A Universal Topic Framework (UniZ) and Its Application in Online Search

Youngchul Cha Keng-hao Chang
Hari Bommaganti Ye Chen Tak Yan
Microsoft, Sunnyvale, CA 94089
{youcha, kenchan, hariprab, yec, takyan}@microsoft.com

Bin Bi
Junghoo Cho
UCLA Computer Science Dept
Los Angeles, CA 90095
{bbi, cho}@cs.ucla.edu

ABSTRACT

Probabilistic topic models, such as PLSA and LDA, are gaining popularity in many fields due to their high-quality results. Unfortunately, existing topic models suffer from two drawbacks: (1) model complexity and (2) disjoint topic groups. That is, when a topic model involves multiple entities (such as authors, papers, conferences, and institutions) and they are connected through multiple relationships, the model becomes too difficult to analyze and often leads to intractable solutions. Also, different entity types are classified into *disjoint* topic groups that are not directly comparable, so it is difficult to see whether heterogeneous entities (such as authors and conferences) are on the same topic or not (e.g., are Rakesh Agrawal and KDD related to the same topic?).

In this paper, we propose a novel universal topic framework (UniZ) that addresses these two drawbacks using "prior topic incorporation." Since our framework enables representation of heterogeneous entities in a *single* universal topic space, all entities can be directly compared within the *same* topic space. In addition, UniZ breaks complex models into much smaller units, learns the topic group of each entity from the smaller units, and then *propagates* the learned topics to others. This way, it leverages all the available signals without introducing significant computational complexity, enabling a richer representation of entities and highly accurate results.

In a widely-used DBLP dataset prediction problem, our approach achieves the best prediction performance over many state-of-the-art methods. We also demonstrate practical potential of our approach with search logs from a commercial search engine.

Categories and Subject Descriptors

H.2.8 [**Information Systems Applications**]: Database Applications—*Data mining*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Clustering*

SAC'15, April 13–17, 2015, Salamanca, Spain
Copyright 2015 ACM 978-1-4503-3196-8/15/04 ...$15.00.
http://dx.doi.org/10.1145/2695664.2695806

General Terms

Experimentation, Performance

Keywords

topic models, universal topic framework, online search, users and context-modeling

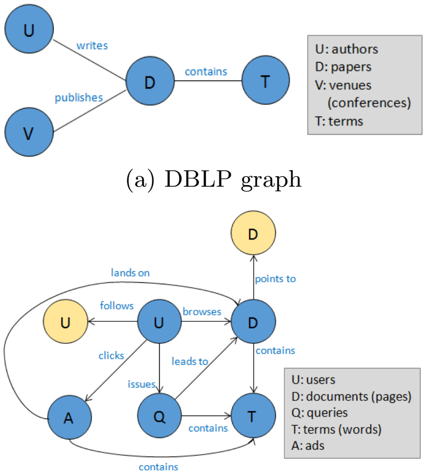
1. INTRODUCTION

The problem with topic modeling is that it is usually limited to a few entity types and not good at handling multiple entity types, which is quite common in real-world applications. For example, consider a topic model for a web graph depicted in Figure 1(b). This model tries to capture web users (U) who issue queries (Q) that contain multiple terms (T), visit relevant web documents (D), and click ads (A). Also, the web users follow other web users, and the web documents have links to other web documents. Unfortunately, a topic model of this complexity is often intractable for analysis. To address this complexity, we may simply decompose the model into multiple segments where each segment contains a subset of entity types, and analyze each segment separately with standard topic models (such as Probabilistic Latent Semantic Analysis (PLSA) [12] and Latent Dirichlet Allocation (LDA) [3]). For example, in Figure 1(b), we may apply LDA to the follow edges between the two U nodes and obtain the topic groups of U. Similarly, we can apply LDA to other edges (such as the click edges between U and A) to obtain estimate of each node's topic groups.

The problem with this approach is that the learned topics from each segment are not directly comparable. That is, the topic No. 1 obtained from the follow edges is totally different from the topic No. 1 obtained from the click edges. In fact, there is no guarantee that topic groups obtained from two LDA applications will be comparable. In principle, when there are N segments, there are N different topic spaces that are completely independent of each other.

In this paper, we propose a novel universal topic framework which enables representation of heterogeneous entities in a *single universal topic space*. Our approach is based on an assumption that there is a hidden interest (topic) for every relationship (edge) between two entities. Based on this assumption, we extend the *follow-edge generative model* [5] developed for social graph mining, and apply topic modeling to any type of edges between any type of entities.

To analyze arbitrarily complex topic models, we take an *incremental approach*, where we decompose a model into smaller segments, learn topic groups from one segment, and



(a) DBLP graph

(b) Simple Web graph

Figure 1: Examples of complex graphs

propagate the learned topic groups to other segments. We make it possible to *incorporate* and *combine* the topic groups from distinct segments using an approach called "prior topic incorporation". More precisely, we propose two extensions to a simple topic model, *mixture model* and *dual-prior model*, which can effectively incorporate topics learned from the entities and relationships from other segments to the current segment. This prior topic incorporation enables the topics to be coherent across all entities and relationships in a complex graph. By representing all the entities in the universal topic space, our framework provides the following benefits: (1) direct topical similarity comparison between any heterogeneous entities (e.g., using simple *cosine similarity*), (2) richer representations of entities by leveraging all available signals (e.g., representing a user by both users she follows and movies she watches), and (3) prediction/recommendation performance improvements. Although the incremental approach was initially introduced in [9], where authors first learn topics with PLSA and propagate the learned topics with Expectation Maximization (EM), our approach incorporates previously learned topics more seamlessly within the LDA framework and produces improved performance.

