunity to transfer learning. Yang et al. [111] used text annotation (e.g., social tags) extracted from Web sites (e.g., Flickr) to enhance image clustering performance. The authors adopted an annotation-based probabilistic latent semantics analysis (aPLSA) algorithm to reveal the latent semantic information shared by text and image features. The clustering function assigned each image to a latent variable.

2.4 Network analysis and its applications

A network is a collection of connected objects. It characterizes the structure, as well as the dynamics, of the relationships between objects. The term "network" is ubiquitous across disciplines. A precise description of a network requires clear definitions on the semantics of the nodes (the objects) and the links (the connections). For example, in a network of friendship, a node is a person and a link is the known friendship and in a network of web, the nodes are a set of webpages connected by hyperlinks. The study of networks defines and analyzes different networks, leveraging the analytical power of networks to solve a scientific research problem.

Mining data to extract useful information and knowledge is one of the most major challenges in industries and scientific communities. Mining relationships between entities helps to discover interesting, or potentially novel patterns of a domain. A network is a graphical representation that captures relational information. Mining network data supports the comprehension of the relational knowledge.

Network analysis is interdisciplinary. Network theory analyzes a graph representation of relations, borrowing analytical power from computer science, graph theory, and target domain knowledge. Applications of network theory extend across numerous disciplines: physics, computer science, sociology, economics, management science, biology, etc. The term "network" is overloaded across disciplines and the prospect of "network analysis" varies from one discipline to another.

Network representations have been widely applied in many successful research applications. This section describes social networks and information networks. Among the four types of networks designated by Newman [71] – social networks, information networks, biological networks, and technological networks – the social network and the information network are most related to information retrieval and text mining. Social networks lend support to browsing and locating relevant content and information networks provide knowledge representation to analyze information space.

2.4.1 Social networks

A social network defines a set of inter-connected actors (usually people or groups of people). The nodes in a social networks represent actors, such as users in a social networking site, and the links indicate specific interactions, such as friendships, family ties, professional relationships, and common interests. Social network analysis (SNA) views actors as nodes connected to each other in a network graph by one or more relationships (e.g., ties, edges, linkages). It conceptualizes relations among actors, establishing linkages between actors as conduits for the "flow" of information [107]. Table 2.3 organizes current research based on different definitions of nodes and links in social networks.

A social network plays a key role in the *information dissemination.* For exam-

Table 2.3: Types (or semantics) of nodes and links in *social networks*

Node Type	**Link Type**	**Application Example**	**Example Dataset**
Persons	Friendship	Analysis of how individuals move between communities[2], the development of the web of trust [87], named entity disambiguation [62]	LiveJournal [2, 21], Epinion [87], Internet Movie Database (IMDB) [62]
	Relationships	Obesity network [18], spread of happiness [31]	Framingham Heart Study cohorts [18]
		Analysis of contagious outbreak [19]	Recorded data by University Health Services (UHS) [19]
Researchers	Co-authorship	Co-authorship network, information retrieval and document ranking [117] [61, 40, 90, 2]	The NIPS paper co-authorship dataset [40, 90], DBLP dataset [2], arXiv e-prints [90, 61], Citeseer [117],NBER Patent Citation Data [41], Web of Science (WOS) [26]
Email address	Communications: sent to and received from	Roles and groups discovery [64], named entity disambiguation [68]	Enron email dataset [52, 68]

ple, Mislove et al. [70] indicated that 80.6% of the views on Flickr were contributed by the user network. In addition to the propagation of information in a network, research also investigated the propagation of obesity [18], happiness [31], contagious disease [19], and academic influence [117, 26]. Research has shown that adjacent users in a social network tends to trust each other or have common interests. It is because individuals tends to link to people who are similar to them (termed *selection* or *homophily* [66]) or gradually become similar to those they link to (termed *social influence* [33]). Crandall et al. [21] modeled the social network on Wikipedia and LiveJournal to study the prediction power of social interaction (the *influence*) and similarity (*homophily*) for future activities of an individual. Their work indicated that social interaction was both an effect and a cause of homophily (selection), and that the similarity of interests among Wikipedia users was not as predictive as social interaction.

Another important research area in social network analysis is *community studies.* The goal of this subarea is to identify community structure in a social network. Previous research discussed the group formation and co-authorship network [2], analyzed the development of the web of trust [87], discovered roles and groups in a network [64], and studied how individuals move between communities [2].

Another use of the social network is *topic identification and prioritization of documents* for improved information retrieval. Research has used social network analysis to discover latent groups and topics from text [106] and prioritize the importance of email messages [114]. In the context of document retrieval, one can assume that relevant documents would exhibit similar network characteristics. Therefore, features extracted from social network structure would be strong predictors for document relevancy. Yoo et al. [114] captured and utilized network features such as personal social roles and social groups to address the problem of personalized email prioritization (PEP). The authors introduced the semi-supervised importance propagation (SIP) algorithm to propagate the importance value of limited labeled email messages (training data) to contact persons and other messages (testing data). Zhou et al. [117] used latent social interactions to estimate the dependency of topics. The assumption is that if social actors found in a given topic (t_a) are closely connected to social actors found in another topic (t_b), these two topic are more likely to be dependent to each other. Ding et al. [26] used path-finding algorithm in an author citation network to analyze scientific collaboration and endorsement patterns of researchers at the topic level.

Social network analysis is also widely used in *named entity disambiguation*. Minkov et al. [68] represented a structured (or semi-structured) dataset of email messages as a graph. Their work used a lazy graph walk to measure similarities between entities to discover relevant results (or documents). They considered the notion that documents are often connected to other objects via meta-data and used it to propagate the similarity across the graph. They modeled the problem as a search task to retrieve a ranked list of entity nodes to disambiguate named entities. Malin [62] used two methods to construct clusters to disambiguate named entities in a relational data set. One method was to transform each source (e.g., document, article) to a Boolean vector of the occurrence of entities (1 if an entity occurs in the source and 0 otherwise). The cosine similarity between sources was used to perform the hierarchical clustering. The other method was to perform random walks between ambiguous entities on a network to compute the network similarity. The random walk approach incorporated the notion of community similarity to take into account the indirect relationships between entities.

2.4.2 Information networks

An information network defines a set of connected *text objects.* In contrast to social networks where a node is a person, a node in an information network is a text-related and information-rich object. Two classic examples of information networks are citation networks and World Wide Web. The definition of nodes and links in information networks varies from one application to another. In a citation network, a node is a scientific research paper and a link indicates one paper citing another paper. In the World Wide Web, a node is a webpage and a link is a hyperlink between pages. Table 2.4 is a tabular view of current research, organized by the semantics of nodes and links.

Citation networks, in contrast with co-authorship networks, emphasize bibliometric studies as opposed to the interactions among researchers. Bibliometric methods analyze texts and information, especially published literature. A citation network is a directed acyclic graph because a paper can only cite papers that existed before it, making it nearly impossible to have closed loops. The inherent topological

Table 2.4: Types (or semantics) of nodes and links in *information networks*

Node Type	**Link Type**	**Application Example**	**Example Dataset**
Papers	Citations: cite and cited by	Citation analysis using a network structure	United States Patent and Trademark Office (USPTO) [60] , Web of Science (WOS) [26]
Web pages, blog posts	Hyperlinks	Web analysis, topic tracking [54], information propagation in blogosphere [55]	Memetracker dataset [54], Usenet blog subset[55], IBMs Patent Server database [11].
Concepts	Semantic relatedness (synonyms, hypernyms etc.)	Use semantic network to disambiguate word sense [97], to improve recommendation for long-tail queries [99]	Wikipedia (e.g., the category hierarchy) [35, 96], Citeseer [28], WordNet [67], Visual Thesaurus [32]
		Ontology, topic/concept map	Wikipedia [80, 98], Dbpedia [1] , Open Directory Project [43]
Terms	Co-occurrence	Improving text retrieval	PubMed [13, 5]
Tags	Co-occurrence	Folksonomy analysis using social tagging datasets [10, 8], information retrieval[118]	CiteULike [8], del.icio.us [10], BibSonomy [10], Flickr [118]
Persons and objects	Purchase or preference	Collaboration filtering, recommender system	Netflix [51, 78], Amazon [53]
Wikipedia articles	Common editor(s)	Qualitative analysis on the edit network, using global network structure characteristics	Wikipedia [3]
Wikipedia categories	Common article(s)	Developing an ontology over the user interests a social network	Wikipedia [43, 115]

nature of a citation network lends power to computing the citation index [38] and hence to indicate the significance of a published paper. Moreover, a citation network provides insight on how a research work is perceived and received by the peer community, which is useful to discover the pattern of citation and current research front [37, 83]. More examples of citation network research include employing features extracted from the citation network to improve patent classification performance [60], using a citation network to assess law reviews influence on judicial decisions [65], and studying the connective thread in a citation network to discover the development of DNA theory [47].

The *World Wide Web* is a network where web pages (the nodes) are interconnected by hyperlinks. Unlike citation networks, the World Wide Web, does not have the constraint to forbid cycles and hence are in general cyclic networks. Similar to citation networks, ranking the nodes is the primary concern of research. Ranking webpages by their relevancy to a user's query is vital for search engines. The network structure of the Web allows the computation of centrality metrics such as HITS [50] and PageRank [72], making the ranking possible.

Another prominent example of information networks is the *semantic network*. A semantic network can be either a directed or an undirected graph, representing semantic relations (the links) among concepts (the nodes). Semantic networks have been used to disambiguate word sense [97] and to improve recommendation for long-tail queries [99]. Common datasets of semantic networks include Wikipedia (e.g., the category hierarchy) [35, 96], Citeseer [28], WordNet [67], Visual Thesaurus [32].

A semantic network is often an incarnation of an ontology. On the other hand, an ontological framework is often represented in the form of semantic network. The two closely related concepts (semantic network and ontology) both define a node as a concept and connect nodes by their semantic relations.

A few other examples of information networks include *folksonomy networks* and *preference networks*. Folksonomy is a user-generated taxonomy used to categorize and retrieve Web pages, photographs, Web links and other Web content using tags. Folksonomy is also known under the names social tagging, collaborative tagging, social indexing, and social classification. Researchers usually represent a folksonomy network as a tri-partite graph whose nodes represent users, tags, and resources connected by tag assignments [45, 10, 8]. Cattuto et al. [10] investigated the clustering coefficient and the characteristic path length (the average length of all shortest paths) of two social tagging systems: del.icio.us and BibSonmy. They introduced a network of tag co-occurrence and analyzed the correlations in node connectivity to detect developing semantics in the folksonomy. Capocci and Caldarelli [8] analyzed tag co-occurrence network from *CiteULike* and used clustering coefficient to discover the semantical patterns among tags. Preference networks are usually constructed as a bipartite graph whose nodes represent individuals and their preferred objects. The network construct provides basis for collaborative filtering and recommender system [51, 78, 53].

2.5 Conclusion

This chapter summarizes the area of study on transfer learning and its applications on information retrieval and data mining. This chapter also exemplifies methods of model-based transfer learning (the reuse of classifiers). Transfer learning framework has been applied to numerous text classification tasks and has been useful for recommender systems. However, additional applications for transfer learning can still be developed. This report suggests applying a transfer learning framework to Wikipedia vandalism detection (Chapter 3) and to problems of entity search and retrieval (Chapter 3) In addition to novel applications, there also remains research opportunities to explore novel approaches to select and manipulate source tasks for effective transfer learning. The proposed *segmented transfer* in Chapter 3 presents two approaches to leverage knowledge from the source tasks. As a machine learning research area, transfer learning aims to improve the performance of "system" as opposed to enhance the understanding of information from the perspective of "users" and "use." Section 2.4 demonstrated numerous successful applications of network analysis. Constructing a knowledge transfer network suggests opportunities to understand the structure of knowledge transfer and the relationship of learning tasks, using transfer learning to create actionable knowledge for domain experts.

CHAPTER 3
SEGMENTED TRANSFER

3.1 Introduction

In this chapter, we discuss the *how* and *what* dimensions in this chapter: first, we introduce the *segmented transfer* approach to determine *how* we can transfer knowledge between the related tasks; second, we explore using Bag-of-Concepts (BoC) to aid understanding of *what* are appropriate transferable objects.

Research in transfer learning explores methods to leverage knowledge acquired from "related" tasks (the source/auxiliary tasks) to the tasks of interest (the target task). A positive transfer occurs when models learned from the source task enhance the performance of the target task. If otherwise, a negative transfer occurs. To address the issue of potential negative transfer, this report proposes *segmented transfer (ST)* [15], a novel algorithm to enrich the capability of transfer learning. The goal of the approach is to identify and learn from the most related segment, a subset from the training samples, from the source task. The motivation comes from two assumptions:

- Not all of the source task is useful, and
- Not all of the target task can benefit from the available source task.

Because the distribution of the feature space is different between the source and target tasks, it is likely that some source task data will not be used. In this chapter, we propose the two approaches – *source task segmented transfer (STST)* and the *target task segmented transfer (TTST)* – aim to transfer knowledge acquired only from the related segment to minimize negative transfer.

We apply the proposed approaches to the problem of Wikipedia vandalism detection and entity search and classification. In order to provide a more thorough background for the experiments of segmented transfer, we elaborate on the problem of Wikipedia Vandalism Detection in the first section (Section 3.2). Section 3.3.2 describes two segmented transfer methods: source task segmented transfer (STST) and target task segmented transfer (TTST). Section 3.4 describes the application of knowledge transfer on the problem of entity search and classification. We use Bag-of-Concepts as a common feature space for the source and the target task to facilitate knowledge transfer.

3.2 Wikipedia Vandalism Detection

Wikipedia, among the largest collaborative spaces open to the public, is also vulnerable to malicious editing – vandalism. Wikipedia defines vandalism as "any addition, removal, or change of content made in a deliberate attempt to compromise the integrity of Wikipedia[1]." The characteristics of Wikipedia vandalism are heterogeneous. It can a be large-scale editing, such as deleting the entire article or replacing the entire article with irrelevant content. It can be some irrelevant, random, or unintelligible text (e.g. *dfdfefefd #$%&@@#, John Smith loves Jane Doe.*) It can be a small change of facts (e.g. *This is true* → *This is not true.*) It can also be an unreg-

[1] http://en.wikipedia.org/wiki/Wikipedia:Vandalism

ulated formatting of text, such as converting all text to the font size of titles. Figure 3.1 illustrates a taxonomy of Wikipedia actions, highlighting the diverse vandalism instances. Table 3.1 describes and exemplifies each type of vandalism.

Wikipedia vandalism detection, an adversarial information retrieval task, is a recently emerging research area. Prior research emphasized methods to separate the malicious edits from the well-intentioned edits [108, 14, 94, 82]. Research has also identified common types of vandalism [103, 84, 82]. Chin and Street [16] explored an unsupervised subclass discovery approach to automatically improve the taxonomy and the categorization of Wikipedia vandalism. The goal of detecting Wikipedia vandalism instances is to determine, for each newly edited revision, whether it could be a vandalism instance and to create a ranked list of probable vandalism edits to alert Wikipedia users (usually the stewards for an article). However, determining if an edit is malicious is challenging and acquiring reliable class labels is non-trivial. To classify a new and unlabeled dataset, it is useful to leverage knowledge from prior tasks.

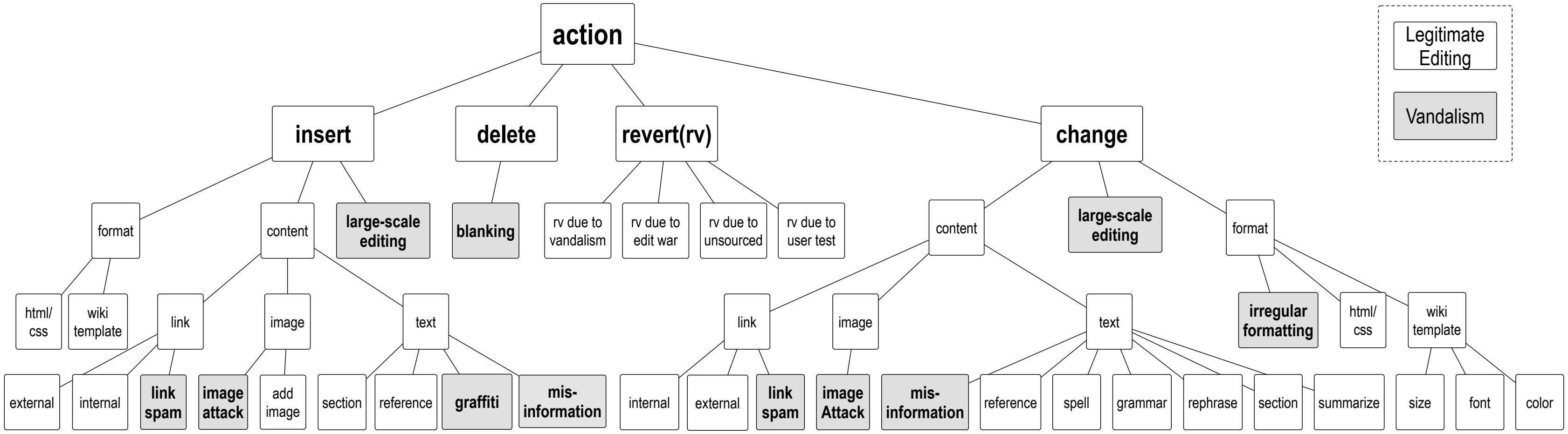


Figure 3.1: Wikipedia Action Taxonomy

Note: The taxonomy groups Wikipedia editing by the four primary actions (change, insert, delete, and revert) and types of change (format and content), considering also the scale of editing. The shaded boxes are types of Wikipedia vandalism.

3.2.1 Vandalism Classification

3.2.1.1 Data and Experimental Setup

We worked with the Wikipedia page history archive from February 24th, 2009[2]. Our corpus includes complete revision histories (note this aspect is unique to our research) for two Wikipedia articles: Abraham Lincoln (8,816 revisions), Microsoft (8,220 revisions). These articles are acknowledged to be among the most vandalized pages[3]. The reason for choosing the most vandalized pages is to acquire an extensive amount of vandalism instances for the analysis. We intentionally chose one article from the "Computing and Internet" category and one article from the "History" to demonstrate the similarity and differences of the vandalism pattern across categories.

Figure 3.2 illustrates the system structure and preprocessing of the revision history. We extracted the two articles from the Wikipedia Dump file and parsed them into individual revisions with the SAX parser. Information such as revision comments, contributors, and timestamp are also extracted from the XML file. We used the Java BreakIterator class to preprocess the revision history. Each revision was processed into one sentence per line to enable diff processing at the sentence level.

We used the CMU-toolkit [20] to build bigram statistical language models for each revision of a page. Moving through the sequence of revisions we adopt the following process. Assuming we are at revision n we compute the diff between it and the previous version *n-1*. This diff is directional in that we record only the new data

[2] http://download.wikimedia.org/enwiki/latest/

[3] http://en.wikipedia.org/wiki/
Wikipedia:Most_vandalized_pages

Table 3.1: Types of Vandalism

Type	**Action Taxonomy**	**Example**
Blanking	Delete(massive)	
Large-scale Editing	Insert (massive), Change (massive)	Replace all the occurrences of “Microsoft” to “Microshaft”.
Graffiti	Insert–Text	• *I like eggs!*
		• *dfdfefefd jaaaei #$%&@@#* • *John Smith loves Jane Doe.* • This ***king program is EVIL!!! • *Buying their computers is totally a waste of your money.*
Misinformation	Change–Text	• *Key Person: John Lennon* (on Microsoft page) • *4,600 million → 4,000 million* • *This is true → This is not true*
Image Attack	Insert–Image, Change–Image	Replace Microsoft logo with a picture of a kitten.
Link Spam	Insert–Link, Change–Link	• *http://www.wierdspot.com Abe’s Personal Diary*
Irregular Formatting	Insert–Format, Change–Format	• Inappropriate use of Wikimarkup such as {{*nonsense*}}

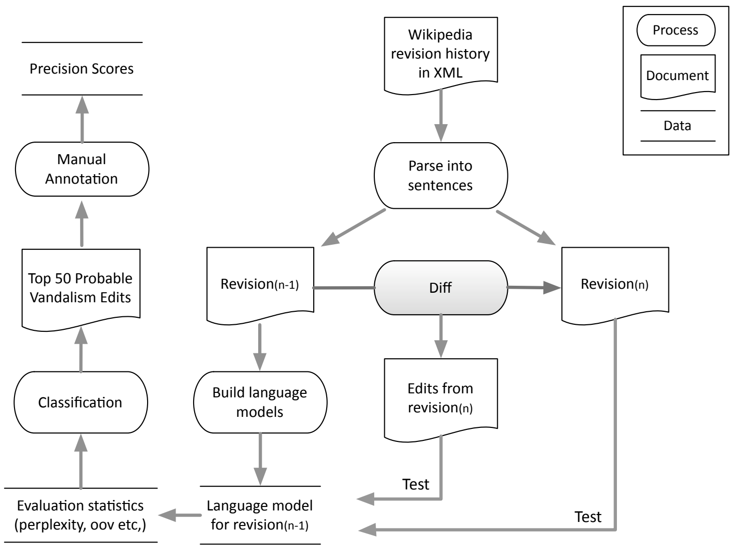


Figure 3.2: Flowchart of experiments.

that is in version n as compared to version *n-1*. The diff data for revision n and the full revision n are then tested using the built model. Each test yields a set of values: perplexity, number of words, number of words that are out of vocabulary, percentage of words that are out of vocabulary, number of bigrams hits and unigram hits, and percentage of bigram and unigram hits.

As vandalism often involves the use of unexpected vocabulary (the "out-of-vocabulary" number from CMU-toolkit *evallm* process) to draw attention, an instance of vandalism would produce high surprise factor when compared with the previous version, i.e., it would produce high perplexity when assessed using the language model of the previous version. Since we built a language model for every individual revision, including vandalized revisions, a follow up revision to revert a vandalism would also have high perplexity compared to the previous vandalism instance. To address the challenge and to identify a non-vandalized revision for the evaluation, we evaluate each diff result and the new added revision n against three language models: the model built from the revision *n-1*, the revision *n-5*, and the revision *n-10.* [4] We would expect an instance of vandalism to have three large out-of-vocabulary results, and a revert to have only one large out-of-vocabulary number. Therefore, from the three results, we select the one with the lowest out-of-vocabulary number, so as to avoid mistaking a legitimate revision for a vandalism instance.

3.2.2 Statistical Language Models and Classification

Statistical language modeling (SLM) [88] computes the distribution of tokens in natural language text and assigns a probability to the occurrence of a string S or a sequence of m words. SLM is commonly applied to many natural language processing tasks such as speech recognition , machine translation , text summarization , information retrieval , and web spam detection [69, 9]. The CMU SLM toolkit [20] allows construction and testing of n-gram language models. The *evallm* tool evaluates the language model dynamically, providing statistics such as perplexity, number of n-grams hits, number of OOV (out of vocabulary), and the percentage of OOV from a given test text.

In our experiments, we built bigram language models with the Good-Turning smoothing method [20]. We used two sets of *evallm* statistics results that were gen-

[4] The choice of *n-5* and *n-10* is based on authors' experiences. It is not uncommon that vandalism actions occur consecutively. If a vandalism occurs at the revision *n-1*, it is likely that the revision *n-2* or *n-3* is also a vandalism instances. Meanwhile, as the language evolves over time, we want to use an old revision that is still similar enough to the current revision. Experience shows that using the revisions *n-5* and *n-10* demonstrates adequate results.

Table 3.2: Definition of Features

Feature	**Definition**
word_num(d)	Number of known words (from *diff*)
perplex(d)	Perplexity value (from *diff*)
entropy(d)	Entropy value (from *diff*)
oov_num(d)	Number of unknown words (from *diff*)
oov_per(d)	Percentage of unknown words (from *diff*)
bigram_hit(d)	Number of known bigrams (from *diff*)
bigram_per(d)	Percentage of known bigrams (from *diff*)
unigram_hit(d)	Number of known unigrams (from *diff*)
unigram_per(d)	Percentage of known unigrams (from *diff*)
ratio_a	Ratio of added text from previous revision
ratio_c	Ratio of changed text from previous revision
ratio_d	Ratio of deleted text from previous revision

erated separately from the diff data for the new revision and the full new revision to build classifiers. In addition to the 18 attributes (9 for each set) generated from SLM, three features: ratio of insertion, ratio of change, and ration of deletion, were added to the set of attributes. We summarize features for the classification in Table 3.2.

We used the Weimar data from Potthast et al. [82] as the baseline to evaluate our features and classification methods. This data includes pairs of consecutive edits from different articles, some of which are vandalism instances. All instances are labeled, allowing a full evaluation of classification accuracy. We used Weka to train classifiers and evaluated them with 10-fold cross-validation. As shown in Table 3.3, boosting with J48 decision trees using our features dramatically outperformed the baseline performance from [82], and both logistic regression and SVMs also achieved better precision than the baseline. The results demonstrate the effectiveness of our features and the potential of three classification methods. However, although boosted decision trees achieved the best performance, the method fails to provide an adequate probability distribution to rank the results. Conversely, both logistic regression and

Table 3.3: Classification Comparison on Weimar Dataset

Classifier	Precision	Recall	F-measure
Baseline	0.860	0.845	0.850
Boosting J48	0.945	0.854	0.897
Logistic	0.876	0.774	0.822
SVMs	0.869	0.774	0.819

SVMs provide satisfactory probability distributions to allow for an accurate ranked list. Therefore, we used logistic regression and SVMs to in our experiments with Wikipedia revision history.

3.2.2.1 Active Learning Models and Annotation

Vandalism instances are not systematically archived by Wikipedia. Previous research [49, 84] typically uses regular expressions matched against revision comments to label vandalism, matching any form of the word "vandal" and "rvv" ("revert due to vandalism"). Studies using this labeling approach showed that vandalism only composed a small portion of edits (1-2%) and was fixed relatively quickly (the mean survival time was 2.1 days, with a median of 11.3 minutes). However, matching against comments is insufficient as vandalism is usually corrected without comments. Moreover, in the case of dual vandalism, in which a user vandalized two or more consecutive revisions and reverted only the last vandalism revision to mislead stewards that the vandalism had been corrected, revision comments were no longer accurate indicators for vandalism instances. Hand-labeling thousands of Wikipedia revisions to obtain an accurate training data is labor intensive. We use a supervised active learning model to address this challenge.

Research [63] has shown that supervised active learning benefited situations in

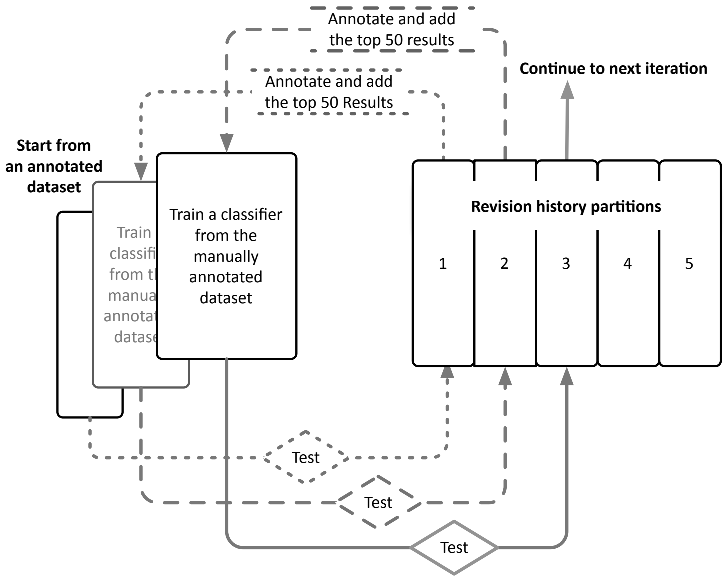


Figure 3.3: Active Learning Models

which labeled training data is sparse and obtaining labels is expensive. In our experiments, we iteratively built classifiers that incorporated the highest-ranked samples from the Wikipedia revision history to detect and rank future vandalism instances. We started with the annotated data provided by Potthast et al. [82] and used it as the baseline dataset. We then divide a revision history into five partitions chronologically. In the first iteration, we built a classifier using the baseline data and tested it on the first partition. The classifier produced a ranked list, and the top 50 results were annotated and added to the existing training pool to build a new classifier for the next iteration. Figure 3.3 illustrates three iterations of active learning.

The annotation process involved labeling whether a revision is a vandalism instance and which type of vandalism it is. An annotator is provided a ranked list of 50 probable vandalism revision identifiers. The annotation interface linked each retrieved identifier to a diff view provided by Wikipedia[5]. An annotator judged from the newly edited content to determine if it is a vandalism instance. An annotator also made the judgement by examining whether the revision was reverted by the next revision[6].

3.2.2.2 Classifiers Performance

Our aim is to classify vandalism instances, providing an accurate ranked list of potential vandalism occurrences. We used a supervised active learning model, learning from the best samples for each of five iterations, to minimize manual effort for the annotation. We used the average precision at 50 revisions that were ranked by classifiers as the most probable vandalism instances to evaluate the performance. Our experiments used two classifiers: logistic regression and SVMs, and worked on two revision histories: “Microsoft” and “Abraham Lincoln”.

Figure 3.4 shows that logistic regression achieved the highest average precision of 0.81 at the 4th iteration for the “Microsoft” dataset and at the 3rd iteration for the “Abraham Lincoln” dataset. SVMs achieved .68 and .76 respectively to “Microsoft” and “Abraham Lincoln” at the third iteration. Both datasets exhibit an increase in average precision from the first to third iteration for either logistic regression or SVMs. The non-monotonic results imply that the underlying distribution of vandalism instances and types varied as a Wikipedia article evolved. One explanation for the decline of the average precision in the last two iterations is the introduction of

[5] http://en.wikipedia.org/w/index.php?diff=prev&oldid=(id)

[6] http://en.wikipedia.org/w/index.php?diff=next&oldid=(id)

Table 3.4: Logistic and SVM Overlap Ratio

	Iteration				
Data	**1**	**2**	**3**	**4**	**5**
Microsoft	0.44	0.29	0.33	0.39	0.54
Lincoln	0.22	0.47	0.25	0.55	0.14

new templates, Wikimarkups, and language links in the later revisions. For example, the insertion and deletion of tags such as {{sprotect}}, {{toolong}}, and {{spilt}} occurred more frequently as the Wikipedia article evolved. Inserting any unseen new tags would increase the perplexity of the current revision and consequently create more false positive instances. Another possibility is that the actual number of vandalism instances decreased in the later revisions.

Our experimental results show that logistic regression and SVMs identified different vandalism instances. Table 3.4 is a tabular view of the overlapping ratio (the intersection over the union) of the two classifiers. This characteristic is most evident at the third iteration for both "Microsoft" and "Abraham Lincoln" data. While both classifiers achieved equivalently high performance, they only overlapped for 0.33 and 0.25 respectively for "Microsoft" and "Abraham Lincoln" data. This, along with the boosting tree results, points to the potential of using ensemble methods for this task.

We observe that classifiers trained from the baseline data can achieve satisfactory performance on the "Microsoft" and "Abraham Lincoln" data. It indicates the potential of training classifiers from heterogeneous sources to use on data from other domains.

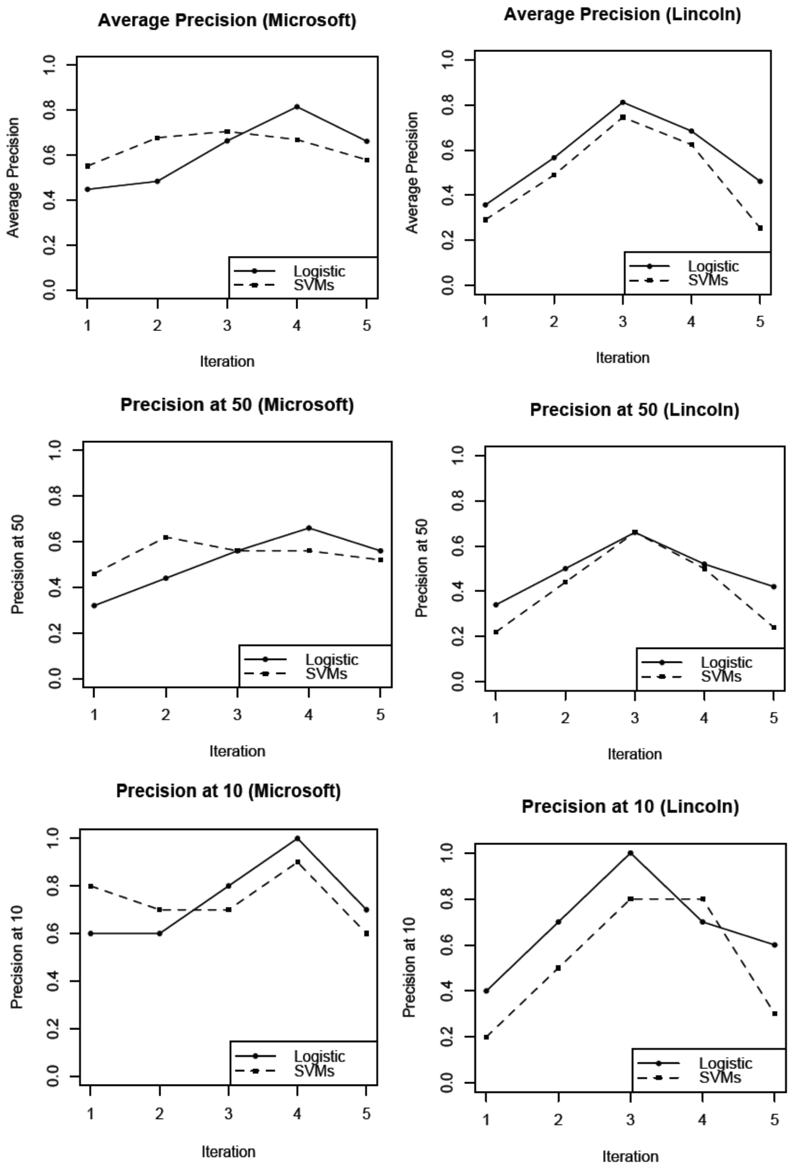


Figure 3.4: Experimental Results for Active Learning

3.3 Divide and Transfer: an Exploration of Segmented Transfer

3.3.1 Motivations

Transfer learning discusses how to transfer knowledge across different data distributions, providing solutions when labeled data are scarce or expensive to obtain. Motivated by the problem of Wikipedia vandalism detection [81, 14], this section investigates the question: *how do we transfer a classifier trained to detect vandalism in one article to another?* We introduce two novel *segmented transfer (ST)* approaches to learn from a labeled but diverse source task, which exhibits a wide-ranging distribution of both positive and negative examples over the feature space, and then selectively transfer the classifier to predict an unlabeled and more uniform target task. Our methods are also tested when transferring between articles with similar distributions.

This work is related to the source task selection problem, investigating methods to enhance transfer learning performance and to minimize negative transfer. We concentrate specifically on transfer at the knowledge level, i.e. the reuse of learned classifiers from a source task, as opposed to transfer at the level of instances, priors, or functions as exemplified by [74]. We investigate two methods to exploit a single source task to predict a target task with no available labels. To improve knowledge transfer, it is useful to identify an effective method to transfer knowledge from the source task to the target task. In this section, we assume that *perhaps not all the source task is useful* and *perhaps not all the target task can learn from the available source task.* This work aims to address the following questions:

- If not all the source task is related to the target task, how do we select the most relevant subset from the source task?
- If not all the target task can be explained or learned from the source task, how do we identify the subset from the target task that can benefit from most the knowledge transfer?

Wikipedia vandalism instances exhibit heterogeneous characteristics. A vandalism instance can be a large-scale editing or a small change of stated facts. Each type of vandalism may demonstrate different feature characteristics and an article may contain more instances of one type of vandalism than others. Moreover, the distribution of different types of vandalism may vary from article to article. For example, the 'Microsoft' article may contain higher ratio of graffiti instances whereas the 'Abraham Lincoln' article may be more vulnerable to misinformation instances. The heterogeneous nature of Wikipedia vandalism detection could potentially introduce negative transfer [89]. It requires a selective mechanism to assure the quality of knowledge transfer, for example, leveraging knowledge about "graffiti" instances from the source task to detect graffiti, as opposed to other types of vandalism instances, in the target task. To resolve the problem of a heterogeneous source task, we introduce two methods to identify the informative segments from the source task in the absence of class labels.

In this section, instead of learning from multiple sources, we focus on the problem setting in which only a single source task is available. Both the source and target task have the same input and output domains, but their samples are drawn

Table 3.5: Tabular comparison of STST and TTST

	STST	**TTST**
Primary assumption:	Not all the source task is useful	Not all the target task can benefit from the available source task
Train cluster models at:	Source task	Target task
Assign cluster membership to:	Target task	Source task
Max number of classifiers:	Number of clusters found in the source task	Number of clusters found in the target task
Transferred object:	Classifiers trained from the source task	

from different populations. Each sample in both the source and target task is a revision of a given Wikipedia article, preprocessed into a feature space representing a collection of statistical language model features. The output labels indicate whether the article is a vandalism instance.

3.3.2 Segmented Transfer

In this section, we propose *segmented transfer (ST)* to enrich the capability of transfer learning and to address the issue of potential negative transfer. The goal of ST is to identify and learn from the most related segment, a subset from the training samples, in the source task. Our motivation comes from two assumptions:

- Not all of the source task is useful, and

- Not all of the target task can benefit from the available source task.

We propose the *source task segmented transfer (STST)* and the *target task segmented transfer (TTST)* approaches to address each assumption and summarize the two approaches in Table 3.5.

3.3.2.1 Source task segmented transfer (STST)

The STST approach clusters the source task, assigning cluster membership to the target task. In Figure 3.5, the labeled source task is first segmented into clusters. Each cluster has its own classifier. We then assign cluster membership to the unlabeled target task and transfer the classifier trained from the corresponding cluster of the source task. Because the distribution of the feature space is different between the source and target tasks, it is likely that some source task data will not be used. The approach aims to transfer knowledge acquired only from the related segment to minimize negative transfer.

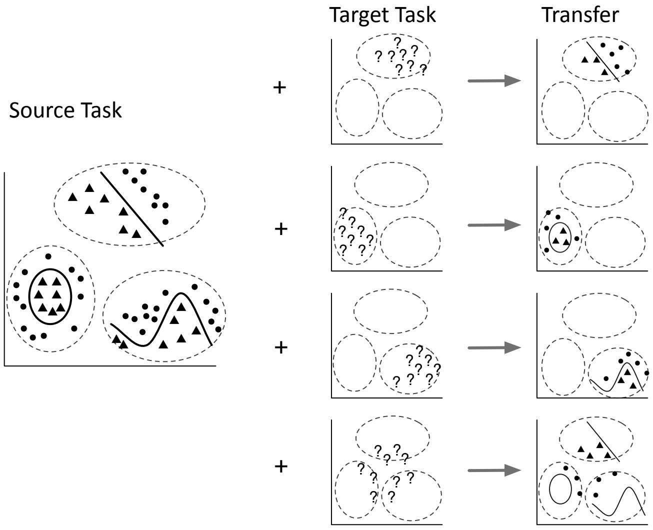


Figure 3.5: Flowchart of source task segmented transfer (STST).