We evaluate the effectiveness of our approach with many state-of-the-art methods using the widely-used Digital Bibliography & Library Project (DBLP) [1] dataset. We also demonstrate a practical potential application of our approach with search logs collected from a commercial search engine, Bing [2].

2. UNIVERSAL TOPIC FRAMEWORK

We propose a novel universal topic framework called "UniZ". Rather than building a complex generative model for a complex graph, which easily becomes intractable as the number of entity types increases, we take a "divide and conquer" strategy and decompose the graph into multiple segments so that we can apply LDA to each segment. However, a simple divide and conquer analysis generates incomparable topic groups, where the topic No. 1 in one segment (e.g. on *music*) is totally different from the topic No. 1 in another segment (e.g. on *politics*). In this section, we propose effective methods to overcome the topic mismatch problem with this simple divide and conquer strategy. We also discuss some major issues in our proposed methods. Before that, we first justify how we can apply topic models to any types of edges (not limited to textual edges or social follow edges) so that we can freely divide the complex graph.

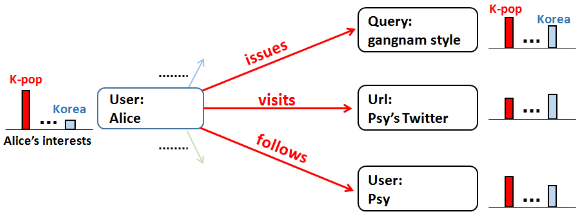


Figure 2: Example of edge labeling

2.1 Edge Generative Model

We extend the follow-edge generative model [5] to different types of edges (including follow edges) between different entities. The assumption behind our approach is that every edge (relationship) is generated because of a hidden interest between two entities regardless of their types [3]. For example, let's say Alice is interested in *K-pop* (Korean pop). She recently heard about a song *Gangnam Style* from her friend and types a query *gangnam style* in a search engine to get more information about the song. From a search results page, she finds out the singer is *Psy* and clicks *Psy*'s Twitter[4] page and follows him in Twitter. Now, we can represent her actions in a single graph as in Figure 2 having three different types of relationships (issues, visits, and follows) among four entities of three different types. The color bars next to each entity denote an interest distribution of that entity. In this example, she took all these actions (not only *following*) because she is interested in *K-pop*. If we know the color (interest or topic) distribution of each entity [5], we may probabilistically label each edge with an appropriate color (in this example *red*, which denotes *K-pop*). Also, the color distribution of each entity is determined by counting the number of colored edges attached to the entity and is updated after coloring newly attached edges. If we continue this cyclic process, we can label all the edges in a graph with appropriate colors in a single palette (i.e., a single topic space). In this way, heterogeneous edges and entities can be represented in a single universal topic space.

Now, we formalize our approach using the framework of LDA [3]. For simplicity, when there is an edge, we denote the starting entity as a reader and the ending entity as a writer as in the follow-edge generative model. When reader r generates an edge to writer w, she first picks interest z (topic) from a distribution $p(z|r)(\theta)$, and then picks a writer from a distribution $p(w|z)(\phi)$. This process is the same with the term selection process for a document in LDA. Thus, we

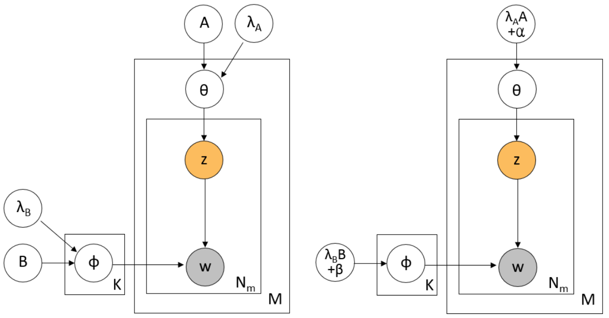
[1] http://www.informatik.uni-trier.de/ ley/db/

[2] http://www.bing.com/

[3] This assumption holds only for an edge whose two connected entities are relevant each other (e.g., causal relationship, containing relationship, following relationship, etc). If an edge is randomly generated or generated by a spammer (to every other entities), this assumption does not hold.

[4] http://www.twitter.com/

[5] More precisely, its importance in an interest group as well.



(a) Mixture model (b) Dual-prior model

Figure 3: Proposed topic models

formulate the probability of a specific reader a to follow a specific writer b based on a certain interest z (or $z_{a,b}$) given all the other conditions as follows:

$$p(z_{a,b}|\cdot) = p(z|r_a)p(w_{a,b}|z), \tag{1}$$

where $\cdot$ denotes values of all the other random variables. We use $w_{a,b}$ to indicate an edge from the reader a to the writer b. By considering Dirichlet priors α and β, constraining θ and ϕ, Equation (1) can be represented as follows:

$$p(z_{a,b}|\cdot) \propto \int p(z|\theta)p(\theta|\alpha)d\theta \times \int p(w_{a,b}|z,\phi)p(\phi|\beta)d\phi. \tag{2}$$

This formula leads to a collapsed Gibbs sampling equation [17]:

$$p(z_{a,b}|\cdot) \propto \frac{R^{-(a,b)}_{a,z_{a,b}} + \alpha_{z_{a,b}}}{R^{-(a,b)}_{a,*} + \alpha_*} \times \frac{W^{-(a,b)}_{w_{a,b},z_{a,b}} + \beta_{w_{a,b}}}{W^{-(a,b)}_{*,z_{a,b}} + \beta_*}, \tag{3}$$

where R denotes an association count matrix between readers and topics and W denotes an association count matrix between writers and topics. The R and W has readers and writers in its rows respectively, and topics in its columns. Thus, $R_{a,z_{a,b}}$ denotes the element of the row a and the column $z_{a,b}$, which is the number of associations between the reader a and the topic $z_{a,b}$. The superscript $-(a,b)$ means that the number does not include the topic assigned to the edge from the reader a to the writer b. We use the symbol $*$ to denote a summation over all possible subscript variables. For example, $\alpha_* = \sum_{z'} \alpha_{z'}$ and $W_{*,z_{a,b}} = \sum_{w'} W_{w',z_{a,b}}$.

In a Gibbs-sampling-based inference, as the number of iterations increases, each R and W gets a better estimate of a joint probability distribution of reader and topic, $p(r,z)$, and a joint probability distribution of writer and topic, $p(w,z)$, respectively. In a sizable number of iterations, R is used to get $p(z|r)$ after being normalized by row with a prior α, and W is used to get $p(w|z)$ after being normalized by column with a prior β, as noted in Equation (3). Thus, we may say that the matrices (R and W) contain topics learned from the Gibbs sampling inference process.

2.2 Incorporating Learned Topics

As explained in the previous section, the learned topics from Gibbs-sampling-based LDA are stored in the reader-topic association count matrix R, and the writer-topic association count matrix W. When we divide a complex graph into multiple segments, we get a pair of R and W per segment. However, each pair is totally different from each other and there is no easy way to combine these multiple pairs. Thus, we need an "incremental" approach of learning initial topics from one segment and incorporate these "old topics" when we learn "new topics" for another segment. We call this approach "prior topic incorporation" (or simply "topic incorporation") and propose two effective topic incorporation models: a *mixture model* and a *dual-prior model*.

2.2.1 Mixture Model

Since we want the new topics to be coherent to the old topics, one possible way of incorporating the old topics is to use a *mixture representation* of new topics and old topics. We modify the *polya-urn model* [1] to pick a topic or a writer from a linear combination of the old and new topic distributions. The original *polya-urn model* was developed to linearly combine a local and a global topic distribution. Now, the Gibbs sampling equation for this approach becomes:

$$\begin{aligned} p(z_{a,b}|\cdot) \propto & \left(\frac{R^{-(a,b)}_{a,z_{a,b}}}{R^{-(a,b)}_{a,*}} + \lambda_A \frac{A_{a,z_{a,b}} + \alpha_{z_{a,b}}}{A_{a,*} + \alpha_*}\right) \\ & \times \left(\frac{W^{-(a,b)}_{w_{a,b},z_{a,b}}}{W^{-(a,b)}_{*,z_{a,b}}} + \lambda_B \frac{B_{w_{a,b},z_{a,b}} + \beta_{w_{a,b}}}{B_{*,z_{a,b}} + \beta_*}\right), \end{aligned} \tag{4}$$

where A/B is the old reader/writer-topic association count matrix (previously learned), and R/W is the new reader/writer-topic association count matrix (to be learned), respectively. There are two scalar weights λ_A and λ_B, which are used as concentration parameters. If λ is high, the new topic distribution becomes similar to the old one. If λ goes to zero, it degenerates to LDA. Figure 3(a) depicts a plate notation of this model. We call this the *mixture model*.

2.2.2 Dual-Prior Model

Another way of incorporating previously learned topics is to use them as "priors". Consider a topic count matrix B, which is learned from relationships between papers and terms in Figure 1(a) and has papers in its rows and topics in its columns. If B_{p_1,z_1} is relatively higher than other values in the row p_1 of the matrix, it suggests that paper p_1 is very likely about topic z_1 (e.g., *data mining*). We can leverage this learned information when we infer topics for relationships between authors and papers. If author a_1 wrote the paper p_1, we can infer that the author a_1 is interested in z_1 (*data mining*). In this way, topics learned from one type of relationships can be used as priors when we infer topics for a different type of relationships. One benefit of LDA is that we can seamlessly incorporate the priors in its equation. Since α and β in Equation (3) are priors, we can extend the equation to:

$$\begin{aligned} p(z_{a,b}|\cdot) \propto & \frac{R^{-(a,b)}_{a,z_{a,b}} + \lambda_A \cdot A_{a,z_{a,b}} + \alpha_{z_{a,b}}}{R^{-(a,b)}_{a,*} + \lambda_A \cdot A_{a,*} + \alpha_*} \\ & \times \frac{W^{-(a,b)}_{w_{a,b},z_{a,b}} + \lambda_B \cdot B_{w_{a,b},z_{a,b}} + \beta_{w_{a,b}}}{W^{-(a,b)}_{*,z_{a,b}} + \lambda_B \cdot B_{*,z_{a,b}} + \beta_*}, \end{aligned} \tag{5}$$

where scalar weights λ_A and λ_B are tunable parameters to normalize the magnitude of different types of relationships, because the number of edges may be largely different among relationships. As there are two types of priors, we call this a *dual-prior model* and depict its plate notation in Figure 3(b). The *dual-prior model* also degenerates to LDA when λ goes to zero.

Note that these two models are originated from LDA but can work (as a framework) with any topic models delivering topic association count matrices. Especially, the *mixture model* can also work with topic models delivering only *conditional distributions* ($p(z|r)$ and $p(w|z)$) without *joint distributions* ($p(r,z)$ and $p(w,z)$, which can be produced from the raw topic count matrices). For example, when there are two types of edges E_A and E_B, and PLSA performs best for E_A and LDA performs best for E_B, it is possible to initially learn topics from E_A using PLSA and incorporate the learned topics when we learn topics for E_B using LDA [6]. In this way, different types of edges can be treated differently.

2.3 Symmetry and Topic Incorporation Order

For proper topic incorporation, there are two things to be considered. We discuss them in this section.