3.3.2.2 Target task segmented transfer (TTST)

The TTST approach clusters the target task, assigning cluster membership to the source task. The goal of the TTST is to differentiate samples that can be better learned from the provided source task. In Figure 3.6, the unlabeled target task is first segmented into clusters. We then assign cluster membership to the labeled source task and train a classifier for each cluster. Finally, the classifiers are transferred to the corresponding clusters in the target task. As shown in Figure 3.6, some data from the target task may not be well learned because of the lack of an appropriate source task.

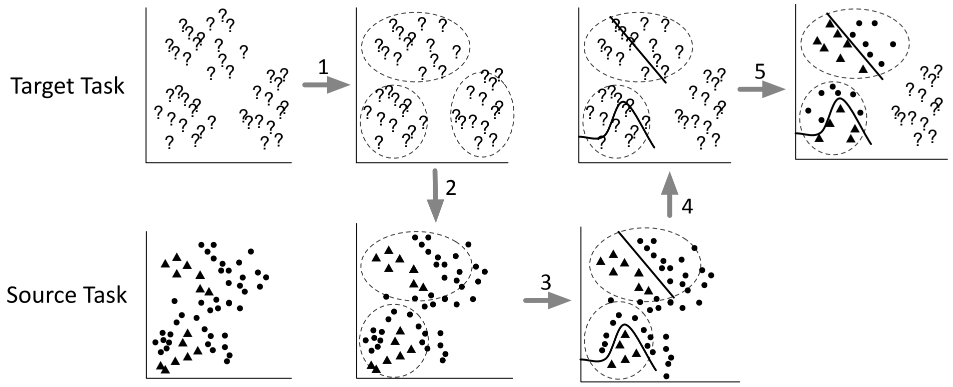


Figure 3.6: Flowchart of target task segmented transfer (TTST).

3.3.3 Experiments

This section describes the datasets used for experiments, the input feature space, the six experimental settings, and the cluster membership assignment distributions for each setting.

3.3.3.1 Dataset description

In four of the experiments, we clustered and trained on the Webis Wikipedia vandalism (Webis) corpus [81] and tested on the revision history of the "Microsoft" and "Abraham Lincoln" articles on Wikipedia [14]. The other two experiments use Microsoft as the source task and transfer to the Lincoln article.

The Webis dataset contained randomly sampled revisions of different Wikipedia articles, drawn from different categories. The Microsoft and Lincoln datasets contained the revision history of those articles. Although class labels were available for both datasets, the class information was ignored during the clustering and was used to build classifiers and to demonstrate the performance of the two methods. Table 3.6 is a tabular description of the three datasets. The AUC and AP scores for the Microsoft and Lincoln dataset were computed by 10-fold cross validation using the provided class labels using an SVM classifier with RBF kernel. The parameters γ and C were chosen empirically to achieve the best performance.

Table 3.6: Dataset description

	Positive	**Negative**	**Total**
Webis	301	639	940
Microsoft	268	206	474
Lincoln	178	223	401

3.3.3.2 Experimental setup and clustering algorithm

Table 3.7 describes six experimental settings. STST and TTST each have three experiments with different combinations of the source and target task. We

Table 3.7: Six experimental settings for STST and TTST

Method	**Exp**	**Source Task**	**Target Task**
STST	1	Webis	Microsoft
	2	Webis	Lincoln
	3	Microsoft	Lincoln
TTST	4	Webis	Microsoft
	5	Webis	Lincoln
	6	Microsoft	Lincoln

used the Weka [42] implementation of clustering, using the Expectation Maximization (EM) algorithm to optimize Gaussian mixture models to cluster the source and target tasks. Using cross validation, the EM algorithm determined the number of clusters to generate. To evaluate the ranked results from the experiments, we used AUC and Average Precision (AP). The ranked list was sorted by the probability of the predictions generated by SVM classifiers.

We used Gaussian mixture model (GMM) optimized with Expectation Maximization (EM) algorithm to assign data to clusters. EM finds clusters by determining a mixture of Gaussians that fit a given dataset. The algorithm is a class of iterative algorithms to estimate maximum likelihood in problems with incomplete data. In our case, the unlabeled target task data are considered incomplete. After training the source task with the EM algorithm, we obtained the cluster assignment of the source task data encoded in means, covariances, and cluster priors in the GMM. We then used EM to assign cluster labels to the target task data. We assigned each data point to the highest probabilistically-weighted cluster label.

3.3.3.3 Cluster Membership Distribution

This paragraph describes the cluster memberships and the distributions of positive and negative instances for the six experimental settings. Tables 3.8 and 3.9 present the cluster assignment distribution for STST. In Experiments 1 and 2, the source Webis dataset is segmented into 16 clusters (see Table 3.8). The target Microsoft and Lincoln datasets are mapped to 9 and 8 of these clusters respectively. The results of cluster assignment confirm the assumption that not all the source task is useful for the target task. However, the source task can still be fully exploited. In Experiment 3, as shown in Table 3.9, all the source task (Microsoft) instances are useful for the target task (Lincoln), both of which were determined to contain three clusters.

Table 3.10 shows the cluster assignment distributions for the TTST approach (Experiments 4, 5, and 6). The distribution shows that sometimes part of the target task would not have available source task to learn from. For example, in Experiment 4, the source task is only useful for cluster 2 of the target task; in Experiment 5, it is only useful for cluster 1.

3.3.4 Experimental results

This section describes the experimental results for STST and TTST. Our results show that the two proposed approaches improved the ranking, moving more actual vandalism instances to the top of the ranked list. Table 3.11 shows the performance of the baseline, a direct transfer without either STST or TTST, using an

Table 3.8: Cluster membership distributions for Experiments 1 and 2

	Source Task	Target Task	
Source cluster	**Webis** **Data Distri.** $(+,-)$	**Microsoft (Exp:1)** **Data Distri.** $(+,-)$	**Lincoln (Exp:2)** **Data Distri.** $(+,-)$
1	75 (9,66)	43 (22,21)	48 (27,21)
2	24 (1,23)	192 (116,76)	85 (41,44)
3	16 (10,6)	153 (80,73)	215 (86,129)
4	25 (8,17)		18 (6,12)
5	46 (24,22)	49 (20,29)	
6	40 (35,5)	16 (16,0)	11 (5,6)
7	41 (3,38)	2 (2,0)	1 (1,0)
8	130 (9,121)		
9	63 (50,13)		
10	43 (9,34)	1 (0,1)	
11	75 (2,73)		
12	43 (6,37)		
13	62 (28,34)	17 (12,5)	22 (11,11)
14	60 (60,0)		
15	149 (8, 141)		
16	48 (39,9)	1 (0,1)	1 (1,0)
Total	940 (301,639)	474 (268, 206)	400 (178,223)

Table 3.9: Cluster membership distribution for Experiment 3.

		Source Task	Target Task
Exp	**Source cluster**	**Microsoft** **Data Distri.** $(+,-)$	**Lincoln** **Data Distri.** $(+,-)$
3	1	344 (186, 158)	357 (146, 211)
	2	125 (80, 45)	42 (30,12)
	3	5 (2,3)	2 (2,0)
	Total	474 (268,206)	401 (178,223)

SVM classifier with linear and RBF kernels. In this section, results that outperform the baseline are marked with a †.

3.3.4.1 STST Evaluation

Table 3.12 shows the experimental results for the STST approach. We compared the performance of STST with the best performance for direct transfer, i.e. train on the source task and transfer directly to the target task, using the SVM classifier with RBF kernel (see Table 3.11). The results indicate that the STST approach

Table 3.10: Cluster membership distribution for Experiments 4, 5, and 6

Exp	**Target cluster**	**Target Task Data Distri.** $(+, -)$	**Source Task Data Distri.** $(+, -)$
4	1	344 (186, 158)	0
	2	125 (80, 45)	940 (301,639)
	3	5 (2,3)	0
	Total	474 (268,206)	940 (301,639)
5	1	56 (36,20)	940 (301,639)
	2	115 (45,70)	0
	3	230 (97,133)	0
	Total	401 (178,223)	940 (301,639)
6	1	56 (36,20)	159 (93,66)
	2	115 (45,70)	121 (56,65)
	3	230 (97,133)	194 (119,75)
	Total	401 (178,223)	474 (268,206)

Table 3.11: Baseline performance.

Exp	**Classifier**	**AUC**	**AP**
1 and 4	SVM w/ linear kernel (C=1)	0.5333	0.6002
	SVM w/ RBF kernel (C=1, $\gamma = 0.1$)	0.5466	0.5862
2 and 5	SVM w/ linear kernel (C=1)	0.5276	0.4528
	SVM w/ RBF kernel (C=0.8, $\gamma = 0.16$)	0.5396	0.4454
3 and 6	SVM w/ linear kernel (C=500)	0.6089	0.6134
	SVM w/ RBF kernel (C=500, $\gamma = 0.02$)	0.6215	0.6021

consistently outperforms the baseline across the three experiments.

Table 3.12: Experiment results for STST

Experiment 1		**Experiment 2**		**Experiment 3**	
AUC	**AP**	**AUC**	**AP**	**AUC**	**AP**
0.5541†	**0.6095**†	**0.5519**†	**0.5063**†	**0.6883**†	**0.6514**†

3.3.4.2 TTST Evaluation

Table 3.13 shows the experimental results for the TTST approach. As shown in Table 3.10, only cluster 2 in Experiment 4 and cluster 1 in Experiment 5 have the source task to learn from. Therefore, presumably, the classifier trained for the assigned cluster in the target task will perform better on the assigned cluster than on other clusters.

The results in Experiment 4 support the assumption. The performance of cluster 2 is much higher than cluster 1 when we used the same classifier trained from the source task for both clusters. Although the cluster 3 in Experiment 4 has high AUC and AP results, it is noted that the size of the cluster is quite small and the results might be insignificant.

Experiment 5 presents mixed results on AUC and AP. We observe that the AP, but not the AUC, is higher in cluster 1, to which all the source task was assigned. In general, AP is more sensitive to the order at the top of the ranked list whereas AUC evaluates the overall number of correctly ranked pairs. In the case that AP is higher but not AUC, it indicates that the algorithm performs better at the top of the list; however, it doesn't create more correctly ranked pairs. To support this observation, we evaluated the results using Normalized Discounted Cumulative Gain (NDCG) at the rank position 5 and 10. Figure 3.7 shows that cluster 1 outperforms the other two clusters. The results suggest the occurrence of negative transfer when the learned classifier was used on less related datasets. The results also demonstrate how negative transfer could be minimized when the target task only learned from more informative segments in the source task.

In Experiment 6, all three clusters from the target task (Lincoln) have assigned instances from the source task (Microsoft). The combined result (the 'Total' row) outperforms the baseline (i.e., direct transfer of a classifier trained from the entire source task).

Table 3.13: Experiment results for TTST, breakdown by cluster

Experiment 4			**Experiment 5**			**Experiment 6**		
#	**AUC**	**AP**	#	**AUC**	**AP**	#	**AUC**	**AP**
1	0.5082	0.5503	1	0.4472	**0.6346**†	1	**0.6792**†	**0.7959**†
2	**0.6569**†	**0.7201**†	2	0.4942	0.3641	2	**0.6288**†	0.495
3	**0.8333**†	**0.8333**†	3	**0.5603**†	0.4393	3	**0.738**†	**0.6637**†
						Total	**0.6627**†	**0.6426**†

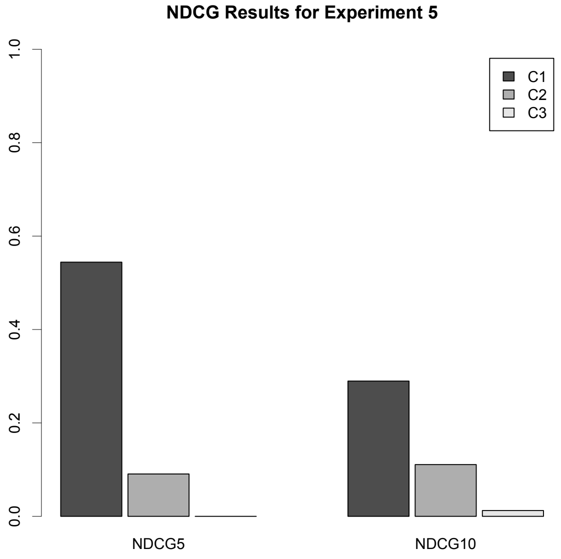


Figure 3.7: NDCG results for Experiment 5

3.4 Entity Search and Classification

We have become dependent on search engines to explore the ever-growing volume of online data. One frequent type of query involves named entities (persons, organizations, locations etc.). Both the Information Retrieval and Semantic Web communities have been studying the problem of entity search, aimed at finding the entity itself instead of merely finding documents that mention the entity. For example, when a search engine receives the query “EU countries,” it would return a list of countries including Germany, France, Netherlands, Great Britain etc. instead of a list of web pages.

One challenge for the problem of entity search is to identify relevant entities for a query that is less common. For example, finding entities for the query “Universities in Kirbati” is a lot more difficult than the query “Universities in the U.S.A.” To address the challenge, we explore using knowledge transfer to leverage knowledge acquired from one entity search topic to another topic. In the experiments, we emphasize the task of entity classification to aid the understanding of entity search.

The experiments used the data collection from INEX XML Entity Ranking (INEX-XER) 2009 track [25]. The track used the Wikipedia 2009 XML data based on a dump of Wikipedia taken on 8 October 2008 and annotated with semantic concepts from the WordNet thesaurus. The entity ranking task aims to return a ranked list of entities for a given query topic. Entities involve countries, persons, novels, movies etc. Examples of the topics include “Science fiction book written in the 1980s,” “Films shot in Venice,” and “Star Trek Captains.” The dataset contains 55 topics with relevance assessments. Our experiments transformed the original entity rank task to an entity classification task using the 55 topics. Each topic has a set of labelled Wikipedia pages, indicating whether the page is about an entity for the topic. For example, the Wikipedia page “James Kirk” is a relevant document for the topic “Star Trek Captains”. In the experiments, we used this dataset to examine knowledge transfer among the topics.

Table 3.14 summarizes the distributions of the number of labelled documents, the number of positive documents, and the number of distinct semantic concepts for the 55 topics. In general, each topic has more than 300 labelled documents. The class label is imbalanced. The average percentage of positive documents for all the 55 topics is 9.9%. Most topics have at least 1,000 distinct semantic concepts annotated in the documents.

3.4.1 Bag-of-Concept (BoC) features

Semantic annotations have shown to be useful for concept-based information retrieval [91]. The INEX 2009 Entity Ranking track also aimed to explore methods that leverage semantic annotations to improve performance for Entity Ranking. In the experiments, we propose constructing bag-of-concepts (BoC) learning models that facilitate knowledge transfer across different but related topics.

Prior research has investigated using BoC approaches to enhance text categorization tasks. Sahlgren and Cöster constructed a concept-based text representations to improve the performance of SVM classifiers, indicating that BoC representations outperformed the Bag-of-Words model for the ten largest text categories. By com-

Table 3.14: INEX-XER Data Distribution

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Doc #	167.0	269.5	316.0	313.9	361.5	438.0
Pos #	8.00	16.00	28.00	30.27	39.00	68.00
Concept #	451	934	1,311	1,267	1,533	2,173

parison, we define concepts differently from the prior works. We adopt the semantic annotations (i.e. the WordNet concepts for the Wikipedia data) as concepts to construct the BoC model.

INEX-XER used Wikipedia 2009 XML data, which has semantic concept annotations from the WordNet thesaurus. There often exist hierarchical structures for the concepts. For example, “American Gods” – a Hugo Awarded best novel – has the semantic concept “novel,” followed by the hypernym concepts “fiction,” “writing,” “written communication,” and “literary composition.” All the concepts are included in the BoC feature set.

Semantic annotations provide a shared feature space for heterogeneous topics. BoC features allow us to identify topics that are related at a higher conceptual level. For example, the term “American Gods” is completely different from “Fahrenheight 451” in a term-vector space. Their cosine similarity is zero in a BoW model. However, since they are both the titles of Hugo awarded best novels, they shared the concepts “fiction,” “writing,” “written communication,” and “literary composition.” A BoC model is hence capable of identifying the high similarity of the two phrases. Such a characteristics captures the conceptual relatedness among topics and facilitates knowledge transfer.

To construct the BoC model, we first extracted the WordNet concepts annotated in the data set. We then represented each document as a vector of concepts, using the tf-idf weights of each concept computed from the entire dataset of 55 topics. The top 1,000 concepts are retained in the feature set. We selected decision trees (J48) and logistic regression models for the experiments. The experiments were implemented with Weka using the default parameter settings. To evaluate the effect of the BoC features, we performed 5 runs of 10-fold cross-validation for the 55 topics and measured the F1 and area under the ROC curve (AUC).

Table 3.15: Performance distribution over the 55 topics using BoC features

Method	Metric	1st Q	Median	Mean	3rd Q	Max
J48	AUC	0.5440	0.6240	0.6366	0.7230	0.9470
	F1	0.1745	0.3000	0.3378*	0.5000	0.8570
Logit	AUC	0.5775	0.6540	0.6437	0.7060	0.9250
	F1	0.1645	0.2580	0.2602	0.3450	0.5110

Note: Each topic is evaluated by 10-fold cross-validation.* indicates that the F1 performance for J48 is significantly higher than logistic regression.

Table 3.16: Top 5 topics ranked by F1

Method	Rank	ID	Topic	F1
J48	1	108	State capitals of the United States of America	0.857
	2	139	Films directed by Akira Kurosawa	0.788
	3	124	Novels that won the Booker Prize	0.691
	4	135	Professional baseball teams in Japan	0.667
	5	91	Paul Auster novels	0.667
Logit	1	139	Films directed by Akira Kurosawa	0.511
	2	86	List of countries in World War Two	0.496
	3	110	Nobel Prize in Literature winners who were also poets	0.462
	4	140	Airports in Germany	0.456
	5	124	Novels that won the Booker Prize	0.442

Table 3.15 shows the distribution of results for the experiments on the 55 topics. With the J48 decision tree model, 50 out of 55 topics have AUC scores higher than 0.5, indicating that the BoC provides informative features for the entity classification problem. The J48 decision model has significantly higher F1 scores

than the Logistic regression model. Table 3.16 describes the top 5 topics of highest F1 scores using BoC. The table shows that decision tree and logistic regression models produce different ranked document lists. However, Topic 139 and Topic 124 are both ranked in the top 5. We used Topic 139 as an example to demonstrate how BoC is used in a decision tree model.