Symmetry: different from textual corpora consisting of asymmetric relationships from documents to terms, there can be many types of symmetric relationships in complex graphs. For example, query-page relationships can be considered in both directions: query-page or page-query. Thus, we initially tried to derive Gibbs sampling equations for the symmetric model as well for the *mixture model* and the *dual-prior model*. However, both the Gibbs sampling equations are equivalent as mentioned in [12]:

$$\begin{aligned} p(z_{a,b}|\cdot) &= p(z|r_a)p(w_{a,b}|z) \\ &\propto p(r_a)p(z|r_a)p(w_{a,b}|z) = p(z)p(r_a|z)p(w_{a,b}|z). \end{aligned}$$

The difference between the asymmetric model and the symmetric model is in the normalization stage after estimating the association count matrices, R and W. While the asymmetric model gets $p(z|r)$ and $p(w|z)$, the symmetric model gets $p(z)$, $p(r|z)$ and $p(w|z)$, from those matrices. As we incorporate those matrices instead of the conditional distributions in the topic incorporation, we do not need to care about symmetry for the topic incorporation.

Incorporation order: there are many possible orders in the topic incorporation. For example, for the DBLP dataset shown in Figure 1(a), we can consider 6 possible incorporation orders: $DT{\rightarrow}UD{\rightarrow}VD$, $DT{\rightarrow}VD{\rightarrow}UD$, $UD{\rightarrow}VD{\rightarrow}DT$, $UD{\rightarrow}DT{\rightarrow}VD$, $VD{\rightarrow}UD{\rightarrow}DT$, and $VD{\rightarrow}DT{\rightarrow}UD$. In terms of a generative model, the order $UD{\rightarrow}DT{\rightarrow}VD$ seems most reasonable and is also chronologically correct. However, for a complex graph like a web graph in Figure 1(b), it is not easy to find an appropriate chronological order because each edge can be generated without any specific order. Thus, instead of the chronological rule, we came up with two effective rules for deciding the incorporation order: (1) denser [7] edges to sparser edges, and (2) textual edges to non-textual edges. Since later topic inferences are largely affected by early-set topics, it is important to select appropriate initial edge types to start with. The denser edges obviously form better topics, and so do the textual edges because they allow multiple edges between each document and term [8]. We will show experimental results on the incorporation order in Section 3.1.2.

3. EXPERIMENTS AND ANALYSES

We evaluate our models with two types of datasets: a bibliographic dataset from DBLP and online search logs from a commercial search engine, Bing. The DBLP dataset is used to fairly evaluate our models with previous state-of-the-art models and online search logs are used to demonstrate usefulness of our approach in a practical environment.

3.1 DBLP Experiment

In this section, we report the prediction accuracy of our models with the DBLP dataset. Through this experiment, we show the followings: (1) the *dual-prior model* performs better than the *mixture model*, (2) our *dual-prior model* achieves the best prediction accuracy, and (3) there are more effective topic incorporation orders.

3.1.1 Dataset

The DBLP dataset has been widely used in evaluating many prediction algorithms. As depicted in Figure 1(a), it consists of four types of entities and three types of relationships among them. We use the same DBLP dataset used in [6, 9]. The numbers of entities and relationships are listed in the column DBLP of Table 1. Moreover, 4,057 authors, 100 papers, and all 20 venues are labeled with one of four categories: *database (DB)*, *data mining (DM)*, *information retrieval (IR)*, and *artificial intelligence (AI)*. We evaluate our models based on the prediction accuracy on these labeled entities.

3.1.2 Accuracy Analysis

To fairly evaluate our models, we follow the same approach in [6, 9]. We compare the prediction accuracy of our models to the following state-of-the-art methods:

- Nonnegative Matrix Factorization (NMF) [13]
- Probabilistic Latent Semantic Analysis (PLSA) [12]
- Laplacian Probabilistic Latent Semantic Indexing (LapPLSI) [4]
- Latent Dirichlet Allocation (LDA) [3]
- Author-Topic Model (ATM) [18]
- Ranking-based Clustering (NetClus) [19]
- Topic Model with Biased Propagation (TMBP) [9]
- Focused Topic Model (FTM) [20]
- Contextual Focused Topic Model (cFTM) [6]

Among these various methods, we briefly explain top-3 performers in Table 2 (excluding ours) in terms of overall accuracy: ATM, TMBP, and cFTM. ATM adds additional author entities into LDA. When there are multiple authors for a paper, it attempts to find out who is the most probable author for each term in the paper. Thus, it has an effect of selecting each term from a more proper topic distribution because each author has her own topic distribution. The cFTM extends FTM, which is developed to deal with sparse (focused) set of topics, and incorporates additional contextual information (authors and venues) when selecting topics. While it automatically finds a proper number of topics due to its non-parametric nature, it involves many

[6] The later topic model should be LDA in our current models.

[7] We measured *density* by simply dividing the number of edges by the multiplication of the number of unique *readers* and that of *writers*.

[8] If multiple edges are allowed, they can be used to measure the strength of the relationship. However, the non-textual edges sometimes do not allow multiple edges between a reader and a writer. For example, a reader cannot follow the same writer more than once in a social graph.

parameters and requires a quite complicated inference process. Different from these two models focusing on enriching θ (topic distribution given other contexts), TMBP takes a different approach of *topic propagation.* After learning topics for papers from the relationships between papers and terms using PLSA, it propagates the learned topics to authors and venues using an Expectation Maximization (EM) algorithm. Compared to ATM and cFTM, our approach is simpler and more flexible. Also, while these models only enrich θ and are limited to a document-centered graph (e.g., DBLP), our approach can be applied to any type of graph because it can enrich ϕ as well as θ. Perhaps TMBP is closest to our approach in the sense that the new topics are inferred with the help of previously learned topics. However, our approach is based on LDA, which solves PLSA's overfitting problem, and more seamlessly incorporate the previously learned topics in the LDA's inference process compared to TMBP, which has two separate processes of PLSA-based topic inference and EM-based topic propagation.

We observe that our *dual-prior model* (UniZ-dual) outperforms all other state-of-the-art methods in terms of overall average AC and NMI. Only cFTM clearly outperforms it in author prediction task. However, our models required only 100 iterations, which is orders of magnitude smaller than that of cFTM $(6,000)$[9]. Also, our models are more general and can be applied to any complex graphs. Our *dual-prior model* is especially good at predicting venue information. Table 3 shows example venue clusters in the four categories. For our experiments, we set $\alpha = \beta = 1$. We first learned topics from DT (edges from D to T) using LDA and incorporated the learned topics to infer topics for VD with $\lambda = 0.1$ $(0.1 \times B^{DT})$. Then, we used a linear combination of two topics, $0.1 \times B^{DT} + 100 \times B^{VD}$, as a prior to infer topics for UD. Note that we used λ to account for magnitude of different types of edges in the graph topology [10]. As explained in Section 2.3, a topic incorporation order of denser edges to sparser edges and textual-edges to non-textual-edges produced a much better result than a chronological order. The incorporation order of $DT \rightarrow VD \rightarrow UD$ produced the best prediction accuracy (reported in Table 3), and the incorporation order of $VD \rightarrow DT \rightarrow UD$ also produced a similar prediction accuracy [11] [12]. It is because later inferences are largely affected by early-set topics. Also, the *dual-prior model* almost always outperformed the *mixture model* in our experiments. It is probably because the former utilizes joint distributions containing more information, while the latter only utilizes conditional distributions as explained in Section 2.2.

As evaluation metrics, we use both prediction accuracy (AC) and normalized mutual information (NMI) [6, 9, 21].

Table 1: Statistics of the datasets

	Meaning	DBLP	B-Log1	B-Log2	C-Log
$\lvert V\rvert$	Venues	20	-	-	-
$\lvert U\rvert$	Users(Authors)	28,702	100,000	10,000	-
$\lvert D\rvert$	Docs(Papers,Pages)	28,569	257,920	27,980	27,980
$\lvert T\rvert$	Terms(Words)	11,771	117,116	24,124	24,124
$\lvert Q\rvert$	Queries	-	281,332	30,242	30,242
$\lvert DT\rvert$	DT edges	2,712,928	-	-	-
$\lvert VD\rvert$	VD edges	28,569	-	-	-
$\lvert UD\rvert$	UD edges	74,632	441,138	42,583	-
$\lvert UQ\rvert$	UQ edges	-	483,839	46,498	-
$\lvert QT\rvert$	QT edges	-	1,362,278	130,841	24,816,679
$\lvert QD\rvert$	QD edges	-	-	-	16,058,804

Table 2: Prediction accuracy on the DBLP dataset. Except ours, all other results are from [6, 9].

Entity	Paper		Author		Venue		Average	
Metric (%)	AC	NMI	AC	NMI	AC	NMI	AC	NMI
NMF	44.55	22.92	-	-	-	-	44.55	22.92
PLSA	59.45	32.75	65.0	37.97	80.0	74.74	68.15	48.49
LapPLSI	61.35	33.93	-	-	-	-	60.70	33.37
LDA	47.00	20.48	-	-	-	-	47.00	20.48
ATM	77.00	52.21	74.13	40.67	-	-	75.57	46.44
NetClus	65.00	40.96	70.82	47.43	79.75	76.69	71.86	55.03
TMBP-RW	73.10	53.13	82.59	67.76	81.75	77.53	79.15	66.14
TMBP-Regu	79.15	59.16	89.81	74.25	82.75	76.56	83.90	69.99
FTM	69.37	43.51	-	-	-	-	69.37	43.51
cFTM	82.73	**62.91**	**92.51**	**76.20**	82.97	76.05	85.73	71.72
UniZ-mix	79.20	53.54	79.40	49.05	90.00	86.18	82.87	62.93
UniZ-dual	**82.75**	59.71	89.00	68.09	**97.25**	**95.57**	**89.67**	**74.46**

The AC is calculated with the following equation:

$$AC = \frac{\sum_{i=1}^{N} \delta(l_i', map(l_i))}{N}, \tag{6}$$

where l_i is labeled category, l_i' is predicted category, and N is the total number of labels. The $\delta(x, y)$ function produces 1 if $x = y$, and 0 otherwise. We need the $map(x)$ function because the predicted category number is usually different from the label category number. The NMI is defined as $MI(C, C')/MI(C, C)$, where C is labeled cluster and C' is predicted cluster. Mutual Information (MI) is calculated as:

$$MI(C, C') = \sum_{c_i \in C, c_j' \in C'} p(c_i, c_j') \log_2 \frac{p(c_i, c_j')}{p(c_i)p(c_j')}, \tag{7}$$

where $p(c_i)$ and $p(c_j')$ are the probabilities of a randomly selected document's belonging to the cluster c_i and c_j' respectively, and $p(c_i, c_j')$ denotes the joint probability that a document belongs to both the clusters. We report both average AC and NMI values from 20 runs in Table 2. Each average value is calculated by equally dividing each sum. We first ran LDA to learn seed topics and incorporated the seed topics when we learn topics for other edges.

3.2 Online Search Experiments

In this section, we report the performance gains of our approach in practical applications using real search logs from Bing. We evaluate our approach in two personalized recom-

[9] One iteration of our models is also much cheaper.