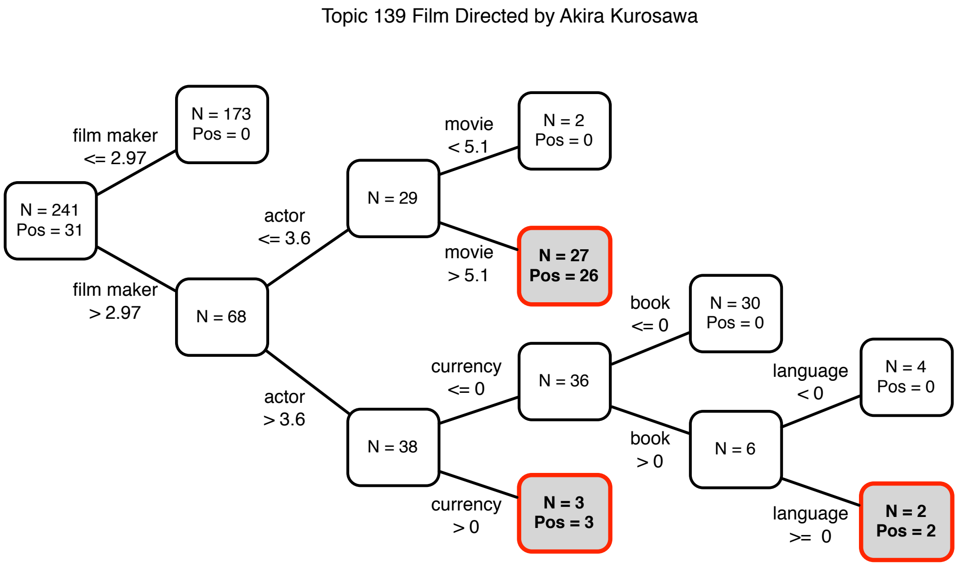


Figure 3.8: Decision Tree for Topic 139

Figure 3.8 shows an example of the decision tree for Topic 139 – Films directed by Akira Kurosawa – using BoC. The top-level node of the decision tree shows that, of the 241 documents for Topic 139, close to 13% (31/241) are positive documents. Under the top-level node, the concept “film maker” is the most predictive feature and is the first concept feature used to split of the documents. If the concept “film maker” has a tf-idf score lower than 2.97, it is highly unlikely the document is relevant to the topic. The second split occurs on the concept “actor.” Most positive documents concentrate in the node where “actor” has a tf-idf score lower than 3.6. The third split on the concept “movie” further improves the precision of the prediction. We obtain a node where “movie” has a tf-idf score higher than 5.1 and and where 97% (26/27) of the documents are positive.

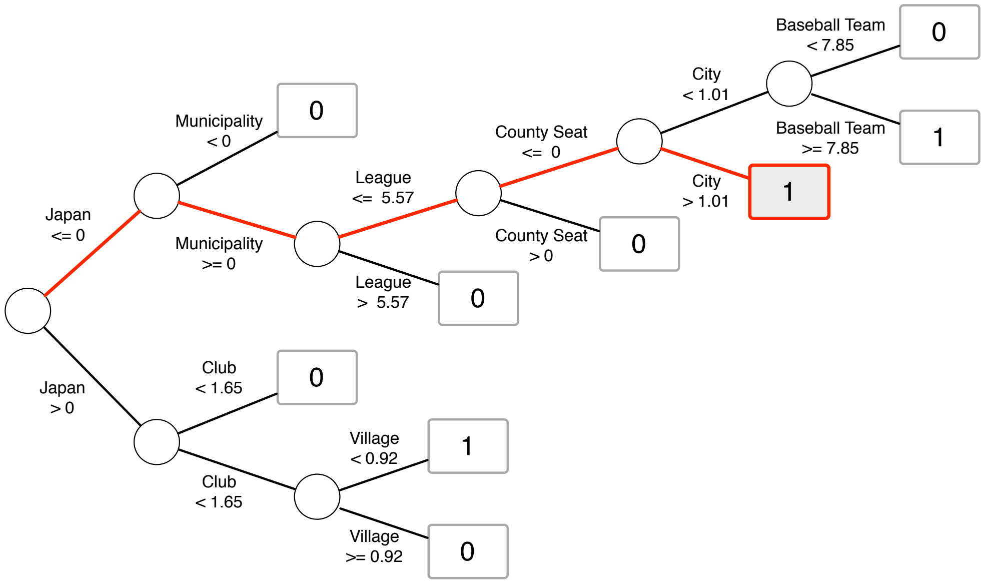


Figure 3.9: Direct transfer between dissimilar topics

Figure 3.9 shows an example of direct transfer of decision tree from the two highly dissimilar topics – “Japanese players in Major League Baseball (Topic 136)” and “List of countries in World War Two (Topic 86).” The decision tree is constructed based on the the topic “Japanese players in Major League Baseball.” The path highlighted in red shows how this decision tree can be useful in identifying “List of countries in World War Two.” This path identifies 17 out of the total 67 positive

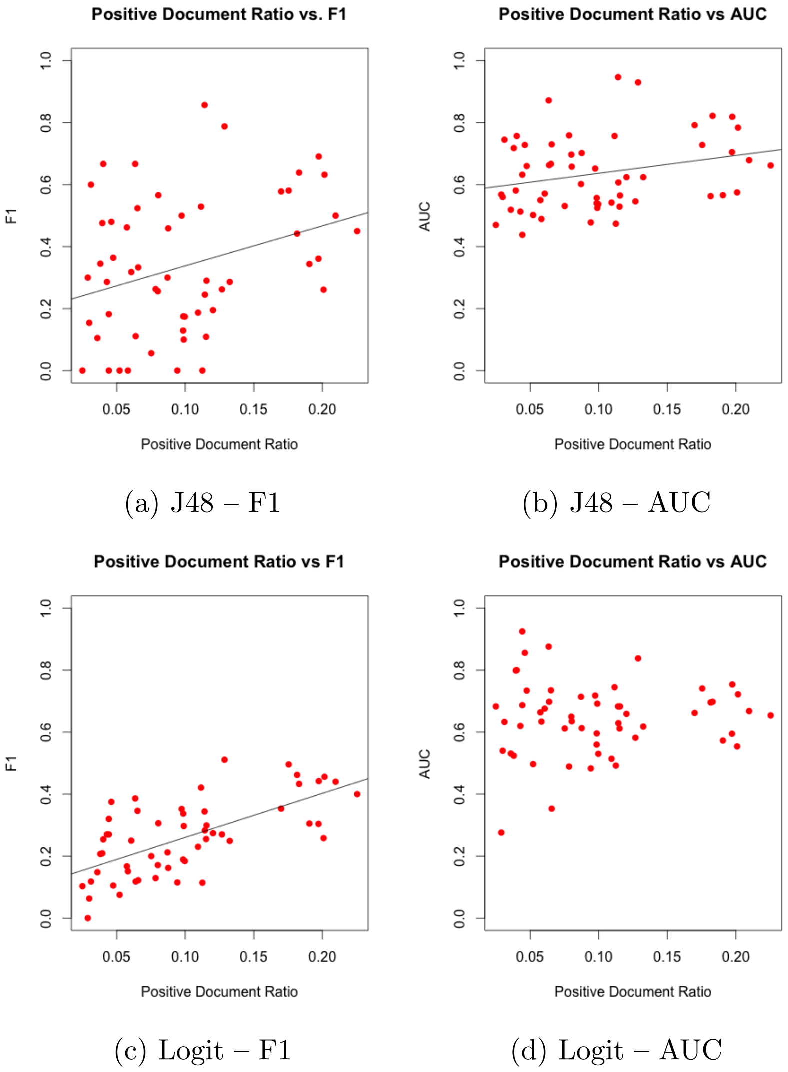
Table 3.17: Correlation Analysis for F1 and AUC

		F1	AUC
J48	Doc #	0.011	0.09
	Pos #	0.317*	0.248
	Pos ratio	0.324*	0.269*
	Concept #	0.092	0.08
Logit	Doc #	-0.096	-0.019
	Pos #	0.606*	0.050
	Pos ratio	0.652*	0.061
	Concept #	0.105	-0.008

instances for Topic 86. The accuracy for the decision path is 17/23. The concepts "Municipality," "League," "County Seat," and "City" have the potential to characterize a country. The discovery of the path (highlighted in red) reveals the obscured relatedness between topics that is unapparent from the surface.

To explore which factors may contribute to the performance of F1 and AUC, we examined whether the four factors – the number of documents, the number of positive documents, the ratio of positive documents, and the number of distinct concepts – are correlated with F1 or AUC. Table 3.17 presents the correlation analysis results for the four factors. Both Table 3.17 and Figure 3.10 show a statistically significant linear correlation between the ratio of positive documents and F1 and AUC for the J48 decision model. We only observed significant correlation between the positive document ratio and F1 for the logistic regression model. However, the number of documents and the number of distinct concepts for each topics show no effect for F1 or AUC.

Early experiments demonstrated that BoC provides valid features for entity classification task. The example decision tree exemplifies how informative concepts can characterize the positive documents. The correlation analysis shows the trend



(a) J48 – F1

(b) J48 – AUC

(c) Logit – F1

(d) Logit – AUC

Figure 3.10: The effect of positive document ratio on F1 and AUC

Note: Figure(d) does not show significant correlation hence the regression line is absent.

that the higher the positive document ratio for a topic, the higher the F1 score. However, the size of the dataset and the number of distinct concepts are not correlated with the performance outcomes.

3.4.2 Direct Transfer

In this section, we investigate the degree of direct transfer of classifiers between topics. As shown in the previous section, a decision tree based on semantic concepts provides explicit knowledge about how to classify a given entity. We therefore examine if such knowledge can be transferred to other related topics, for example, whether the classifier trained from the topic "Novels that won the Booker Prize" can be reused to classify the topic "Hugo awarded best novels." In the experiments, we first explore the similarity among topics (1,485 topic pairs) in our dataset. Second, we examine the direct transfer relationship for all the possible 2,950 topic pairs for the 55 topics.

3.4.2.1 Topic Similarity

To evaluate the similarity between a pair of topics, we first built a vector of all the concepts occurring in the positive documents for each topic. We then computed the cosine similarity of between the two topics with the two concept vectors. Table 3.18 describes the top 10 most similar topic pairs and their categories. The results demonstrates that the method is an effective method to determine topic similarities.

In the experiments, we computed the cosine similarity using concept vectors from positive documents for each topic. For 55 topics, we computed the cosine similarity for 1,485 topic pairs. Although several topic pairs are highly similar as we observe in Table 3.18, the 55 topics used in the experiments are, in general, heterogenous. Figure 3.4.2.1 shows that the majority of the topic pairs has similarity scores lower than 0.4. However, for topics of the same categories, the similarity scores are higher than 0.8.

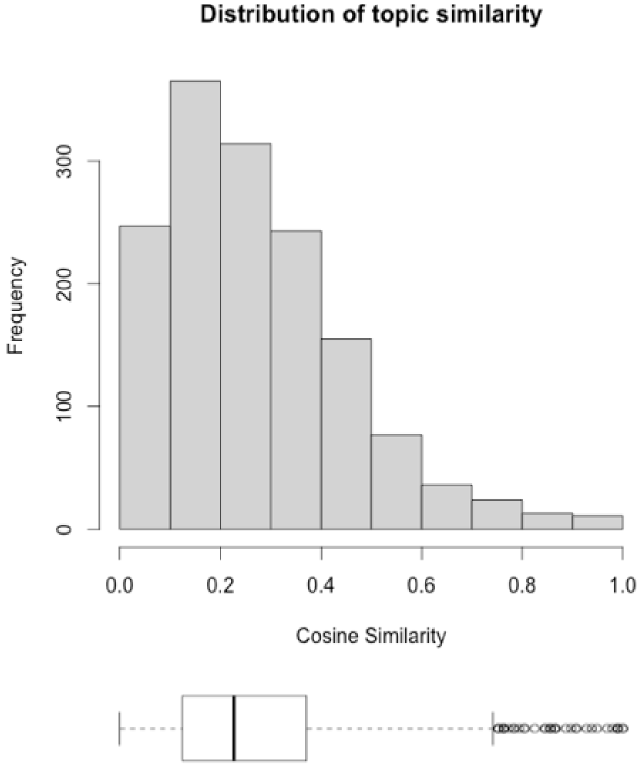


Figure 3.11: Similarity distribution for 1,485 distinct topic pairs.

3.4.2.2 Experimental Results

In the experiments, we considered all possible 2,950 transfer relationships for the 55 topics. We directly applied classifiers trained from the source task to the target task. The goal of the experiments is to identify factors that can influence the

Table 3.18: Top 10 most similar topics

	ID	Topic	Category	ID	Topic	Category	Sim
1	86	List of countries in World War Two	Countries	87	Axis powers of World War II	Countries	0.99
2	86	List of countries in World War Two	Countries	133	EU countries	Countries	0.99
3	87	Axis powers of World War II	Countries	133	EU countries	Countries	0.98
4	63	Hugo awarded best novels	Novels	129	Science fiction book written in the 1980	Novels	0.98
5	94	Hybrid cars sold in Europe	Vehicles	118	French car models in 1960's	Manufacturers	0.98
6	125	countries which have won the FIFA world cup	Countries	133	EU countries	Countries	0.98
7	86	List of countries in World War Two	Countries	125	countries which have won the FIFA world cup	Countries	0.97
8	87	Axis powers of World War II	Countries	125	countries which have won the FIFA world cup	Countries	0.96
9	63	Hugo awarded best novels	Novels	124	Novels that won the Booker Prize	Novels	0.93
10	135	professional baseball team in Japan	Baseball	136	Japanese players in Major League Baseball	Baseball	0.92

Table 3.19: Top 5 target topics benefiting the most from direct transfer.

Source Topic	Target Topic	Sim	F1 (x-val)	F1 (DT)
List of countries in World War Two	Axis powers of World War II	0.99	0.175	0.405
Formula 1 drivers that won the Monaco Grand Prix	Pacific navigators Australia explorers	0.22	0.100	0.310
Nobel Prize in Literature winners who were also poets	Pacific navigators Australia explorers	0.49	0.100	0.304
Living nordic classical composers	Nordic authors who are known for children's literature	0.65	0.056	0.250
Chess world champions	Circus mammals	0.26	0.105	0.296

Note: F1 (x-val) shows the F1 score resulting from the 10-fold cross validation using only the target topic. F1 (DT) shows the F1 score for direct transfer from the source to the target topic.

performance of direct transfer.

Table 3.20 shows the distribution of F1 and AUC performance for direct transfer. We applied J48 decision tree and linear regression model on the 2,950 source and target topic pairs. The results of a Wilcoxon test indicate that logistic regression models significantly outperforms J48 decision trees for direct transfer. It implies that logistic regression models have better transfer capability.

The results also show that the some topics have higher F1 score in the direct transfer experiments than in the cross-validation experiments described in the previous section. We observe that 25 topics show improved results using a transferred decision tree, and 30 topics show improvements on F1 using a logistic regression model. Table 3.19 presents the top 5 source and target topic pairs that have highest improvement on F1 from directly transferring a J48 decision tree from the source to the target topic. We observe from Table 3.19 that effective knowledge transfer can occur not only between similar topics but also dissimilar topics.

Table 3.20: Performance distribution over the 2,950 topic transfer pairs.

Method	Metric	1st Q	Median	Mean	3rd Q	Max
J48	AUC	0.4920	0.5000	0.5006	0.5070	0.8720
	F1	0.00000	0.00000	0.03415	0.03400	0.67100
Logit	AUC	0.4072	0.5020	0.4960	0.5870	0.9240
	F1	0.0450	0.1070	0.1236*	0.1800	0.5950
ZeroR	AUC	0.5000	0.5000	0.5000	0.5000	0.5000
	F1	0.0000	0.0000	0.0000	0.0000	0.0000

Note: Each topic is evaluated by 10-fold cross-validation.* indicates that logistic regression model significantly outperforms the decision tree model and the baseline zeroR model on F1 scores.

We applied correlation analysis to identify influential factors for direct transfer. Table 3.21 shows the results of correlation analysis on five factors – the similarity between the source and the target topic (Sim), the ratio of positive documents in the source topic (Src posRatio), the ratio of positive documents in the target topic (Tar posRatio), the F1 scores from the cross-validation experiments of the source topic (Src F1), and the F1 scores from the cross-validation experiments of the target topic (Tar F1).

The results indicate that similarity between the source and the target topic has positive correlation to F1 scores for both J48 decision tree and logistic regression model. The positive document ratio for both the source and the target topic is also positively correlated to the F1 scores of direct transfer. In addition, we observe positive correlation between the F1 scores from the cross-validation experiments of the target topic and the F1 scores of direct transfer experiments. However, the positive correlation between the F1 scores of the source task from the cross-validation experiments and the F1 scores of direct transfer experiments only exhibits on the J48 decision tree.

As shown in Table 3.22, whether topic similarity contributes to transferability could be dependent on the types of topics. For example, direct transfer contribute to significant performance improvements for topics related to countries and films. However, we do not observe a significant positive influence for topics about novels.

In summary, we conclude that the more similar the source and the target task, the better the direct transfer performance. We obtained improved experimental

Table 3.21: Correlation Analysis for direct transfer

		F1	AUC
J48	Similarity	0.1268*	0.0684*
	Src posRatio	0.2422*	-0.0237
	Tar posRatio	0.0451*	-0.0048
	Src F1	0.1099*	0.1123*
	Tar F1	0.0962*	-0.0065
Logit	Similarity	-0.0544*	0.0009
	Src posRatio	0.0995*	-0.0120
	Tar posRatio	0.3860*	0.0463*
	Src F1	0.0220	-0.0149
	Tar F1	0.3006*	0.0288

Table 3.22: Similarity correlation analysis by categories

		F1	AUC
J48	Country	0.2997*	0.2724*
	Novel	0.1512	-0.0509
	Film	0.0254	0.2054*
Logit	Country	0.0292	-0.0490
	Novel	-0.0454	0.0198
	Film	-0.0644	0.0442

results if more positive documents are available for both source and target topics. Target topics that benefit from BoC features (according to the cross-validation results) also have higher direct transfer outcome. However, source topics that benefit from BoC features do not necessarily transfer well to target topics. Results shown in Table 3.21 provide the following insights:

- Similarity between topics contributes to the transferability. Although we observe negative correlation between similarity and transferability for logistic regression model, it may due to the noise in the data;

- Higher positive document ratio for the source and the target topics contributes to the transferability.

3.4.3 Relative importance of the factors

We used multiple regression to evaluate the relative importance of the factors of transferability. We standardized the three factors – the ratio of positive documents (PosRatio) for the source topic, the similarity between the source and the target topic, and the ratio of positive concepts (PosConRatio) for the source topic – in order to compare their relative importance using the coefficients.

Table 3.23 and Table 3.24 show the results of regression analysis. We analyzed the transferability in AUC an F1 respectively. We first note that, though the three factors are all significant attributes for both cases, the order of their relative importance is different. Table 3.23 shows that PosConRatio is the most important factor contributing to the transferability measured in AUC. However, PosRatio is the most influential factor for the transferability measured in F1 as shown in Table 3.24. The collinearity between PosConRatio and PosRatio explains the negative coefficient for PosRatio in Table 3.23. While AUC evaluates the performance of classifiers without a cutoff threshold, F1 evaluates classifiers with a threshold at 0.5. Therefore, the results imply that PosConRatio aids in correct ranking of the results while PosRatio helps identify correct instances.

Figure 3.13 and Figure 3.12 are the visualizations of the importance comparison of the three factors. The order of relative importance for the factors are consistent across the four metrics – LMG, first, last, and Betasq. We also noted both models have low R^2 scores, indicating that the three factors – PosRatio, Similarity, PosConRatio – are insufficient to explain the transferability between topics. Future work will investigate additional factors that translate into transferability.

Table 3.23: Transferability factors analysis for AUC

Factor	**Coef.**	**P-value**
PosRatio	-4.727e-02	0.01379*
Similarity	5.913e-02	0.00129*
PosConRatio	9.043e-02	2.59e-06*

Table 3.24: Transferability factors analysis for F1

Factor	**Coef.**	**P-value**
PosRatio	2.229e-01	< 2e-16*
Similarity	1.332e-01	6.06e-14*
PosConRatio	9.063e-02	9.76e-07*

3.4.4 Segmented Transfer

This section describes segmented transfer experiments on the entity classification task. In the experiments, we adopted the source task segmented transfer (STST) approach described in the Section 3.3.2. For each source topic, we used the EM clustering algorithm to create three clusters. We then assigned the cluster membership to every data sample (i.e. documents for each topic) of the target topic. Two classification models – J48 decision tree and logistic regression model – are trained from each of the three clusters of a source topic and then tested on the target topic. We transferred only the classifier learned from one cluster of the source topic to the data samples assigned to the same cluster.