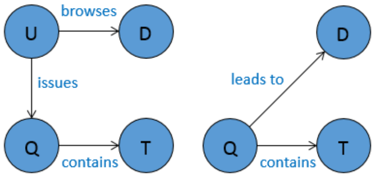
[10] For λ, we tried 0.01, 0.1, 1, 10, and 100 and selected the one producing the best results.

[11] We measured *density* by dividing the number of edges by the matrix size. For example, in the DBLP dataset, $density(DT) = 0.008 = 2712928/(28569 * 11771)$, $density(VD) = 0.05$, and $density(UD) = 0.000091$. Though DT has lower density than VD, it produced similar or better results because DT is a textual dataset and allows multiple edges for each document and term.

[12] The overall accuracy achieved by a chronological order ($UD \rightarrow DT \rightarrow VD$) was 72.16.

Table 3: Venue clusters

DB	DM	IR	AI
VLDB	KDD	SIGIR	IJCAI
ICDE	PAKDD	WWW	AAAI
SIGMOD	ICDM	CIKM	ICML
PODS	PKDD	ECIR	CVPR
EDBT	SDM	AAAI	ECML



(a) B-Log (user behavior log) (b) C-Log (click log)

Figure 4: Structures of two types of search logs

mendation tasks: (1) query recommendation, and (2) page recommendation. We also conduct topic granularity analysis. Finally, we attempt to propagate topics across two very disparate datasets.

3.2.1 Datasets

We use two types of search logs for our experiments. The first search log depicted in Figure 4(a) is a per-user search and browsing log (we call this B-Log from now on), which contains users, their search queries, and pages visited. Terms are simply individual words in a query. We collected this log for $100,000$ users (B-Log1). When we collected this log, we first sampled users and included all the queries issued by the users, and all the pages visited by them. We also collected another user behavior log with $10,000$ users (B-Log2) for another experiment which will be described shortly. The statistics of these logs are listed in Table 1. The density of each log is very low.

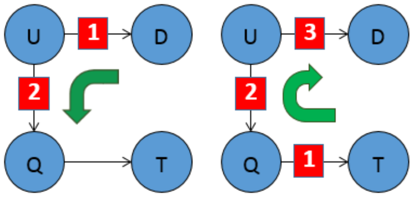
Another type of a search log is an aggregated query-page click log (we call this C-Log from now on) depicted in Figure 4(b). While B-Log is a collection of queries and pages for individual users, C-Log consists of triplets of (query, page, click count), where the click count is the number of clicks between the query and the page across all users (not limited to the users sampled in B-Log). Thus, C-Log has very different statistical characteristics compared to B-Log. Since C-Log is believed to be a very strong signal and is used in various fields of online search (e.g., query intent mining), we attempt to incorporate topics learned from C-Log to B-Log2 in Section 3.2.4. The statistics of C-Log are also listed in Table 1.

3.2.2 Performance Analysis

Since we do not have any judged labels for the search logs, we use the widely-used *perplexity* metric [3,11,22] to measure prediction performance of our models. It is defined as:

$$perplexity(E_{test}) = \exp^{-\frac{\sum_{e\in E_{test}} \log p(e)}{|E_{test}|}}, \qquad (8)$$

where E_{test} denotes all the edges in a test dataset and $p(e)$ denotes an edge prediction probability (i.e., $p(q|u)$ for the query recommendation task and $p(d|u)$ for the page recommendation task). The perplexity quantifies the prediction power of a trained model by measuring how well the model handles unobserved test data. Since the exponent part of Equation (8) is a minus of the average log prediction probability over all the test edges, a lower perplexity means stronger prediction power of the model. We calculated the perplexity for a separate 10% randomly held-out dataset after training a model on the remaining 90% dataset.



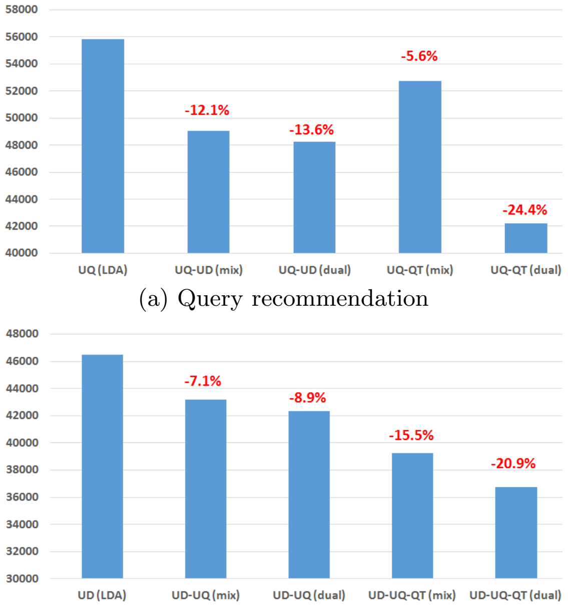
(a) $\underline{U}Q\leftarrow UD$ (b) $\underline{U}D\leftarrow U\underline{Q}\leftarrow QT$