Table 3.25 shows the distribution of F1 and AUC performances for the STST

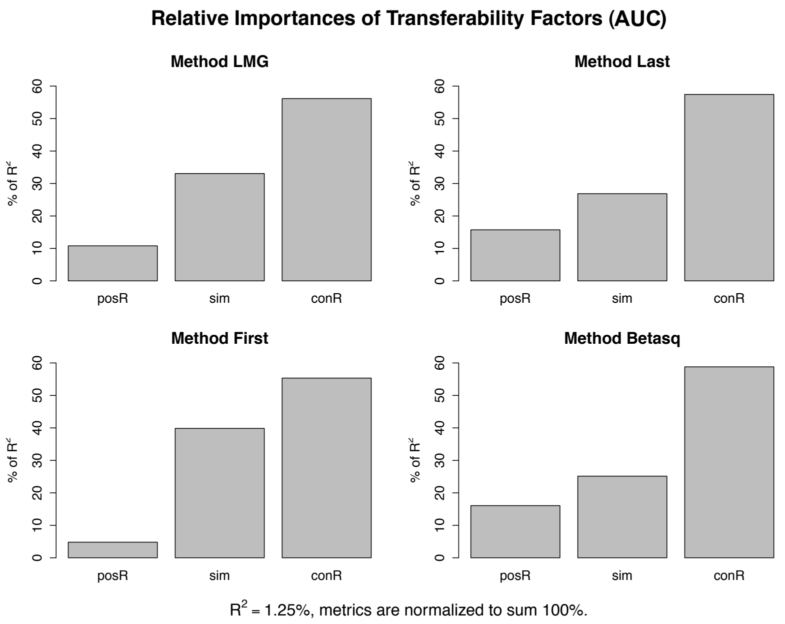


Figure 3.12: Relative importances of transferability factor (AUC)

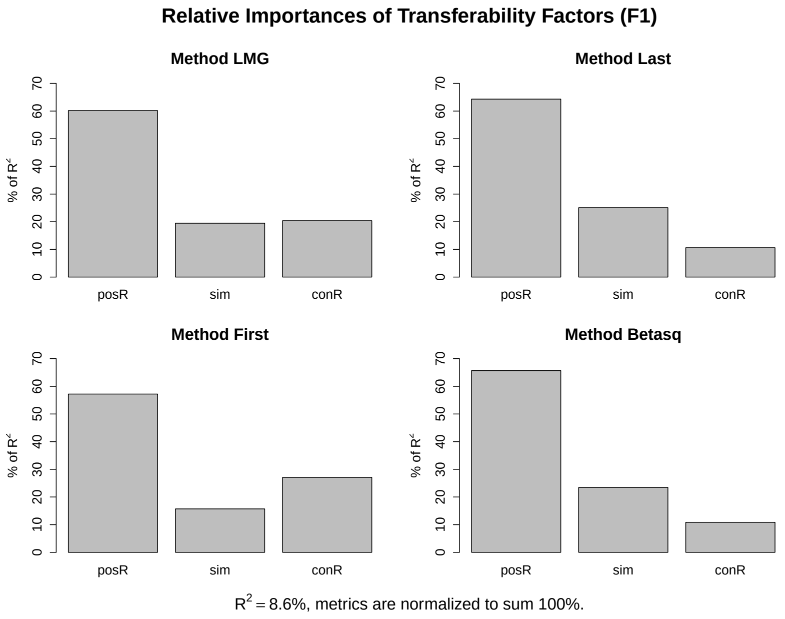


Figure 3.13: Relative importances of transferability factor (F1)

experiments. A paired Wilcoxon test on the performance for J48 decision tree and logistic regression indicates that the logistic regression outperforms J48 decision tree.

Table 3.25: STST performance distribution over the 2,950 topic transfer pairs.

Method	Metric	1st Q	Median	Mean	3rd Q	Max
J48	AUC	0.4106	0.4882	0.4901	0.5720	0.9389
	F1	0.0000	0.0000	0.0308	0.0156	0.5345
Logit	AUC	0.4634	0.5036	0.5030*	0.5475	0.8693
	F1	0.0000	0.0217	0.0623*	0.0769	0.5652

Table 3.26 compares the performances between direct transfer and STST. The paired Wilcoxon test shows that direct transfer outperforms STST by a small margin. However, among the 2,950 topic pairs, 1,769 topic pairs show no performance difference between direct transfer and STST, and 536 topics pairs show improved F1 scores for STST.

Table 3.26: F1 performance distribution for Direct Transfer (DT) vs. STST

Method		1st Q	Median	Mean	3rd Q	Max
J48	DT	0.00000	0.00000	0.03415*	0.03400	0.67100
	STST	0.0000	0.0000	0.0308	0.0156	0.5345
Logit	DT	0.0450	0.1070	0.1236*	0.1800	0.5950
	STST	0.0000	0.0217	0.0623	0.0769	0.5652

The mixed performance of STST may also result from the high dimensionality of feature space. The high-dimensional BoC feature space makes the distance between data samples large. A cluster groups related objects based on observations of their feature values. However, given a large number of features in both the source and the target topics, concept features found in the target topic may not be meaningful to the clusters formed in the source topic. Therefore, the cluster assignments between the source and the target tasks become less accurate. We observe the effect of the "curse of dimensionality" in the STST experiments. We will explore using Principal component analysis (PCA) to reduce the dimensionality of the concept feature space to address the problem.

3.4.5 Summary

This chapter demonstrates a novel *segmented transfer* approach used to address the *how* dimension of transfer learning. The experimental results for the problem of Wikipedia vandalism detection shows how to select and learn from the most related portion in the source task to improve the predicting performance of the target task. In this chapter we also explore the *what* dimension of knowledge transfer by introducing the use of Bag-of-Concepts (BoC) as a common feature space for the source and the target task. The experimental results show that the BoC provide informative features for the problem of entity search and classification. However, despite the fact that BoC largely reduced the dimensionality of features compared to Bag-of-Words models, it remains a sparse feature space and creates challenges for clustering algorithms. Although we did not observe performance improvements from segmented transfer approach using BoC, we believe using dimensionality reduction methods such as Principal Component Analysis may shed lights for the problem.

The next chapter extend the discussion to the *why* dimension. We suggest constructing a knowledge transfer network to visualize and analyze the relationship among different types of vandalism.

CHAPTER 4
KNOWLEDGE TRANSFER NETWORK

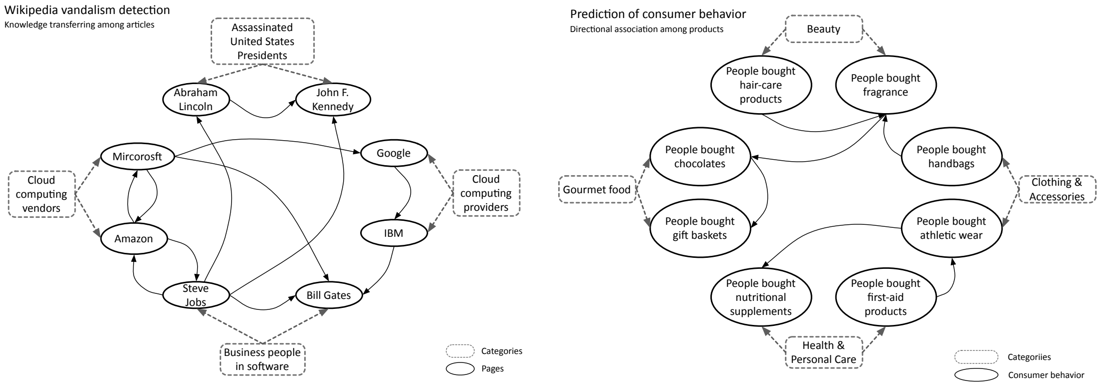
4.1 Introduction

The proposed knowledge transfer network (KTN) aims to address the *why* dimension of knowledge transfer. In Chapter 3, we proposed Segmented Transfer to build a better predictive model using transfer learning. In this chapter, we propose knowledge transfer network to aid in understanding the phenomenon of knowledge transfer. We use KTN to approach the following two questions: How can we identify topics of higher transferability? And, what are the characteristics of topics of high transferability?

To address the *why* dimension, this chapter attempts to develop a knowledge transfer network to discover and characterize knowledge of high transferability – knowledge that can be re-used across various tasks. Inspired by research in network science, the proposed knowledge transfer network is a visualized knowledge representation to describe the dynamic process of transfer learning. In a real-world job market, possessing transferable knowledge makes a job applicant competitive. Transferable knowledge is the previously learned abilities that are applicable in a variety of work settings. For example, the abilities of multi-tasking, time management, and event planning are highly transferable skills that help one become successful in most job settings. Similarly, some knowledge is more central and generalizable to multiple tasks. For example, it would be faster for an apprentice to learn how to make most of the beverages in a coffee shop if the person first masters the making of latte. It is because the task of making a latte involves all the essential techniques (e.g. brewing espresso and frothing milk) for the making of most other beverages.

A knowledge transfer network views the transfer learning relationship as a directed graph, where the nodes represent learning problems, and the directed edges between node pairs represent knowledge flow between problems. The proposed knowledge transfer network would be a novel type of network, using different semantics of nodes and edges from the existing types of networks (e.g. social networks and information networks described in Section 2.4). In a knowledge network, a directed link from a node A to a node B represents the flow of knowledge, that is, the potential of using the knowledge acquired from A to solve the problem for node B.

Figure 4.1 describes two prototypes of knowledge transfer networks to exemplify the potential utility of knowledge transfer networks in real-world applications. In Figure 4.1, each node is a learning problem. The solid directional links indicate the knowledge flow from one node to another, and the dashed lines indicates the categories of nodes, providing informative context for learning problems. The network indicates that the classifiers learned from the page of *Microsoft* can be transferred to the articles *Amazon*, *Google*, and *Bill Gates*. It can also acquire knowledge from the *Amazon* article. Figure 4.1b describes a knowledge transfer network of consumer behavior prediction problem. Each node represents a learning problem for a particular consumer behavior and each link indicates that a learner trained to predict on one behavior can be reused to predict another behavior. For example, the learning model trained from the problem of predicting customers who bought chocolates can be reused to predict customers who would buy gift baskets.


Wikipedia vandalism detection
Knowledge transferring among articles
Assassinated United States Presidents
Abraham Lincoln
John F. Kennedy
Mircorosft
Google
Cloud computing vendors
Cloud computing providers
Amazon
IBM
Steve Jobs
Bill Gates
Business people in software
Categories
Pages
Prediction of consumer behavior
Directional association among products
Beauty
People bought hair-care products
People bought fragrance
People bought chocolates
People bought handbags
Gourmet food
Clothing & Accessories
People bought gift baskets
People bought athletic wear
People bought nutritional supplements
People bought first-aid products
Health & Personal Care
Categoriies
Consumer behavior

(a) Vandalism Detection

(b) Consumer Behavior

Figure 4.1: Knowledge transfer network prototype for two example applications.

The purpose of developing a knowledge transfer network is to discover and characterize tasks or subtasks of higher transferability. An example application is the product recommendation system. In the example knowledge transfer network, each node is a classification task to determine which customers would buy a given product and an edge denotes whether or how much a learned classifier can be reused from one task (node) to another. Therefore, having tasks of higher centrality in the knowledge transfer network implies that knowledge acquired from the task has higher transferability in general. The network also informs the business owner which products are transferable to each other. The benefit of it is to know how to perform the most efficient market survey or estimate the market performance of a new product more precisely.

In this chapter we explore building knowledge transfer network for two applications: Wikipedia Vandalism Detection and Entity Search and Classification. Both sections compare a similarity network to one or more knowledge transfer networks. We use network centrality metrics to approximate the transferability of tasks. Moreover, we analyze what characteristics of tasks may contribute to the its centrality in a knowledge transfer network.

4.2 KTN for Wikipedia Vandalism Detection

In this section we investigate the transfer of learners among different types of Wikipedia vandalism, defined as malicious editing intended to compromise the quality of articles. Common types of vandalism include Blanking, Large-scale Editing, Graffiti, and Misinformation [14]. We investigate how well knowledge acquired to detect one type of vandalism can be reused on another type.

The success of transfer learning requires finding relevant and related source tasks. Similar source tasks induce positive transfer that improves learning in the target task. Too dissimilar source tasks may cause negative transfer that degenerates performance in the target task, i.e. negative transfer [89]. The goal of source task selection is to most efficiently leverage previously-acquired knowledge to learn the target task faster and better. Prior research has suggested selecting source tasks based on their "relatedness" to the target task. Several approaches exist to measure the relatedness between the source and the target task. However, it remains unclear how to properly measure relatedness, or how it translates to actual transfer performance.

To understand how to select appropriate source tasks, we create a *knowledge transfer network* to visualize the transfer relationship between learning problems, using a graphic representation to provide insights on source task selection. We compare the constructed knowledge transfer network to a similarity network and discuss the characteristics for the two networks. The contributions of this section are trifold:

- We use a one-class SVM classifier to characterize a given type of vandalism, investigating how well the classifier can be reused on other types;
- We compute likelihood ratio to compare the performance of a single classifier trained from one type of vandalism to other types;
- We introduce and construct a knowledge transfer network, inspired by network analysis, and compare it with a similarity network, and an existing vandalism taxonomy, to better understand the transfer relationship between different types of vandalism.

We proceed as follows. Each vandalism class is learned using a one-class SVM, using various values of the parameter ν. We then apply the resulting model to the other vandalism classes, and evaluate how well it separates each target class from the negative instances (i.e., legitimate edits). Finally, the results are used to construct the knowledge transfer network. These steps are described in the following subsections.

4.2.1 One-class SVM

One-class classification learns a characteristic function for the target class, defining a classification boundary around the positive (or target) class, such that it includes as many instances as possible from the positive class, while it minimizes the chance to include non-positive instances. One-class classification is useful when the negative class is either absent or improperly sampled, so only the boundary of the target class can be determined definitively by using the data. For example, in order to construct a classifier to detect a given type of vandalism, collecting proper samples of non-vandalized (negative training examples) is very challenging because the negative concept (the legitimate edits) lacks a uniform representation. The purpose of using a one-class classifier in the experiments is to study how well the characteristics learned from one vandalism type can be reused to another one.

We train a one-class SVM using Weka LibSVM (WLSVM) [29], an implementation of the LibSVM [12][1] using the algorithms proposed by Sholkopf et al. [92]. We chose a linear kernel and varied the parameter ν (0.1 to 0.9), which controls the allowable percentage of outliers, to find the best value for transfer. Default values were used for other parameters. We trained the one-class classifier using only the positive data (a given type of vandalism) and tested on a dataset comprised of one other selected vandalism type, plus the legitimate edits (negative examples).

4.2.2 Likelihood Ratio

In evaluating the ability of a learned one-class model to transfer from one task to another, we find that standard measures such as recall (sensitivity) and precision (positive predictive value) are inadequate. Consider a model that predicts all cases

[1] The tool is available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

as positive. The recall on the target task will be 1.0, indicating perfect performance despite the fact that no useful learning has taken place. Precision, meanwhile, is dramatically affected by the prior probability of the positive class, and will appear artificially higher for more populous classes (such as Graffiti) than for others.

To overcome the limit of recall and precision measures, we adopt *likelihood ratio* to measures the change in probability. That is, the increase of likelihood ratio indicates an increase of probability that a predicted positive instance is, in fact, positive. Likelihood ratio is defined as

$$\text{Likelihood Ratio} = \frac{\text{sensitivity}}{1 - \text{specificity}} = \frac{\frac{\text{TP}}{\text{TP+FN}}}{\frac{\text{FP}}{\text{FP+TN}}}$$

where the counts TP = true positives, TN = true negatives, FP = false positives, and FN = false negatives. Values of likelihood ratio greater than one indicate positive transfer; that is, the model learned on the source task successfully increases the probability of predicting the target task. Values less than one indicate negative transfer, and those near one indicate that the source is neither helpful nor harmful to the target.

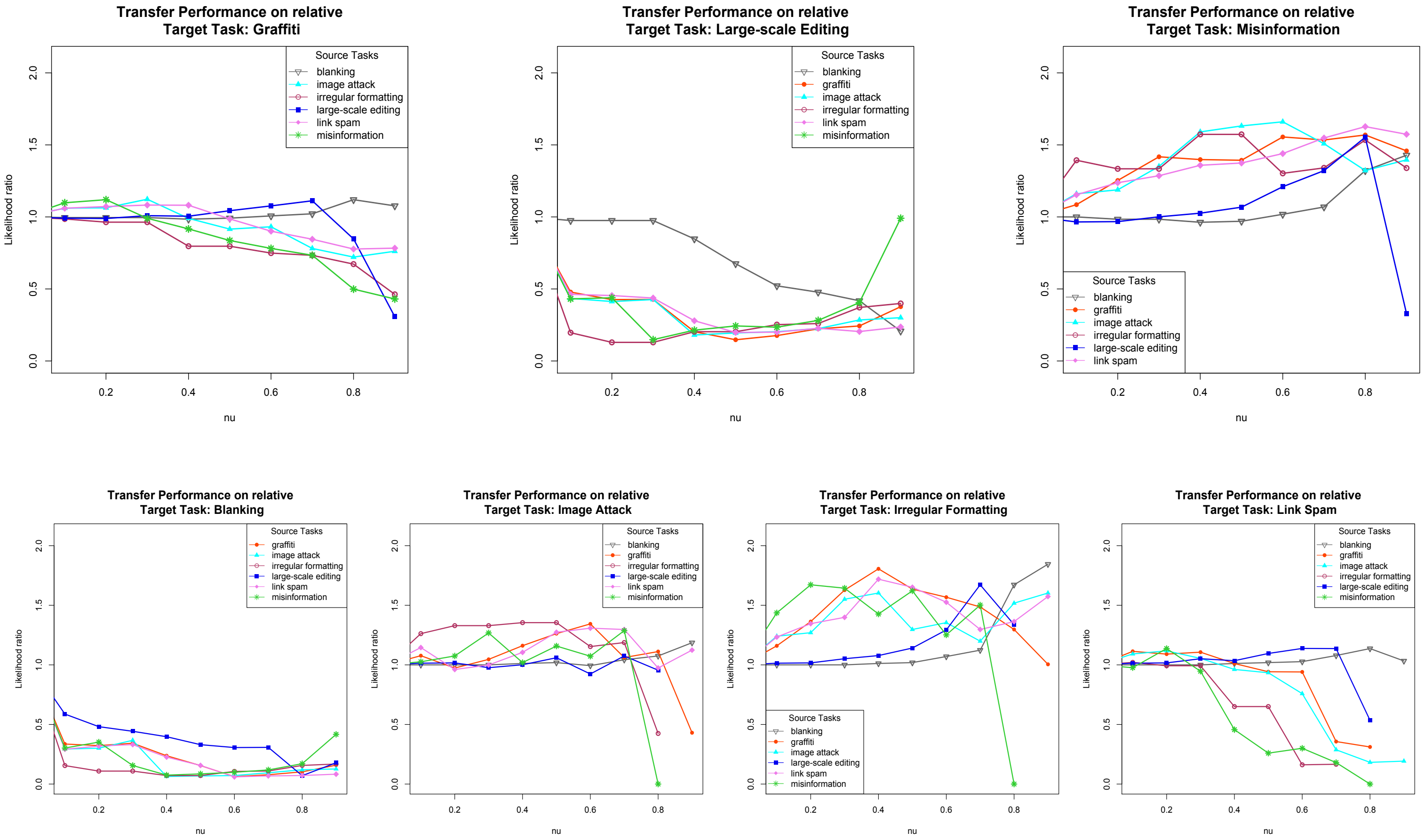


Figure 4.2: Experimental results on likelihood ratio for the seven source tasks.

Figure 4.2 shows the results of likelihood ratio over the nine different choices of ν for each target task. Each line in a panel represents a source task. The results show that knowledge acquired from other source tasks is only moderately transferable to the graffiti and the link spam types. All source tasks introduce negative transfer to the blanking and the large-scale editing. The top-right panel shows that the type Image Attack and Irregular Formatting have higher transferability to the misinformation type. The types Misinformation, Image Attack and Irregular Formatting would benefit from transfer learning.

Table 4.1: Likelihood Ratio performance between vandalism types.

Source Task	**Target Task**						
Blanking (B)		1.1196	1.1859	1.8448	0.9750	1.1368	1.4282
Graffiti (G)	0.3426		1.3438	1.8059	0.4784	1.1135	1.5689
Image Attack (I)	0.3672	1.1227		1.6030	0.4342	1.1176	1.6602
Irregular Formatting (F)	0.1681	0.9860	1.3544		0.3992	1.0209	1.5728
Large-scale Editing (L)	0.5871	1.1120	1.0747	1.6718		1.1378	1.5532
Link Spam (S)	0.3312	1.0822	1.3078	1.7188	0.4625		1.6268
Misinformation (M)	0.4171	1.1196	1.2872	1.6718	0.9907	1.1368	
	B	**G**	**I**	**F**	**L**	**S**	**M**

Table 4.2: The choice of ν for one-class SVM to achieve the highest likelihood ratio.

Source Task	**Target Task**						
Blanking (B)		0.8	0.9	0.9	0.1	0.8	0.9
Graffiti (G)	0.3		0.6	0.4	0.1	0.1	0.8
Image Attack (I)	0.3	0.3		0.4	0.1	0.2	0.6
Irregular Formatting (F)	0.9	0.1	0.4		0.9	0.1	0.4
Large-scale Editing (L)	0.1	0.7	0.7	0.7		0.6	0.8
Link Spam (S)	0.3	0.3	0.6	0.4	0.1		0.8
Misinformation (M)	0.9	0.2	0.7	0.2	0.9	0.2	
	B	**G**	**I**	**F**	**L**	**S**	**M**

Table 4.1 shows the values of likelihood ratio for each source task / target task

pair. These values are best-case results, obtained with the values of ν indicated in Table 4.2. The distribution of ν indicates that the values of ν vary to achieve the best performance on the likelihood ratio. The choice of ν is dependent on the pair of the source and the target task.

4.2.3 Constructing the Knowledge Transfer Network

In a knowledge transfer network, a node represents a learning problem (e.g. how to classify graffiti instances, the target class) and the in-links of a node indicate the direction of transfer learning. Figure 4.3 visualizes all the positive transfer pairs from Table 4.1. The more in-links a node has, the more choices of source tasks it has.

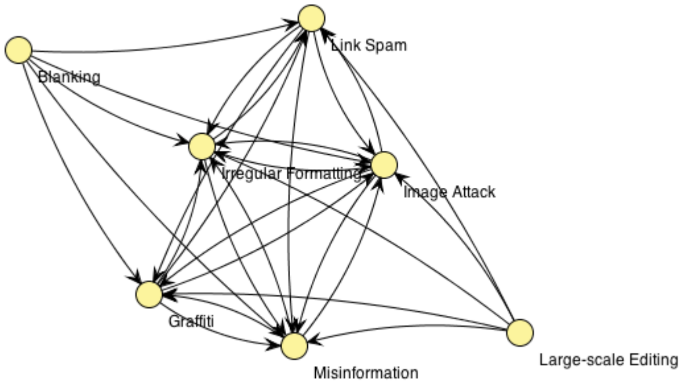


Figure 4.3: Knowledge Transfer Network for Wikipedia vandalism detection

Some of the smallest – and most difficult to detect – classes, such as Misinformation and Image Attack, can benefit most generally from transfer. Conversely,

Table 4.3: Network analysis of transfer network

Types	In-D	Out-D	Degree	Hub	Authority	Pagerank
Blanking (B)	0	5	5	0.4614	0.0257	0.0214
Graffiti (G)	5	4	9	0.3776	0.4049	0.1676
Image Attack (I)	6	4	10	0.3659	0.4615	0.2151
Irregular Formatting (F)	6	3	9	0.2785	0.4785	0.2033
Large-scale Editing (L)	0	5	5	0.4614	0.0257	0.0214
Link Spam (S)	5	4	9	0.3776	0.4049	0.1676
Misinformation (M)	6	3	9	0.2785	0.4785	0.2033

Blanking and Large-Scale Editing do not benefit at all from transfer, but do transfer well to other classes. In the cases where Blanking and Large-Scale Editing transfer, the value of ν is high, indicating that it is only the central core of the concept that transfers from, e.g., Blanking to Misinformation.

Table 4.3 quantifies the knowledge transfer network. Concepts with high hub scores, such as blanking and large-scale editing, are easier to differentiate. They are vandalism types that are easier to detect and they are appropriate source tasks in transfer learning. Concepts with higher authority scores, such as image attack, irregular formatting, and misinformation, are intrinsically harder to detect as observed from the results predicted by one-class SVM. Observations from Table 4.3 indicate the higher the centrality (e.g. pagerank) of a target concept, the more diverse it is, and the more it would benefit from transfer learning.

The results from Table 4.3 also match the handcrafted taxonomy in Figure 3.1. Concepts with higher centrality in the knowledge transfer network (e.g. image attack, irregular formatting, and misinformation) are the leaves in the taxonomy. On the other hand, concepts with lower centrality (e.g. blanking and large-scale editing) have higher hierarchy in the taxonomy.

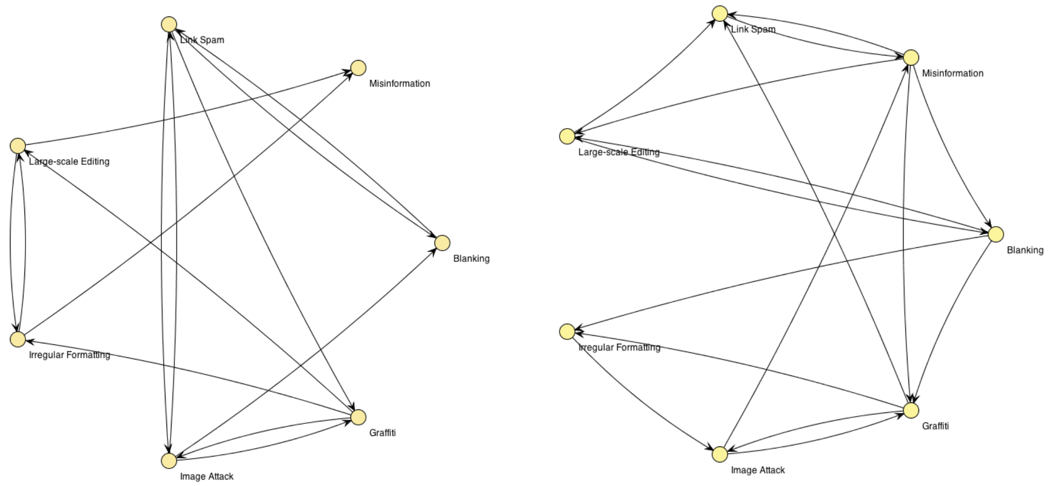
Table 4.4: P-value similarity between vandalism types

Blanking (B)	1						
Graffiti (G)	0.2989	1					
Image Attack (I)	0.6991	0.8657	1				
Irregular Formatting (F)	0.2989	0.4436	0.4410	1			
Large-scale Editing (L)	0.2029	0.4147	0.4666	0.7621	1		
Link Spam (S)	0.7155	0.6918	0.9219	0.3854	0.3499	1	
Misinformation (M)	0.0157	0.0307	0.0452	0.3676	0.1367	0.0284	1
	B	**G**	**I**	**F**	**L**	**S**	**M**

4.2.4 Constructing the Similarity Network

This section describes how we construct a similarity network to represent the transfer learning relationship. In a similarity network for knowledge transfer, each node is a learning problem and each directed link indicates the similarity of the two problems. For each target task (a vandalism type) we select the two source tasks of highest similarity values to draw the two directed links pointing to the target task. The measure of similarity is the P-value computed from Student's T-test to estimate the divergence of two sample distributions. The higher the P-value is for the two target tasks, the more similar they are. Figure 4.4a shows the similarity between each source / target task pairs.

Figure 4.4b is the knowledge transfer network described in the previous section. In order to have a fair comparison with the similarity network, we present the two in-links of the task by selecting the two source tasks of highest likelihood ratio for each target task (a vandalism type). Figure 4.4a and Figure 4.4b are almost completely distinct, indicating that the similarity between tasks does not necessarily translate into an opportunity for transfer learning. Misinformation, for example, is not highly similar to any other classes, but still transfers well.



(a) Similarity network (b) Transfer network

Figure 4.4:
Comparison between similarity network and knowledge transfer network

4.3 Knowledge Transfer Network for Entity Search and Retrieval

In this section we applied KTN to the problem of entity search and classification. Figure 4.5 shows an example of the knowledge transfer among different topics and from categories to instances, e.g. reusing the models from “Germany” to “Airports in Germany”. It could be that the learned models are transferable among nodes of the same categories (e.g. between “Airports in Germany” and “Universities in Bavaria”) and that models learned from “Airports in Germany” are transferable to topics different categories (e.g. the topic “Universities in Catalunya” under the “Spain” category).

The purpose of a KTN is to discover and characterize topics of higher transferability. Applying network analysis on a KTN would reveal nodes of high centrality, measuring the importance of certain types of topics. For example, a topic of higher degree centrality implies that knowledge acquired from the topic has higher transfer-

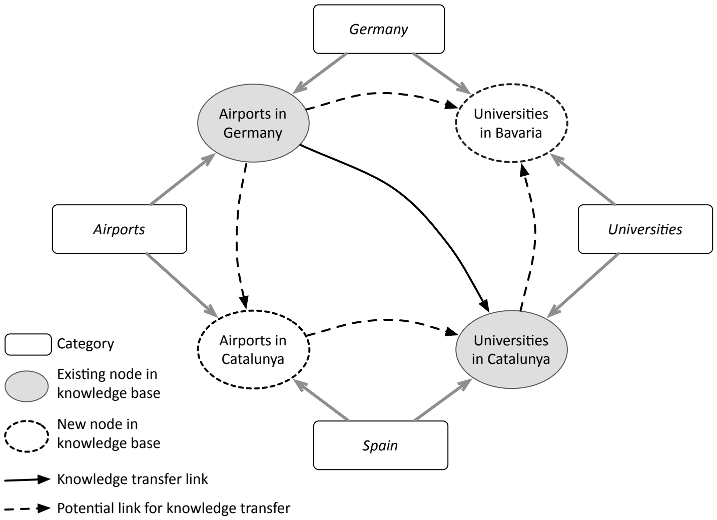


Figure 4.5: Example of Knowledge Transfer Network (KTN).

Each node is a learning problem (e.g. topic). The solid directional links indicate the observed knowledge flow from one node to another, and the dashed lines are the to-be-learned knowledge flow.

ability. Similarly, topics with higher betweenness centrality have more control on the knowledge flow in the network. Patterns observed from a KTN would help determine how to efficiently allocate annotation effort. For example, it could be more productive to obtain more labelled training data for "Airports in Germany" than "Universities in Catalunya" if the knowledge acquired from the topic is highly transferable.

KTN is not only a knowledge representation, but also a method to derive heuristics to improve learning. For example, we may observe that a source topic with more positive documents in the dataset transfers well to more target topics. Although KTN may resemble a top-down ontology in appearance, the goal of KTN is to visualize the knowledge transfer relationships among problems, extracted in a bottom-up fashion, and the interactions among predefined categories. KTN may provide insights on identifying ontological connections that were initially obscured between entity retrieval problems.

4.3.1 Similarity Network

Figure 4.3.1 shows a simplified topic similarity network with the cosine similarity greater than 0.7. We color-coded the topics associated with the top 3 popular pre-defined categories (see Table 4.5) in the dataset – countries, novels, and films. Figure 4.3.1 demonstrates that topics in the same category are closely connected in the graph. However, the clusters of topics of the same categories are less evident in a similarity network of lower cosine similarity threshold.

Figure 4.7 shows the topic similarity network of cosine similarity threshold equal to 0.5 (Figure 4.7a) and the discovered topic communities that contain topics from multiple categories (Figure 4.7b). Figure 4.7a shows a large and dense cross-connected topics, indicating that numerous topics, regardless of their pre-defined categories, are similar. Therefore, a network community discovery method can be useful to identify natural groupings of topics. Figure 4.7b visualizes network communities discovered by random walks [79]. The method assumes that short random walks tend to stay in the same community. Hence the method can identify densely connected subgraphs in a sparse graph. The example results (Table 4.6 and Table 4.7) shows that the method is effective in grouping related topics across different categories.

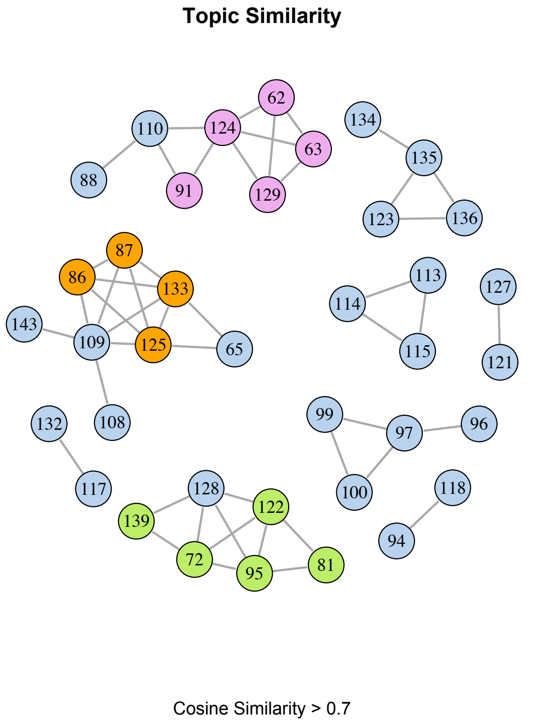


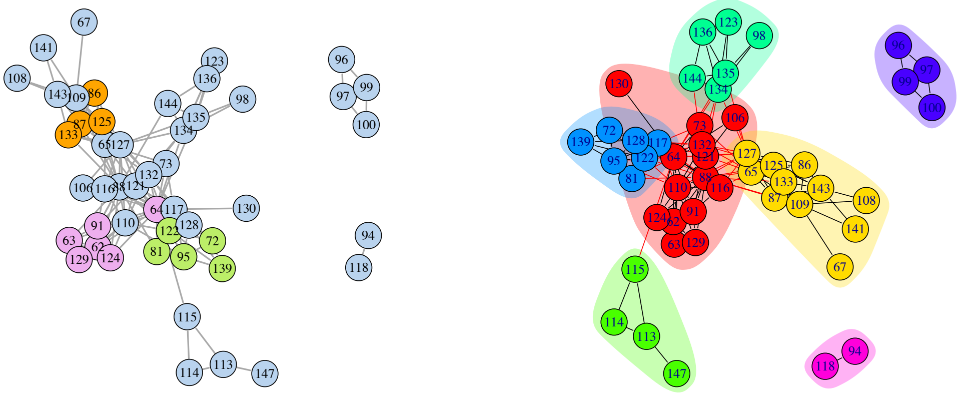
Figure 4.6: Topic similarity network (Cosine Similarity > 0.7)

Table 4.5: Top 3 categories and the associated topics

Category	ID	Topic
Countries	86	List of countries in World War Two
	87	Axis powers of World War II
	125	countries which have won the FIFA world cup
	133	EU countries
Novels	62	Neil Gaiman novels
	63	Hugo awarded best novels
	91	Paul Auster novels
	124	Novels that won the Booker Prize
	129	Science fiction book written in the 1980
Films	72	Films shot in Venice
	81	Movies about English hooligans
	95	Tom Hanks movies where he plays a leading role
	122	Movies with eight or more Academy Awards
	139	Films directed by Akira Kurosawa

Table 4.6: Example of topic community – sport related topics

ID	Topic	Category
98	Makers of lawn tennis rackets	Tennis
123	FIFA world cup national team winners since 1974	Football
134	record-breaking sprinters in male 100-meter sprints	Sprinters
135	professional baseball team in Japan	Baseball
136	Japanese players in Major League Baseball	Baseball
144	chess world champions	Chess



(a) Topic similarity network (Cosine Similarity > 0.5)

(b) Topic community via random walk

Figure 4.7: Topic category vs. community

Table 4.7: Example of topic community – movie related topics

ID	Topic	Category
72	Films shot in Venice	Films
81	Movies about English hooligans	Films
95	Tom Hanks movies where he plays a leading role	Films
117	Musicians who appeared in the Blues Brothers movies	Musicians
122	Movies with eight or more Academy Awards	Films
128	Bond girls	Film actors
139	Films directed by Akira Kurosawa	Films

4.3.2 Direct Transfer Network

In this section, we construct knowledge transfer network using the results from direct transfer experiments in Section 3.4.2. The direct transfer experiments investigate two classification methods – J48 decision tree and logistic regression model – and use the area under ROC curve (AUC) statistic for model comparison. Minimally, classifiers should perform better than AUC of 0.5. Predictors with an AUC less than 0.5 are negative predictors, and predictors with a ROC area between 0.5 and 1 are positive predictors. Therefore, we determine if a direct transfer occurs when the predictors trained from the source task has an AUC greater than 0.5 when they are tested on the target task.

Figure 4.8 and 4.9 visualize and compare a topic type-centric similarity network and a type-centric direct transfer network. A type-centric network visualizes only nodes connecting to certain types of topics. In our examples, we use three types (categories) of topics: countries (orange-colored nodes), novels (pink-colored nodes), and films (green-colored nodes). We observe from the two figures that the knowledge transfer networks are quite different from the similarity network. A notable example is the topic 64 "Alan Moore graphic novels adapted to film." The topic is similar to four topics in the films categories. However, only the topic 72 "Films shot in Venice" is transferable to the topic 64.

Community discovery method is also applicable to transfer network. Figure 4.10 shows an example of community detection on a transfer network. The examples from Table 4.8 show which topics can be grouped in the same community in the

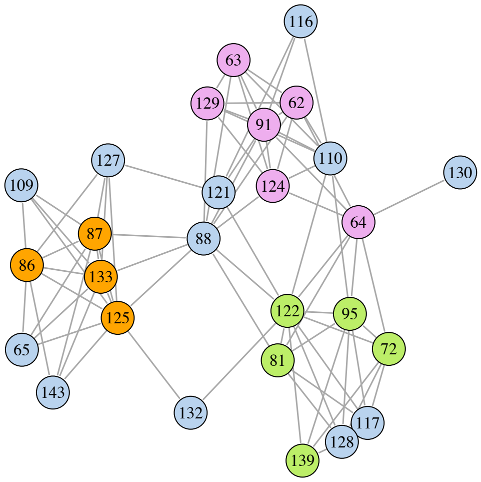


Figure 4.8: Type-centric similarity network

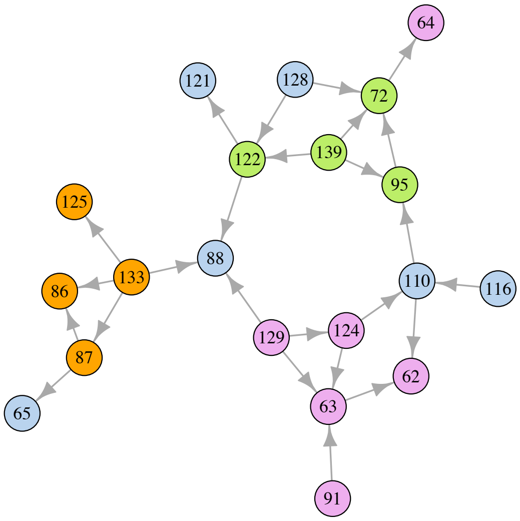


Figure 4.9: Type-centric J48 direct transfer network

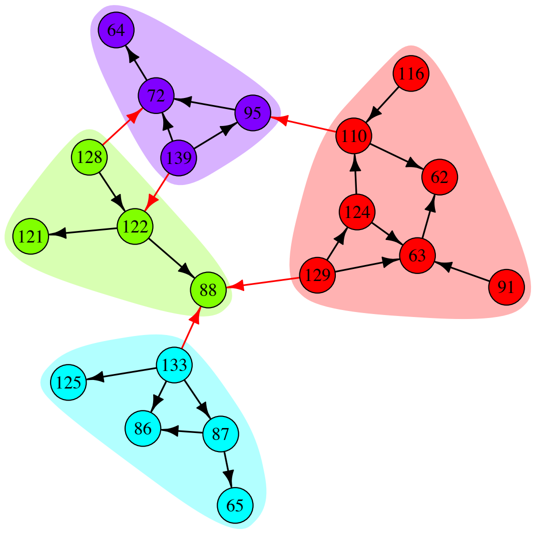


Figure 4.10: Type-centric J48 direct transfer network

transfer network.