Figure 5: Examples of topic incorporation orders

Based on this perplexity metric, we evaluate our proposed models (the *mixture model* and the *dual-prior model*) on the two recommendation tasks: (1) query recommendation for a user using UQ (user-query edges), and (2) page recommendation for a user using UD (user-page edges). If the perplexity of a model is lower, it means that the model is better at recommending queries or pages to a user (predicting more probable queries or pages). We use LDA as a baseline for both these tasks [13]. We tried two topic incorporation orders for each task (four in total). Figure 5 illustrates two example incorporation orders. In Figure 5(a), topics learned from UD are incorporated when learning topics for UQ for the query recommendation task. We use the notation $\underline{U}Q\leftarrow UD$ to denote this incorporation order (Note that we use the reverse arrow ($\leftarrow$) to put a recommendation task (UQ in this case) on the left). The bar under U indicates that the topics are incorporated onto users. Similarly, $\underline{U}D\leftarrow U\underline{Q}\leftarrow QT$ in Figure 5(b) denotes a topic incorporation from QT to UQ to UD for the page recommendation task. When there is no ambiguity, we also use a simpler notation without the underbar and the arrow. With the simple notation, the former becomes UQ-UD and the latter becomes UD-UQ-QT. For our experiments, we set the parameters as $|Z| = 100$, $\alpha = 0.01$, $\beta = 0.1$. We also set $\lambda_A = \lambda_B = 1$ because the number of edges is not different by orders of magnitude each other. The number of iterations is 100. We averaged perplexity values from five runs with different random seeds.

We report perplexity values of LDA and our two models in Figure 6. We tested two topic incorporation orders for each model: (1) UQ-UD and UQ-QT for the query recommendation task, and (2) UD-UQ and UD-UQ-QT for the page recommendation task. For example, the UQ-UD(mix) in the x-axis denotes the topic incorporation of $\underline{U}Q\leftarrow UD$ in the *mixture model*. The bars show perplexity values and the minus numbers show perplexity drop rates compared to LDA. The left most bar shows LDA which does not benefit from the topic incorporation. Our models seem to be very effective in lowering perplexity because they leverage all the available signals. We also observe that the *dual-prior*

[13] Although we acquired the source code of TMBP, we could not fine-tune it to produce good results on our machine. The cFTM is too expensive and cannot handle the topology of our search logs.



(a) Query recommendation

(b) Page recommendation

Figure 6: Performance Analysis

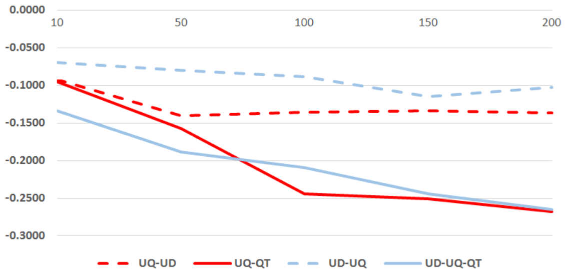


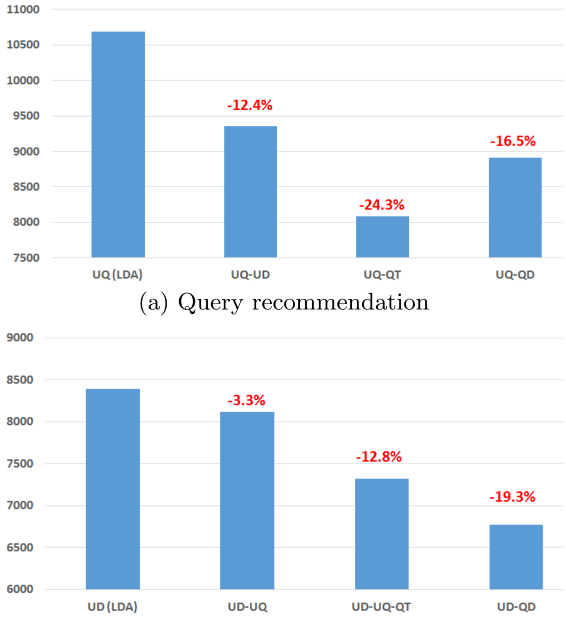
Figure 7: Topic Granularity Analysis

model always performs better than *mixture model* as in Section 3.1.2. Especially, *UQ-QT* and *UD-UQ-QT* achieved the best results (-24.4% and -20.9%) for each task.

3.2.3 Topic Granularity Analysis

We perform an analysis on the number of topics. The number of topics is related to how granular the learned topics are. If the topic granularity is high, it means we can get finer clusters and achieve more accurate targeting. Thus, more granular topics are usually preferred. In our experiments, the perplexity of LDA, which provides seed topics in our framework, did not show a noticeable decrease (i.e., performance improvement) when the number of topics tends toward 100. Rather, it showed an increase (i.e., performance degradation) when the number of topics increases from 100 to 150.

However, when we incorporated topics from other types of edges, we could observe the perplexity drops even when the number of topics increases beyond 150. Figure 7 generated from B-Log1 shows changes in perplexity drop rates (compared to LDA's perplexity) when the number of topics increases from 10 to 200. We observe that *UQ-QT* and *UD-UQ-QT* have consistently lower perplexity values as the number of topics increases. Note that our framework is about the topic incorporation (not LDA) and can work with more granular topic models than LDA.



(a) Query recommendation

(b) Page recommendation

Figure 8: Incorporation of C-Log (the last bars show results using C-Log)

3.2.4 Incorporating a Query-Page Click Log

In this section, we investigate the effect of topic incorporation from C-Log to B-Log. Because C-Log is query-page click counts across all users, it is considered a very strong signal between queries and pages, and has been widely used in modeling query intent in online search. However, the characteristics of C-Log are very different from that of B-Log (aggregated, denser, and extremely power-law). We incorporate topics learned from C-log to B-Log to improve performance in our proposed recommendation tasks. Because the number of edges in C-Log for $100,000$ users was too large to handle in one machine, we reduced the number of users to $10,000$ and prepared B-Log2. The C-Log dataset is collected so that all the queries and pages in B-Log2 are included.