Figure 4.11 and 4.12 visualize direct knowledge transfer where the AUC is greater than 0.7 and 0.8 . The two AUC thresholds are selected in order to create direct knowledge transfer networks of the size equivalent to the similarity network shown in Figure 4.3.1. We observe that the knowledge transfer networks are quite different from the similarity network. Moreover, knowledge transfer networks created by different predictive models also vary.

Table 4.9 compares the network properties for the two complete direct knowledge transfer networks created by each predictive model. The logistic regression model creates a larger and denser knowledge transfer network than J48 decision tree.

We examine the correlations between centrality metrics (i.e. degree, in-degree,

Table 4.8: Example community (red)

ID	Topic	Category
62	Neil Gaiman novels	novels
63	Hugo awarded best novels	novels
91	Paul Auster novels	novels
110	Nobel Prize in Literature winners who were also poets	nobel laureates
116	Italian nobel prize winner	nobel laureates
124	Novels that won the Booker Prizes	novels
129	Science fiction book written in the 1980	novels

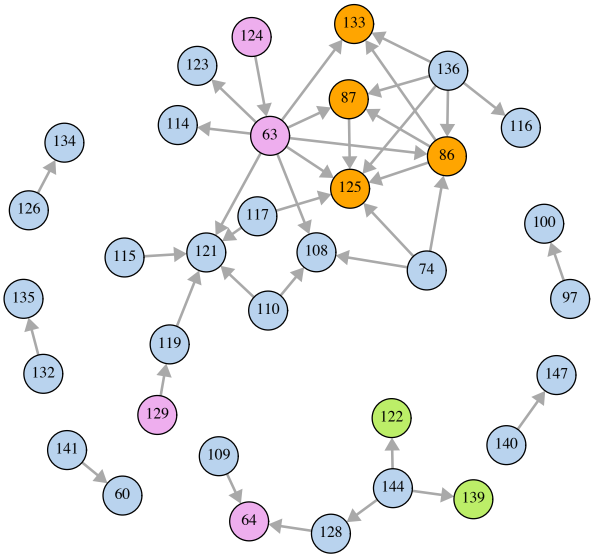


Figure 4.11: Direct Transfer Network with J48 Decision Tree

Table 4.9: Direct knowledge transfer network properties

Network Property	Sim	J48	Logit
Num of nodes	48	55	55
Num of edges	161	499	1,413
Density	0.1427	0.1680	0.4757
Avg. path length	2.3755	2.009	1.524

Direct knowledge transfer network properties – J48 decision tree and logistic regression

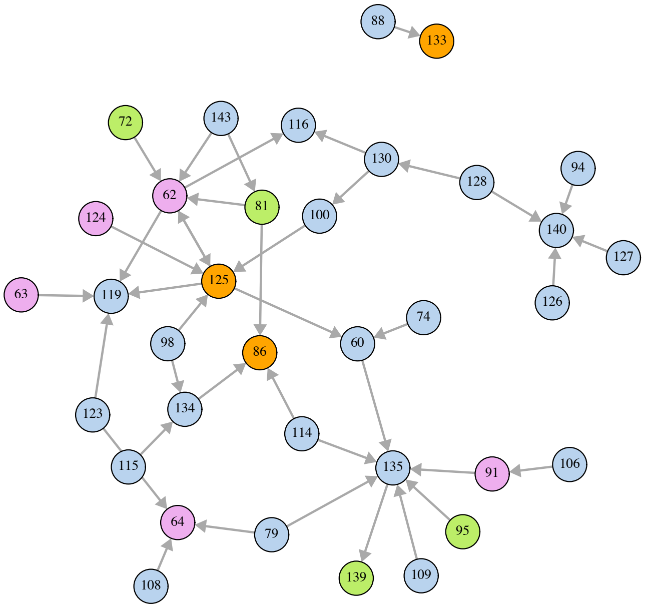


Figure 4.12: Directed Transfer Network with logistic regression model

out-degree, closeness, betweenness, and page rank) and the topic features (i.e. the number of documents annotated for each topic, the ratio of positive documents, the number of distinct concepts, and the ratio of positive concepts). Table 4.10 shows the results for the correlation analysis. The results indicate that most topic features are not correlated to topic centrality in the network created by logistic regression model, except for the positive document ratio is highly correlated with the pagerank of a topic. However, for the network created using J48 decision tree, the positive document ratio is correlated with degree and closeness centrality. The positive correlation between positive document ratio and out-degree indicates that the more positive documents a topic has in the dataset, the higher transferability it has. On the other hand, we observe high positive correlation between positive concept ratio and in-degree, in-

Table 4.10: Correlation analysis for node centrality

		Degree	In-degree	Out-degree	Closeness	Betweenness	PageRank
J48	# Docs	0.2032	0.3299*	0.0898	0.0433	0.1146	0.2533
	PosDoc Ratio	0.4076*	0.1207	0.4537*	0.3402*	0.5157	-0.0503
	# Concepts	0.2167	0.5350*	0.0032	0.1302	0.0585	0.5654*
	PosCon Ratio	0.5834*	0.7046*	0.3804	0.3143*	0.2520	0.6939*
Logit	# Docs	0.0394	0.1329	-0.0460	0.1129	0.0551	-0.0891
	PosDoc Ratio	0.2567	0.2059	0.1866	-0.1619	-0.4139*	0.7209*
	# Concepts	-0.1954	-0.0735	-0.2034	0.1637	0.0165	0.0142
	PosCon Ratio	0.0896	0.1157	0.0328	0.1465	-0.4026*	0.6068

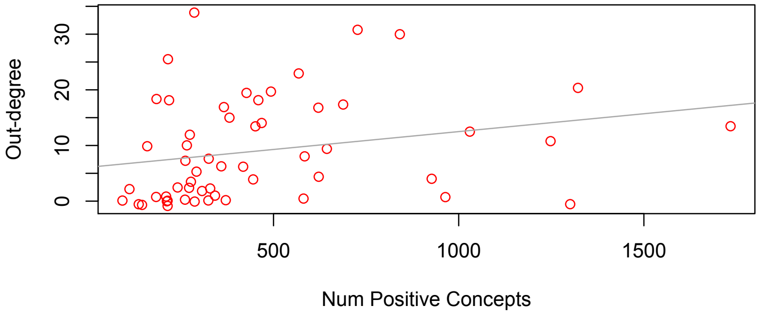


Figure 4.13: Out-degree and positive concept ratio

dicating that the more distinct concepts in the positive documents, the more the topic can benefit from knowledge transfer. We also observe the same pattern between the positive concept ratio and the PageRank.

Table 4.11 and Table 4.12 show the top 5 topics of high transfer centrality (i.e. high out-degree) and the top 5 topics benefit most from knowledge transfer (i.e. high in-degree). In general, as shown in Figure 4.13 and Figure 4.14, we observe that high ratio of positive documents and high ratio of positive concepts contribute to high centrality.

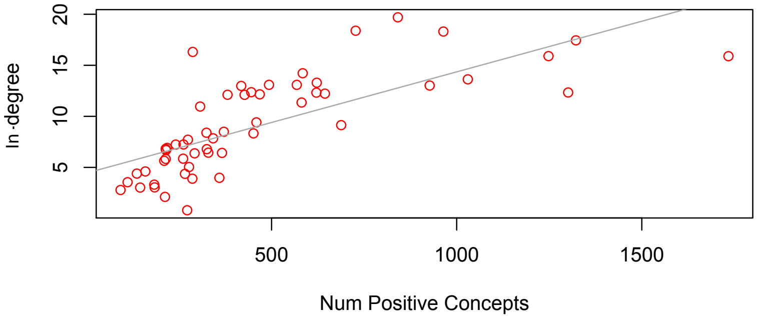


Figure 4.14: In-degree and positive concept ratio

Table 4.11: Top 5 topics of high transfer centrality

ID	Topic	Out-degree
63	Hugo awarded best novels	31
110	Nobel prize in literature winners who were also poets	30
133	EU countries	20
129	Science fiction book written in the 1980	19
117	Musicians who appeared in the Blues Brothers movies	18

Table 4.12: Top 5 topics benefit most from knowledge transfer

ID	Topic	Out-degree
110	Nobel prize in literature winners who were also poets	19
63	Hugo awarded best novels	18
125	Countries which have won the FIFA world cup	18
133	EU countries	17
87	Axis powers of World War II	16

Table 4.13: Correlation between topic difficulty (AAP) and performance.

Experiment	**AUC**	**F1**
J48 X10	0.3116*	0.2096
J48 Direct (src)	-0.0351	-0.0541*
J48 Direct (tar)	0.0009	-0.1316*
Logit X10	0.3486*	0.3119*
Logit Direct (src)	-0.0378*	-0.0347
Logit Direct (tar)	0.0656*	-0.0137

* indicates the significant level of 0.05

4.4 Task Difficulty and Performance

We compute the topic difficulty using the Average Average Precision (AAP) – the average AP of all submitted runs for given topic. An easier topic has a higher AAP. Table 4.13 shows the correlation between topic AAP and the performances for both 10-fold cross validation results and direct transfer. In general, the easier a topic is, the higher the performance for the 10-fold cross validation results. However, we do not observe the same effect for transfer learning. On the contrary, the more difficult a topic is (the lower AAP of a topic), the higher the transferability. As shown in Figure 4.15, topics that are more difficult (the lower AAP) are richer in concepts, allowing more opportunities to transfer the learner to other topics.

4.5 Summary

In this chapter, we developed knowledge transfer networks to study the problem of Wikipedia vandalism detection and the problem of entity search and classification.

In Section 4.2, we built a knowledge transfer network of different types of Wikipedia vandalism instances, investigating how the task similarity relates to transfer learning. The purpose of the knowledge transfer network is to reveal the connec-

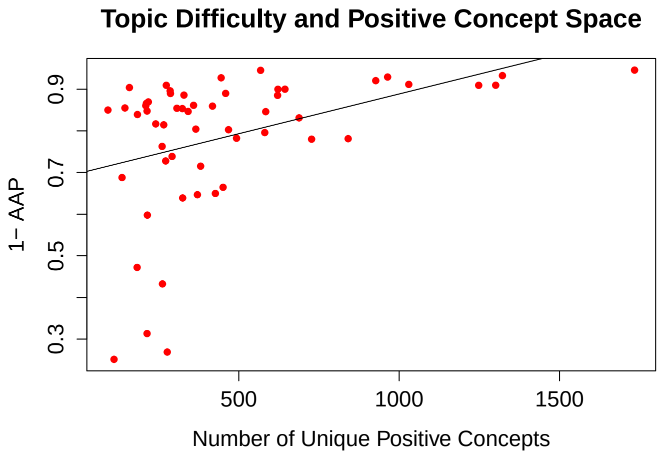


Figure 4.15: Topic difficulty and positive concept space

tions between learning problems from the perspective of transferability. The experimental results shows that similarity networks may not correspond to the transfer network. It requires experiments at a larger scale to discover which task characteristics influence the transferability and the centrality of a task in a knowledge transfer network.

In Section 4.3, we extended the analytic framework of knowledge transfer network to the problem of entity search and classification. Experiments with INEX-XER 2009 track dataset show that the similarity between tasks may not necessarily contribute to the knowledge transfer. The visualization of knowledge transfer networks indicates the following findings:

First, the choice of predictor affects the knowledge transfer relationship. Figure 4.11 and Figure 4.12 exhibit distinctly different linkages among the nodes. For example, a decision tree trained from Topic 63 can be transferred to seven topics in Figure 4.11. However, a logistic regression model trained from the same topic can only be transferred to one topic in Figure 4.11. Table 4.9 also shows the two networks vary in size and density. The results indicate that we would need to account for the selection of predictors in order to select appropriate source task.

Second, we observe that topics of higher centrality (transferability) are the topics with more positive documents and more positive concepts in the data set. In addition, topics that benefit more from knowledge transfer are the topics with more concept features. The results provide a heuristics to select appropriate source topics – topics that have higher ratio of positive documents in the dataset.

CHAPTER 5
CONCLUSIONS AND FUTURE WORK

5.1 Three Dimensions of Knowledge Transfer

The intuition of knowledge transfer comes from the experience that it is easier to learn Spanish after having learned French or that it is easier to learn freestyle ice skating after having learned ballet. In the context of machine learning, knowledge transfer is particularly useful when labelled data is absent or difficult to acquire. The discussion of how to efficiently use available information makes knowledge transfer valuable to information science studies. We position knowledge transfer in an interdisciplinary context of machine learning and information science. We investigate how to apply a machine learning method — transfer learning — to study problems in the field of information science.

In this thesis, we investigate three dimensions of knowledge transfer: what, how, and why. We present and elaborate on these questions: What are the transferable knowledge objects? How should we transfer knowledge across entities? Why do we observe a certain pattern of knowledge transfer? The goal of knowledge transfer is discovering explicit knowledge that can be effectively transferred between the source and the target tasks as well as balancing the goal to improve the performance of learning models.

To address the *what* dimension of knowledge transfer, we explore using Bag-of-Concepts (BoC) as the common feature space for the source and the target tasks. The

BoC provides informative features for the problem of entity search and classification. Building a decision tree with BoC features shows an explicit knowledge representation about a predictive problem. It allows us to analyze how knowledge transfer takes place between the source and the target task. However, as a transferable object, BoC still suffers from the curse of dimensionality, which compromises the experimental results of using BoC features to cluster related data samples.

To address the *how* dimension of the knowledge transfer, we introduce Segmented Transfer – a novel knowledge transfer model – to identify and learn from the most informative partitions from prior tasks. The proposed approach selects and learns from the most related portion in the source task to improve the predictive performance of the target task. Experimental results show improved performance using segmented transfer for the problem of Wikipedia vandalism detection. However, we do not observe the same performance improvements using segmented transfer for the problem of entity search and classification. A possible explanation is that the high dimensionality of BoC feature space has negative influences on the cluster mapping between the source and the target task. Future work will investigate the effect of feature dimensionality on knowledge transfer.

To address the *why* dimension of knowledge transfer, we propose the Knowledge Transfer Network (KTN), a novel type of network describing transfer learning relationships among problems. In a knowledge transfer network, a node is a learning problem (e.g. a topic in Entity Ranking); a directed link from a node A to a node B represents the flow of knowledge, that is, the potential of using the knowledge

Table 5.1: Summary of future work

	Proposed Strategy	**Intended Contribution**
What	Behavior-centric object (i.e. Kaplan-Meier survival curve)	New transferable object type
How	Force-directed Transfer	New transfer learning algorithms
Why	Structural hole discovery	Understanding the knowledge flow among learning problems

acquired from A to solve the problem for node B. Representing knowledge transfer relationships in a network allows us to apply network analysis on knowledge transfer networks to measure and analyze the centrality of a predictive problem. This novel type of network provides insights on identifying ontological connections that were initially obscured. For example, in our experiments, we observe that knowledge transfer can occur among dissimilar tasks.

5.2 Future Work

In this thesis, we adopt interdisciplinary perspectives, crossing the domains of machine learning and information science, to approach knowledge transfer. Our proposed methods aim to build a better predictive transfer learning model as well as to better understand the happening of knowledge transfer. Future studies will continue working on the dimensions of *what*, *how* and *why* of knowledge transfer, moving toward creating actionable knowledge that could help information holders understand a specific knowledge domain (e.g. clinical data analysis or search engine click log analysis).

In this section we outline three future work directions. Table 5.1 summarizes the intended contributions of the proposed future work.

5.2.1 Extract transferable features from user click patterns

This strategy aims to address the problem of *what* can be extracted from a click log as a transferable object between the target and source task.

Click logs provides a rich resource about user satisfaction with their search results. Click patterns often indicate relevance information for queries. Research has attempted to model and characterize user behaviors using click logs. In [17], we apply the Kaplan-Meier estimator to study click patterns. The visualization of click curves demonstrates the interaction between the relevance and the rank position of URLs. The observed results demonstrate the potential of using click curves to predict the quality of the top-ranked results.

The survival functions can be a transferable object between the source and the target task. Future work may explore methods to transfer a survival curve or function to enhance the quality of ranked search results. For example, for queries that occur less frequent, we may transfer from frequent queries about the knowledge of the clicking pattern and search result quality. We may also transfer the clicking pattern knowledge from one region to another region.

5.2.2 A dynamic framework for transfer learning

Future work may investigate *force-directed transfer* approach to exploit knowledge from less related source tasks. Transfer learning is inspired by the notion of “following advice from people similar to you” or “learning from good examples.” However, it is also intriguing to explore the notion of “how to handle knowledge from people dissimilar to you" and "how can we learn from bad examples." In order to maximize the use of available information, this report explores methods to exploit not only "related" source tasks but also "unrelated" or "less relevant" tasks. For example, to solve a multiple choice problem, one can determine an answer either by knowing the correct choice or by eliminating the choices that are unlikely to be true. This strategy, inspired by force-directed algorithm [34], aims to answer the question of "how to incorporate the related domain (the positives) as the attraction force and the unrelated domain (the negatives) as the repulsion force in a knowledge transfer framework to enhance learning performance?" Current transfer learning research investigated approaches to identify and learn from good guidance. However, learning to identify and proactively reject negative influences may also be valuable. Although research in the area has recognized the effect of negative transfer [89], no algorithm attempted to exploit the information in unrelated (or negatively related) source domain. Therefore, this strategy suggests a transfer learning framework that incorporates both the "good guidance"and "bad influences" to maximize the value of available information.

5.2.3 Examine structural holes property

Section 2.4 demonstrates the utility and the construction of a wide variety of networks. To approach the *why* dimension of the three stated problems, this strategy suggests constructing ego-centric knowledge networks. A knowledge transfer network views the transfer learning relationship as a directed graph, where the nodes represent learning problems, and the directed edges between node pairs represent knowledge flow between problems. In a knowledge network, a directed link from a node A to a node B represents the flow of knowledge, that is, the potential of using the knowledge acquired from A to solve the problem for node B. An ego-centric knowledge network uses the target query as the focal node (i.e. the "ego") to observe and analyze the structure of the network.

Structural holes [4] describes a network property that measures the number of non-redundant ties to indicate a brokering position in a network. An ego network with a lot of structural holes indicates that the focal node has advantages over the flow of information. In the setting of a knowledge transfer network, structural holes can indicate whether a task contains indispensable information and knowledge in the network. This proposed strategy aims to answer the question of *how to identify the data samples or learning tasks of higher transferability? What are their characteristics?*

REFERENCES

[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. *The Semantic Web*, page 722–735, 2007.

[2] Lars Backstrom, Dan Huttenlocher, Jon Kleinberg, and Xiangyang Lan. Group formation in large social networks: membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, page 44–54, New York, NY, USA, 2006. ACM. ACM ID: 1150412.

[3] Ulrik Brandes, Patrick Kenis, Jürgen Lerner, and Denise van Raaij. Network analysis of collaboration structure in wikipedia. In *Proceedings of the 18th international conference on World wide web*, WWW '09, page 731–740, New York, NY, USA, 2009. ACM. ACM ID: 1526808.

[4] Ronald S Burt. *Structural Holes: The Social Structure of Competition*, volume 58. Harvard University Press, 1992.

[5] Jan Buzydlowski, Howard White, and Xia Lin. Term co-occurrence analysis as an interface for digital libraries. In Katy Börner and Chaomei Chen, editors, *Visual Interfaces to Digital Libraries*, volume 2539 of *Lecture Notes in Computer Science*, pages 133–144. Springer Berlin / Heidelberg, 2002. 10.1007/3-540-36222-3_10.

[6] Pedro Henrique Calais Guerra, Adriano Veloso, Wagner Meira, and Virglio Almeida. From bias to opinion. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*, page 150, San Diego, California, USA, 2011.

[7] Bin Cao, Nathan Nan Liu, and Qiang Yang. Transfer learning for collective link prediction in multiple heterogenous domains. Haifa, Israel, June 2010.

[8] Andrea Capocci and Guido Caldarelli. Folksonomies and clustering in the collaborative system citeulike. *Journal of Physics A: Mathematical and Theoretical*, 41(22):224016, June 2008.

[9] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: Web spam detection using the web topology. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, page 430. ACM, 2007.

[10] Ciro Cattuto, Christoph Schmitz, Andrea Baldassarri, Vito D.P. Servedio, Vittorio Loreto, Andreas Hotho, Miranda Grahl, and Gerd Stumme. Network properties of folksonomies. *AI Communications*, 20(4):245–262, January 2007.

[11] Soumen Chakrabarti, Byron Dom, and Piotr Indyk. Enhanced hypertext categorization using hyperlinks. *SIGMOD Rec.*, 27(2):307–318, 1998.

[12] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:127:27, May 2011. ACM ID: 1961199.

[13] Hao Chen and Burt Sharp. Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics*, 5(1):147, 2004.

[14] S.-C. Chin, W. N. Street, P. Srinivasan, and D. Eichmann. Detecting Wikipedia vandalism with active learning and statistical language models. In *Proceedings of the 4th Workshop on Information Credibility*, WICOW '10, page 3–10, 2010.

[15] Si-Chi Chin and W. Nick Street. Divide and transfer: an exploration of unsupervised knowledge transfer. In *Proc. The 28th Int. Conf. on Machine Learning (ICML-2011): Workshop on on Unsupervised and Transfer Learning*, 2011.

[16] Si-Chi Chin and W. Nick Street. Enhanced wikipedia vandalism taxonomy via subclass discovery. In *Proc. Twenty-second Int. Joint Conf. on Artificial Intelligence (IJCAI-11): Workshop on Discovering Meaning On the Go in Large Heterogeneous Data*, 2011.

[17] Si-Chi Chin and W. Nick Street. Survival analysis of click logs. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, page 11491150, New York, NY, USA, 2012. ACM.

[18] Nicholas A. Christakis and James H. Fowler. The spread of obesity in a large social network over 32 years. *New England Journal of Medicine*, 357(4):370–379, July 2007.

[19] Nicholas A Christakis and James H Fowler. Social network sensors for early detection of contagious outbreaks. *1004.4792*, April 2010.

[20] Philip Clarkson and Ronald Rosenfeld. Statistical language modeling using the CMU-Cambridge toolkit. In *Fifth European Conference on Speech Communication and Technology*, pages 2707—2710, September 1997.

[21] David Crandall, Dan Cosley, Daniel Huttenlocher, Jon Kleinberg, and Siddharth Suri. Feedback effects between similarity and social influence in online communities. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, page 160–168, New York, NY, USA, 2008. ACM. ACM ID: 1401914.

[22] Wenyuan Dai, Ou Jin, Gui-Rong Xue, Qiang Yang, and Yong Yu. EigenTransfer: a unified framework for transfer learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 193200, New York, NY, USA, 2009. ACM.

[23] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, page 193200, New York, NY, USA, 2007. ACM.

[24] Hal Daum III. Frustratingly easy domain adaptation. *arXiv:0907.1815*, July 2009. ACL 2007.

[25] Gianluca Demartini, Tereza Iofciu, and Arjen de Vries. Overview of the INEX 2009 entity ranking track. In Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors, *Focused Retrieval and Evaluation*, volume 6203 of *Lecture Notes in Computer Science*, pages 254–264. Springer Berlin / Heidelberg, 2010.

[26] Ying Ding. Scientific collaboration and endorsement: Network analysis of coauthorship and citation networks. *Journal of Informetrics*, 5(1):187–203, January 2011.

[27] E. Eaton and M. desJardins. Selective transfer between learning tasks using task-based boosting. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[28] P. Edwards, G. A.A Grimnes, and A. Preece. An empirical investigation of learning from the semantic web. *Semantic Web Mining*, page 71, 2002.

[29] Yasser EL-Manzalawy and Vasant Honavar. WLSVM: Integrating LibSVM into Weka environment, 2005. Software available at http://www.cs.iastate.edu/ yasser/wlsvm.

[30] G. Forman. Tackling concept drift by temporal inductive transfer. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, page 252–259, 2006.

[31] James H Fowler and Nicholas A Christakis. Dynamic spread of happiness in a large social network: longitudinal analysis over 20 years in the framingham heart study. *BMJ (Clinical Research Ed.)*, 337:a2338, 2008. PMID: 19056788.

[32] Richard H Fowler, Wendy A. L Fowler, and Bradley A Wilson. Integrating query thesaurus, and documents through a common visual representation. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '91, page 142–151, New York, NY, USA, 1991. ACM. ACM ID: 122874.

[33] Noah E. Friedkin. *A Structural Theory of Social Influence*. Cambridge University Press, September 1998.

[34] T.M.J. Fruchterman and E.M. Reingold. Graph drawing by force-directed placement. *Software: Practice and experience*, 21(11):1129–1164, 1991.

[35] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, page 6–12, 2007.

[36] Sheng Gao and Haizhou Li. A cross-domain adaptation method for sentiment classification using probabilistic latent analysis. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, page 10471052, New York, NY, USA, 2011. ACM.

[37] E. Garfield. The use of citation data in writing the history of science. Technical report, Philadelphia: Institute for Scientific Information, 1964.

[38] Eugene Garfield. Citation analysis as a tool in journal evaluation. *Science*, 178:471–479, November 1972.

[39] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the Twenty-eight International Conference on Machine Learning, ICML*, 2011.

[40] Anna Goldenberg and Andrew W Moore. Bayes net graphs to understand co-authorship networks? In *Proceedings of the 3rd international workshop on Link discovery*, LinkKDD '05, page 1–8, New York, NY, USA, 2005. ACM. ACM ID: 1134272.

[41] Bronwyn H. Hall, Adam B. Jaffe, and Manuel Trajtenberg. The NBER patent citation data file: Lessons, insights and methodological tools. Working Paper 8498, National Bureau of Economic Research, October 2001.

[42] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: An update. *SIGKDD Explor. Newsl.*, 11:10–18, November 2009.

[43] Mandar Haridas and Doina Caragea. Exploring wikipedia and DMoz as knowledge bases for engineering a user interests hierarchy for social network applications. In Robert Meersman, Tharam Dillon, and Pilar Herrero, editors, *On the Move to Meaningful Internet Systems: OTM 2009*, volume 5871 of *Lecture Notes in Computer Science*, pages 1238–1245. Springer Berlin / Heidelberg, 2009. 10.1007/978-3-642-05151-7_35.

[44] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):553, January 2004.

[45] A. Hotho, R. J\äschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. *The Semantic Web: Research and Applications*, page 411–426, 2006.

[46] Derek Hao Hu, Vincent Wenchen Zheng, and Qiang Yang. Cross-domain activity recognition via transfer learning. *Pervasive and Mobile Computing*, 7(3):344–358, June 2011.

[47] Norman P. Hummon and Patrick Dereian. Connectivity in a citation network: The development of DNA theory. *Social Networks*, 11(1):39–63, March 1989.

[48] Toshihiro Kamishima, Masahiro Hamasaki, and Shotaro Akaho. TrBagg: a simple transfer learning method and its application to personalization in collaborative tagging. In *Data Mining, IEEE International Conference on*, volume 0, pages 219–228, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[49] Aniket Kittur, Bongwon Suh, Bryan A. Pendleton, and Ed H. Chi. He says, she says: Conflict and coordination in Wikipedia. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 453–462, San Jose, California, USA, 2007. ACM.

[50] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[51] Jon M Kleinberg. Challenges in mining social network data: processes, privacy, and paradoxes. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, page 4–5, New York, NY, USA, 2007. ACM. ACM ID: 1281195.

[52] B. Klimt and Y. Yang. Introducing the enron corpus. In *First conference on email and anti-spam (CEAS)*, 2004.

[53] Jure Leskovec, Lada A Adamic, and Bernardo A Huberman. The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, 1, May 2007. ACM ID: 1232727.

[54] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, page 497–506, New York, NY, USA, 2009. ACM. ACM ID: 1557077.

[55] Jure Leskovec, Mary McGlohon, Christos Faloutsos, Natalie Glance, and Matthew Hurst. Cascading behavior in large blog graphs. *0704.2803*, April 2007.

[56] B. Li, Q. Yang, and X. Xue. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *Proceedings of the 21st international jont conference on Artifical intelligence*, page 20522057, 2009.

[57] Bin Li, Qiang Yang, and Xiangyang Xue. Transfer learning for collaborative filtering via a rating-matrix generative model. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 617624, New York, NY, USA, 2009. ACM.

[58] Shoushan Li and Chengqing Zong. Multi-domain sentiment classification. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, HLT-Short '08, page 257260, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[59] Tao Li, Yi Zhang, and Vikas Sindhwani. A non-negative matrix tri-factorization approach to sentiment classification with lexical prior knowledge. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, page 244252, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[60] Xin Li, Hsinchun Chen, Zhu Zhang, and Jiexun Li. Automatic patent classification using citation network information: an experimental study in nanotechnology. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '07, page 419–427, New York, NY, USA, 2007. ACM. ACM ID: 1255262.

[61] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58(7):1019–1031, May 2007.

[62] B. Malin. Unsupervised name disambiguation via social network similarity. In *Workshop on Link Analysis, Counterterrorism, and Security*, pages 93–102, Newport Beach, CA, 2005.

[63] A. McCallum and K. Nigam. Employing EM in pool-based active learning for text classification. In *Proceedings of ICML-98, 15th International Conference on Machine Learning*, pages 350–358, 1998.

[64] A. McCallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research*, 30:249–272, 2007.

[65] M. D McClintock. The declining use of legal scholarship by courts: An empirical study. *Okla. L. Rev.*, 51:659–727, 1998.

[66] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, August 2001.

[67] George A Miller. WordNet: a lexical database for english. *Communications of the ACM*, 38:39–41, November 1995. ACM ID: 219748.

[68] Einat Minkov, William W Cohen, and Andrew Y Ng. Contextual search and name disambiguation in email using graphs. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, page 27–34, New York, NY, USA, 2006. ACM. ACM ID: 1148179.

[69] Gilad Mishne, David Carmel, and Ronny Lempel. Blocking blog spam with language model disagreement. In *1st International Workshop on Adversarial Information Retrieval on the Web*, 2005.

[70] Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, page 29–42, New York, NY, USA, 2007. ACM. ACM ID: 1298311.

[71] M. E.J Newman. The structure and function of complex networks. *Arxiv preprint cond-mat/0303516*, 2003.

[72] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. 1999.

[73] Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, WWW '10, page 751760, New York, NY, USA, 2010. ACM.

[74] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[75] W. Pan, N.N. Liu, E.W. Xiang, and Q. Yang. Transfer learning to predict missing ratings via heterogeneous user feedbacks. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[76] W. Pan, E.W. Xiang, N. Liu, and Q. Yang. Transfer learning in collaborative filtering for sparsity reduction. In *Proceedings of the 24rd AAAI Conference on Artificial Intelligence*, 2010.

[77] Weike Pan, Erheng Zhong, and Qiang Yang. Transfer learning for text mining. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, pages 223–257. Springer US, 2012.

[78] Gerold Pedemonte and May Lim. Tracking the dynamic variations in a social network formed through shared interests. *Science Diliman*, 21(1), March 2010.

[79] Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks (long version). *arXiv:physics/0512106*, December 2005.

[80] S. P Ponzetto and M. Strube. Deriving a large scale taxonomy from wikipedia. In *Proceedings of the national conference on artificial intelligence*, volume 22, page 1440, 2007.

[81] M. Potthast and R. Gerling. *Wikipedia Vandalism Corpus Webis-WVC-07*. 2007.

[82] Martin Potthast, Benno Stein, and Robert Gerling. Automatic vandalism detection in Wikipedia. In *Advances in Information Retrieval*, pages 663–668. Springer Berlin / Heidelberg, 2008.

[83] D. J Price. Networks of scientific papers. *Science (New York, NY)*, 149:510, 1965.

[84] Reid Priedhorsky, Jilin Chen, Shyong (Tony) K. Lam, Katherine Panciera, Loren Terveen, and John Riedl. Creating, destroying, and restoring value in Wikipedia. In *Proceedings of the International ACM Conference on Supporting Group Work*, pages 259–268, Sanibel Island, Florida, USA, 2007. ACM.

[85] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, pages 759–766, New York, NY, USA, 2007. ACM.

[86] Parisa Rashidi and Diane J. Cook. Activity knowledge transfer in smart environments. *Pervasive and Mobile Computing*, 7(3):331–343, June 2011.

[87] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. *The SemanticWeb-ISWC 2003*, page 351–368, 2003.

[88] R. Rosenfeld. Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, 88(8):1270–1278, 2000.

[89] M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich. To transfer or not to transfer. In *NIPS05 Workshop, Inductive Transfer: 10 Years Later*, 2005.

[90] Purnamrita Sarkar and Andrew W Moore. Dynamic social network analysis using latent space models. *ACM SIGKDD Explorations Newsletter*, 7:31–40, December 2005. ACM ID: 1117459.

[91] Ralf Schenkel, Fabian Suchanek, and Gjergji Kasneci. YAWN: a semantically annotated wikipedia XML corpus. In *Technology and the Web of the German Socienty for Computer science, BTW 2007*, pages 277–291, 2007.

[92] Bernhard Schlkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a High-Dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.

[93] D. L. Silver and R. E. Mercer. *Selective transfer of neural network task knowledge*. PhD thesis, 2000.

[94] K. Smets, B. Goethals, and B. Verdonk. Automatic vandalism detection in Wikipedia: Towards a machine learning approach. In *AAAI Workshop on Wikipedia and Artificial Intelligence: An Evolving Synergy*, pages 43–48, 2008.

[95] Avar Stewart, Matthew Smith, and Wolfgang Nejdl. A transfer approach to detecting disease reporting events in blog social media. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia*, HT '11, page 271280, New York, NY, USA, 2011. ACM.

[96] M. Strube and S. P Ponzetto. WikiRelate! computing semantic relatedness using wikipedia. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, page 1419, 2006.

[97] Michael Sussna. Word sense disambiguation for free-text indexing using a massive semantic network. In *Proceedings of the second international conference on Information and knowledge management*, CIKM '93, page 67–74, New York, NY, USA, 1993. ACM. ACM ID: 170106.

[98] Z. Syed, T. Finin, and A. Joshi. Wikipedia as an ontology for describing documents. In *Proceedings of the Second International Conference on Weblogs and Social Media*, 2008.

[99] Idan Szpektor, Aristides Gionis, and Yoelle Maarek. Improving recommendation for long-tail queries via templates. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 47–56, New York, NY, USA, 2011. ACM.

[100] Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *The Journal of Machine Learning Research*, 10:1633–1685, December 2009. ACM ID: 1755839.

[101] L. Torrey and J. Shavlik. Transfer learning. *Handbook of Research on Machine Learning Applications. IGI Global*, 3:17–35, 2009.

[102] Vishvas Vasuki, Nagarajan Natarajan, Zhengdong Lu, Berkant Savas, and Inderjit Dhillon. Scalable affiliation recommendation using auxiliary networks. *ACM Trans. Intell. Syst. Technol.*, 3(1):3:13:20, October 2011.

[103] Fernanda B. Viégas, Martin Wattenberg, and Kushal Dave. Studying cooperation and conflict between authors with history flow visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 575–582, Vienna, Austria, 2004. ACM.

[104] Xiaojun Wan. Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, page 235243, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[105] Hua Wang, Heng Huang, Feiping Nie, and Chris Ding. Cross-language web page classification via dual knowledge transfer using nonnegative matrix tri-factorization. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, page 933942, New York, NY, USA, 2011. ACM.

[106] Xuerui Wang, Natasha Mohanty, and Andrew McCallum. Group and topic discovery from relations and text. In *Proceedings of the 3rd international workshop on Link discovery*, pages 28–35, Chicago, Illinois, 2005. ACM.

[107] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, January 1995.

[108] Andrew G West, Sampath Kannan, and Insup Lee. Detecting Wikipedia vandalism via spatio-temporal analysis of revision metadata. In *Proceedings of the Third European Workshop on System Security*, EUROSEC '10, page 2228, New York, NY, USA, 2010. ACM. ACM ID: 1752050.

[109] Rong Yan and Jian Zhang. Transfer learning using task-level features with application to information retrieval. In *Proceedings of the 21st international jont conference on Artifical intelligence*, IJCAI'09, page 13151320, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[110] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive SVMs. In *Proceedings of the 15th International Conference on Multimedia*, MULTIMEDIA '07, page 188–197, New York, NY, USA, 2007.

[111] Qiang Yang, Yuqiang Chen, Gui-Rong Xue, Wenyuan Dai, and Yong Yu. Heterogeneous transfer learning for image clustering via the social web. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, page 19, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[112] Tianbao Yang, Rong Jin, Anil K. Jain, Yang Zhou, and Wei Tong. Unsupervised transfer classification: application to text categorization. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, page 11591168, New York, NY, USA, 2010. ACM.

[113] Weilong Yang, Yang Wang, and Greg Mori. Learning transferable distance functions forhuman action recognition. In Liang Wang, Guoying Zhao, Li Cheng, and Matti Pietikinen, editors, *Machine Learning for Vision-Based Motion Analysis*, pages 349–370. Springer London, London, 2011.

[114] Shinjae Yoo, Yiming Yang, Frank Lin, and Il-Chul Moon. Mining social networks for personalized email prioritization. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 967–976, Paris, France, 2009. ACM.

[115] T. Zesch and I. Gurevych. Analysis of the wikipedia category graph for NLP applications. *TextGraphs-2: Graph-Based Algorithms for Natural Language Processing*, page 1, 2007.

[116] Y. Zhang, W.N. Street, and S. Burer. Sharing classifiers among ensembles from related problem domains. In *Fifth IEEE International Conference on Data Mining*, pages 522–529. IEEE Computer Society, 2005.

[117] Ding Zhou, Xiang Ji, Hongyuan Zha, and C. Lee Giles. Topic evolution and social interactions: how authors effect research. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 248–257, Arlington, Virginia, USA, 2006. ACM.

[118] Ning Zhou, Jinye Peng, Xiaoyi Feng, and Jianping Fan. Towards more precise social Image-Tag alignment. In Kuo-Tien Lee, Wen-Hsiang Tsai, Hong-Yuan Liao, Tsuhan Chen, Jun-Wei Hsieh, and Chien-Cheng Tseng, editors, *Advances in Multimedia Modeling*, volume 6524 of *Lecture Notes in Computer Science*, pages 46–56. Springer Berlin / Heidelberg, 2011. 10.1007/978-3-642-17829-0_5.