In Figure 8, the leftmost bar is the result from LDA and the next two are from only B-Log2 and the last one is from the combination of C-Log and B-Log2. We set $\lambda = 0.01$ when we incorporate C-Log to B-Log2 due to huge difference in the numbers of edges. We observe that adding C-Log signal produces second best result (-16.5%) in query recommendation task, and the best result (-19.3%) in page recommendation task. It shows that our framework is effective even between very disparate datasets.

4. RELATED WORK

In Section 3.1.2, we introduced some state-of-the-art topic models. We briefly review more topic models in three categories: topic models for authorship, connectivity, and graph.

One of the most popular forms of HINs is documents (consisting of terms) and their authors. To deal with these types of HINs, researchers incorporated authors and their relationships in topic models. These topic models attempt to group documents and authors by assuming that a document is created by authors sharing common topics. The concept of authors (users) was initially introduced by Steyvers et al. [18] in the Author-Topic model (ATM). McCallum et al. [14] also extended ATM and proposed the Author-Recipient-Topic model (ARTM) and the Role-Author-Recipient-Topic model (RARTM) to analyze e-mail networks.

Another popular form of HINs is documents (consisting of terms) and their connectivity information, which is frequently observed in academic bibliography networks and the Web. Topic models for this type of HINs analyze two types of entities (documents and terms) and two types of relationships (contains and links to). Cohn et al. [8] initially introduced a topic model combining PLSA [12] and PHITS [7]. Later, PLSA in this model was replaced with LDA [3] by Erosheva et al. [10]. Nallapati et al. [15] extended Erosheva's model and proposed Link-PLSA-LDA model which applies PLSA and LDA to cited and citing documents, respectively.

Topic models in the previous categories are rooted in the relationships between documents and terms. They enrich topic modeling by adding additional information such as authorship and connectivity. However, there are topic models which do not rely on any textual information and purely depend on structural information (linkage) in a graph. Since they only focus on the graph structure, they can be easily applied to a variety of datasets but there has been relatively less research in this category. Airoldi et al. [2] proposed the Mixed Membership Stochastic Block (MBB) model to analyze pairwise measurements such as social networks and protein interaction networks. Zhang et al. [22] and Henderson et al. [11] dealt with the issues in applying LDA to academic social networks. High popularity issue was addressed by Steck [16]. Our work is in this category and attempts to analyze any complex graph by decomposing it into smaller components.

5. CONCLUSION

In this paper, we introduced a universal topic framework called "UniZ", which represents various types of entities and their edges in a single topic space. By incorporating previously learned topics, UniZ improves prediction and recommendation performance. We also proposed two novel and effective topic models in this framework: the *mixture model* and the *dual-prior model*. In a DBLP prediction task, one of our models performed better than all other state-of-the-art methods. They also achieved significant improvements in query and page recommendation tasks performed with real search logs. We also demonstrated great potential of our approach in dealing with granular topics and disparate datasets.

6. REFERENCES

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

[2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. In *J. Mach. Learn. Res.*, 2008.

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

[4] D. Cai, Q. Mei, J. Han, and C. Zhai. Modeling hidden topics on document manifold. CIKM '08, pages 911–920, New York, NY, USA, 2008. ACM.

[5] Y. Cha, B. Bi, C.-C. Hsieh, and J. Cho. Incorporating popularity in topic models for social network analysis. ACM SIGIR '13, New York, NY, USA, 2013. ACM.

[6] X. Chen, M. Zhou, and L. Carin. The contextual focused topic model. ACM SIGKDD '12, pages 96–104, New York, NY, USA, 2012. ACM.

[7] D. Cohn and H. Chang. Learning to probabilistically identify authoritative documents. ICML '00, pages 167–174, 2000.

[8] D. Cohn and T. Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In *NIPS '00*, 2000.

[9] H. Deng, J. Han, B. Zhao, Y. Yu, and C. X. Lin. Probabilistic topic models with biased propagation on heterogeneous information networks. ACM SIGKDD '11, New York, NY, USA, 2011. ACM.

[10] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed membership models of scientific publications. In *Proceedings of the National Academy of Sciences*, 2004.

[11] K. Henderson and T. Eliassi-Rad. Applying latent dirichlet allocation to group discovery in large graphs. In *ACM SAC '09*, 2009.

[12] T. Hofmann. Probabilistic latent semantic analysis. In *In Proc. of Uncertainty in Artificial Intelligence, UAIâĂŹ99*, pages 289–296, 1999.

[13] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *In NIPS*, pages 556–562. MIT Press, 2001.

[14] 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.

[15] R. M. Nallapati, A. Ahmed, E. P. Xing, and W. W. Cohen. Joint latent topic models for text and citations. ACM SIGKDD '08, New York, NY, USA, 2008. ACM.

[16] H. Steck. Item popularity and recommendation accuracy. RecSys '11, pages 125–132. ACM, 2011.

[17] M. Steyvers and T. L. Griffiths. Probabilistic topic models. *Handbook of Latent Semantic Analysis*, 2007.

[18] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *SIGKDD*, 2004.

[19] Y. Sun, Y. Yu, and J. Han. Ranking-based clustering of heterogeneous information networks with star network schema. ACM SIGKDD '09, New York, NY, USA, 2009. ACM.

[20] S. Williamson, C. Wang, K. A. Heller, and D. M. Blei. The ibp compound dirichlet process and its application to focused topic modeling. In *ICML*, pages 1151–1158, 2010.

[21] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. ACM SIGIR '03, New York, NY, USA, 2003. ACM.

[22] H. Zhang, B. Qiu, C. L. Giles, H. C. Foley, and J. Yen. An lda-based community structure discovery approach for large-scale social networks. In *IEEE ISI'07*, 2007.